ASPAI
TENERIFE•2023

Advances in Signal Processing and Artificial Intelligence

Proceedings of the 5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2023)

Edited by Sergey Y. Yurish

IFSA

Advances in Signal Processing and Artificial Intelligence:

Proceedings of the 5th International Conference on Advances in Signal Processing and Artificial Intelligence

7-9 June 2023
Tenerife (Canary Islands), Spain

Edited by Sergey Y. Yurish

IFSA Publishing

Sergey Y. Yurish, *Editor*
Advances in Signal Processing and Artificial Intelligence
ASPAI' 2023 Conference Proceedings

E-mail (for orders and customer service enquires): ifsa.books@sensorsportal.com

Visit our Home Page on https://sensorsportal.com/ifsa_publishing.html

ASPAI Conference Website: https://aspai-conference.com/

ISSN: 2938-5350
ISBN: 978-84-09-48561-1
BN-20230602-XX
BIC: UYQ

Contents

Foreword

On behalf of the ASPAI' 2023 Organizing Committee, I introduce with pleasure these proceedings devoted to contributions from the 5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2023), 7-9 June 2023, Tenerife (Canary Islands), Spain.

Advances in artificial intelligence (AI) and signal processing are driving the growth of the artificial intelligence market as improved appropriate technologies is critical to offer enhanced drones, self-driving cars, robotics, etc. Today, more and more sensor manufacturers are using machine learning to sensors and signal data for analyses. The machine learning for sensors and signal data is becoming easier than ever: hardware is becoming smaller and sensors are getting cheaper, making Internet of things devices widely available for a variety of applications ranging from predictive maintenance to user behavior monitoring. Whether we are using sounds, vibrations, images, electrical signals or accelerometer or other kinds of sensor data, we can build now richer analytics by teaching a machine to detect and classify events happening in real-time, at the edge, using an inexpensive microcontroller for processing - even with noisy, high variation data.

The global **artificial intelligence market** size was valued at **US $ 136.55** billion in 2022 and is projected to expand at a compound annual growth rate (CAGR) of 37.3 % from 2023 to 2030. Artificial intelligences currently transforming the manufacturing industry. Virtual reality, automation, Internet of Things (IoT), and robotics are some important features of AI that are benefitting the manufacturing industry. AI has been one of the fastest-growing technologies in recent years. The market growth is mainly driven by factors such as the increasing adoption of cloud-based applications and services, growing big data, and increasing demand for intelligent virtual assistants. The major restraint for such market is the limited number of AI technology experts.

The Series of ASPAI Conferences have been launched to fill-in this gap and to provide a forum for open discussion and development of emerging artificial intelligence and appropriate signal processing technologies focused on real-word implementations by offering Hardware, Software, Services, Technology (Machine Learning, Natural Language Processing, Context-Aware Computing, Computer Vision and Signal Processing). The goal of the conference is to provide an interactive environment for establishing collaboration, exchanging ideas, and facilitating discussion between researchers, manufacturers and users. The first ASPAI conference has taken place in Barcelona, Spain in 2019, the second (2020) and the third (2021) – in the virtual format due to the COVID-19 pandemic – in Berlin, Germany and Porto, Portugal. In 2022 we have returned to the in-person conference format in Corty, Greece.

The conference is organized by the International Frequency Sensor Association (IFSA) - one of the major professional, non-profit association serving for sensor industry and academy since 199, in technical cooperation with media partners – IOS Press (journal *'Integrated Computer-Aided Engineering'*) and World Scientific (*International Journal of Neural Systems*). The conference program provides an opportunity for researchers interested in signal processing and artificial intelligence to discuss their latest results and exchange ideas on the new trends.

I hope that these proceedings will give readers an excellent overview of important and diversity topics discussed at the conference.

We thank all authors for submitting their latest work, thus contributing to the excellent technical contents of the Conference. Especially, we would like to thank the individuals and organizations that worked together diligently to make this Conference a success, and to the members of the International Program Committee for the thorough and careful review of the papers. It is important to point out that the great majority of the efforts in organizing the technical program of the Conference came from volunteers.

Prof., Dr. Sergey Y. Yurish
ASPAI' 2023 Conference Chairman

(003)

Estimation of Selected Geometric Dimensions during Manufacturing of Aircraft Accessory Gearboxes on a CNC Machine Using ANFIS

P. Gierlak [1], **M. Muszyńska** [1], **A. Burghardt** [1], **D. Szybicki** [1], **K. Kurc** [1] **and B. Bomba** [2]
[1] Rzeszów University of Technology, Faculty of Mechanical Engineering and Aeronautics,
al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
[2] Pratt & Whitney Rzeszów S. A., ul. Hetmańska 120, 35-078 Rzeszów, Poland
Tel.: + 48 178651854
E-mail: pgierlak@prz.edu.pl

Summary: The subject of this paper includes quality control of the most important geometric dimensions of an accessory gearbox for a PW1000G series aircraft engine. The main objective of the research described in this paper was to develop an inference tool aimed at eliminating the need for coordinate measuring machines, in favor of using geometric measurements taken on a CNC machine, in the production inspection of aircraft accessory gearboxes casing covers. The research presented in the paper focused on the most important geometric dimensions, namely the positions of the bearing seat axes. For this purpose, a method was developed to determine the deviation of bearing seat positions from nominal values based on measurements on the CNC machine using neuro-fuzzy models. These models take into account the relationship between the results of measurements in different states of clamping on the CNC machine and coordinate measuring machine.

Keywords: Accessory gearbox, On-machine measurements, CNC, CMM, ANFIS.

1. Introduction

The subject of this paper includes quality control of the most important geometric dimensions of an accessory gearbox (AGB) for a PW1000G series aircraft engine. This AGB provides the drive of all external accessories of the engine. The AGB consists of induction-hardened gears supported in conventional roller bearings built into the gearbox housing. The aircraft accessory gearbox housing is designed for clamping and mutual arrangement of various components, assemblies and mechanisms, mainly bearings, shafts and gears. The gearbox housing is made of two elements, which are the casing and the cover. Both components are made as aluminum alloy castings and are classified as thin-walled components due to the small ratio of wall thickness to external dimensions. The AGB casing covers are mainly prone to deformation under low loads and therefore their manufacturing, machining and measurement methods are among the more difficult issues in aerospace manufacturing and they require a special approach [1].

One of the most important geometric characteristics of an AGB is the arrangement of bearing seats. It has a direct impact on factors that determine the service life of aircraft gearboxes, such as [2]: change in the clearances of a pair of mating gears, improper gear meshing, surface flaking and the risk of damage or tooth breakage. Therefore, the AGB housing after machining must meet the following requirements [3]. The high required accuracy entails the necessity to minimize the influence of the human factor on the technological process of gearbox manufacturing and geometric measurements. For this reason, in the manufacturing process of gearbox components, eliminating human factor from the process is sought.

In the case of the discussed AGB casing cover, the number of dimensional and shape-related characteristics produced for each piece is about 100. Due to the required manufacturing accuracy, the dimensional and shape-related characteristics are obtained as standard using a coordinate measuring machine (CMM). Taking such measurements requires unclamping the AGB casing cover from the CNC machine, pressure cleaning the workpiece, drying it, transporting it to the measurement laboratory and setting it up on the CMM. The biggest drawbacks of this type of approach are the high cost of purchasing and maintaining a CMM, additional time, and the introduction of the human factor into the process. The measurement laboratory is also a so-called bottleneck in the production process. This is due to the fact that a control cell equipped with a CMM usually supports multiple production lines.

In recent years, probing packages with dedicated software to support them have become standard equipment in CNC machining centers. This made it possible to take basic measurements in the machining center and save the results as CNC system variables or measurement reports. It has also opened a new field for research and engineering works known as on-machine measurement [4] or on-machine probing [5]. The core feature of this subject is that it allows reliable measurements of workpieces to be carried out during and after production using CNC machines [6].

Research in this area addresses both basic and more advanced issues. Authors [7] studied the uncertainty of an inspection probe based on point coordinate measurement, length measurement using one and simultaneously two CNC machine's axes, and length measurement using multiple measurement strategies. In paper [8] authors analyzed the possibility of using a measuring probe on CNC machine to determine selected linear dimensions of workpieces during the machining process, as well as after it. Authors in [9] investigated the use of a CNC machining center with a touch probe as a measuring device. The capabilities of the measuring system were verified based on a combined method with length measurement using a simple touch probe in combination with the use of a laser interferometer.

In all of the above-mentioned works, high-stiffness workpieces that do not undergo significant deformation in the machining process were measured. A gap that needs to be filled is the final inspection of thin-walled components after the machining process with the use of CNC machines. The main objective of the research described in this paper was to develop an inference tool aimed at eliminating the need for CMMs, in favor of using geometric measurements taken on a CNC machine, in the production inspection of AGB casing covers for a PW1000G series aircraft engine. The research presented in the paper focused on the most important geometric dimensions, namely the positions of the bearing seat axes.

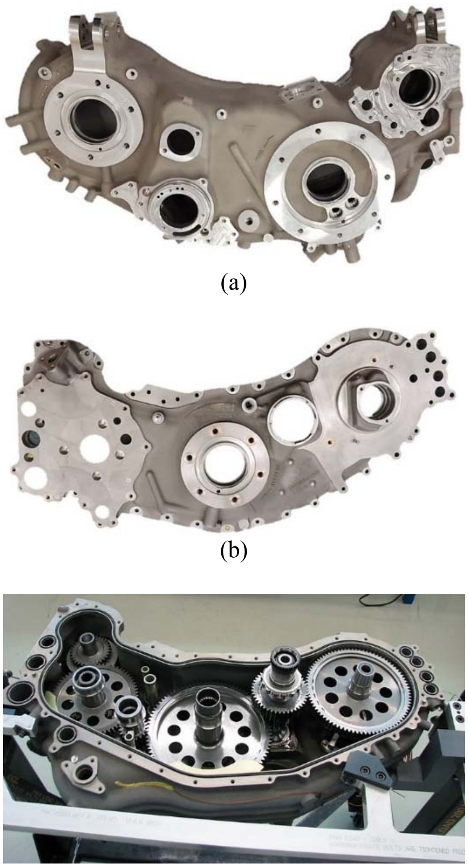
2. Material and Methods

2.1. PW1000G Engine Accessory Gearbox

A view of the PW1000G accessory gearbox is shown in Fig. 1. Fig. 1a shows the AGB housing, Fig. 1b shows the AGB casing cover, and Fig. 1c shows the interior view of the accessory gearbox, which is a kinematic chain consisting of gears.

The key features of the AGB housing, from the point of view of gearbox quality, are the positions of the bearing seats in which the gear shafts are fixed. A particularly difficult element to manufacture is the AGB casing cover (Fig. 1b) owing to the fact that it undergoes significant deformation after unclamping after the machining process, which results from the release of stresses. The deformation of the front surface of the AGB casing cover reaches up to 0.6 mm [3], which affects the spacing of the bearing seats. The inspection of the proper spacing of the bearing seats in the AGB casing cover relative to the base elements is usually a factor in determining the AGB casing cover's compliance with technical requirements, and was chosen as the subject of this research. Fig. 2 shows a model of AGB casing cover with the base holes (9) and (1) and bearing seats (2-8) marked. The center of hole B is the origin of the reference system in both the CMM as well as the CNC machine. The X axis of the reference system passes through the center of the base hole A. The position tolerance of the bearing seat axes is 75 µm.

Most of the AGB components are measured on the CMM in a freely supported state. The AGB casing cover is an exception, as it is measured in its clamped state on the CMM. This is due to the fact that after the gearbox is assembled, the cover, as a prone part, adapts to the face of the stiffer casing. The way the cover is clamped on the CMM corresponds to the clamping of the cover to the casing, while the way it is clamped during machining on the CNC is quite different, due to the need for machining operations.


(a)

(b)

(c)

Fig. 1. A view of the PW1000G engine accessory gearbox: a) AGB housing, b) AGB casing cover, c) kinematic chain.

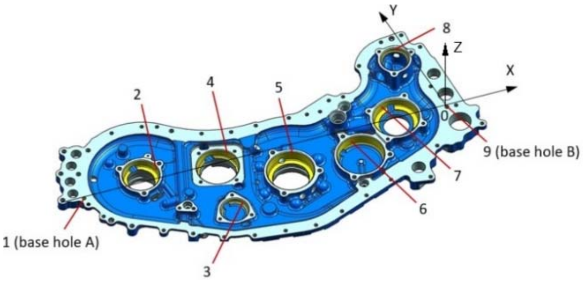
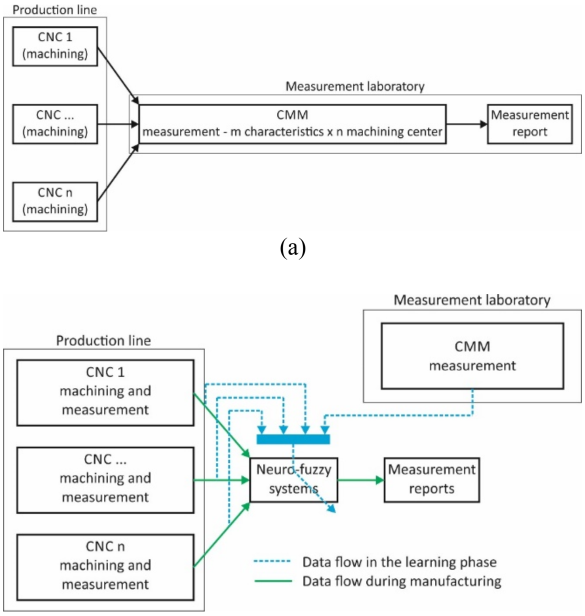


Fig. 2. AGB casing cover with marked base holes and bearing seats (in yellow) and reference system.

2.2. Quality Control System

The problem to overcome when trying to replace the CMM with a CNC machine is the dissimilarity of conditions when taking measurements. This dissimilarity is due, among other things, to the different clamping condition of the cover on the CNC machine and on the CMM. The change in the clamping condition affects the release of stresses, which in turn causes the workpiece to deform. Thus, the results of measurements on the CNC machine and on the CMM differ significantly. For this purpose, a method was developed to determine the deviation of bearing seat positions from nominal values based on measurements on the CNC machine using neuro-fuzzy models. These models take into account the relationship between the results of measurements in different states of clamping on the CNC machine and CMM.

In Fig. 3 two options for a quality control system are presented. The first option (Fig. 3a), which can be called conventional, relies on measurements by a single CMM of workpieces produced on multiple CNC machines. The second option (Fig. 3b) relies on the Adaptive Neuro Fuzzy Inference System (ANFIS), which generates outputs based on measurements from the CNC machine only, which are the values of deviations of bearing seat positions from nominal values. Data from the CMM is needed only in the learning phase of the system. In Fig. 3b, the blue color indicates the data flow during learning, while the green color indicates the data flow during the manufacturing.



(b)

Fig. 3. Quality control of production: a) carried out entirely in the measurement laboratory; b) carried out entirely on the production line.

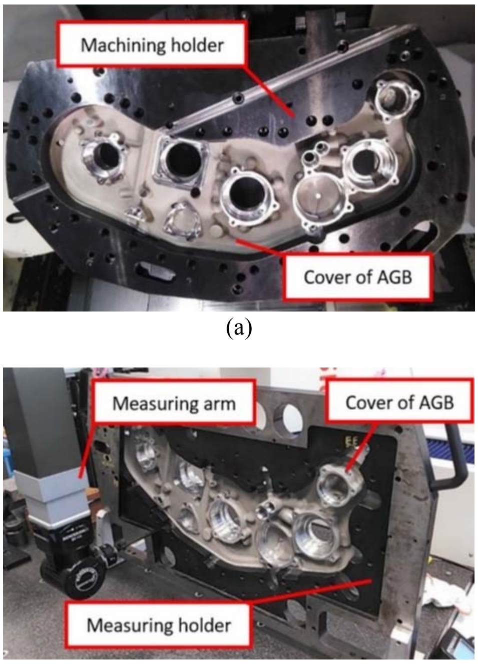
Neuro-fuzzy systems were chosen for processing measurement results from the CNC machine, as they combine the advantages of artificial neural networks (ANN) and fuzzy systems. The main advantages of the ANN include: the ability to learn nonlinear mappings from a dataset, the properties of generalizing the information contained in the data, and robustness to errors and data noise [10].

The main drawback is that the relationship between input and output data cannot be represented in the form of clear principles, and the parameters of the ANN do not have an interpretation related to the subject being solved. In contrast, a feature of fuzzy systems [11] is the clear representation of knowledge in the form of inference rules, and the ability to operate on the basis of not only numerical, but also linguistic data; thus, it is imprecise. It was shown in [12], that a fuzzy system can be converted into an ANN structure by creating an ANFIS that combines the advantages of both types of structures. This allows the possibility of learning nonlinear mappings from a dataset, and at the same time obtaining an inference rule base that is a readable representation of knowledge.

Test of accuracy of the five-axis Okuma MU6300V machining center is presented in paper [13], and accuracy of the machine measurement system based on Renishaw RMP600 probes used in research is presented in paper [14].

2.3. Measurement Data

In order to acquire data for training ANFIS, the positions of the bearing seats (holes 2 to 8 in Fig. 2) were measured during the machining of 97 covers in a normal production cycle. The measurements were taken on a CNC machine and on a CMM (Fig. 4). After data collection was completed, the accuracy of the CNC machine was re-evaluated and no significant change in geometric error values was registered. The data acquisition process took 8 months, due to the fact that the monthly production of AGBs is a dozen pieces.



(b)

Fig. 4. AGB casing cover: a) in a fixture in a CNC machine, b) in a measurement laboratory on a CMM.

On the basis of measurement results, hole position deviations (HPDs) for bearing seats with respect to the nominal values specified in the documentation were determined. Deviations determined from measurements on the CMM were designated as TPCMMji, whereas deviations determined from measurements on the CNC machine were designated as TPCNCji, where j = 2, 3, ..., 8, is the bearing seat number, and i = 1, 2,...,97 is the cover number. Position deviations for base holes A numbered 1 were determined in a similar manner. In the next step, the correlation between the deviations obtained on the CMM and the CNC machine was determined by determining the Pearson correlation coefficient (Table 1). The axis of hole 1 is the origin of the coordinate system for the measurements on the CNC machine and the CMM and is therefore not included in Table 1. The values of the correlation coefficient are in the negligible to moderate range. In addition, the measurement errors on the CNC machine were determined, defined for each hole as:

$$E_{ji} = |TP_{CMMji} - TP_{CNCji}|, \tag{1}$$

followed by the maximum error E_{MAX} and the average error E_{AVE} of the measurement on the CNC machine for each hole:

$$E_{MAXj} = \max_{i}(E_{ji}), \tag{2}$$

$$E_{AVEj} = \frac{1}{97}\sum_{i=1}^{97} E_{ji} \tag{3}$$

The maximum and average error values obtained are shown in Table 2.

Table 1. Correlation of bearing seat position deviations measured using CNC and CMM.

Bearing seat number j	Value of the Pearson correlation coefficient
1	0.13
2	0.09
3	0.28
4	0.02
5	0.43
6	0.30
7	0.18
8	0.22

Table 2. Values of measurement errors E_{MAX} and E_{AVE}.

Bearing seat number j	E_{MAX}	E_{AVE}
	mm	
1	0.036	0.013
2	0.032	0.012
3	0.028	0.010
4	0.030	0.010
5	0.024	0.009
6	0.028	0.004
7	0.014	0.003
8	0.067	0.009

The maximum values of measurement errors take values close to the half of positional tolerance for each hole, and for the bearing seat 8, a value close to the maximum tolerance of its position. The presented values of correlation and measurement errors indicate that the measurements on the CNC alone are unreliable, since they are not strongly correlated with the results of measurements on the CMM. This is due to the fact that the way the cover is clamped in the fixture on the CMM corresponds to the clamping of the cover to the casing and is different from that on the CNC machine. A different state of stress, resulting in deformation of the cover, is responsible for the difference in the results of measurements on the CNC machine and CMM.

The base face of the cover, in which the base hole 1 is located (Fig. 2), is firmly clamped in the fixture. Thus, it can be assumed that the position of the face of the cover does not change during its machining and measurement on the CNC machine, or the changes in position are negligibly small. The bearing seats, on the other hand, are located on a free surface (diaphragm, blue color in Fig. 2), which is not supported by anything during the machining of the seats. This surface, along with the bearing seats, is subject to relaxation and displacement when the cutting forces disappear. Thus, it was assumed that this observation provides important information about the process of measuring the positions of the bearing seats on the CNC machine, and that this information can provide an important premise in fuzzy inference, and will make it possible to realize the scheme shown in Fig. 3b.

The error TP_{CNCji}, which is determined for each characteristic in relation to the nominal value, does not contain any information about local cover deformations. Thus, it is necessary to obtain a variable describing the relationship between the measurement results of the two selected elements. A reference of measurement results not to a global system, but to another measured element, will provide information on cover deformations, and indirectly on the released stress. As a result of numerous tests carried out, it turned out that all hole position deviations show the highest correlation with the deviation of hole 1 (Table 3).

Table 3. Correlation of bearing seat position deviations with deviation of base hole 1 – according to measurements on CNC.

Correlation coefficient	Value of the Pearson correlation coefficient
R(2,1)	0.80
R(3,1)	0.62
R(4,1)	0.67
R(5,1)	0.75
R(6,1)	0.36
R(7,1)	0.01
R(8,1)	0.000044

A variable described by the following relationship was proposed:

$$E_{1i-ji} = \left|TP_{CNC1i} - TP_{CNCji}\right|, \tag{4}$$

where T_{PCNC1i} is the deviation of the base hole 1 position from the nominal value determined by measurements on the CNC machine.

Summing up the above considerations, it should be stated that the results of measurements carried out on the CNC machine alone cannot be used to generate an appropriate measurement report. To overcome this difficulty, an application of ANFIS is provided, in which inputs will include two values: TP_{CNCj} which is the deviation of a given bearing seat expressed with respect to nominal values, and E_{1i-ji}, which is the deviation expressed relatively. This second input to ANFIS, combined with the output pattern (actual deviation) determined from measurements on the CMM, provides information about the relationship between the results of measurements made on the CMM and CNC. It should be made clear that only the output pattern, and only at the learning phase of ANFIS, is used from measurement data from the CMM. The inputs, on the other hand, come only from measurement data from the CNC machine

3. Calculations

The aviation industry is characterized by the fact that production usually involves up to dozens of engines and gearboxes of a given type manufactured per year. Consequently, the time to collect the required amount of data from the manufacturing process is long. ANFIS learning and testing, on the other hand, requires having as much data as possible, which additionally needs to be divided into a learning, validation and test set. In contrast, in the case under review, only 97 samples were included in the dataset. To solve this inconvenience, an approach based on generating a synthetic learning dataset was used [15]. Synthetic data was generated using interpolation procedures for real data. Using interpolation methods, curves passing through all nodal points [16], which are measurement points, were obtained. Since there are various interpolation methods, these methods were evaluated in the context of generating synthetic data in the analyzed problem. The following methods were tested: linear interpolation, polynomial interpolation, spline interpolation and Akima interpolation. The analysis shows that the best representation of the original measured data was achieved using the Akima method for holes 1, 3, 4, 6, 7, 8, and the spline for holes 2 and 5. The synthetic data obtained in this way was used in the ANFIS learning process. On the other hand, the procedure of validation and testing of the obtained neuro-fuzzy models was carried out using a set of real measurement data.

A Takagi-Sugeno model, separate for each bearing seat under consideration, was provided for solving the problem of generating actual deviation of bearing seats positions. In each case, the same general structure of the neuro-fuzzy system was provided (Fig. 5).

The differences in the models, depending on the bearing seat number, concerned the number of fuzzy sets, singletons and shapes of membership functions. These parameters were selected on the basis of the tests conducted, in which the quality of the system's outputs was examined for the number of fuzzy sets for input variables from 2 to 4, the number of singletons from 4 to 16, and the following membership functions: triangular, trapezoidal, Gaussian, bell-shaped, bell-shaped asymmetric, π-shaped. The selected parameters of the system, along with the shape of the model surface, are presented in Table 4.

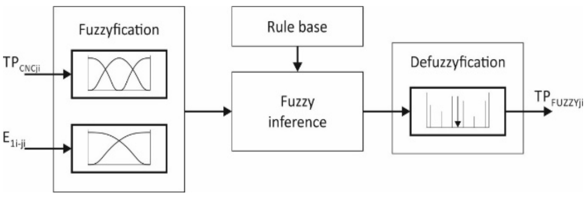


Fig. 5. A diagram of a neuro-fuzzy system.

Table 4. Parameters of ANFIS models selected for individual bearing seats: NMF1 is the number of membership functions for input 1, NMF2 is the number of membership functions for input 2, NS is the number of singletons.

Bearing seat number j	**Model parameters**	
2	Membership function	Bell-shaped
	NMF1	2
	NMF2	3
	NS	6
3	Membership function	Bell-shaped asymmetric
	NMF1	4
	NMF2	4
	NS	16
4	Membership function	Gaussian
	NMF1	4
	NMF2	4
	NS	16
5	Membership function	Gaussian
	NMF1	3
	NMF2	2
	NS	6
6	Membership function	Bell-shaped
	NMF1	3
	NMF2	3
	NS	9
7	Membership function	Bell-shaped
	NMF1	4
	NMF2	3
	NS	12
8	Membership function	Bell-shaped
	NMF1	3
	NMF2	2
	NS	6

The complexity of the ANFIS model, related to the number of fuzzy sets and singletons**,** is not of great importance, since the computation time is not a critical criterion and is of negligible value compared to the time of manufacturing and measurement on the CNC machine. The advantage of using bell-shaped and Gaussian functions is the smoothness of the model surface, which is not always the case with, for example, trapezoidal or triangular functions.

4. Results

Trained ANFIS were used to estimate actual bearing seat deviations from CNC machine data. For each hole, the estimation errors of bearing seat deviations determined by ANFIS models were determined:

$$\delta_{ji} = |TP_{CMMji} - TP_{ANFISji}|, \tag{5}$$

followed by the maximum error δ_{MAX} and the average error δ_{AVE} estimated by ANFIS models for each hole:

$$\delta_{MAXj} = \max_{i}(\delta_{ji}), \tag{6}$$

$$\delta_{AVEj} = \frac{1}{97}\sum_{i=1}^{97}\delta_{ji} \tag{7}$$

The maximum and average error values obtained are shown in Table 5.

Table 5. δ_{MAX} and δ_{AVE} error values.

Bearing seat number j	δMAX	δAVE
	mm	
2	0.012	0.003
3	0.015	0.003
4	0.011	0.002
5	0.014	0.004
6	0.010	0.003
7	0.014	0.003
8	0.013	0.006

It can be seen from Tables 5 and 2 that the reduction in errors is significant, especially in the case of bearing seats, for which the measurement results from the CNC were subject to large errors.

5. Conclusions

The main findings of the research are as follows:

1. Using properly trained ANFIS based on knowledge extracted from the data, the need for CMM can be eliminated in favor of CNC for geometric inspection.

2. Training ANFIS requiring large amounts of data, which is difficult to obtain in low-volume manufacturing, was made possible by expanding the dataset by generating synthetic data.

3. A separate ANFIS must be built for each bearing seat, and the systems created differ significantly in the number and shape of membership functions, the number of singletons, and the surface of the model.

4. ANFIS models are not universal so their application is limited to the CNC machine on which the machining and measurements were carried out. However, the approach to the problem itself is universal and can be applied to other machines and machined parts, with the entire process presented in Fig. 3.

References

[1]. H. Wang, M. X. Zhou, W. Z. Zheng, Z. B. Shi, H. W. Li, 3D machining allowance analysis method for the large thin-walled aerospace component, *International Journal of Precision Engineering and Manufacturing*, Vol. 18, 2017, pp. 399-406.

[2]. S. P. Radzevich, Dudley’s Handbook of Practical Gear Design and Manufacture, *CRC Press*, 2016.

[3]. G. Bomba, P. Gierlak, Dimensional Control of Aircraft Transmission Bodies Using CNC Machines and Neuro-Fuzzy Systems, *Applied Sciences*, Vol. 9, 2019, 4094.

[4]. B. Wen, Y. Shimizu, Y. Watanabe, H. Matsukuma, W. Gao, On-machine profile measurement of a micro cutting edge by using a contact-type compact probe unit, *Precision Engineering*, Vol. 65, 2020, pp. 230-239.

[5]. S. Sepahi-Boroujeni, J. R. R. Mayer, F. Khameneifar, Repeatability of on-machine probing by a five-axis machine tool, *International Journal of Machine Tools and Manufacture*, Vol. 152, 2020, 103544.

[6]. S. Li, L. Zeng, P. Feng, D. Yu, An accurate probe pre-travel error compensation model for five-axis on-machine inspection system, *Precision Engineering*, Vol. 62, 2020, pp. 256-264.

[7]. E. Jacniacka, L. Semotiuk, Experimental methods for determining uncertainty of measurement using inspection probes, *Maintenance and Reliability*, Vol. 15, 2013, pp. 246-252.

[8]. B. Kamieńska-Krzowska, L. Semotiuk, M. Czerw, Analysis of possibility of objective sound application to active checking on FV580A vertical machining centre, *Acta Mechanica et Automatica*, Vol. 1, 2007, pp. 19-24.

[9]. M. Holub, R. Jankovych, O. Andrs, Z. Kolibal, Capability assessment of CNC machining centres as measuring devices, *Measurement*, Vol. 118, 2018, pp. 52-60.

[10]. K. Gurney, An Introduction to Neural Networks, *CRC Press*, 2018.

[11]. H. T. Nguyen, C. Walker, E. A. Walker, A First Course in Fuzzy Logic, *CRC Press*, 2018.

[12]. H.-X. Li, C. L. P. Chen, The equivalence between fuzzy logic systems and feedforward neural networks, *IEEE Transactions on Neural Networks*, Vol. 11, 2000, pp. 356-365.

[13]. G. Bomba, P. Gierlak, Assessment of geometric accuracy of a 5-axis CNC machine in the context of machining aircraft transmission housings, *Universal Journal of Mechanical Engineering*, Vol. 8, 2020, pp. 258-264.

[14]. G. Bomba, A. Ornat, P. Gierlak, Geometric measurements on a CNC machining device as an element of closed door technology, *Sensors*, Vol. 21, 2021, 4852.

[15]. K. El Emam, L. Mosquera, R. Hoptroff, Practical Synthetic Data Generation: Balancing Privacy and the Broad Availability of Data, *O’Reilly Media*, 2020.

[16]. A. Ralston, P. Rabinowitz, A First Course in Numerical Analysis, *Dover Publications*, 2001.

5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2023),
7-9 June 2023, Tenerife (Canary Islands), Spain

(007)

Building Kalypso: The Navigation System

Nikos Vasilopoulos, Ergina Kavallieratou and Efstathios Stamatatos
Department of Information and Communication Systems Engineering
University of the Aegean, Karlovasi, Samos, Greece
E-mail: kavallieratou@aegean.gr

Summary: This paper describes the navigation system of a robotic vehicle named "Kalypso" which was developed to inspect nets in fish farms. Kalypso can distinguish between clean areas on the net and areas that are either torn or covered in algae. The need for underwater works in fish farms is being increased. These works include the supervision of the operation of the facilities, the maintenance and repair of faults as well as the expansion, upgrading and construction of new units. Most tasks are particularly demanding and time-consuming and are traditionally carried out by divers, who are exposed to fatigue and other risks. The last decades, AUVs have replaced divers in many cases, reducing costs and accidents [1]. In addition, AUVs undertake tasks that cannot be performed by humans, such as exploring at great depths or unknown and dangerous environments.

Keywords: Fish farms, Navigation, Ultrasonic sensor, Artificial intelligence, Autonomous Unmanned Vehicle (AUV).

1. Introduction

Over the past few decades, there has been a growing demand for underwater activities in fish farms. These activities encompass overseeing facility operations, addressing maintenance issues, repairing faults, and even constructing new units or expanding existing ones. Many of these tasks are highly challenging and time-consuming, typically assigned to divers who face fatigue and various risks. The accurate navigation of Autonomous Underwater Vehicles (AUVs) frequently proves vital in accomplishing these tasks successfully, as precise coordinates are necessary for data collection and any necessary interventions. The systems used initially for underwater navigation were mainly based on depth and orientation sensors (like magnetic compasses), as well as acoustic range systems. In recent decades, new sensors with higher accuracy and lower cost have become available. In many cases AUVs combine multiple sensors, which complement each other's functionality and improve overall accuracy. However, errors due to noise and drift can always occur. To overcome these problems, most systems integrate a method for stochastic state estimation [1].

Whether one single or many sensors are used, AUV position is calculated from continuously updated measurements. In most cases a state estimation algorithm processes the data in order to output more reliable results. Assuming gaussian state distribution, Kalman filtering can be applied. Many sensors fusion based systems use an extension of Kalman filter, the Extended Kalman Filter (EKF) that is slower but accommodates non-linear measurement models [2-4]. Systems that rely on acoustic range positioning often combine triangulation with an EKF [5]. Another flavor of Kalman filter is the Unscented Kalman Filter (UKF) that is even slower than EKF but reduces linearization errors further [6]. Invariant Extended Kalman Filter (InEKF) is one more extension of the EKF that has been applied for longer-term localization and faster convergence [7]. In case non-gaussian state distributions and non-linear models need to be incorporated, non-parametric estimation representations like Particle Filters (PFs) are an option [8]. Artificial Neural Networks (ANNs) have also been used in order to approximate non-linear conditions [9].

Localization is achieved as long as precise measurements match to specific points on a map. When a-priori maps are not available, the vehicle has to construct them. An AUV can build the map of the environment, while estimating its position at the same time. This process is called Simultaneous Localisation and Mapping (SLAM) and is being extensively applied in underwater robot navigation since 2000 [10]. SLAMs compare images and features that can be extracted either from acoustic systems or from cameras. In the case of an AUV inside a fishery cage, maps are either available or can be easily constructed if needed by calculating the distance between the vehicle and the nets with an acoustic distance sensor [1] or by optically recognizing visual markers attached on the net.

In this paper, the navigation experiments for Kalypso (Fig. 1), a robot for fisheries, are presented. In Section 2, the navigation system is presented in detail, while the experimental results and the conclusion is given in Sections 3 and 4, respectively.

Fig. 1. The AUV Kalypso.

2. The Navigation System

The main objective of the navigation system is to enable the robot to autonomously navigate through the mesh structure of the fishing cages (Fig. 2) and inspect their condition. To achieve this, various navigation methods were explored and tested during the development process. In more detail, the used techniques are based on variety of sensors and they are described in the following paragraphs.

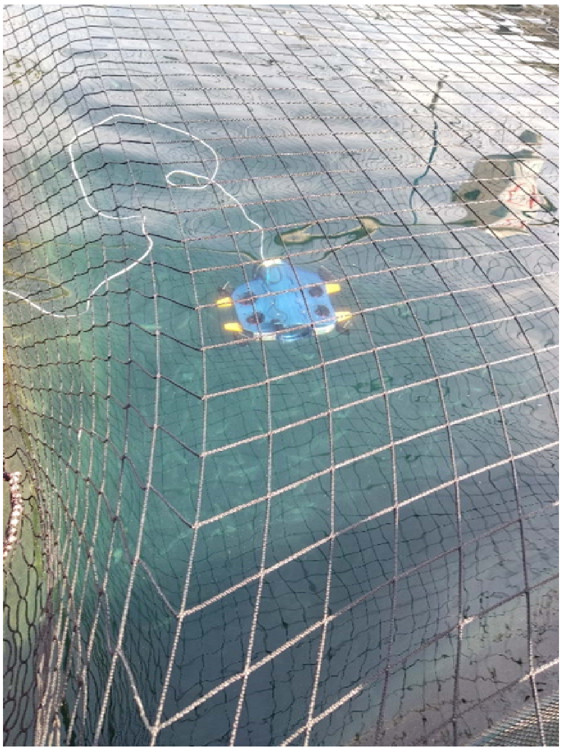


Fig. 2. Navigating close to the nets.

2.1. Navigation with IMU

The code had already been tested in previous missions. The robot would start from a corner of the net and move parallel to the net (facing it) for a specific period of time. The time had been experimentally calculated and was approximately the duration required for the robot to reach the next corner. At that point, it would make a 90-degree left turn and then repeat the process four times (for all sides of the cage) before submerging. The most significant problem with this method is that the robot cannot maintain its distance and orientation relative to the net due to the compass (used to calculate the turn) having a large error.

2.2. Navigation with IMU and Distance Sensor

The distance sensor is placed looking in front. This code was also ready and had been tested before. It successfully addresses the issue of the robot's distance from the net (important for video quality by keeping the robot relatively close to the net). While the sensor effectively calculates and adjusts the distance, it does not correct the problem of the continuously increasing orientation error.

2.3. Navigation with Two Distance Sensors

In this case, two sensors were tested, positioned vertically between them (front and left). The sensors operated at the same frequency and had a long range (50 meters), resulting in receiving signals from multiple reflections and often providing incorrect values. It was not possible to rely on the sensors for controlling the distance from the net or for the robot's rotation (the accuracy of the measurements decreased further as the robot turned).

2.4. Navigation with IMU and Two Distance Sensors

This case is the same as the previous one, with the difference that compass is used instead of sensors for rotation. The problems stemming from both orientation error and sensor interaction still persist.

In order to arrive at a solution that would address the aforementioned problems, we started with the latest code and the logic of the two sensors and made the following changes:

1) Measurements from the sensors are taken into account only when the robot is close to the net (we experimentally observed that in this case, the values are usually reliable);
2) The robot turns manually, with a rotation speed and timing that we adjusted through testing (we found this method to be the most accurate by far);
3) After each correction of the distance from the net, the robot moves backwards (Fig. 3) briefly (this allows it to return to a normal position in case it has touched the net due to sensor inaccuracies);
4) After a sufficient period of lateral movement and three failed attempts to measure the distance to the left side of the net, the robot moves to the right for a short period and then turns (this addresses the scenario where it touches the net due to sensor inaccuracies).

To address these issues, the final version (Fig. 4) of the navigation code involved taking measurements from the sensors only when the robot was close to net (Fig. 5), as it was experimentally observed that in this case, the values were usually reliable. The robot was rotated (by telecontrolling it), with a rotation speed and time that were adjusted after testing, as this was found to be the most accurate method.

The code that was checked and eventually worked satisfactorily operates as follows: The robot starts at a depth of one meter, adjusting its distance from the net, and then moves laterally to the left for a specific period of time. At intervals, it checks its distance from the net. When the time is up, it checks its distance from the side of the net on its left. If the distance is small, the robot rotates 90 degrees to the left. Otherwise, it continues to move laterally for some more time. The process is repeated, with the robot increasing the depth it moves by one meter after four consecutive rotations.

3. Experimental Results

The testing phase of Kalypso involved evaluating the vehicle's ability to identify damaged areas on the net using the algorithm partitioning. The results showed that the vehicle was able to distinguish between clean areas on the net and areas that were either torn or covered in algae with high accuracy. The leak sensor was also evaluated during the initial testing, and it proved to be a critical component in detecting water leakage in the watertight housing and alerting the operator.

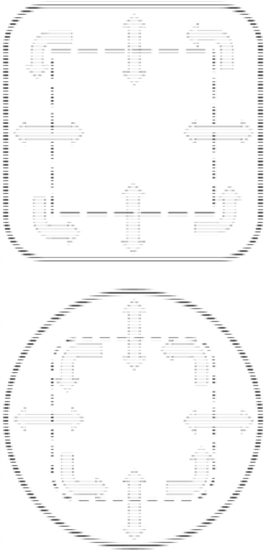


Fig. 3. Navigation in rectangular and circular cage.

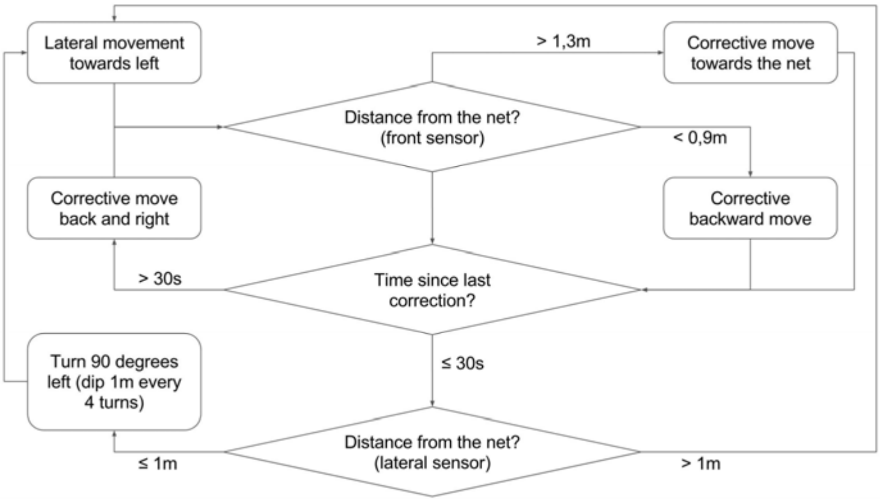


Fig. 4. The navigation procedure for the fisheries.

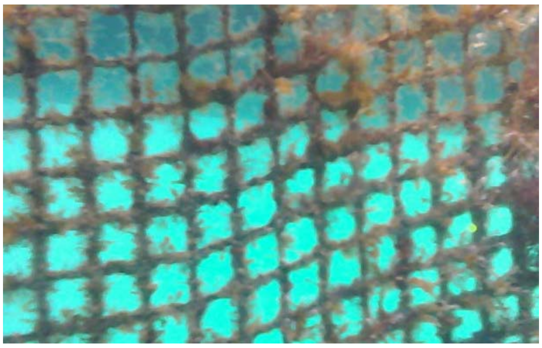


Fig. 5. Image from the ROV.

4. Conclusions

The experimental results from the initial testing phase of Kalypso demonstrated the effectiveness and potential of the robotic vehicle in the fish farming industry. The vehicle's stable design, advanced sensors, and precise movement capabilities make it a cost-effective and efficient solution for inspecting nets in fish farms. The ability to identify potential issues and prevent the spread of diseases in the fish population is critical to the success of the industry, and Kalypso's success in this area provides a promising model for future innovations in the field.

Acknowledgements

This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship, and Innovation, under the call RESEARCH – CREATE – INNOVATE. (Project code: T2EDK- 02504).

References

[1]. M. Vasileiou, N. Manos, N. Vasilopoulos, A. Douma, E. Kavallieratou, Kalypso AUV: A 3D-printed Underwater vehicle for inspection at Fisheries, *Journal of Mechanisms and Robotics*, Vol. 16, Issue 1, 2023, 011012.

[2]. B. Butler, V. Den Hertog, Theseus: A cable-laying AUV, in *Proceedings of the OCEANS Conference*, Victoria, BC, 1993, pp. I210-I213.

[3]. X. Yun, E. R. Bachmann, A. K. Suatarslan, R. B. Mcghee, An inertial navigation system for small autonomous underwater vehicles, *Advanced Robotics,* Vol. 15, Issue 5, 2001, pp. 521-532.

[4]. L. Zhao, W. Gao, The experimental study on GPS/INS/DVL integration for AUV, in *Proceedings of the Position Location and Navigation Symposium*, Monterey, California, USA, 2004, pp. 337-340.

[5]. J. Bellingham, C. Goudey, T. Consi, J. Bales, D. Atwood, J. Leonard, C. Chryssostomidis, A second generation survey AUV, in *Proceedings of the Symposium on Autonomous Underwater Vehicle Technology*, Cambridge, Massachusetts, 1994, pp. 148-155.

[6]. M. Liu, J. Hu, W. Li, Research on integrated navigation for autonomous underwater vehicle based on an improved unscented Kalman filter, *Acta Armamentarii*, Vol. 2, Issue 25, 2011.

[7]. E. Potokar, K. Norman, J. Mangelson, Invariant extended Kalman filtering for underwater navigation, *IEEE Robotics and Automation Letters*, Vol. 6, Issue 3, 2021, pp. 5792-5799.

[8]. G. T. Donovan, Position error correction for an autonomous under water vehicle inertial navigation system (INS) using a particle filter, *IEEE J. Oceanic Eng*., Vol. 37, Issue 3, 2012, pp. 431-445.

[9]. N. Shaukat, A. Ali, M. Moinuddin, P. Otero, Underwater vehicle localization by hybridization of indirect Kalman filter and neural network, in *Proceedings of the 7th International Conference on Mechatronics and Robotics Engineering (ICMRE'21)*, 2021, pp. 111-115.

[10]. S. B. Williams, P. Newman, G. Dissanayake, H. Durrant-Whyte, Autonomous underwater simultaneous localisation and map building, in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'2000)*, San Francisco, CA, USA, pp. 1793-1798.

(008)

Explainable Similarity Measures with Categorical Attributes

M. V. Aarset
Norwegian University of Science and Technology – NTNU,
Department of Ocean Operations and Civil Engineering, Larsgårdsvegen 2, 6009 Ålesund, Norway
TERP as, Research department, Torvvikvegen 48, 4250 Kopervik, Norway
Tel.: + 47 45402225
E-mails: magne.aarset@ntnu.no, maa@terp.no

Summary: We have entered an era with an increasingly demand for safe and effective monitoring and control of complex systems. In this line of development, artificial intelligence (AI) has proved to be a powerful tool to produce decision support to decision makers. As humans, sometimes correctly, are reticent to apply decision support they don't understand a demand for explainable artificial intelligence (XAI) has emerged. To further enable human users to understand, trust and manage an artificial intelligent partner, we are introducing a new similarity measure in cluster analysis to treat asymmetric categorical attributes. This measure is further developed to adapt to categorical attributes that naturally may be organized in a hierarchical structure. These measures naturally generalise the traditional similarity measure of "simple matching".

Keywords: Explainable artificial intelligence (XAI), Machine learning, Cluster analysis, Asymmetric categorical attributes, Hierarchical categorical attributes.

1. Introduction

Cluster analysis, a branch of unsupervised learning within the field of machine learning, is the art of finding groups in data. Let O = $\{o_1, o_2, ..., o_N\}$ be a set of objects (e.g., persons or mechanical failures). A partition divides O into subsets (clusters) O = $\{O_1, O_2, ..., O_k\}$ that satisfy

$$O_i \cap O_j = \Phi \ (\forall\ i \neq j), \tag{1}$$

and

$$O_1 \cup O_2 \cup ... \cup O_k = O \tag{2}$$

The objective is to find clusters in such a way that objects in the same cluster are similar to each other, while objects in different clusters are as dissimilar as possible. Such a partition can be presented as decision support to a decision maker as it can suggest that objects in the same cluster can be treated similarly.

Suppose there are N objects to be clustered by means of J attributes (e.g., personality and age, or cause and effect of mechanical failures). The data can be arranged in an (NxJ) matrix, where the rows corresponds to the objects and the columns to the different attributes.

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1J} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{NJ} \end{bmatrix} \tag{3}$$

1. In cluster analysis human intervention is (most often) called for in the following four steps;
2. Selection of attributes to characterize the objects;
3. Selection of metrics to quantify the different attributes;
4. Selection of (dis-)similarity to measure the distance between objects, between objects and clusters, and between clusters;
5. Selection of algorithm to create the clusters.

The actual choices made in each of these steps will influence the final classification, and thereby the validity of the decision support system. In some applied analyses, surprisingly little attention has been placed on steps 1-3, though. This lack of attention should come as no surprise with all the heuristics coming into action regarding human attention and human perception [1]. Another reason may be a focus on artificial intelligence (AI) instead of explainable artificial intelligence (XAI). Explainable artificial intelligence produces details or reasons to make its functioning clear and easy to understand to a certain audience [2]. Here, we will focus on step 3.

2. Similarities for Categorical Attributes

2.1. Traditional Approach

As the similarity is a measure that quantifies the degree of natural association between objects, this is what we will utilise to identify a possible underlying unknown clustering structure in the data.

Suppose there are N objects to be clustered by means of J categorical attributes $A_1, \ldots, A_J$. Assume $A_1, \ldots, A_J$ to be a finite number of discrete stochastic variables where each takes on some categorical value from the finite sets $A_j = \{0, \ldots, M_j-1\}$ $(j = 1, \ldots, J)$.

The similarity measure between two objects X and Y, both characterized by $A_1, \ldots, A_J$, is traditionally defined by the relative number of matches of the corresponding attribute values of the two objects [11],

$$s_T(X,Y) \;=\; \sum_{j\,=\,1}^{J} \omega_j s_j(x_j,y_j), \tag{4}$$

where $\omega_j \;=\; \frac{1}{J}$, and $s_j\big(x_j,y_j\big) \;=\; \begin{cases} 1, & x_j = y_j \\ 0, & x_j \neq y_j \end{cases}$.

$s_j\big(x_j,y_j\big)$ may be understood as a *partial similarity* with respect to attribute A_j.

Several alternative similarity functions have been suggested focusing on different characteristics of the attributes (e.g., [3, 5, 7, 8], and [14]). Brossier [4] are discussing partial (dis-)similarities. Zeinedine et al. [15] are among other techniques applying ensemble methods, which give good predictions that may be harder to explain. Of special interest here are the works by Goodall [9] and by Spärck Jones [12], who both considered relative frequencies of the categories. Sulc and Rezankova [13] gives an excellent overview of what has been presented in the literature of cluster analysis regarding (dis-)similarities based on nominal characteristics.

2.2. A Similarity Measure Reflecting Occurrence

This section studies a situation where the possible outcomes of some of the attributes characterizing the objects are not equally important (i.e., some of the attributes are asymmetric). An example of an asymmetric binary attribute is "presence of intelligent life" on planets somewhere in space. While it can be said that two planets with intelligent life have something in common, it is not as clear if the same should be said of two planets without intelligent life.

We seek to improve the explain ability of the artificial intelligence providing decision support by claiming that if two objects are equal with respect to some asymmetric attribute, the more unlikely it is to find such values the more we prefer to assign the objects to the same cluster. Such a cluster may typically need special attention (e.g., persons constituting a terrorist cell, or failure modes that can create disasters). This generalises naturally from binary to nominal categorical attributes (i.e., attributes taking values with no natural ordering of the different outcomes).

Basically, there are two different roads to follow to handle objects characterized by asymmetric attributes. The traditional way is to give extra weight to the attribute which has an unlikely outcome when defining the similarity [8]. When the actual attribute is binary, this may be a better procedure than when the attribute is nominal. Although a particular outcome of a nominal attribute may be extremely rare and important, it may not be beneficial to place additional weight on the more common outcomes of the same attribute. Therefore, we will focus on the partial similarities of the different outcomes of the attribute.

Let the least probable outcome of a categorical attribute A_j be coded $M_j – 1$, the second least probable $M_j–2$, and so on. We define the partial similarity between two objects X and Y to be 1 if $x_j = y_j = M_j–1$ and 0 if $x_j \neq y_j$. Reflecting that the less probable an outcome is the more similar are the objects, we would like to expand the possible values of this partial similarity by suggesting the following similarity measure.

The similarity between two objects X and Y characterized by J categorical attributes is defined by

$$s_M(X,Y) \;=\; \sum_{j\,=\,1}^{J} \omega_j s_j(x_j,y_j), \tag{5}$$

where $\omega_j \;=\; \frac{1}{J}$ and $s_j\big(x_j,y_j\big)$, the partial similarity with respect to attribute A_j (j = 1, 2,, …, J), is defined as the probability of an object to be different from the respective outcomes, standardised to the 0–1 scale. That is,

$$\begin{aligned} s_j\big(x_j,y_j\big) \;&=\; \frac{1-p_{M_j-1}}{1-p_{M_j-1}} = 1 \text{ if } x_j = y_j = M_j–1, \\ s_j\big(x_j,y_j\big) \;&=\; \frac{1-p_{M_j-2}}{1-p_{M_j-1}} \text{ if } x_j = y_j = M_j–2, \\ &\dots \\ s_j\big(x_j,y_j\big) \;&=\; \frac{1-p_0}{1-p_{M_j-1}} \text{ if } x_j = y_j = 0, \\ s_j\big(x_j,y_j\big) \;&=\; 0 \text{ otherwise,} \end{aligned} \tag{6}$$

where $p_{Mj-m} = P\{A_j = M_j\text{-}m\}$, m = 1, …, M_j.

Instead of just counting the number of attributes where two objects are equal, we add their weighted partial similarities. By adding partial similarities to create a similarity between objects, categorical attributes may directly be mixed with (standardised) non-categorical attributes. Observe also, that under the assumption that all outcomes of a categorical attribute are equally probable (i.e., the attribute is symmetric), this similarity measure will be equal to the traditional similarity measure of “simple matching”.

This measure reflects that the less probable it is for two objects to be equal with respect to a specific variable – the more equal they are (if they actually are equal). Let for example two objects be characterised by one nominal attribute only (A), with only three possible outcomes a, b, and c, and let P(A = a) = 0.01, P(A = b) = 0.20 and P(A = c) = 0.79. Then, the (partial) similarities of two objects, X and Y, will become as presented in Table 1.

Table 1. Partial similarities based on probability.

Probability	Outcome	Partial similarity	
P(A = a) = = 0.01	x = y = a	s(x, y)	= 1
P(A = b) = = 0.20	x = y = b	s(x, y) = = (1-0.2)/(1-0.01)	= 0.81
P(A = c) = = 0.79	x = y = c	s(x, y) = = (1-0.79)/(1-0.01)	= 0.21
	x ≠ y	s(x, y)	= 0

Lingoes [10] has suggested a somewhat similar measure for a simpler situation without generalising to attributes organised in a hierarchical structure as we will in Section 2.3.

2.3. A Similarity Measure Reflecting Hierarchy

In for example a survey among passengers or while clustering failure modes in a risk analysis, it is common that the attributes characterizing the objects naturally split the objects into two or more groups (see Table 2). It is also common that within each group there may be several subgroups, where each subgroup again might be divided into several items, and so on. This hierarchical structure may hold important information to a decision maker.

It is natural to claim that two objects are more similar if they are in the same group than in different groups (e.g., A: Business traveler or B: Pleasure traveler). Within each group, we would also claim that objects in the same subgroups are more similar than objects in different subgroups. And so on. Furthermore, some of the alternatives may be common and some rare (i.e., the attribute can be asymmetric).

Table 2. Hierarchical structure of a nominal attribute.

Group level (3)	**A: Business traveler**		**B: Pleasure traveler**		
Sub-group level (2)	**A1: Education**	**A2: Contracts**	**B1: Holiday**	**B2: Visiting**	**B3: Adventure**
Item	A11: Course	A21: Sales	B11: Short stay	B21: Friends	B31: Sports
level (1)	A12: Conference	A22: Purchasing	B12: Longer stay	B22: Relatives	B32: Culture
	A13: Congress			B23: Hospital	
				B24: Ex-wife	

Observe that traditionally the term hierarchical clustering methods reflects the hierarchical structure of the clusters, while we in this subchapter are discussing the hierarchical structure of the attributes.

The similarity between two objects characterized by categorical attributes, each organized in a hierarchical structure, is defined in a similar way as suggested in equation (6). This new similarity measure is taking into account the probabilities of an attribute taking values in the different groups, subgroups, and items.

The similarity between two objects X and Y characterised by J categorical attributes all organised in a hierarchical structure is defined by

$$s_H(X,Y) \;=\; \sum_{j\,=\,1}^{J} \omega_j s_j(x_j, y_j), \tag{7}$$

where $\omega_j \;=\; \frac{1}{J}$ and s_j, the partial similarity with respect to attribute A_j (j = 1, 2,, …, J), is defined as the probability of each of the objects to be different from the respective outcome, standardised to the 0 – 1 scale.

That is:

1. If two objects X and Y have equal values at item level (level 1), the partial similarity between the two objects is defined by

$$\begin{gathered} s_j(x_j, y_j) \;=\; \frac{1-p_{Mj-1}}{1-p_{Mj-1}} = 1 \text{ if } x_j = y_j = M_j\text{–}1, \\ s_j(x_j, y_j) \;=\; \frac{1-p_{Mj-2}}{1-p_{Mj-1}} \text{ if } x_j = y_j = M_j\text{–}2, \\ \dots \\ s_j(x_j, y_j) \;=\; \frac{1-p_0}{1-p_{Mj-1}} \text{ if } x_j = y_j = 0, \end{gathered} \tag{8}$$

where $p_{Mj\text{-}m} = P\{A_j = M_j\text{-}m\}$, m = 1, …, M_j.

2. If two objects have non-equal values at item level, but belong to the same subgroup sg (level 2), the partial similarity between the two objects is defined by

$$s_j(x_j, y_j) \;=\; \frac{1-p_{sg}}{1-p_{Mj-1}}, \tag{9}$$

when $x_j \neq y_j$, $x_j \in$ sg and $y_j \in$ sg, where $p_{sg} \;=\; \sum_{i:x_i \in sg} p_i$.

3. If two objects have non-equal values at subgroup level, but belong to the same group g (level 3), the partial similarity between the two objects is defined by

$$s_j(x_j, y_j) \;=\; \frac{1-p_g}{1-p_{Mj-1}}, \tag{10}$$

when $x_j \in sg_k$, $y_j \in sg_l$, $sg_k \neq sg_l$, $x_j \in$ g, and $y_j \in$ g, where $p_g \;=\; \sum_{i:x_i \in g} p_i$.

4. If the two objects have non-equal values at group level the partial similarity between the two objects is defined by

$$s_j(x_j, y_j) \;=\; 0 \tag{11}$$

The number of levels in the hierarchical structure may of course be expanded.

This definition of similarity is introducing a more nuanced structure than the traditional number of matches (with some kind of weighting) and may also improve explain ability. By adding partial similarities to create a similarity between objects like introduced here, also categorical attributes organized in a hierarchical manner may directly be mixed with (standardised) non-categorical attributes. In Table 3, partial similarities are calculated based on some suggested probabilities as an example.

3. Estimation

A naïve estimator for $p_{Mj\text{-}m}$ is the relative frequency

$$\hat{p}_{M_j-m} \;=\; \frac{\sum_{n\,=\,1}^{N} \delta_{1n}}{\sum_{n\,=\,1}^{N} \delta_{2n}}, \tag{12}$$

where:

- $\delta_{1n} = 1$ if $x_j = M_j – m$, $m = 1, …, M_j$, and 0 otherwise;
- $\delta_{2n} = 1$ if the value for object n is non-missing, and 0 otherwise;
- N is the number of objects to be clustered.

Alternatively, some prior knowledge about the parameter p_{Mj-m} may be available, for example that

$$p_{Mj-m} \sim \text{Beta}(\alpha, \beta) \qquad (13)$$

Table 3. Probabilities and respective similarities of the hierarchical nominal attribute in Table 1.

Outcome	**Probability (level 1)**	**Probability (level 2)**	**Probability (level 3)**	**$s_j(x_j,y_j)$ Equal at level 1**	**$s_j(x_j,y_j)$ Equal just at level 2**	**$s_j(x_j,y_j)$ Equal only at level 3**	**$s_j(x_j,y_j)$ Not equal at level 3**
A11	0.05	0.2001	0.75	0.95	0.80	0.25	0
A12	0.0001			1.00			
A13	0.15			0.985			
A21	0.35	0.55		0.65	0.45		
A22	0.20			0.80			
B11	0.08	0.09	0.25	0.92	0.91	0.75	
B12	0.01			0.99			
B21	0.02	0.050000001		0.98	0.95		
B22	0.01			0.99			
B23	0.02			0.98			
B24	0.00000001			1.00			
B31	0.1	0.1098999		0.90	0.89		
B32	0.0098999			0.99			

After observing the sufficient stochastic variables

$$\sum_{n=1}^{N} \delta_{1n} \text{ and } \sum_{n=1}^{N} \delta_{2n} \qquad (14)$$

The posterior distribution in a Bayesian framework will be [6]

$$p_{Mj-m} \mid \sum_{n=1}^{N} \delta_{1n}, \sum_{n=1}^{N} \delta_{2n} \sim \\ \sim \text{Beta}(\alpha+\sum_{n=1}^{N} \delta_{1n}, \beta+\sum_{n=1}^{N} \delta_{2n}-\sum_{n=1}^{N} \delta_{1n}), \qquad (15)$$

and the posterior expectation

$$E(p_{Mj-m} \mid \sum_{n=1}^{N} \delta_{1n}, \sum_{n=1}^{N} \delta_{2n}) = \\ = (\alpha+\sum_{n=1}^{N} \delta_{1n})/(\alpha+\beta+\sum_{n=1}^{N} \delta_{2n}) \qquad (16)$$

With a "non-informative" rectangular prior Beta($\alpha = 1$, $\beta = 1$), a reasonable estimate would be the posterior expectation

$$E(p_{Mj-m} \mid \sum_{n=1}^{N} \delta_{1n}, \sum_{n=1}^{N} \delta_{2n}) = \\ = (1+\sum_{n=1}^{N} \delta_{1n}) / (2+\sum_{n=1}^{N} \delta_{2n}) \qquad (17)$$

4. Conclusions

The introduced similarity measures are both easily explainable and are reflecting the occurrence of the different outcomes of the attributes. This may increase the value of the decision support.

Acknowledgements

This research was funded in part by The Research Council of Norway [310123]. A CC BY or equivalent license is applied to any Author Accepted Manuscript (AAM) version arising from this submission, in accordance with the grant's open access conditions.

References

[1]. M. V. Aarset, R. Glomseth, P. C. Juvkam, Situational awareness during a crisis in Norway: Seeing the forest, but not the trees, in Enhancing Police Service Delivery. Global Perspective and Contemporary Policy Implications (J. F. Albrecht, G. den Heyer (Eds.), *Springer International Publishing,* Cham, 2021, pp. 29-54.

[2]. A. B. Arrieta, N. Diaz-Rodriguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, F. Herrera, Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI, *Information Fusion,* Vol. 58, 2020, pp. 82-115.

[3]. H. Bustince, A. Jurio, A. Pradera, R. Mesiar, G. Beliakov, Generalization of the weighted voting method using penalty functions constructed via faithful restricted dissimilarity functions, *European Journal of Operational Research*, Vol. 225, 2013, pp. 472-478.

[4]. G. Brossier, Partial dissimilarities with application to clustering, *Journal of Classification*, Vol. 11, 1994, pp. 37-58.

[5]. F. Cao, J. Liang, D. Li, L. Bai, C. Dang, A dissimilarity measure for the k-Modes clustering algorithm, *Knowledge-Based Systems*, Vol. 26, 2012, pp. 120-127.

[6]. M. H. DeGroot, Optimal Statistical Decisions, *McGraw-Hill,* New York, 1970.

[7]. K. El Hindi, B. A. Shawar, R. Aljulaidan, H. Al Salam, Improved distance functions for instance-based text classification, *Computational Intelligence and Neuroscience*, Vol. 2020, 2020, 4717984.

[8]. J. Grim, EM cluster analysis for categorical data, in in *Proceedings of the Structural, Syntactic, and Statistical Pattern Recognition Conference (SSPR/SPR'06)*, 2006, pp. 640-648.

[9]. D. W. Goodall, A new similarity index based on probability, *Biometrics*, Vol. 22, Issue 4, 1966, pp. 882-907.

[10]. J. C. Lingoes, The multivariate analysis of qualitative data, *Multivariate Behavioral Research*, Vol. 3, Issue 1, 1968, pp. 61-94.

[11]. M. K. Ng, M. J. Li, J. Z. Huang, Z. He, On the impact of dissimilarity measure in k-modes clustering algorithm, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, Issue 3, 2007, pp. 503-507.

[12]. K. Spärck Jones, A statistical interpretation of term specificity and its application in retrieval, *Journal of Documentation*, Vol. 28, 1972, pp. 11-21.

[13]. Z. Sulc, H. Rezankova, Comparison of similarity measures for categorical data in hierarchical clustering, *Journal of Classification*, Vol. 36, 2019, pp. 58-72.

[14]. D. R. Wilson, T. R. Martinez, Improved heterogeneous distance functions, *Journal of Artificial Intelligence Research*, Vol. 6, 1997, pp. 1-34.

[15]. H. Zeineddine, U. Braendle, A. Farah, Enhancing prediction of student success: Automated machine learning approach, *Computers & Electrical Engineering*, Vol. 89, Issue 4, 2021, 106903.

(010)

Abnormal Behavior Recognition based on Multiple Instance Learning

Zhanhe Yu and Yuanyao Lu
North China University of Technology, School of Information Science and Technology,
100144 Beijing, P.R. China
E-mail: luyy@ncut.edu.cn

Abstract: Intelligent monitoring system plays an important role in social security and criminal case detection. However, most of the traditional intelligent surveillance systems cannot automatically identify abnormal behaviors and automatically record abnormal information, which leads to a long time for the relevant personnel to retrieve the abnormal behaviors after the occurrence of abnormal situations. In this paper, the multiple instance learning model is selected to identify 13 types of abnormal behaviors according to the characteristics of abnormal behavior data. The model uses the idea of multiple instance learning to use normal video and abnormal video as bags, and video clips as instances in multiple instance learning, and then predict abnormal video clips. To obtain better training results, we add the dense block, soft thresholding algorithm, and Bi-LSTM to the feature extraction network, and add sparsity constraints and smoothness constraints to the model to more accurately locate the time of anomaly occurrence. We also design the anomalous behavior recognition system based on the model. Through experiments, our model achieves 94.19% recognition accuracy on the UCF-Crime dataset. The comparison experiments can show that our model is competitive in terms of recognition accuracy.

Keywords: Abnormal behavior recognition, Multiple instance learning, Residual network, Dense block, Soft thresholding.

1. Introduction

In today's social security situation has become complex and diversified, and people also gradually began to pay attention to their own life and property security. However, for the traditional surveillance cameras currently in use on the market, because of the limited ability of the system to discriminate independently, these products usually simply play and save the incoming images from the camera, and do not analyze the video. This leads to an analysis process that still relies on manual surveillance to identify anomalies in the video footage. However, in a video surveillance room, the number of displays is large and the number of surveillance personnel is limited, so that it is difficult to catch the useful information in the video with human eyes alone. More importantly, when people stare at multi-screen surveillance images for a long time, human attention will be distracted and judgment and concentration will be significantly reduced. According to a study, when people stare at a surveillance screen for more than 20 minutes, 95 % of the information is discarded intentionally or unintentionally. On the other hand, when a violent incident occurs, the surveillance personnel is not the first to find the abnormality in the video screen, which leads to both failure to stop the violence in time, and the difficulty of tracing the incident after it happens, resulting in unnecessary property losses. Therefore, the development of an efficient, accurate, real-time, highly robust behavior recognition system has important economic value and social significance.

In recent years, behavior recognition based on deep learning has developed rapidly, and current deep learning methods [1, 2] have made many improvements on the basis of traditional artificial neural networks. Among them, the convolutional neural network has become the mainstream method of deep learning, and behavior recognition is also mostly based on continuous improvement and optimization. Ji et al. proposed the 3D CNN [3], where they extended the 2D convolution to 3D. This network performs spatial convolution while also capturing feature information in the temporal dimension as a way to establish connections between different video frames of a video stream. The network has a simple structure, and feature extraction is performed in the horizontal and vertical directions for the 2D spatial domain and optical domain respectively. This network performs very well on the KTH dataset, which also opens up new solution ideas for subsequent behavior recognition. Simonyan et al. proposed a Two-Stream network structure [4]. They divided the static single-frame video and the stacked optical stream into two independent branches for training and feature extraction, which resulted in a network structure with a significant effect on behavior recognition. Qiu et al. proposed P3D [5]. They found that although 3D convolution can capture behavioral features well, the computational effort is much larger than 2D convolution. Therefore, they proposed to replace the original 3D convolution with a spatio-temporal asymmetric convolution kernel and added residuals to the 3D convolution, which enables the model to extract spatio-temporal features efficiently. Tran et al. proposed the R(2+1)D model [6]. They used 2D convolution and 1D convolution instead of 3D convolution, which has the advantage of introducing more nonlinear activation layers and making the model more representable.

Although behavior recognition has achieved good results after decades of development. However, developing efficient and accurate algorithms is still very challenging due to the variability of abnormal behaviors and the complexity of the environment. Developing efficient and robust behavioral recognition algorithms with high accuracy is also the goal of researchers.

In this paper, we mainly design the anomalous behavior recognition system based on deep learning with the premise of real scene use and the idea of multiple instance learning. The system does not need to modify the traditional surveillance video architecture, but only needs to connect the video stream to the behavior recognition analysis system to be able to use it normally, which is easy and fast to deploy. At the same time, we use accurate, efficient, and highly robust algorithms instead of manual monitoring of video walls to make up for the shortcomings of traditional surveillance video that is not notified in the first place when anomalies occur and is difficult to find afterward, which greatly saves labor costs. The main work of this paper is as follows:

(1) To obtain better training results, we analyzed the characteristics of anomalous behavior. We add sparsity restriction and smoothness restriction to the multiple instance learning algorithm according to the characteristics of anomalous behavior, so that the model can locate the time when the anomalous behavior occurs and ends more accurately.

(2) We choose the residual network as the backbone network of the feature extraction network, and optimize and improve it. To better extract the spatio-temporal features in the video, we use 3D convolution instead of 2D convolution, and use the dense block and soft thresholding to reduce the number of parameters of the model and improve the computational efficiency of the network.

(3) We tested the abnormal behavior recognition system based on multiple instance learning and verified that the system can perform the expected task well. The system can identify the time period when the abnormal behavior occurs in the video based on the content of the input video.

2. Proposed Method

2.1. Multiple Instance Learning

Since the determination of anomalous behavior is often subjective, and the nature of the behavior varies with the actual situation and may be ambiguous depending on our subjective view, it is difficult to define. Therefore, it is not easy to distinguish anomalous behaviors directly by using positive and negative labels, and it is not possible to distinguish them precisely in time. Therefore, we constructed an abnormal behavior recognition model based on multiple instance learning [7, 8], and the structure of the model is shown in Fig. 1. In this model, we consider anomalous behavior recognition as a regression problem. Before training, the surveillance videos in the dataset are divided into video clips of the same length according to the number of frames. The spatio-temporal features of each video clip are extracted by the feature extraction network, and then all video clips in each video are rated separately. Finally, the video clip with the highest rating in each video is selected. The result of the training is that all the abnormal video clips are expected to have higher ratings than the normal video clips among all the selected video clips.

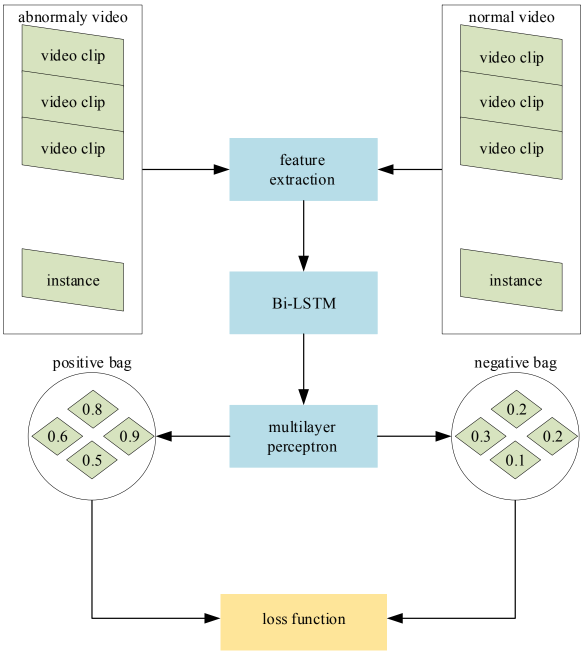


Fig. 1. Multiple instance learning model.

The most intuitive way to make the score of anomalous video clips exceed that of normal video clips is to use a ranking loss function. It can motivate the anomalous video clips to achieve higher ratings, as shown in Equation (1):

$$f(V_p) > f(V_n), \tag{1}$$

where V_p and V_n are denoted as abnormal video clips and normal video clips respectively, and $f(V_p)$ and $f(V_n)$ denote the scores of both respectively. However, since we are not sure of the precise time point of abnormal behavior, we cannot use Equation (1) for ranking, but instead use the ranking loss function shown in Equation (2):

$$\max_{i \in B_p} f(V_p^i) > \max_{i \in B_n} f(V_n^i), \tag{2}$$

where max denotes the range of all video clips in a given video. Instead of selecting all the video clips in a video to sort, we sort only the two video clips with the highest abnormal scores in the abnormal video and the normal video. The clip with the highest anomaly score in the anomaly video, which is the most likely to contain truly anomalous behavior, is a positive instance. The clip with the highest anomaly score in the normal video is the clip that looks most like anomalous behavior, but is actually a negative instance. This negative instance can also be seen as a confusing instance, which may be misrepresented during anomalous behavior identification. By using Equation (2), we want to separate positive and negative instances as much as possible in anomalous behavior recognition, so we use Equation (3) as the loss function of the model.

$$l(B_p,B_n)=\max\left(0,1-\max_{i\in B_p} f\left(V_p^i\right)+\max_{i\in B_n} f\left(V_n^i\right)\right) \tag{3}$$

However, the disadvantage of the loss function mentioned above is that it ignores the temporal nature of the anomalous behavior. In practice, anomalous behaviors tend to appear only in a short period of time. Under this premise, we should consider that the distribution of high-rated instances in anomalous videos is relatively sparse, indicating that only a few video clips contain anomalous behaviors. Secondly, since the videos are time-continuous, the anomalous scores of video clips should also be smoothly varying. Therefore, we reduce the anomalous behavior score difference of adjacent video clips in the same video to make the anomalous score of adjacent video clips change more smoothly. Equation (4) is the ranking loss function with the addition of sparsity and smoothness constraints.

$$\begin{aligned} l(B_p,B_n)=&\max\left(0,1-\max_{i\in B_p} f\left(V_p^i\right)+\max_{i\in B_n} f\left(V_n^i\right)\right)\\ &+\lambda_1 \overbrace{\sum_{i}^{m-1}\left(f\left(V_p^i\right)-f\left(V_p^{i+1}\right)\right)^2}^{\text{①}}+\lambda_2 \overbrace{\sum_{i}^{m} f\left(V_p^i\right)}^{\text{②}} \end{aligned} \tag{4}$$

where ① denotes the temporal smoothing term and ② denotes the sparse term. In this ranking loss function for multiple instance learning, the model parameters are back-propagated from the highest rated video clip in the positive and negative packets to optimize and iterate the model.

2.2. Feature Extraction Network

Based on previous deep learning experiments, we can conclude that the recognition accuracy of the model is related to the number of layers of the network. The more layers of the network, the better the recognition ability of the model. However, if the number of layers of the network is too many, the network may suffer from gradient disappearance. This causes the parameters of the network to stop updating and leads to a decrease in recognition accuracy. To solve this problem, we use the residual network as the backbone network of the feature extraction network. However, traditional residual networks use 2D convolution. This can only extract the spatial features in the image, but not the temporal features in the video. In order for the network to better understand the information in the video, we also need to make the network have the ability to acquire temporal features. To solve the above problem, we use 3D convolution instead of 2D convolution. In 3D convolution, we set the number of layers of the filter convolution kernel to be much smaller than the number of channels of the input data, so that the filter can slide over the length, width, and height dimensions of the input data. After the filter slides over the entire input data, the output result is still 3D. This allows the output to include the spatio-temporal characteristics of the input data.

2.3. Soft Thresholding

Soft thresholding [9] is the basis of many information noise reduction techniques, which eliminates the features with values below the threshold and adjusts the features with values above the threshold, as in Equation (5):

$$y=\begin{cases} x-\tau & x>\tau \\ 0 & -\tau\le x\le\tau \\ x+\tau & x<-\tau \end{cases} \tag{5}$$

The derivatives of soft thresholding are only 0 and 1. This property of the soft thresholding function is consistent with the ReLU activation function, so soft thresholding can also reduce the phenomenon of gradient dispersion and gradient explosion of the model in deep learning to some extent. Since the noise content of different samples in the same dataset tends to be different, it is desirable for us to allow different samples to automatically adjust the threshold for their own noise content if we want to improve the generalization ability of the model. Therefore, we design and use the network in Fig. 2 to achieve the autonomous adjustment of soft thresholds.

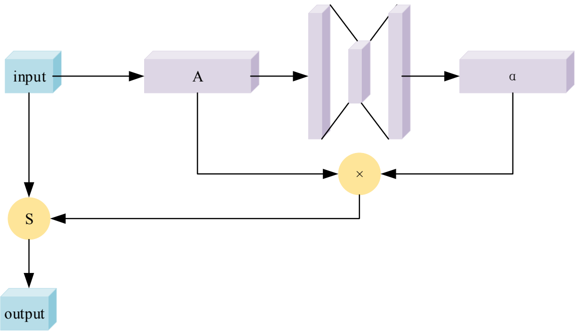


Fig. 2. Soft thresholding.

In this network, we first take the absolute values of the input information and then pool them by the global mean to derive a new set of feature values, denoted as A.Then we designed a small fully-connected network and used the Sigmoid activation function in the output layer to adjust the output values to between 0 and 1. The final threshold is given by α×A.In this way, each sample is guaranteed to have its own unique threshold value. In a sense, this algorithm can be understood as a unique attention mechanism. It can pay attention to features that are irrelevant to the current task, set them to 0 using soft thresholds, and pay attention to features that are relevant to the current task, adjust them accordingly, and keep them.

2.4. Dense Block

The principle of the dense block [10, 11] is similar to that of the residual network. The principle of the residual network is to establish short-circuit connections between the input and output features, which facilitates the backward gradient transfer during the training process, and therefore allows the training of deep convolutional neural networks. The dense block is characterized by making the features connected in the channel dimension to achieve feature reuse. This allows the network to achieve better performance than the residual module with fewer parameters and computational costs. Therefore, we incorporate the dense connection module in the residual module, as shown in Fig. 3.

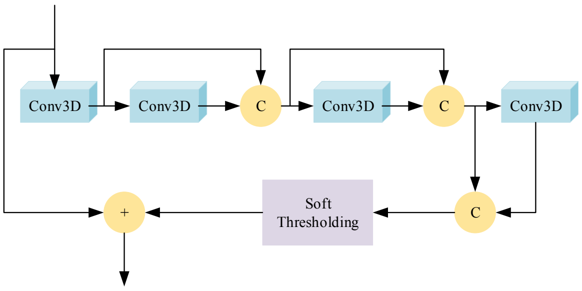


Fig. 3. Dense block.

Before the input data enters the convolutional layer, we let each group of feature maps is stitched in the channel dimension instead of summing up. Compared with the summation, stitching in the channel dimension does not introduce additional computation, which makes the network significantly more computationally efficient. In order to show the process of building the network in this paper more intuitively, we summarize the specific internal structure of the network in the following table, as shown in Table 1.

Table 1. Feature extraction network.

Layer name	Output size	Network
Conv1	56×56×32	$\begin{bmatrix}1\times1\times1\\3\times3\times3\\1\times1\times1\end{bmatrix}\times 3$
Conv2	28×28×16	$\begin{bmatrix}1\times1\times1\\3\times3\times3\\1\times1\times1\end{bmatrix}\times 4$
Conv3	14×14×8	$\begin{bmatrix}1\times1\times1\\3\times3\times3\\1\times1\times1\end{bmatrix}\times 6$
Conv4	7×7×4	$\begin{bmatrix}1\times1\times1\\3\times3\times3\\1\times1\times1\end{bmatrix}\times 3$

Since the network reuses the same feature maps many times, this makes the network prone to overfitting during training. Therefore, we use Dropout to deactivate some parameters randomly during the training process. In the forward pass process of the network, we let some neurons stop working with a certain probability, which can effectively avoid the overfitting phenomenon of the network. Since Dropout will make two neurons not necessarily work at the same time in every computation, this makes the update of weights no longer depend on the joint action between two hidden nodes with fixed meanings, thus preventing the model from working properly only when a particular feature is identified, and making the model have more robust performance.

2.5. Bi-LSTM and Multilayer Perceptron

Although LSTM [12] can analyze the content of the later information according to the content of the previous information, in some cases, the perception of the current information may be related not only to the previous information but also to the later information. Therefore, we use a Bi-LSTM to further extract temporal features from the video. The Bi-LSTM consists of two layers of LSTM combined top and bottom together. The final result is determined by the output of these two LSTM layers together.

At the end of the model, we use a multilayer perceptron [13] with three hidden layers to predict the anomalous behavior of the extracted video features and to output the anomalous behavior score. First, we input the extracted features into the multilayer perceptron sequentially, and in each of the three hidden layers, the features are L2-normalized. The first hidden layer has a total of 1024 neurons, the second hidden layer has 512 neurons, the last hidden layer has 32 neurons, and the output layer has only 1 neuron, and a 60 % Dropout is used between all the hidden layers. Also, the ReLU activation function was used in all three hidden layers to introduce nonlinearity, while in the output layer, we used the Sigmoid activation function to adjust the predicted values.

Finally, the highest rated instance in the positive and negative bags are used as input to the loss function, and the loss values are counted and passed backward to adjust the model. At the same time, the method also adjusts the learning rate in training by an Adagrad optimizer with an initial learning rate of 0.001. Finally, our trained model can output anomalous behavior scores in real-time based on the input video, and we can determine whether there is anomalous behavior in the video based on the output scores, and when the anomalous behavior starts and ends.

3. Experimental Results

After the model is trained, the first thing we need to focus on is how the model converges during the training process. If the value of hyperparameters or the model is not constructed reasonably, the model will be difficult to converge, which in turn will cause the model not to work. Therefore, we recorded and plotted the loss function curves of the model on the training and test sets at each stage of the training process. Using this curve, we can determine whether the model proposed in this paper is able to learn the characteristics of the abnormal behavior, and then judge the convergence of the algorithm. In order to see the trend of the loss values, we performed 70 iterations of the experiment and recorded the results once for each iteration. Fig. 4 shows the change curve of the loss value in different periods.

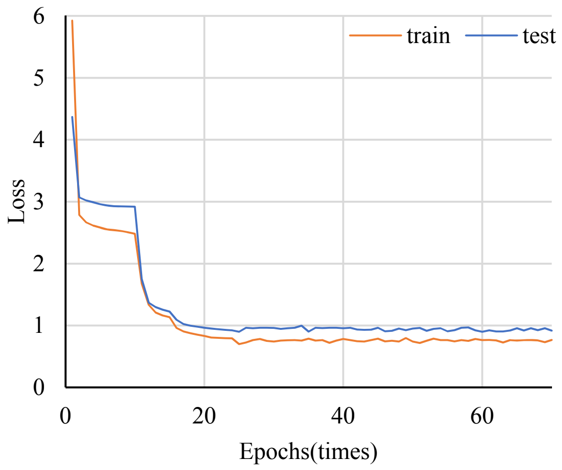


Fig. 4. Loss curve for training and testing.

In the figure, we can find that the loss value has stabilized after 25 training sessions. This indicates that the model has been trained and the loss value continues to oscillate during the next training, indicating that the performance of the model will not be improved by further training. During the training process, the loss values of both the training and test sets converge gradually with the training, indicating that our model performs well in the abnormal behavior recognition task. At this point, we can conclude that our model is workable.

After verifying that the model has converged, we conducted ablation experiments on the model in order to compare the change in performance before and after the model improvement. As seen in Table 2 and Fig. 5, for the UCF-Crime dataset used in this paper, the recognition accuracy of the model using only the residual network as the feature extraction network is 71.15 %, which is not good. In contrast, the recognition accuracy using the dense block and using soft thresholding is 86.35 % and 85.40 %. Both methods led to a certain degree of improvement in the performance of the model. The best result was achieved by using both of these improved methods, with a recognition accuracy of 94.19 %.

We chose videos of different durations to demonstrate the recognition effect of the abnormal behavior recognition system in this paper, as shown in Fig. 6. After running, the system can output the predicted scoring curve of abnormal behavior and play it simultaneously with the video, which verifies that this system can greatly reduce the working time of relevant staff and improve the working efficiency.

Table 2. Ablation experiment.

Model	Dense block	Soft thresholding	Accuracy
R3D			71.15 %
Dense-R3D	√		86.35 %
Soft-R3D		√	85.40 %
Dense-Soft-R3D	√	√	94.19 %

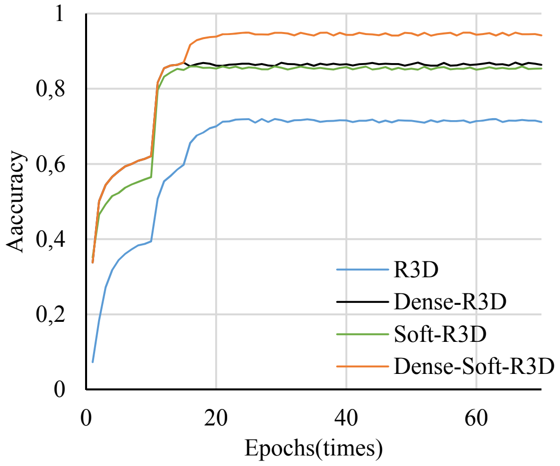


Fig. 5. Accuracy curve of ablation experiment.

Fig. 6. Abnormal behavior recognition system.

4. Conclusions

Abnormal behavior recognition algorithms have extremely rich practical application needs in both computer vision and deep learning fields. The multiple instance learning model combines the latest methods in the field of deep learning for anomalous behavior recognition, so that the algorithm proposed in this paper can improve the effectiveness of anomalous behavior recognition. In this paper, we mainly study and analyze the algorithm of multiple instance learning, and analyze and improve the loss function to lay the foundation for system development. Also, we analyze and improve the feature extraction network. We use the residual network as the backbone network of the feature extraction network based on our previous research. We also used 3D convolution to optimize the problem of poor model recognition caused by 2D convolution due to the inability to extract temporal features. Then, the number of network parameters and computational effort in surveillance video feature extraction is effectively reduced by using dense block and soft thresholding. Finally, the Bi-LSTM also solves the long-term dependency problem in video understanding.

The abnormal behavior recognition system based on multiple instance learning proposed in this paper has basically accomplished the expected functions and can realize the recognition of traffic abnormal behavior. However, there are still some shortcomings in the research of this paper. We found through experiments that the computation time occupied by the bidirectional LSTM network during the testing of our model accounts for more than 50 % of the total computation time. Therefore, we need to consider whether to replace the structure of the recurrent neural network, which is research with a high difficulty factor, which is not only a breakthrough in the technology of abnormal behavior recognition, but also a breakthrough in the research of models with temporal feature extraction in deep learning. Moreover, according to the current research results, we have been able to identify the time point when the abnormal behavior appears and disappears, but we cannot identify the type of abnormal behavior yet. If we can classify the anomalous behaviors and identify the types of current anomalous behaviors, it will make the whole system more meaningful.

References

[1]. Y. Longhui, G. Jianjun, P. Yuhai, C. Honglei, L. Jingbin, L. Shuangyin, N. Jing, G. Jianbing, Y. Shuo, Z. Hangxing, X. Yalei, W. Jianglin, W. Kang, A Recognition Method of Ewe Estrus Crawling Behavior Based on Multi-Target Detection Layer Neural Network, *Animals*, Vol. 13, Issue 3, 2023, pp. 413-423.

[2]. C. Xiangang, Z. Chiyu, W. Peng, W. Hengyang, H. Shikai, L. Hu, Unsafe Mining Behavior Identification Method Based on an Improved ST-GCN, *Sustainability*, Vol. 15, Issue 2, 2023, pp. 1041-1051.

[3]. J. Shuiwang, Y. Ming, Y. Kai. 3D convolutional neural networks for human action recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35, Issue 1, 2013, pp. 221-231.

[4]. K. Simonyan, A. Zisserman, Two-Stream Convolutional Networks for Action Recognition in Videos, *Advances in Neural Information Processing Systems*, Vol. 28, 2014, pp. 919-927.

[5]. Z. Qiu, T. Yao, T. Mei, Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks, in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 34-39.

[6]. D. Tran, H. Wang, L. Torresani, A Closer Look at Spatiotemporal Convolutions for Action Recognition, *Conference on Computer Vision and Pattern Recognition*, 2018, pp. 33-38.

[7]. Y. Qixiang, W. Fang, L. Chang, H. Qingming, J. Xiangyang, Continuation Multiple Instance Learning for Weakly and Fully Supervised Object Detection, *IEEE Transactions on Neural Networks and Learning Systems*, 2021, pp. 5452-5466.

[8]. H. Yang, S. Qu, F. Zhu, Z. Zheng, Robust objectness tracking with weighted multiple instance learning algorithm, *Neurocomputing*, Vol. 288, 2018, pp. 43-53.

[9]. C. Rui, W. Yuzhe, W. Jia, M. Mingming, N. Yi, S. Guangming. Adaptive feature denoising based deep convolutional network for single image super-resolution, *Computer Vision and Image Understanding*, Vol. 223, 2022, pp. 96-101.

[10]. F. Barzegar, S. Seydi, S. Farzaneh, M. Sharifi. Oil spil detection in the Caspian Sea with a SAR image using a densenet model, *Remote Sensing and Spatial Information Sciences*, 2023, pp. 95-100.

[11]. P. Zhu, J. Dai, H. Chang, Y. Xu, Z. Zhang. DenseNet network-based surface defect detection algorithm for strip steel, *International Journal of Frontiers in Engineering Technology*, Vol. 4, Issue 10, 2022, pp. 34-40.

[12]. D. Sarupa, S. Ranjan, A. Twum, L. Jinfeng, Input-Output Selection for LSTM-Based Reduced-Order State Estimator Design, *Mathematics*, Vol. 11, Issue 2, 2023, pp. 400-410.

[13]. O. Emenike, H. Qiong, L. Yun, N. Chukwunonso, Automatic measurement and prediction of Chinese Grown Pigs weight using multilayer perceptron neural networks, *Scientific Reports*, Vol. 13, Issue 1, 2023, pp. 2573-2583.

(014)

A Novel Approach Combining Deep Learning and Stochastic Modeling to Retrieve 3D Properties of Multiphase Flows from 2D Projections

K. Dia [1,2], **F. Lamadie** [1] **and J. Debayle** [2]
[1] CEA, DES, ISEC, DMRC, Univ Montpellier, Marcoule, France
[2] MINES Saint-Etienne, SPIN/LGF UMR CNRS 5307, 158 cours Fauriel, Saint-Etienne, France
Tel.: +33 (0)4 66 73 30 80
E-mail: kassem.dia@cea.fr

Summary: The measurement of morphological properties in complex multiphase flows is a challenging task in many areas of science and industry, and is particularly difficult in dense environments with limited optical access. This work introduces a novel approach capable of extracting three-dimensional (3D) information of spherical particle systems based only on two-dimensional (2D) projections. The approach consists of generating synthetic images of the system using a stochastic geometrical model with the same geometrical features as the system under study. They are then projected into 2D images and labeled with the appropriate 3D information. A convolutional neural network (CNN) is then trained from this set of images and applied on both synthetic and experimental images. Validation results show that this technique successfully predicts the 3D properties of the studied systems, even for high particle overlapping, with high computational efficiency.

Keywords: Multiphase flows, Convolutional neural network, 3D stochastic model, Image processing, 3D properties.

---

1. Context

Imaging-based method combined with image processing techniques are one of the most extensively used tool for monitoring complex particle systems. These systems are generally defined as mixtures of two or more substances where one is suspended in another (liquid-gas, emulsions, liquid-liquid, …). Acquiring properties of these systems, such as the particle size distribution, the mean characteristic diameters, the volume fraction of the dispersed phase, or morphological information, is essential to optimize and improve the performance of many processes involving multiphase flows.

Various image processing techniques have been employed, in particular deterministic methods that allow to separate and segment overlapping particles [1, 2]. More recently, the uprise of deep learning algorithms in the field of image processing has led researchers to rely more extensively on these techniques [3]. In this field, 3D information can be retrieved from multiple viewpoints provided that a complex setup with multiple optical access points is used. However, most experimental setups are limited to 2D projected images that do not allow access to 3D information.

In this communication, we present a new approach for extracting 3D properties such as volume fraction (ϕ) and particle size distribution (PSD) from 2D projections of arbitrary systems by combining a 3D stochastic model and a convolutional neural network (see Fig. 1) [4].

2. Main Workflow

The proposed approach relies on three main steps. First, a 3D stochastic geometrical model that consist of a hard-core model of spherical particles that take into account all particle-particle and particle-wall interactions [5], is used to generate a large dataset of 2D synthetic images. The full set of several thousand of images is then split into three parts: training, validating, and testing. Each one of these images are then labeled with their corresponding 3D geometrical properties and feed to the CNN so that it would be trained to extract 3D properties (Fig. 1b). Once trained, the network can be used directly to extract the desired 3D properties from images after just a binarization procedure (Fig. 1c).

3. Results and Discussion

This approach was first validated on digital images before being tested on real images acquired on particle suspensions of calibrated PMMA beads submerged in brine stirred in a rectangular mockup. Pictures were acquired using a telecentric backlight setup to limit perspective effects. Results, presented in Fig. 1c and Table 1 prove that all 3D quantities (volume fraction ϕ and the particle size distribution) can be measured using this approach from a single optical access, even for flows with high-dispersed phase volume fractions. The approach remains effective in the presence of high particle overlap, where traditional image-processing techniques typically fail.

It is also computationally efficient, with the analysis of thousands of images typically complete in less than 30 min. A second validation on a liquid-liquid flow in a perfectly stirred reactor confirmed the relevance of the method by allowing the simultaneous determination of the holdup and the drop size distribution from simple images.

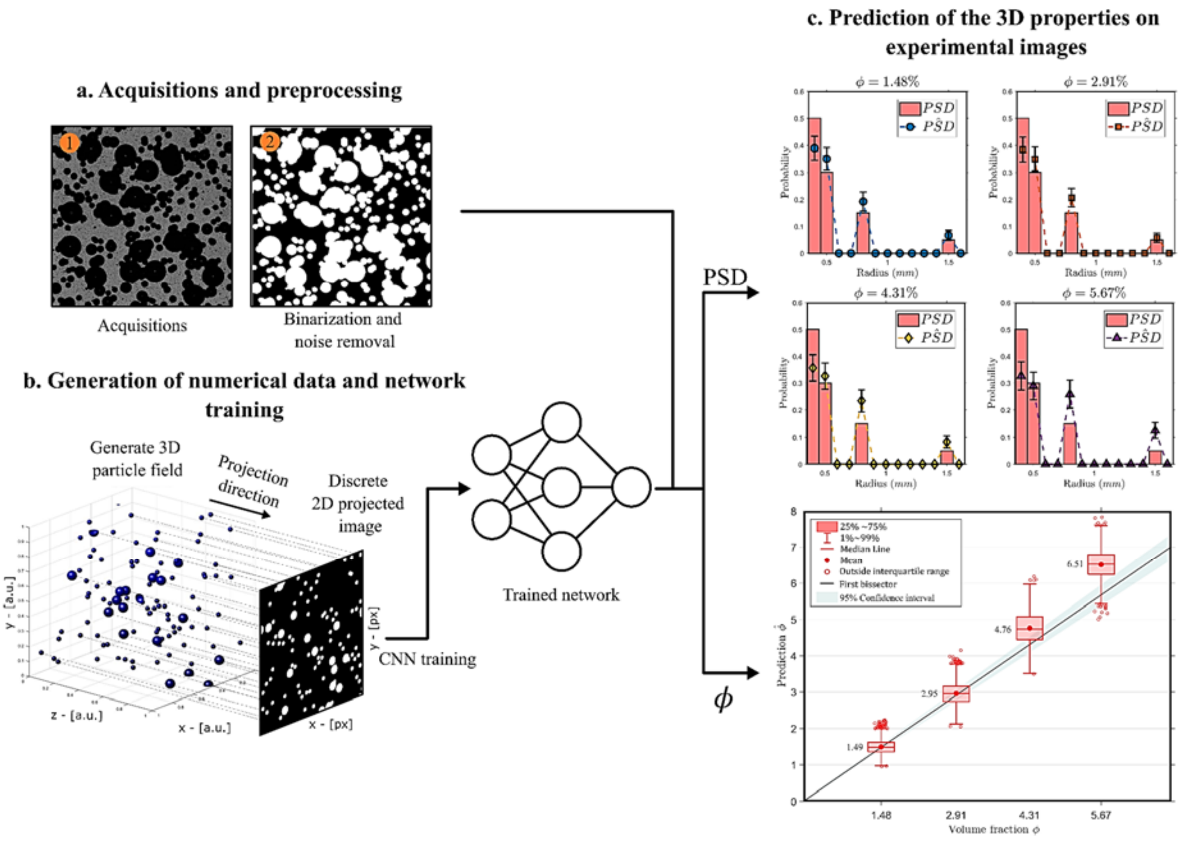


Fig. 1. Schematic outline of the main steps of the workflow. a- Acquisition and preprocessing of experimental images. b- Generation of a simulated dataset and training of the neural network. c- Results of the prediction of 3D properties ($\boldsymbol{\phi}$ and PSD) from experimental data.

Table 1. Comparison between the average predicted value of the volume fraction ($\hat{\phi}$) and the actual value (ϕ); std, standard deviation; MAE, mean absolute error.

Exp.	$\boldsymbol{\phi}$ **(%)**	$\hat{\boldsymbol{\phi}}$ **(%)(± std)**	***MAE* (± std)**
1.	1.48	1.49 ± 0.19	0.15± 0.12
2.	2.91	2.95 ± 0.31	0.25± 0.19
3.	4.31	4.76 ± 0.44	0.52 ± 0.37
4.	5.67	6.51 ± 0.39	0.84 ± 0.38

4. Conclusion and Perspectives

In this communication a new image processing technique for dense flow imaging, combining a stochastic geometrical model with convolutional neural networks to retrieve 3D properties of particle systems using only 2D projected images is introduced. Using this approach, the volume fraction of the dispersed phase and the particle size distribution are predicted directly from 2D images in a computationally efficient manner.

Based on these encouraging results, future studies will focus on improving the stochastic model by incorporating spatial inhomogeneities and introducing more complexly shaped particles that are more common in real-life applications.

References

[1]. T. Zou, T. Pan, M. Taylor, H. Stern, Recognition of overlapping elliptical objects in a binary image, *Pattern Analysis and Applications*, Vol. 24, Issue. 3, 2021, pp. 1193-1206.

[2]. M. de Langlard, H. Al-Saddik, S. Charton, J. Debayle, F. Lamadie, An efficiency improved recognition algorithm for highly overlapping ellipses: Application to dense bubbly flows, *Pattern Recognition Letter*, Vol. 101, 2018, pp. 88-95.

[3]. T. Haas, C. Schubert, M. Eickhoff, H. Pfeifer, BubCNN: Bubble detection using Faster RCNN and shape regression network, *Chemical Engineering Science*, Vol. 216, 2020, 115467.

[4]. K. Dia, F. Lamadie, J. Debayle, Using deep learning to retrieve 3D geometrical characteristics of a particle field from 2D projected images: Application to multiphase flows, in *Proceedings of the* 12^{th} *International Conference on Pattern Recognition Systems (ICPRS'22)*, Saint Etienne, France, Jun. 2022, pp. 1-7.

[5]. M. de Langlard, F. Lamadie, S. Charton, J. Debayle, A 3D stochastic model for geometrical characterization of particles in two-phase flow applications, *Image Analysis & Stereology*, Vol. 37, Issue 3, 2018, pp. 233-247.

(015)

A Convolutional Neural Network for Recognition of Coffee Leaf Disease

Nameer Baht and Enrique Domínguez
Dept. of Computer Science University of Malaga, Malaga, Spain
E-mails: nameer.m.b.88@gmail.com, enriqued@lcc.uma.es

Summary: Early detection of plant diseases is one of the most important challenges to limit the spread of diseases and reduce farm losses. The coffee plant is considered one of the most important plants in several economic aspects. also, the high price and is one of the most used daily drinks. Convolutional neural networks are a great technology and one of the most efficient and fast detection algorithms for classifying and detecting objects compared to classical methods. The images of the coffee leaves were analyzed, based on the datasets JMuBEN and JMuBEN1, which contain five classes of the most important and dangerous ones that affect the Increase production of coffee. The model CNN with Maxpool 2D &Batch Normalization achieved a result of an accuracy of 99 % to diagnose diseases affecting coffee.

Keywords: CNN, Deep learning, Arabica coffee, Image dataset, Disease diagnosis.

1. Introduction

The field of agriculture has been an important sector for AI researchers in recent years. The goal is to reduce losses and increase the production of plants and fruits with high quality and quantity [1]. The coffee plant is of economic importance to many countries that are famous for growing coffee. The traditional methods of detecting coffee diseases are observation with the naked eye, which is expensive, time-consuming and requires great expertise. Therefore, it is important to identify diseases automatically without the need for experts. The most dangerous diseases that affect coffee diseases are four Cerscospora, Phoma, Rust, and Miner and have many causes, including weather and climate is not suitable. The area of Artificial Intelligence research has shown significant growth in the few years concerning the applications of Machine Learning, which led to the emergence of a new category of models called deep Learning [2]. Deep Learning methods have shown state-of-the-art performance when compared to traditional machine Learning approaches in different domains. amid deep learning methods, convolutional neural networks (CNNs) have shown outstanding performance in image classification and object detection and tasks [3]. CNN automatically learns appropriate features from the training dataset while traditional methods rely on handcrafted features, i.e. features are calculated based on prior knowledge of the problem. In addition, the segmentation step is intrinsic to convolutional CNN filters, which further simplifies their use. CNNs are powerful tools that deal well with the problem of plant diseases. However, there are still many challenges associated with automatic diagnosis of plant diseases.

2. Materials and Methods

2.1. Data Materials

The leaves of Arabica coffee are called JMuBEN and JMuBEN1, and this data was collected in the country of Kenya by experts in this field under natural conditions and using a digital camera. The two volumes contain (58555) images 128×128 pixels size across five classes (Phoma, Cescospora, Rust, Healthy, Miner,) [4]. Fig. 1 show the number of images for every class.

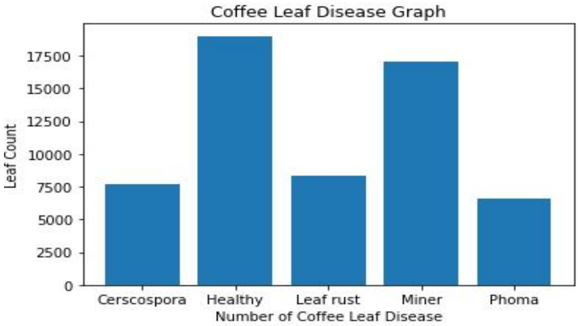


Fig. 1. Diseases of the leaves with the number of images for each disease.

2.2. Convolution Neural Network

We proposed a CNN structure with different layers and different numbers of neurons for data filtration to increase the accuracy. The network contains only three convolutional layers, which are followed by a non-linearity ReLU filter and a max-pooling layer. To complement the network performance optimization, the structure that Batch normalization is applied after the max-pooling layer. Therefore, we chose to apply this normalization to each output of the max-pooling layers. At this output, we applied a dropout of 25 %.

3. Results

Using MaxPooling 2D with batch normalization, we are dividing the dataset into train and test sets, with the application utilizing 80 % of the images for training and 20 % for testing. The results of the proposed CNN model using MaxPooling2D with batch normalization achieved an accuracy of 99.8 % and have been analyzed and compared using standard metrics as accuracy, precision, recall, F-score, confusion matrix, and ROC graph. Table 1 shows the preliminary results of these standard metrics.

Table 1. Preliminary results of the proposed model.

Accuracy	99.88
Precision	99.85
Recall	99.84
F-score	99.84

Fig. 2 shows the Confusion Matrix and Roc Curve on test data using the proposed model and we got its accuracy of 99 %.

4. Conclusions

In order to help coffee farmers boost output and quality, this study looked at computer approaches for recognizing illnesses in photos of the coffee leaf. We performed studies for recognizing five output classes: Cercospora, Rust, Miner, Phoma, and healthy leaves. The proposed model achieved an accuracy of 97 % Our findings demonstrated the viability of approaches for creating a system to detect diseases in coffee leaves that may be used to aid coffee farmers. Future research employing this method should investigate not just new coffee illnesses, but also their effects on the stem and root of the plant. Additionally, other sorts of texture properties, such as spectral attributes should be validated. Nevertheless, more tests with other picture samples are recommended.

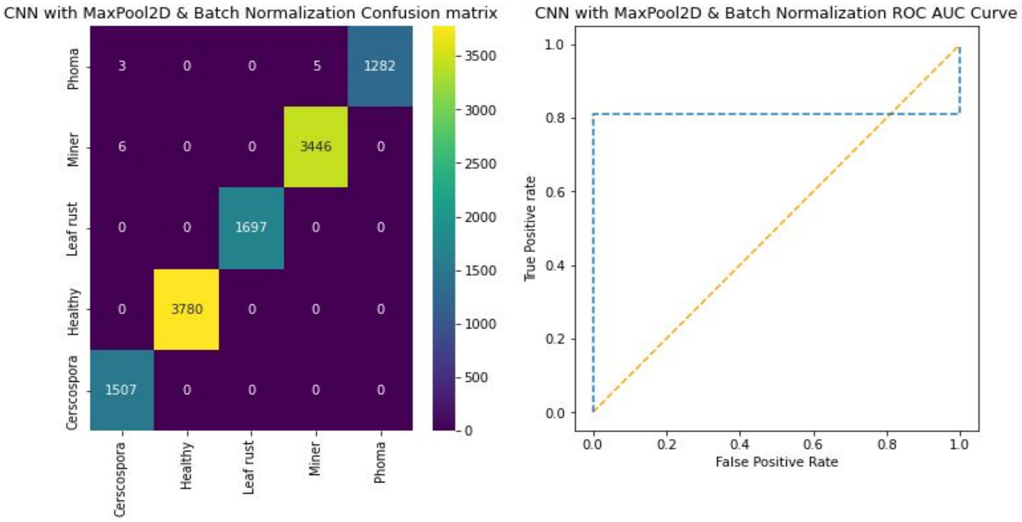


Fig. 2. Confusion matrix and roc curve perform prediction of coffee.

References

[1]. N. Baht, detection of plant diseases based on convolutional neural network approach: A review, in *Proceedings of the 14th International Conference on Electronics, Computers and Artificial Intelligence (ECAI'22)*, Ploiesti, Romania, 2022, pp. 1-4.

[2]. P. Kaur, et al., Recognition of leaf disease using hybrid convolutional neural network by applying feature reduction, *Sensors*, Vol. 22, Issue 2, 2022, 575.

[3]. M. Yebasse, et al., Coffee disease visualization and classification, *Plants*, Vol. 10, Issue 6, 2021, 1257.

[4]. J. Jepkoech, et al., Arabica coffee leaf images dataset for coffee leaf disease detection and classification, *Data in Brief*, Vol. 36, 2021, 107142.

(016)

Radar Target DOA Estimation using Deep Learning

Yiyang Li and Webert Montlouis
Johns Hopkins University, Electrical and Computer Engineering Department,
Baltimore, MD 21218 USA
E-mails: yli302@jhu.edu, wmontlouis@jhu.edu

Summary: In the last ten years, many Analog-to-Digital Converters (ADC) have had faster sampling rates, eliminating the need for multiple down conversion stages. These advances are beneficial to designing robust systems but produce a large amount of data. If not properly handled, these data create a bottleneck in the data and signal processors. As a result, techniques that can extract relevant information from high data rates are an important research area. Trained deep learning networks are known for quickly extracting relevant features from complex data, making them a modern approach to sensing and processing such data. This paper uses a Convolutional Neural Network (CNN) design to extract the Direction of Arrival (DOA) of targets from radar return signals. To minimize delay in the signal processor, the CNN is configured uniquely, with In-phase and Quadrature (IQ) components data fed directly to the trained network without preprocessing. The design shows high performance, and the results are presented.

Keywords: Artificial intelligence, Deep learning, Direction of arrival, Radar, Signal processor.

1. Introduction

The current environment requires that we make decisions for complex systems at machine speed. The proliferation of deep learning architectures provides the framework to tackle many problems that we thought impossible to solve a decade ago [1]. The introduction of faster CPU, GPU, and parallel processing make the application of deep leaning feasible. With advances in GPU, more sophisticated deep learning architectures are developed. Therefore, it is logical to replace traditional signal processing techniques with deep leaning.

Deep learning is a specific type of machine learning (ML) type that utilizes neural networks to make predictions about certain types of data [2]. It is called “deep” learning because these networks often have great depth, propagating data through many layers to perform classification. Deep learning applications include facial recognition, natural language processing, and self-driving cars, to name a few. In our case, deep learning will be used to complete the task of radar signal processing.

More specifically, it performs angle of arrival classification on a section radar signal.

Sequence-to-sequence classification takes a temporal data sequence and classifies each time step using a label [3-6]. In this paper, the trained CNN network is applied on each radar receive window. In this case, the classification of time steps of a reflected radar signal is performed. The complex data from the received window is transformed into real sequences to accomplish this classification process, which is more efficient than further adjusting the deep leaning model to accommodate complex datasets [7].

First proposed in [8], CNN has shown its ability to train a multi-layer network using gradient descent for complex, high-dimensionality, and a large amount of data. More specifically, the CNN model uses multiple layers of trainable filters to convolve with test samples, extracting local features in the lower and global features in the higher layers. More recent applications of CNN involve facial emotion recognition [9], PET image reconstruction [10], and many other classifications of image processing applications. CNN’s capability in determining the angle of arrival is explored by Kokkins et al. [11]. Yet, the authors use a sample correlation matrix transformation on the input data, demanding additional computational power to transform the raw input signals.

Data transformation is a common step in deep learning signal processing, but it can be problematic in applications where hardware is limited and system requirements demand low latency. In radar, for instance, the computational burden and time delay associated with these transformations can significantly impact some radar modes and render the system inefficient. To address this issue, we develop a CNN model that analyzes the direction of arrival of a target using the raw signal at the output of a Uniform Linear Array (ULA) and evaluates its performance. This modern approach using the trained network replaces traditional radar signal processors, providing a faster response time.

The paper is organized as follows. In Section 2, the radar data model is formulated. In Section 3, the CNN training methodology is described. In Section 4, the network model developed in the paper is shown. In Section 5, the simulation results are presented. We conclude in Section 6.

2. Radar Model

The signal model is developed using the constant low-velocity target model [12]. The radar uses a Linear Frequency Modulated (LFM) signal, and each transmitted pulse signal is given by

$$s(t) = rect\left(\frac{t}{T}\right) e^{j2\pi\left(f_0 t + \frac{1}{2}Kt^2\right)}, \tag{1}$$

where $u(t)$ represents the envelope of the LFM wave, T represents pulsewidth, and K is the linear modulation rate. Down-converting this transmit signal gives the baseband signal

$$s_b(t) = rect\left(\frac{t}{T}\right) e^{j\pi Kt^2} \tag{2}$$

Under the constant and low-velocity target model assumptions, the target's movement is modeled as static when the signal reflects off the target. Therefore, the following equation describes the reflected baseband signal

$$s_{rb}(t) = rect\left(\frac{t-t_r}{T}\right) e^{j\pi Kt^2} + n(\sigma) \tag{3}$$

In the equation $t_r = 2R/c$ represents the time delay resulting from the signal propagating to and from the target and the radar receiver, and the noise of the environment is modeled by the random Gaussian noise $n(\sigma)$. Since, for our purposes, the desire is to study the detection of the target, there is a need to set up the return signal amplitude for our single pulse system.

If we consider a ULA of M Omni-directional sensors receiving D independent narrowband reflected signals from targets in the field of view $(D < M)$, the output of the antenna array system becomes

$$\boldsymbol{x}(t) = \mathbf{A}(\theta)s_{rb}(t) + \boldsymbol{N}(t), \tag{4}$$

where $\boldsymbol{x}(t) \in \mathbb{C}^M$ is the antenna array output at each sample time t. $s_{rb}(t) = [s_{rb1}(t) \cdots s_{rbD}(t)]^T \in \mathbb{C}^D$ denotes the reflected target signals, $\boldsymbol{N}(t) \in \mathbb{C}^M$ is the sensor noise vector, and the Direction of Arrival (DOA) of the reflected signals are $\boldsymbol{\theta} \triangleq [\theta_1 \cdots \theta_D]^T$. The array manifold is given by

$$\mathbf{A}(\theta) = [a(\theta_1) \cdots a(\theta_D)], \tag{5}$$

where $\boldsymbol{a}(\theta_d) = \left[1\ e^{-j2\pi sin(\theta_d)d/\lambda} \ldots e^{-j2\pi sin(\theta_d)d/\lambda}\right]^T \in \mathbb{C}^D$ is the steering vector, λ denotes the carrier wavelength, and $d = \lambda/2$ is the antenna element spacing.

The intensity of the return follows the Radar Range Equation (RRE). For the single pulse case, the RRE is given by

$$SNR = \frac{P_t G^2 \lambda^2 \sigma}{(4\pi)^3 R^4 k T_0 BFL}, \tag{6}$$

with the parameters: P_t is the Peak transmit power, G is the Array gain, λ is the Transmit signal wavelength, σ is the Target radar cross section, k is the Boltzman constant, B is the Radar bandwidth, F is the Noise figure, L is the Total system losses.

We then assume that a 13 dB signal-to-noise ratio is enough from the return signal for a general detection. It is further assumed that the radar is properly sized to detect a target at the instrumented range with a given radar cross-section.

3. Training Methodology

The training dataset consists of samples generated by sampling the receiver output as described in the radar model discussed in equation (4), with 100 sample signals generated for each angle in the -90 to 90 degrees range. The samples are generated using a randomly chosen range between 200 km to 300 km to keep some generality in the target distance. The target velocity was kept constant at 10 meters/second. The receiving antenna was simulated with 40 channels and a 5 km range window (or 1.67e-5 second time window), resulting in a 40×40 complex-valued matrix. To avoid complex-valued weights in the deep learning networks, each signal sample is further partitioned into a corresponding real and imaginary 3D matrix with dimensions $40 \times 40 \times 2$ to reduce complexity in the training of the deep learning network.

There are several details worth noting in the training process as well, one of which is the selection of learning rates. The learning rate is a hyperparameter that dictates how much adjustment is made to the deep learning model weights after each iteration. Higher learning rates converge faster but could overlook essential details or optimal outcomes. In comparison, lower learning rates are more accurate with further information at the cost of slower convergence and the potential risk of being stuck at local gradient minimums [13]. For the current model, the learning rate is set high to 0.001 in the beginning to ignore the noise patterns in the training dataset, as is often the case in managing noisy training datasets in the training process. The learning rate is then reduced by (10 %) for every ten epochs to extract more acceptable attributes of target signals in the later stages of training.

Overfitting is often encountered in the training process; batch normalization is a mechanism that helps to avoid such problems [14]. With mini-batches, gradient update after each iteration contains small noises that accumulate into a more generalized model that is robust towards variations. A batch size of 128 samples is used in the training process as it reduces the time needed to train while still providing enough robustness [15]. It is not desirable to use more min-batches because it would increase the training time.

The last hyperparameter to tune is the epoch number, which directly correlates to the training time of the model. A longer epoch takes longer to train and also suffers the potential risk of overfitting after convergence is reached. The training is initialized with 100 epochs as a starting point, and early stopping is implemented when convergence is achieved around 40 epochs. Several runs are done to ensure that 40 epochs are optimal as it captures the convergence without being too long to induce overfitting.

4. Network Architecture

Despite the temporal nature of radar signals, a close examination of the dataset and expected outcomes revealed that CNN is well-suited for this type of data. A block matrix can represent an excerpt of a radar signal with dimensions of 40 × 40 × 2 as mentioned in section III, and the output range of -90 to 90 degrees can be mapped into 181 categories, with each angle being a distinct category. Other time series models, such as LSTM [16] could be viable and could be further examined and compared, but this is beyond the scope of this paper.

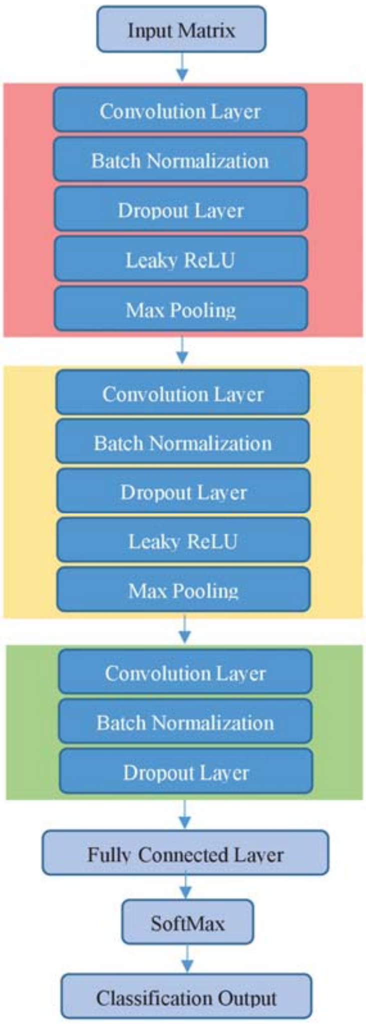


Fig. 1. CNN Architecture.

The CNN model mainly consists of 3 repeating layers following the input layer. Each layer consists of a convolutional learning layer, a batch normalization layer, and a dropout layer to prevent overfitting, a Leaky ReLU layer to avoid vanishing gradient [17], and max pooling in the first two layers to ensure the extraction of features. It is worth noting that the input data has large dimensions, and the angle of arrival requires a receptive field that covers multiple channels over a time interval**.** Therefore, a larger filter 5 × 5 is used in the first convolutional layer, while 3 × 3 filters are used in later layers. The results are then passed through a fully connected layer and a SoftMax layer for the final classification output.

5. Simulation Results

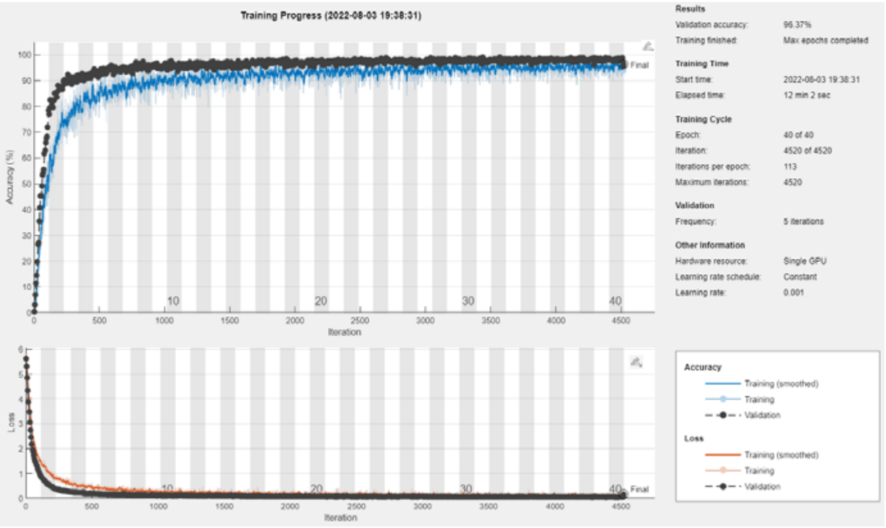
The resulting loss and accuracy curves for test and validation converged quickly, indicating that a simpler model or a shorter signal window would also produce a trained network that captures all the features. But, in the interest of generating a high-fidelity detector or signal processor, it may be necessary to keep the current model. In the current setup, the radar signal processor is used to indicate the existence of a target. When the hypothesis is true, other signal-processing functions are triggered to extract additional features from the target return.

To test the accuracy of the model, a separate dataset is often generated. In this example, 1000 samples are randomly created in the range of -90 to 90 degrees to ensure that we cover all possible cases for a ULA. A prediction is considered accurate if it is within +/-1 degree of the actual angle, and the results are shown in Fig. 3.

The accuracy criteria can be further reduced if the application requires a tighter limit. For the 1000 samples tested, 975 samples were correctly predicted, leading to a 97.5 % accuracy. It is worth noting that many wrong predictions are found towards either end of the range, i.e., < -80 degrees and > 80 degrees. This corresponds to the fact that angle changes are rather minuscule at such ranges. It is also worth noting that some false predictions exist when the actual angle is at 90 degrees, as the data there are very similar to the data from -90 degrees. Aside from these errors, the model performed extraordinarily well in the middeep learning e section from -80 to 80 degrees, showing the model’s reliability and potential for further improvements in angle accuracy.

6. Conclusion

With the introduction of faster analog-to-digital converters, radar receiver designs can be less complex, more efficient, and less costly. As such, the technology trend is converging toward sampling the radar return at RF. This gain comes with some challenges that must be addressed; otherwise, the computational complexity will be high for the signal processor. A novel radar target direction of arrival classification scheme was investigated using deep learning convolutional neural network. The results show that our approach is accurate and can be easily implemented in the radar signal processor. The classification scheme presented in this paper indicates that target DOA can be estimated in real-time using the raw radar data without preprocessing.



Results
Validation accuracy: 96.37%
Training finished: Max epochs completed

Training Time
Start time: 2022-08-03 19:38:31
Elapsed time: 12 min 2 sec

Training Cycle
Epoch: 40 of 40
Iteration: 4520 of 4520
Iterations per epoch: 113
Maximum iterations: 4520

Validation
Frequency: 5 iterations

Other Information
Hardware resource: Single GPU
Learning rate schedule: Constant
Learning rate: 0.001

Fig. 2. Training Performance.

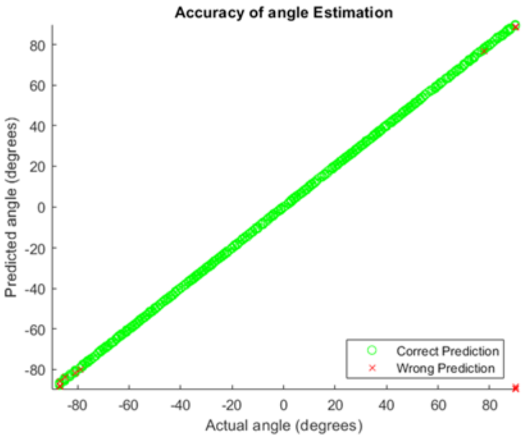


Fig. 3. Plot of angle classification accuracy.

References

[1]. S. Yang, *et al.*, Learning COVID-19 Pneumonia Lesion Segmentation From Imperfect Annotations via Divergence-Aware Selective Training, *IEEE Journal of Biomedical and Health Informatics*, Vol. 26, Issue 8, 2022, pp. 3673-3684.

[2]. H. C. Kaskavalci, S. Gören, A Deep Learning Based Distributed Smart Surveillance Architecture using Edge and Cloud Computing, in *Proceedings of the International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML'19)*, 26-28 Aug. 2019, pp. 1-6.

[3]. H. Phan, F. Andreotti, N. Cooray, O. Y. Chén, M. D. Vos, SeqSleepNet: End-to-End Hierarchical Recurrent Neural Network for Sequence-to-Sequence Automatic Sleep Staging, *IEEE Transactions on Neural Systems and Rehabilitation Engineering,* Vol. 27, Issue 3, 2019, pp. 400-410.

[4]. Y. Tang, J. Xu, K. Matsumoto, C. Ono, Sequence-to-Sequence Model with Attention for Time Series Classification, in *Proceedings of the IEEE 16th International Conference on Data Mining Workshops (ICDMW'16)*, 12-15 Dec. 2016, pp. 503-510.

[5]. Y. Deng, L. Wang, H. Jia, X. Tong, F. Li, A Sequence-to-Sequence Deep Learning Architecture Based on Bidirectional GRU for Type Recognition and Time Location of Combined Power Quality Disturbance, *IEEE Transactions on Industrial Informatics,* Vol. 15, Issue 8, 2019, pp. 4481-4493.

[6]. S. Arniriparian, M. Freitag, N. Cummins, M. Gerczuk, S. Pugachevskiy, B. Schuller, A Fusion of Deep Convolutional Generative Adversarial Networks and Sequence to Sequence Autoencoders for Acoustic Scene Classification, in *Proceedings of the 26th European Signal Processing Conference (EUSIPCO'18)*, 3-7 Sept. 2018, pp. 977-981.

[7]. C. Lee, H. Hasegawa, S. Gao, Complex-Valued Neural Networks: A Comprehensive Survey, *IEEE/CAA Journal of Automatica Sinica,* Vol. 9, Issue 8, 2022, pp. 1406-1426.

[8]. Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proceedings of the IEEE,* Vol. 86, Issue 11, 1998, pp. 2278-2324.

[9]. E. Pranav, S. Kamal, C. S. Chandran, M. H. Supriya, Facial Emotion Recognition Using Deep Convolutional Neural Network, in *Proceedings of the 6th International Conference on Advanced Computing and Communication Systems (ICACCS'20)*, 6-7 March 2020, pp. 317-320.

[10]. K. Gong, *et al.*, Iterative PET Image Reconstruction Using Convolutional Neural Network Representation, *IEEE Transactions on Medical Imaging,* Vol. 38, Issue 3, 2019, pp. 675-685.

[11]. G. Kokkinis, Z. D. Zaharis, P. I. Lazaridis, N. V. Kantartzis, Direction of Arrival Estimation Applied to Antenna Arrays using Convolutional Neural Networks, in *Proceedings of the 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC'22)*, 30 May-4 June 2022, pp. 1-4.

[12]. M. A. Richards, Fundamentals of Radar Signal Processing, 2nd Ed., *McGraw-Hill Education LLC*, Chicago, Ill., 2014.

[13]. J. Konar, P. Khandelwal, R. Tripathi, Comparison of Various Learning Rate Scheduling Techniques on Convolutional Neural Network, in *Proceedings of the IEEE International Students' Conference on*

Electrical, Electronics and Computer Science (SCEECS'20), 22-23 Feb. 2020, pp. 1-5.

[14]. P. R. Tupe, P. M. Vibhute, M. A. Sayyad, An Architecture Combining Convolutional Neural Network (CNN) with Batch Normalization for Apparel Image Classification, in *Proceedings of the IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC'20)*, 16-17 Dec. 2020, pp. 1-6.

[15]. R. Lin, Analysis on the Selection of the Appropriate Batch Size in CNN Neural Network, in *Proceedings of the International Conference on Machine Learning and Knowledge Engineering (MLKE'22)*, 25-27 Feb. 2022, pp. 106-109.

[16]. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.,* Vol. 9, Issue 8, Nov. 15 1997, pp. 1735-1780.

[17]. J. Xu, Z. Li, B. Du, M. Zhang, J. Liu, Reluplex made more practical: Leaky ReLU, in *Proceedings of the IEEE Symposium on Computers and Communications (ISCC'20)*, 7-10 July 2020, pp. 1-7.

(017)

Enhancing Early Detection of Schizophrenia through Multi-modal EEG Analysis: A Fusion of Wavelet Transform, Reconstructed Phase Space, and Deep Learning Neural Networks

A. Al Fahoum and A. Zyout
Biomedical Systems and Informatics Engineering Dept., Hijjawi Faculty for Engineering Technology,
Yarmouk University, Irbid, 21163, Jordan
E-mail: afahoum@yu.edu.jo

Summary: This article aims to describe a reliable expert system for the early diagnosis of schizophrenia using only EEG signals. EEG is an inexpensive, safe, and non-invasive method for diagnosing neurological diseases. The study employs the reconstructed phase space (RPS) and the continuous wavelet transform to better reveal the hidden interrelations of EEG nonstationarities and provide valuable insights. The resulting images are then fed into a deep-learning network for better classification. The proposed algorithm was tested using 70 % of the data for training, 15 % for validation, and 15 % for testing. After 50 attempts, the DL-CNN achieved an accuracy of 97.09 ± 2.22 %, and the MCC was 94.19 ± 4.40 %. Various metrics were applied to validate the precision and robustness of the results, leading to a minimum mean value of 96.17 % and a range of standard deviations of 2.22 % to 3.47 %. The study's outcomes lay the foundation for researchers and technology professionals to develop more accurate methods for identifying brain illnesses and abnormalities. The results demonstrate a comprehensive and stable system implementation that can automatically differentiate between schizophrenic patients and healthy individuals.

Keywords: Reconstructed phase space (RPS), Deep learning (DL), Continuous wavelet transform (CWT), Schizophrenia (SZ) classification.

---

1. Introduction

Schizophrenia (SZ) is a highly complex psychiatric disorder that affects millions of individuals worldwide. While research on the early detection of SZ is limited compared to other forms of neurodegeneration, recent studies have started focusing on identifying the disorder at its earliest stages [1].

One promising approach in SZ research involves analyzing electroencephalogram (EEG) signals, specifically using auto-regressive (AR) coefficients and vector autoregressive (VAR) models [2]. The least-squares (LS) approach is used to estimate VAR coefficients, and the student t-test is employed to rate the essential characteristics of an SZ study [3]. Phase locking value (PLV) and coherence have also been utilized as valuable tools to build brain networks and detect SZ [4]. Additionally, researchers have measured decision-making abilities using P300, a potential endogenous component of event-related potential (ERP).

Classification methods have been applied to almost every research problem, and machine learning (ML) algorithms have been chosen often and perform well in supervised learning [5, 6]. For example, a new multi-domain connectome CNN (MDC-CNN) has been used in an SZ study. This proposed model extracts connectivity features and performs classification based on a CNN [2]. Another promising approach involves using time-frequency analysis and a convolutional neural network (CNN) using three press-button tasks to segregate normal subjects from SZ patients [7]. EEG signals were analyzed using continuous wavelet transform, short-time Fourier transform, and smoothed pseudo-Wigner-Ville distribution (SPWVD) techniques to obtain scalograms, spectrograms, and SPWVD-based time-frequency representation (TFR) plots, respectively. These 2-D plots were fed to pre-trained AlexNet, VGG16, ResNet50, and CNN, achieving an accuracy of 93.36 % using the SPWVD-based TFR and CNN models [7].

Early detection of SZ with precision is essential. Current methods proposed in the literature for identifying SZ entail separating rhythms, extracting features such as time, frequency, entropy, spectral, and nonlinear, and classifying them with reasonable accuracy. Notable is the fact that selecting appropriate features can be laborious and time-consuming. Moreover, conventional methods require multiple tuning parameters for decomposition, feature extraction, and classification. In addition, relying solely on the empirical selection of these tuning parameters may lead to information loss and an increase in misclassification. Therefore, there is an urgent need to devise a novel method for the precise and efficient detection of SZ. In this paper, a novel method is presented that, for the first time, integrates the capabilities of RPS and the CWT to maximize the differences between N and SZ characteristics and use these differences as input to a high-performance pre-trained DL network.

After a concise explanation of the methodology, the article proceeds to showcase its principal discoveries. Furthermore, it undertakes a comparative analysis of these findings with those of recent studies and concludes with a brief discussion of the outcomes.

2. Methodology

The EEG signals used in this study were obtained from a publicly available database [8]. To obtain detailed time-frequency representations and information about the phase from the EEG signals, the RPS and CWT were employed. The deep-learning CNN network was then trained to test images for the presence of SZ automatically. The proposed methodology is presented in Fig. 1.

Only specific channels (C3, C4, P3, and P4) are utilized for further processing in both N and SZ to enhance the detection of N and SZ EEG signals. It was found that these channels can better amplify the differences between N and SZ EEG signals [4]. For each subject, only the average of the aforementioned channels is processed. To ensure accuracy, each signal undergoes a band-pass filtering operation in the 0.5-40 Hz region using a Butterworth IIR digital filter to minimize electrical interference and high-frequency noise. Furthermore, other noises are eliminated by applying smooth filtering to low and higher frequencies of the processed EEG signal using the Daubachies wavelet filtering method. The data is normalized in the next step to establish a baseline for various patients and datasets. This preprocessing step is necessary for generating reliable and consistent signals [9].

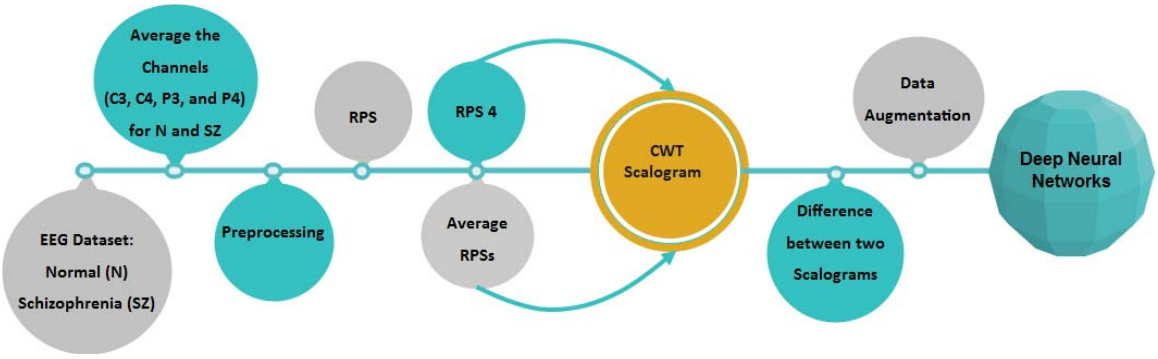


Fig. 1. Proposed automatic classification model.

All the processed average signals from both N and SZ are then further analyzed by the average mutual information function and the false nearest neighbor method to extract the best time lag (L) and dimension (d) for the RPS attractors. The achieved maximum time delay lag was ($L = 10$) and the best embedding dimension was found to be four ($d = 4$). The generated RPS attractors are then used to extract two signals. The first is the fourth vector of the RPS matrix, and the second is the average vector of all vectors of the RPS matrix. The two signals are then fed into the CWT, whose mother wavelet is the Morlet filter, to produce two different scalograms for each subject. The outcomes of CWT will produce cross-term interferences in the time-frequency domain [10]. The tradeoff between time–frequency localization and the inter-phase relations will reveal distinct patterns in the signals of N and SZ subjects [10]. The difference between the two scalograms maximizes their distinguishability. The final difference images for all subjects are then used as training and testing datasets for the DL network. In this work, ResNet 101 was considered due to its excellent performance. 70 % of the data is used as a training set, 15 % for validation, and 15 % for testing. The ResNet 101 parameters were as follows: the solver used was stochastic gradient descent with momentum, the minimum batch size was 64, the maximum epochs were 80, the learning rate was 1e-4, the validation frequency was 3, and Shuffle the data every epoch.

3. Results and Conclusions

The DL network trained and tested the RPS version of the EEG signals and the CWT version to understand the significance of the combination of RPS and CWT processing. The resulting accuracies were 74.32 % and 75.35 %, respectively. Accordingly, the RPS-CWT is proposed. The RPS-CWT is judged based on six performance parameters: sensitivity (SEN), accuracy (ACC), specificity (SPE), F-1 score, Mathew's correlation coefficient (MCC), and precision (PRC). The mathematical formulation of these parameters is obtained from [7]. Table 1 shows the performance parameters for the tested dataset using the RPS-CWT approach. The automatic classifier is robust when dealing with all calculated parameters, achieving an accuracy of 97.09 % with a standard deviation (SD) of 2.22 % (CI 0.95, $p<0.05$). The minimum average parameter for N is SPE at 96.17 %, while the maximum is sensitivity (Recall) at 98 %. Meanwhile, the minimum average parameter for SZ subjects is Recall at 96.17 %, while the maximum is PRC at 98.03 %. The average MCC parameter between N and SZ is 94.19±4.40 %. Furthermore, the SD for N ranged from 2.18 % to 3.47 % for other metrics, while the standard SD for SZ subjects ranged between 2.22 % and 3.47 %.

Table 2 shows a comparison between the proposed algorithm and recently published literature. The proposed algorithm has provided higher ACC, SPE, MCC, F1-Score, and PRC than most available algorithms in the recently published articles. Additionally, its performance parameters are robust.

Table 1. Performance parameters for the RPS_CWT proposed model for 50 iterations of the deep learning CNN network.

CM	Mean (%)		Standard deviation (%)	
	N	SZ	N	Sz
Recall	98.00	96.17	2.67	3.47
ACC	97.09		2.22	
SPE	96.17	98.00	3.47	2.67
MCC	94.19		4.40	
PRC	96.35	98.03	3.25	2.62
F-1 Score	97.12	97.05	2.18	2.27

In conclusion, this work accurately detects SZ in its early stages, leading to better management and treatment of the disorder. The proposed method showed accurate, stable, and robust performance parameters, with very close variances and means and the highest accuracy. Further research would enhance the results and contribute to the field significantly.

References

[1]. V. Jahmunah, S. Oh, V. Rajinikanth, E. Ciaccio, K. Cheong, N. Arunkumar, U. Acharya, Automated detection of schizophrenia using nonlinear signal processing methods, *Artif. Intell. Med.*, Vol. 100, Issue 9, 2019, pp.1-18.

[2]. C. Phang, F. Noman, H. Hussain, C. Ting, H. Ombao, A multi-domain connectome convolutional neural network for identifying schizophrenia from EEG connectivity patterns, *IEEE J. Biomed. Health Inform.*, Vol. 24, Issue 5, 2019, pp. 1333-1343.

[3]. S. Siuly, S. Khare, V. Bajaj, H. Wang, Y. Zhang, A computerized method for automatic detection of schizophrenia using EEG signals, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 28, Issue 11, 200, pp. 2390-2400.

[4]. F. Li, J. Wang, Y. Liao, C. Yi, Y. Jiang, Y. Si, W. Peng, D. Yao, Y. Zhang, W. Dong, et al., Differentiation of schizophrenia by combining the spatial EEG brain network patterns of rest and task P300, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 27, Issue 4, 2019, pp. 594-602.

[5]. M. Baradits, B. Kakuszi, S. Bálint, M. Fullajtár, L. Mód, I. Bitter, P. Czobor, Alterations in resting-state gamma activity in patients with schizophrenia: A high-density EEG study, *Eur. Arch. Psychiatry Clin. Neurosci.*, Vol. 269, Issue 4, 2019, pp. 429-437.

[6]. A. S. Al Fahoum, A. O. Abu Al-Haija, H. A. Alshraideh, Identification of coronary artery diseases using photoplethysmography signals and practical feature selection process, *Bioengineering*, Vol. 10, Issue 2, 2023, 249.

[7]. S. K. Khare, V. Bajaj, U. R. Acharya, SPWVD-CNN for automated detection of schizophrenia patients using EEG Signals, *IEEE Transactions on Instrumentation and Measurement*, Vol. 70, Issue 4, 2021, pp. 1-9.

[8]. EEG Database – Schizophrenia. EEG Database – Schizophrenia, brain.bio.msu.ru/ eeg_schizophrenia.htm

[9]. A. S. Al-Fahoum, A. M. Qasaimeh, A practical reconstructed phase space approach for ECG arrhythmias classification, *Journal of Medical Engineering & Technology*, Vol. 37, Issue 7, 2013, pp. 401-408.

[10]. A. S. Al-Fahoum, I. Howitt, Combined wavelet transformation and radial basis neural networks for classifying life-threatening cardiac arrhythmias, *Medical & Biological Engineering & Computing*, Vol. 37, Issue 5, 1999, pp. 566-573.

[11]. Y. Wu, M. Xia, X. Wang, Y. Zhang, Schizophrenia detection based on EEG using recurrent auto-encoder framework, *arXiv Preprint*, 2022, arXiv: 2207.04262.

[12]. A. Keihani, S. Sajadi, M. Hasani, F. Ferrarelli, Bayesian optimization of machine learning classification of resting-state EEG microstates in schizophrenia: A proof-of-concept preliminary study based on secondary analysis, *Brain Sciences*, Vol. 12, Issue 11, 2022, 1497.

[13]. E. Lillo, M. Mora, B. Lucero, Automated diagnosis of schizophrenia using EEG microstates and deep convolutional neural network, *Expert Systems with Applications*, Vol. 209, Issue 12, 2022, 118236.

[14]. C. Devia, R. Mayol-Troncoso, J. Parrini, G. Orellana, A. Ruiz, P. E. Maldonado, J. I. Egana, EEG classification during scene free-viewing for schizophrenia detection, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 27, Issue 6, 2019, pp. 1193-1199.

Table 2. Comparison of recent studies about EEG-based detection of SZ.

Ref.	Classifier	Algorithm	Acc (%)
[1]	SVM	Nonlinear features	92.91
[2]	MDC- CNN	Connectomic feature extraction	91.69
[3]	SVM	EMD	89.59
[4]	SPN, SVM, LDA	HT, REST, PLV, P300	90.48
[5]	HLM	Butterworth filter, ICA, FFT	$p < 0.05$
[7]	CNN	CWT, STFT, SPWVD	93.36
[11]	RAE		81.81
[12]	G- SVM	Microstate analysis	90.93
[13]	CNN	Random walk generation	93.00
[14]	LDA, Rule-based LDA	utterworth filter, ERP, WRS	71.00
Proposed	CNN	RPS and CWT	97.09

5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2023),
7-9 June 2023, Tenerife (Canary Islands), Spain

(018)

Iterative Laser Measurement of an Aircraft Engine Blade in Robotic Grinding Process

K. Kurc, A. Burghardt, P. Gierlak, M. Muszyńska and D. Szybicki
Rzeszów University of Technology, Faculty of Mechanical Engineering and Aeronautics,
al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
Tel.: + 48178651814
E-mails: kkurc@prz.edu.pl, andrzejb@prz.edu.pl, pgierlak@prz.edu.pl, magdaw@prz.edu.pl,
dszybicki@prz.edu.pl

Summary: Aircraft engine blades are manufactured by a number of methods. One of them is precision electrochemical machining, after which the blades are subjected to a manual grinding process. When grinding aircraft engine blades, high precision is required and iterative dimensional control is necessary. This control includes the thickness of the blade's feathering in the required sections, the radii of the leading and trailing edges, the chords, the twist angles, etc. This paper presents the developed robotic blade grinding process with proprietary iterative laser measurement of geometric parameters. A measuring device with laser heads was designed and fabricated, calibrated and checked for repeatability. The measurement data were used in the process of selecting the machining feed rate and tool speed by a fuzzy logic decision-making system. The proposed solution was verified on twenty aircraft engine blades.

Keywords: Industrial robot, Laser scanners, Software, Decision-making system, Fuzzy logic, Communication.

---

1. Introduction

One of the basic methods of manufacturing complex-shaped parts made of difficult-to-machine materials is electrochemical machining (ECM). The machining method is based on the phenomenon of anodic dissolution of the machined surface. The workpiece (anode) is connected to the positive pole, while the working electrode (cathode) is connected to the negative pole. The inter-electrode space is filled with flowing electrolyte. The excess is removed atom by atom by electrochemical reactions. An important advantage of electrochemical machining is that there is no wear on the tool (the working electrode – the cathode), since the reaction equivalent to the removal of material from the workpiece is the release of hydrogen on the surface of the cathode, which is removed by the electrolyte flowing through the machining area. Because of the above, electrochemical machining is widely used in the aerospace, aviation, automotive and electromechanical industries. The accuracy of machining depends on a number of interrelated factors: the geometry of the working electrode, the machining parameters (electrical, hydrodynamic), the machining time, the electrochemical properties of the electrolyte, the initial geometry of the machined surface, etc. [1-7]. The aggregate accuracy during hollowing reaches values of 0.1-0.3 mm.

Because of this, manufactured parts with complex shapes require grinding and polishing. These activities are mostly performed manually by workers. During the grinding of the blade, they repeatedly check the obtained effect on specially prepared measuring gauges in a visual and organoleptic way. This activity is tedious, monotonous requiring a lot of manual skills, and in the current labor market it is increasingly difficult to find and train a worker in this profession. For the above reasons, attempts have been made to automate and robotize this task [8-10].

In the article [8, 9], the authors implement the process in a robotic way, however, the measurement with an optical 3D scanner greatly increases the measurement time and there is also the need for an anti-reflective agent. In the article [10], the authors measure the thickness of the blade profile only in three places with tactile sensors without the ability to measure, for example, the radii of the leading and trailing edges.

In article [11], the authors present the process of robotic grinding of complex components and focus on process performance and challenges, solutions and applications.

The authors [12] present the industrial robot approach as a new idea for manufacturing complex components. Compared to multi-axis CNC machine tools, robots are attractive due to their large workspace and competitive price, making them a cost-effective solution for machining complex components, especially large-sized parts.

Over the past two decades, research results on robotic grinding of complex-shaped parts have mainly focused on feasibility studies of robotic machining [13-15], or modeling and analysis of machining dynamics [16-18]. Research is mainly conducted on aspects of robot position optimization [19-21], robot calibration [22-27], robot path planning [28-30], allowance removal control [17, 31-33], and force control [34-37].

In the article [38], the authors reviewed the research and development of turbine blade profile measurement methods, describing common and advanced measurement methods. They described the principles of measurement, key technical issues and analyzed the characteristics of these methods.

Geometric and optical measurement conditions, such as surface quality, surface color, measurement distance and surface orientation, affect the measurement accuracy of laser displacement sensors (LDS). Measurement errors caused by these factors have been extensively studied in works [39-43]. Non-contact methods, thanks to their numerous advantages such as good stability, large measurement range, high accuracy and speed, have become widely available in geometric inspection of parts [44], product quality control [45, 46], three-dimensional shape measurements [47, 48]. Some traditional measurement systems, such as the well-known coordinate measuring machine (CMM), often fail for some large free-form surfaces [49]. The paper [50] proposes an adaptive grinding method for precision cast blades with geometric deviation to improve precision and machining performance. Precision-cast blades are essential components of aircraft engines, and the precision of their machining has a significant impact on the performance of an aircraft engine.

Currently, the measurement of aircraft engine blades is mainly carried out using coordinate measuring machines. This method makes it possible to measure most aircraft engine blades by direct contact. However, the complex surface, chord width and high twist of the blade profile lead to measurement difficulties. The article [51] proposed a new method for obtaining over-dimension data based on a point cloud of spatial scanning with blue light similarly to the article [52].

An aircraft engine blade is a typical thin-walled workpiece with low stiffness. The article [53] presents a new cantilever grinding process for high-precision machining of aircraft engine blades. Only one end of the blade is fixed, and the other end is free to eliminate deformation of the blade caused by excessive stiffening.

Reviewing the above solutions, it can be concluded that the grinding and polishing process as well as the measurement of aircraft engine blades is a developmental issue and there is still much room for improvement, such as the time of acquiring geometric information on the basis of which process parameters are selected.

2. Developed Process

Turbine machines have an extremely wide range of applications in aviation, aerospace, automotive, power generation and many other industries. Measuring the profile of a blade is a difficult issue especially because of its complex shape.

2.1. Robotic Station Design

Two KEYENCE LJ-X8080 laser profilometers and the KEYENCE LJ-X8000 blade profile measurement management controller were used to measure the blade profile. This controller has a system for writing complex measurement programs (Fig. 1).

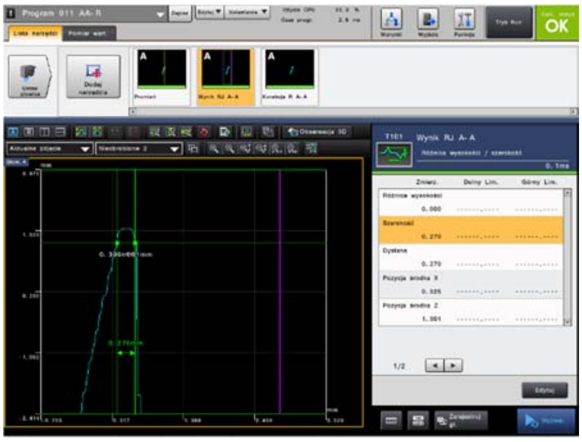


Fig. 1. An application for writing measurement programs.

The KEYENCE LJ-V7200 laser profilometer and KEYENCE XG-X2800 controller were used to correct the repeatability of the ABB IRB 2400 robot, which holds the blade during measurement in different cross sections (Fig. 2).

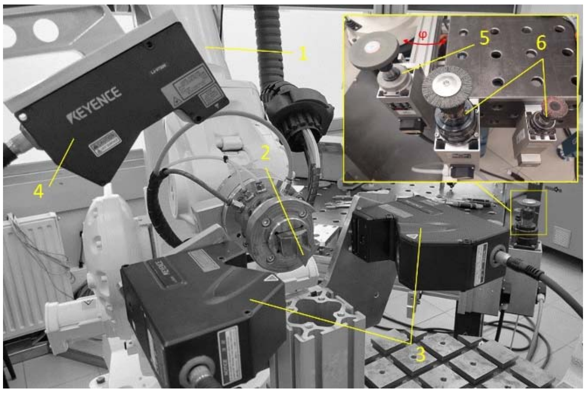


Fig. 2. Robotic station for the iterative measurement and machining process. 1 is the ABB IRB 2400 robot, 2 is the blade profile in the process of grinding, 3 is the two KEYENCE LJ-X8080 laser profilometers, 4 is the KEYENCE LJ-V7200 laser profiler, 5 is the spindle with Lukas P6SE1 grinding wheel on compensator (φ – compensator deflection angle), 6 is the two spindles (one with flap wheel, the other with polyamide brush).

A compensator with a deflection angle φ has two springs in its design. The role of the springs is to provide a constant clamping force of approximately 5 N (Fig. 3) and to compensate for tool wear.

Communication between (Fig. 4) the KEYENCE controllers and ABB's PLC and robot controller is provided by the Profinet protocol.

A PLC works as the management system. It receives information from two measurement systems. The first is a KEYENCE XG-X2800 controller together with a KEYENCE LJ-V7200 laser profilometer, whose task is to correct the repeat position of the robot holding the blade during measurement. The second system is a KEYENCE LJ-X8000 controller and two KEYENCE LJ-X8080 laser profilometers, which provides measurement of the allowances on the convex and concave of the blade profile in selected sections.

Fig. 3. Measurement of the achieved force.

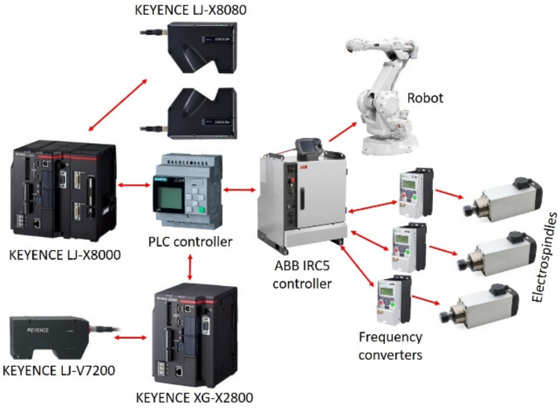


Fig. 4. Information flow between devices.

2.2. Measuring System

KEYENCE laser profilometers and controllers were used to build the designed measurement system. After many measurement tests, it was decided to position the heads opposite each other with the measurement zone in the middle. This arrangement makes it possible to simultaneously measure the allowances on both the blade convex and blade concave, as well as the blade thickness at the controlled sections (Fig. 5). The blade is delivered into the measurement zone in pre-programmed sequences by an ABB IRB 2400 robot. A reference blade with nominal parameters was used to write the measurement program. The profiles of the reference vane held by the ABB robot in sections A-A, B-B, C-C, D-D (Fig. 5) were stored in the controller's memory and served as a reference element (Fig. 6).

The currently measured values are compared to these stored reference profiles and the deviations (allowances) on the convex as well as concave of the blade profile and blade thickness at the controlled locations are obtained.

The required tolerance of the blade thickness, depending on the measuring point of the blade, is approximately ±0.1 mm. The aim is to iteratively grind the blade to approximately the same dimensions as the reference blade.

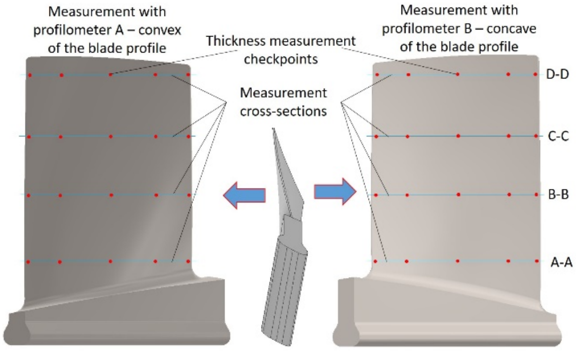


Fig. 5. Measurement concept.

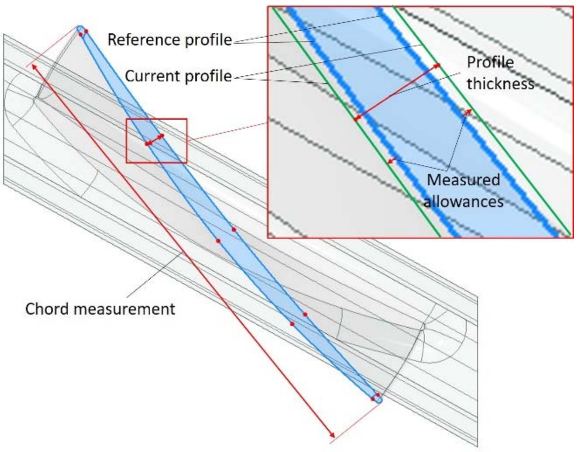


Fig. 6. View of D-D section of blade profile.

A certain difficulty during measurement is the robot's repeatability of ±0.05 mm. With this repeatability, the robot is able to deliver the blade into the measurement zone once again. For this reason, the KEYENCE XG-X2800 controller with the KEYENCE LJ-V7200 laser profiler was used to correct the repeatability of the robot, so that the measured blade is deposited in the same place every time. Without this correction, deviations were obtained, e.g. plus towards the concave of the blade and minus towards the convex of the blade or vice versa.

2.3. Selection of Process Parameters

In order to select the optimum parameters (feed rate) in the blade machining process, a solution based on fuzzy logic systems was proposed. The proposed solution was simulated in MATLAB, using the 'Fuzzy Logic Toolbox' library (Fig. 7). A Mamdani-type fuzzy model, a fuzzy implication of type (MIN) and a sharpening method of type (CA) were adopted to determine the dependence of the tool feed rate on the amount of collected material (allowance).

Twenty regularly spaced measurement points were adopted, where measurements were taken to determine the allowance value. Six feed rates ranging from 3 to 18 [mm/s] were used to build the fuzzy model. The proposed fuzzy model consists of three input signals *x, y, N and* one output signal (velocity). Where *x, y* are the coordinates of the position of the measurement points on the blade surface. Each input signal was divided into an appropriate number of fuzzy sets. A uniform division of the consideration space by the fuzzy sets was assumed. The fuzzy sets were adopted in the form of triangular sets. The consideration space for signal *x was divided* into 5 fuzzy sets from A_1 to A_5 (Fig. 8).

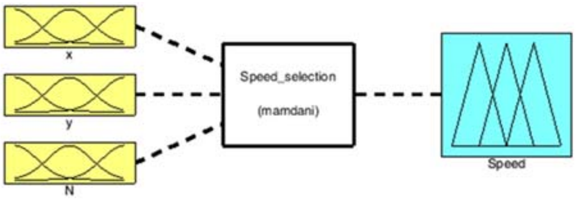


Fig. 7. Fuzzy model proposed for the blade machining process.

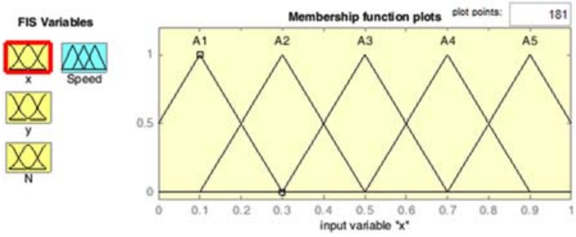


Fig. 8. 'Membership Function Editor' window showing the subdivision of the consideration space for the input signal *x*.

For the *y* signal, 4 fuzzy sets from B_1 to B_4 were adopted. This decomposition is due to the division of the blade into corresponding fragments. For signal *N* (collected allowance), the space was divided into 6 fuzzy sets ranging from 0 to 0.18 [mm]. These values are related to the amount of the allowance collected by the tool (sets from n_1 to n_6). For the output signal, 6 fuzzy sets were adopted. These are triangular sets with centres 3, 6, 9, 12, 15, 18 with an assumed tolerance of 0.5 [mm/s]. The graphical interpretation of the fuzzy rules is shown in Fig. 9.

At the beginning of the process, sharp values are introduced where their blurring takes place, i.e. the calculation of the degree of membership of the individual fuzzy sets A_i, B_j, N_k. When fuzzy inference is realised, it is required to assess the degree to which the individual rules are satisfied.

The higher the degree of fulfilment of a premise, the higher the membership of a rule. In the proposed fuzzy model, a premise consists of three simple premises connected by a logical conjunction AND. The next step is the inference of the proposed fuzzy model. Using the input membership degrees, the resulting membership function of the model output is calculated. In this block, a rule base is defined, which includes:

$$If\ (x\ is\ A_1) and\ (y\ is\ B_1)\ and\ (N\ is\ n_2)\ then\ (Speed\ is\ S_5)$$

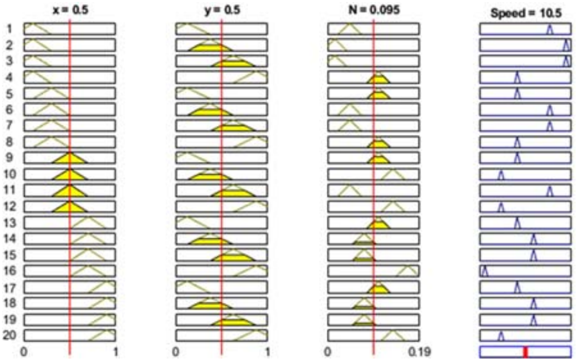


Fig. 9. Fragment of a graphical interpretation of fuzzy rules.

The rule is interpreted as follows: if the measuring point is defined by the set A_1 and B_1 and the amount of collected allowance is n_2 then the feed rate of the tool should be 15 [mm/s]. The rest of the rule runs as follows:

$$If\ (x\ is\ A_1) and\ (y\ is\ B_2)\ and\ (N\ is\ n_1)\ then\ (Speed\ is\ S_6)$$
$$If\ (x\ is\ A_1) and\ (y\ is\ B_3)\ and\ (N\ is\ n_1)\ then\ (Speed\ is\ S_6)$$
$$If\ (x\ is\ A_1) and\ (y\ is\ B_4)\ and\ (N\ is\ n_4)\ then\ (Speed\ is\ S_3)$$
$$\vdots$$
$$If\ (x\ is\ A_5) and\ (y\ is\ B_4)\ and\ (N\ is\ n_5)\ then\ (Speed\ is\ S_2)$$

A fuzzy model surface was also obtained (Fig. 10).

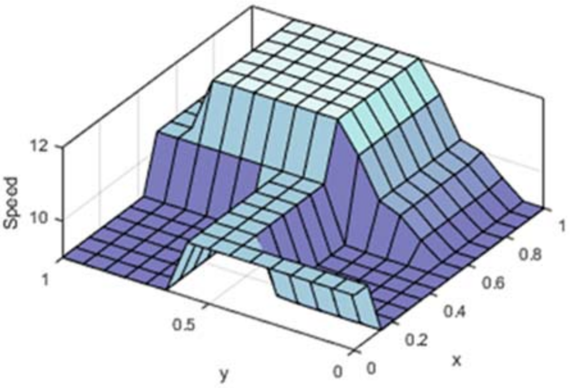


Fig. 10. Graphical interpretation of the fuzzy model surface.

An intelligent controller based on fuzzy logic generates tool speed values depending on the value of the material allowance present. The proposed solution was implemented on the controller using PLC Coder.

3. Iterative Process Algorithm

The process algorithm for iterative measurement of an aircraft engine blade in a robotic grinding process is shown in Fig. 11.

The start of the process is the retrieval of an aircraft blade from the details magazine by the developed robot gripper. The blade has sharp edges from previous processes, so the blade is pre-treated with robotic deburring brushes. The blade is then taken to the measuring position by the robot gripping the blade, and the blade is measured and aligned to a reference blade stored in the program. The next step is to measure the blade in the required cross sections and determine the allowances. This is followed by a decision as to whether the blade is in accordance with the reference and the process is complete or whether corrections are required. If so, the process parameters are selected by the fuzzy logic system. The feed rate of the grinding process and the selection of the programmed blade-to-tool contact path are selected (Fig. 12).

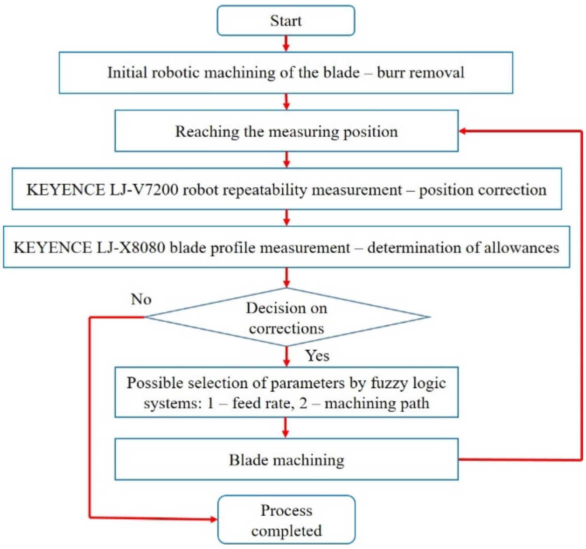


Fig. 11. Process algorithm.

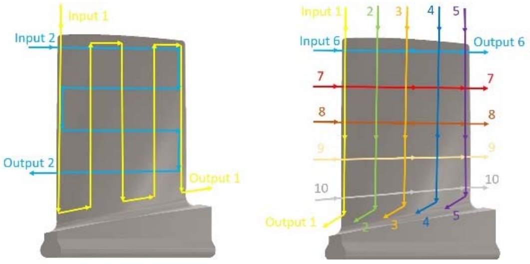


Fig. 12. Examples of blade processing paths.

The next stage involves machining the blade and arriving at the measuring positions. The blade is measured and ground as many times as necessary until, in the decision-making process, it conforms within the specified tolerance to the reference blade.

4. Conclusions

The aim of the study was to develop a robotic grinding process that takes into account the individual geometric parameters of each blade based on iterative measurements with laser heads. Two measurement systems were used, whereby the repeatability of the robot was corrected with respect to the master measurement and the blade was measured in programmed sections. The measured allowances with respect to the reference profiles allowed the feed rate to be determined by a fuzzy logic system. The proposed fuzzy model consists of three input signals *x, y, N and* one output signal. The proposed iterative machining process, consisting of measurements and grinding of the blade, is carried out until the objective is reached, i.e. grinding of the blade in accordance with the compared model.

References

[1]. H. Hardisty, A. R. Mileham, H. Shirvarni, A. N. Bramley, A finite element simulation of the electrochemical machining process, *CIRP Annals*, Vol. 42, Issue 1, 1993, pp. 201-204.

[2]. J. Kozak, Mathematical models for computer simulation of electrochemical machining processes, *Journal of Materials Processing Technology*, Vol. 76, Issues 1-3, 1998, pp. 170-175.

[3]. J. Kozak, K. P. Rajurkar, Y. Makkar, Selected problems of micro-electrochemical machining, *Journal of Materials Processing Technology*, Vol. 149, Issues 1-3, 2004, pp. 426-431.

[4]. B. Bhattacharyya, S. Mitra, A. K. Boro, Electrochemical machining: new possibilities for micromachining, *Robotics and Computer-Integrated Manufacturing*, Vol. 18, Issues 3-4, 2002, pp. 283-289.

[5]. D. Rajkumar, K. Palanivelu, Electrochemical treatment of industrial wastewater, *Journal of Hazardous Materials*, Vol. 113, Issues 1-3, 2004, pp. 123-129.

[6]. C. Trellu, H. O. Vargas, E. Mousset, N. Oturan, M. A. Oturan, Electrochemical technologies for the treatment of pesticides, *Current Opinion in Electrochemistry*, Vol. 26, 2021, 100677.

[7]. A. Reza, L. Chen, Electrochemical treatment of livestock waste streams. A review, *Environmental Chemistry Letters*, Vol. 20, Issues 3, 2022, pp. 1863-1895.

[8]. K. Kurc, A. Burghardt, P. Gierlak, M. Muszyńska, D. Szybicki, A. Ornat, M. Uliasz, Application of a 3D scanner in robotic measurement of aviation components, *Electronics*, Vol. 11, 2022, 3216.

[9]. A. Burghardt, D. Szybicki, P. Gierlak, K. Kurc, M. Muszyńska, Robotic grinding process of turboprop engine compressor blades with active selection of contact force, *Tehnički Vjesnik*, Vol. 29, Issue 1, 2022, pp. 15-22.

[10]. D. Szybicki, A. Burghardt, K. Kurc, P. Gierlak, Device for contact measurement of turbine blade geometry in robotic grinding process, *Sensors,* Vol. 20, 2020, 7053.

[11]. D. Zhu, X. Feng, X. Xu, Z. Yang, W. Li, S. Yan, H. Ding. Robotic grinding of complex components: A step towards efficient and intelligent machining-challenges, solutions, and applications, *Robotics and Computer-Integrated Manufacturing*, Vol. 65, 2020, 101908.

[12]. F. Leali, A. Vergnano, F. Pini, M. Pellicciari, G. Berselli, A workcell calibration method for enhancing accuracy in robot machining of aerospace parts, *The International Journal of Advanced Manufacturing Technology*, Vol. 85, 2016, pp. 47-55.

[13]. X. Ren, B. Kuhlenkötter, H. Müller, Simulation and verification of belt grinding with industrial robots, *International Journal of Machine Tools and Manufacture*, Vol. 46, Issues 7-8, 2006, pp. 708-716.

[14]. X. Y. Ren, H. Mueller, B. Kuhlenkoetter, Surfel-based surface modeling for robotic belt grinding simulation, *Journal of Zhejiang University-Science,* Vol. A7, 2006, pp. 1215-1224.

[15]. X. Ren, M. Cabaravdic, X. Zhang, B. Kuhlenkötter, A local process model for simulation of robotic belt grinding, *International Journal of Machine Tools and Manufacture*, Vol. 47, Issue 6, 2007, pp. 962-970.

[16]. S. Nahavandi, H. J. Uddin, Y. Nasu, H. Trinh, M. Saadat, Automated robotic grinding by low-powered manipulator, *Robotics and Computer-Integrated Manufacturing*, Vol. 23, Issue 5, 2007, pp. 589-598.

[17]. S. Wu, K. Kazerounian, Z. Gan, Y. Sun, A simulation platform for optimal selection of robotic belt grinding system parameters, *The International Journal of Advanced Manufacturing Technology*, Vol. 64, 2013, pp. 447-458.

[18]. F. Rafieian, B. Hazel, Z. Liu, Vibro-impact dynamics of material removal in a robotic grinding process, *The International Journal of Advanced Manufacturing Technology*, Vol. 73, 2014, pp. 949-972.

[19]. G. Zhihui, L. Xiaodong, B. Yushu, Structural dimension optimization of robotic belt grinding system for grinding workpieces with complex shaped surfaces based on dexterity grinding space, *Chinese Journal of Aeronautics*, Vol. 24, Issue 3, 2011, pp. 346-354.

[20]. Y. Lin, H. Zhao, H. Ding, Posture optimization methodology of 6R industrial robots for machining using performance evaluation indexes, *Robotics and Computer-Integrated Manufacturing*, Vol. 48, 2017, pp. 59-72.

[21]. A. Burghardt, K. Kurc, D. Szybicki, M. Muszyńska, J. Nawrocki, Software for the robot-operated inspection station for engine guide vanes taking into consideration the geometric variability of parts, *Tehnički Vjesnik,* Vol. 24*,* Issue 2*,* 2017, pp. 349-353.

[22]. J. Li, J. Zhu, Y. Guo, X. Lin, K. Duan, Y. Wang, Q. Tang, Calibration of a portable laser 3-D scanner used by a robot and its use in measurement, *Optical Engineering*, Vol. 47, Issue 1, 2008, 017202.

[23]. Y. Sun, D. J. Giblin, K. Kazerounian, Accurate robotic belt grinding of workpieces with complex geometries using relative calibration techniques, *Robotics and Computer-Integrated Manufacturing*, Vol. 25, Issue 1, 2009, pp. 204-210.

[24]. X. Xu, D. Zhu, H. Zhang, S. Yan, H. Ding, TCP-based calibration in robot-assisted belt grinding of aero-engine blades using scanner measurements, *The International Journal of Advanced Manufacturing Technology*, Vol. 90, 2017, pp. 635-647.

[25]. W. L. Li, L. P. Zhou, S. J. Yan, A case study of blade inspection based on optical scanning method, *International Journal of Production Research*, Vol. 53, Issue 7, 2015, pp. 2165-2178.

[26]. W. L. Li, A. Wu, Z. C. Li, G. Zhang, W. Y. Yu, A new calibration method between an optical sensor and a rotating platform in turbine blade inspection, *Measurement Science and Technology*, Vol. 28, Issue 3, 2017, 035009.

[27]. D. Szybicki, A. Burghardt, K. Kurc, P. Pietruś, Calibration and verification of an original module measuring turbojet engine blades geometric parameters, *Archive of Mechanical Engineering*, Vol. 66, 2019, pp. 97-109.

[28]. H. Huang, Z. M. Gong, X. Q. Chen, L. Zhou, Robotic grinding and polishing for turbine-vane overhaul, *Journal of Materials Processing Technology*, Vol. 127, Issue 2, 2002, pp. 140-145.

[29]. W. Wei, Y. Chao, A path planning method for robotic belt surface grinding, *Chinese Journal of Aeronautics*, Vol. 24, Issue 4, 2011, pp. 520-526.

[30]. T. Zhang, J. Su, Collision-free planning algorithm of motion path for the robot belt grinding system, *International Journal of Advanced Robotic Systems*, Vol. 15, Issue 4, 2018, 1729881418793778.

[31]. Y. Song, W. Liang, Y. Yang, A method for grinding removal control of a robot belt grinding system, *Journal of Intelligent Manufacturing*, Vol. 23, 2012, pp. 1903-1913.

[32]. S. Yixu, L. Hongbo, Y. Zehong, An adaptive modeling method for a robot belt grinding process, *IEEE/ASME Transactions on Mechatronics*, Vol. 17, Issue 2, 2011, pp. 309-317.

[33]. A. Ornat, M. Uliasz, G. Bomba, A. Burghardt, K. Kurc, D. Szybicki, Robotised geometric inspection of thin-walled aerospace casings, *Sensors,* Vol. 22*,* 2022, 3457.

[34]. T. Zhang, Y. Yu, Y. Zou, An adaptive sliding-mode iterative constant-force control method for robotic belt grinding based on a one-dimensional force sensor, *Sensors*, Vol. 19, Issue 7, 2019, 1635.

[35]. A. E. K. Mohammad, J. Hong, D. Wang, Design of a force-controlled end-effector with low-inertia effect for robotic polishing using macro-mini robot approach, *Robotics and Computer-Integrated Manufacturing*, Vol. 49, 2018, pp. 54-65.

[36]. X. Xiaohu, Z. Dahu, H. Zhang, Y. Sijie, D. Han, Application of novel force control strategies to enhance robotic abrasive belt grinding quality of aero-engine blades, *Chinese Journal of Aeronautics*, Vol. 32, Issue 10, 2019, pp. 2368-2382.

[37]. F. Chen, H. Zhao, D. Li, L. Chen, C. Tan, H. Ding, Robotic grinding of a blisk with two degrees of freedom contact force control, *The International Journal of Advanced Manufacturing Technology*, Vol. 101, 2019, pp. 461-474.

[38]. H. Junhui, W. Zhao, G. Jianmin, Y. Yu, Overview on the profile measurement of turbine blade and its development, *Proceedings of SPIE*, Vol. 7656*,* 2010, 76560L.

[39]. B. Sun, B. Li, Laser displacement sensor in the application of aero-engine blade measurement, *IEEE Sensors Journal*, Vol. 16, Issue 5, 2015, pp. 1377-1384.

[40]. N. Vukašinović, D. Bračun, J. Možina, J. Duhovnik, The influence of incident angle object colour and distance on CNC laser scanning, *The International Journal of Advanced Manufacturing Technology*, Vol. 50, Issues 1, 2010, pp. 265-274.

[41]. K. C. Fan. A non-contact automatic measurement for free-form surface profiles, *Computer Integrated Manufacturing Systems*, Vol. 10, Issues 4, 1997, pp. 277-285.

[42]. N. Van Gestel, S. Cuypers, P. Bleys, J. P. Kruth, A performance evaluation test for laser line scanners on CMMs, *Optics and Lasers in Engineering*, Vol. 47, Issue 3, 2009, pp. 336-342.

[43]. H. Y. Feng, Y. Liu, F. Xi, Analysis of digitizing errors of a laser scanning system, *Precision Engineering*, Vol. 25, Issue 3, 2001, pp. 185-191.

[44]. T. Miyasaka, H. Okamura, Dimensional change measurements of conventional and flowable composite resins using a laser displacement sensor, *Dental Materials Journal*, Vol. 28, Issue 5, 2009, pp. 544-551.

[45]. D. Bračun, V. Gruden, J. Možina, A method for surface quality assessment of die-castings based on laser triangulation, *Measurement Science and Technology*, Vol. 19, Issue 4, 2008, 045707.

[46]. C. Hu, C. Tanaka, T. Ohtani, Locating and identifying splits and holes on SUGI by the laser displacement sensor, *Journal of Wood Science*, Vol. 49, Issue 6, 2003, pp. 492-498.

[47]. J. Sandak, C. Tanaka, T. Ohtani, Evaluation of surface smoothness by a laser displacement sensor II: Comparison of lateral effect photodiode and multielement array, *Journal of Wood Science*, Vol. 50, Issues 1, 2004, pp. 22-27.

[48]. S. Oya, Measurement of the vibrating shape of a bimorph deformable mirror using a laser displacement sensor, *Optical Engineering*, Vol. 48, Issue 3, 2009, 033601.

[49]. B. Li, F. Li, H. Liu, H. Cai, X. Mao, F. Peng, A measurement strategy and an error-compensation model for the on-machine laser measurement of large-scale free-form surfaces, *Measurement Science and Technology*, Vol. 25, Issue 1, 2014, 015204.

[50]. M. Zhang, T. Chen, Y. Tan, J. Lu, An adaptive grinding method for precision-cast blades with geometric deviation. *The International Journal of Advanced Manufacturing Technology*, Vol. 108, 2020, pp. 2349-2365.

[51]. G. Xiao, S. Liu, Y. Zhang, Y. Wu, B. Chen, S. Song, A measurement method of the belt grinding allowance of hollow blades based on blue light scanning, *The International Journal of Advanced Manufacturing Technology*, Vol. 116, 2021, pp. 3295-3303.

[52]. A. Burghardt, K. Kurc, D. Szybicki, M. Muszyńska, T. Szczęch, Robot-operated inspection of aircraft engine turbine rotor guide vane segment geometry, *Technical Gazette*, Vol. 24, Issue 2, 2017, pp. 345-348.

[53]. M. Fan-Jun, L. Xun, C. Zhi-Tong, W. Xiao-Wen, Study on the cantilever grinding process of aero-engine blade, *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, Vol. 228, Issue 11, 2014, pp. 1393-1400.

(019)

Velocity Time of Arrival in the Tip-timing Analysis of Steam Turbine Rotor Blades

R. Rzadkowski [1,2] **and J. Manerowski** [1]
[1] Department of Aircraft Engines, Air Force Institute of Technology, Warsaw 01-494,
Ksiecia Boleslawa 6, Poland
[2] Aeroelasticity Department, Institute of Fluid-Flow Machinery Polish Academy of Sciences,
Gdańsk 80-231, Fiszera 14, Poland
Tel.: + 48585225169
E-mail: z3@imp.gda.pl

Summary: The rotor blade velocity obtained from the measured time of rotor blade arrival is used in the functional of the Least Square Technique, which is a novelty. An additional technique has been devised to determine the frequencies of blade vibration coupled modes, which is also a novelty. The approach requires only two sensors in the casing for the algorithm to find blade multi-mode vibration components. For comparison, three and four sensors have also been used. Verification of the algorithm is based on experimental and numerical analyses of steam turbine Low Pressure last stage rotor blades vibrating during run-down in a vacuum spin chamber, for synchronous vibrations, and during nominal work at 3000 rpm in a real working turbine for asynchronous vibrations. A multimode analysis is carried out for blade vibration at a nominal rotation speed of 3000 rpm. The considered steam turbine rotor blade is found to vibrate with two-mode shapes.

Keywords: Tip-timing, Steam turbine, Last squares technique.

---

1. Introduction

Blade tip-timing (BTT) is a non-contact method for measuring blade vibration. Tip-timing methods were presented at ASME Turbo-Expo 2012, Copenhagen [1], Rzadkowski et al. [2].

The first tip-timing models assumed that each blade vibrates with one frequency [3].

Ao et al. [4] proposed a multimode model of the BTT signal. The signal of the blade was decoupled and recovered as of single-mode response.

In our paper, the Least Squares Technique is used to find the multimode of synchronous and nonsynchronous blade vibrations from the tip-timing velocity time of arrival, which is a novelty. The measured time of rotor blade arrival is used to find the rotor blade velocities. The rotor blade velocities are used in the functional of the Least Square Technique. A new technique is presented to determine the frequencies of the coupled modes in the blade vibrations, which is also a novelty.

2. Methodology

The presented model uses the j-th blade arrival times ($t_{1,n}$, $t_{2,n}$, …, $t_{l,n}$, $t_{l+1,n}$, …, $t_{s,n}$), where s is the number of sensors and n is the total number of rotations. The angles between the probes determine the actual velocity of the blade tips (Fig. 1).

The linear velocity of the vibrating j-th blade tip $V_c(t)$ is the sum of rotor velocity $V_\Omega(t)$ and the velocity of rotor blade vibration.

The measured velocities of blade vibration v in ($t_{1,n}$, $t_{2,n}$, …, $t_{j,n}$, $t_{j+1,n}$, …, $t_{l,n}$), rotation velocity Ω_n in successive n rotations (n = 1, 2, …, N, N is the total number of rotation), between l^{th} and $(l+1)^{th}$ sensors mounted in the casing are:

$$\begin{gathered} t_{v1,n} = (t_{2,n} + t_{1,n})/2 \quad v_{1,n} = R\,(\varkappa_2 - \varkappa_1)/(t_{2,n} - t_{1,n}) - \Omega_n R, \\ t_{v2,n} = (t_{3,n} + t_{2,n})/2 \quad v_{2,n} = R\,(\varkappa_3 - \varkappa_2)/(t_{3,n} - t_{2,n}) - \Omega_n R, \\ \ldots, \\ t_{vl,n} = (t_{l+1,n} + t_{l,n})/2 \quad v_{l,n} = R\,(\varkappa_{l+1} - \varkappa_l)/(t_{l+1,n} - t_{l,n}) - \Omega_n R \end{gathered} \qquad (1)$$

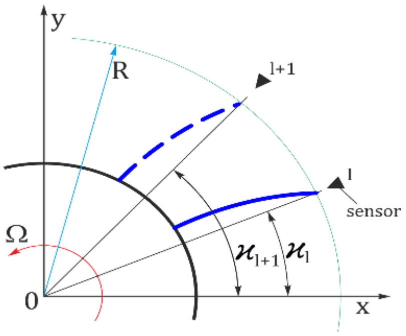


Fig. 1. Blade tip position.

The assumed blade vibration velocity is:

$$V(t_v) = \sum_{i=1}^{k}\left(a_i \sin(\omega_i\, t_v) + b_i \cos(\omega_i\, t_v)\right) = \mathbf{s}\,\mathbf{a}, \qquad (2)$$

where $t_v = (t_{v1,n}, t_{v2,n}, \ldots, t_{v(s-1),n})$, s-number of sensors, k is the number of modes.

For example, $t_v = t_{v1,1}$ (blade velocity time between sensor 1 and 2 in first rotation)

$$\begin{gathered} \mathbf{s} = [\sin(\omega_1 t_{v1,1}), \cos(\omega_1 t_{v1,1}), \sin(\omega_2 t_{v1,1}), \\ \cos(\omega_2 t_{v1,1}), \ldots, \sin(\omega_k t_{v1,1}), \cos(\omega_k t_{v1,1})], \\ \mathbf{a} = [a_1, b_1, a_2, b_2, \ldots, a_k, b_k]^T \end{gathered} \qquad (3)$$

The blade displacement is found by integrating Eq. (2):

$$X(t_v) = C+\sum_{i=1}^{k}(-a_i(\omega_i)^{-1}\cos(\omega_i\, t_v) + + b_i(\omega_i)^{-1}\sin(\omega_i\, t_v)), \tag{4}$$

where C is the constant, in the case of calculation of vibration parameters C = 0.

To find vibration velocities, V Eq. (2) for ($t_{v1,n}$, $t_{v2,n}$, …, $t_{v(s-1),n}$) matrix **a** must be found for (n = 1, 2, …, N).

To find matrix **a,** the Least Squares Technique uses functional F for various t_{vm}:

$$F = \sum_1^M (V(t_{vm}) - v(t_{vm}))^2, \tag{5}$$

where $V(t_{vm})$ is the blade tip vibration velocity and $v(t_{vm})$ is the measured velocity, M = N *(s-1).

Taking into account Eq. (2) and Eq. (1) and assuming that the first derivative of F(**a**) is zero:

$$\mathbf{a} = (\sum_1^M s_m^T)^{-1}\,(\sum_1^M s_m^T\, v_m) \tag{6}$$

The blade amplitude can be found by integrating Eq.(2):

$$X(t_{vm}) = C + \mathbf{s}_X(t_{vm})\,\mathbf{a}, \tag{7}$$

where

$$\mathbf{s}_X = \mathbf{s}_X(t_{vm}) = [-\omega_1^{-1}\cos(\omega_1\, t_{vm}), \omega_1^{-1}\sin(\omega_1\, t_{vm}), -\omega_2^{-1}\cos(\omega_2\, t_{vm}), \omega_2^{-1}\sin(\omega_2\, t_{vm}),\ldots, -\omega_k^{-1}\cos(\omega_k\, t_{vm}), \omega_k^{-1}\sin(\omega_k\, t_{vm})], \tag{8}$$

and C are the unknown values.

The presented algorithm requires the number of blade vibration components to be assumed (Eq. (2)). Therefore, the first thing to do is to use the measured times of blade arrival to determine the number of harmonics. The presented method of determining the number of coupled modes and frequencies in the mode of blade vibrations is also a novelty. It is assumed that the blade vibration velocity is:

$$V(t_v) = a \sin(\omega\, t_v) + b \cos(\omega\, t_v) \tag{9}$$

By varying the frequencies ω and using the presented here algorithm, the blade amplitude A_w may be found for various ω, where local extremes can be identified, which is a novelty. These local extremes indicated the number of blade vibration components and their frequencies in vibration velocity. The same algorithm can be used with these frequencies to calculate the blade amplitude and phase of each mode.

3. Experimental Results for the Nominal Regime

The experiment was carried out in the last stage of a 380 MW low-pressure steam turbine at Ω = 50 Hz rotation speed in a nominal regime to measure asynchronous blade vibration. Three sensors were placed in the inner casing at 0°, 10° and 15°. With the Least Squares Technique, two velocity times of arrival can be found using three sensors. The numerical velocity amplitude and phase are found for 50 rotations and four mode components. These components are found according to the above methodology for determining blade vibration frequencies. The four nondimensional frequencies were found: 3.5, 5.47, 7.89, and 9.95 for varying the frequencies ω up to 10 in Eq. (9) using the presented algorithm.

The blade amplitudes, velocities, and phases for three sensors are calculated (for k = 2, 3, 4 (Eq. (2)), which shows that the last stage rotor blade vibrates with dominated non-dimensional frequencies of 3.5 (0.37 mm amplitude) and 5.47 (0.24 amplitude). The other frequencies have smaller amplitudes: 7.89 (0.06 mm) and 9.95 (0.02 mm) (see Tables 1, 2). This shows that the rotor blades vibrate predominately with two components.

The above method can identify the component modes of vibration in steam turbine LP last stage blades based on experimental measurements using three sensors.

Fig. 2 presents blade vibration velocity V (blue line) and displacement X (red line) obtained using the least squares technique with three sensors. The maximal blade velocity V is 0.61 m/s and the maximum displacement is 0.62 mm with simultaneous vibration non-dimensional frequencies of 3.5 and 5.47. The experimental v is marked with green crosses. Here, the numerical values V differ slightly from the experimental ones. This is because there are not in resonance vibrations and, therefore, the blade displacements are smaller (0.62 mm, Fig. 2) than those of blades in resonance (3.97 mm).

Small blade displacements cause considerable measurement errors. On the other hand, the numerical results using the single-degree-of-freedom fitting method for the measured time of blade arrival provide a blade non-dimensional frequency of 3.53 and an amplitude of 0.7 mm, which is close to 0.62. The numerical results compare well with some modes obtained from other single-mode numerical analyses. However, our results also show that additional blade vibration components may appear.

The above results are for analyses using three sensors. Table 1 presents the blade amplitudes and phases and the number of blade harmonics k = 4, Table 2 for k = 2. The angle between them is 6°. Table 3 presents the blade amplitudes and phases using only two sensors and the number of blade harmonics k = 4, Table 4 for k = 2. The angle between them is 6°.

Tables 3, 4 and Tables 1, 2 show that non-dimensional frequencies for two sensors in comparison with three sensors are ±1 % different. Therefore, for two sensors, the accuracy of calculations is very good and comparable with results using three sensors. Tables 1, 2 and 3, 4 clearly show that, in this particular case, blades vibrate with two harmonics.

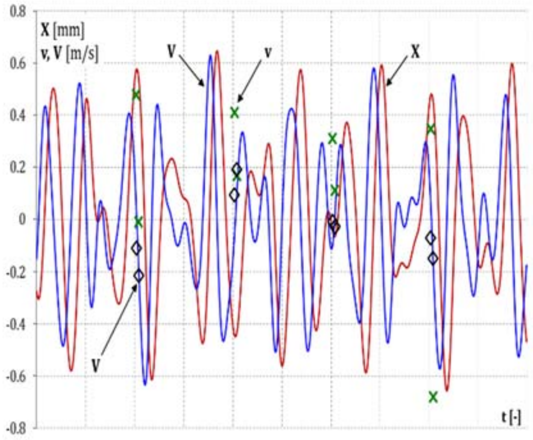


Fig. 2. Numerical vibration velocity V, experimental velocity v and blade displacements X for blade vibration and rotor velocity during nominal work at Ω = 50 Hz.

Table 1. Velocity amplitude A, velocity phase φ, and blade amplitude A_x for k = 4 with three sensors.

ω (–)	A (m/s)	φ (deg)	A_i (mm)
3.5	0.27	57	0.37
5.47	0.28	83.4	0.24
7.89	0.10	51.4	0.06
9.95	0.04	45.5	0.02

Table 2. Velocity amplitude A, velocity phase φ, and blade amplitude A_x for k = 2 with three sensors.

ω (–)	A (m/s)	φ (deg)	A_i (mm)
3.5	0.26	42.3	0.36
5.47	0.28	85.2	0.21

Table 3. Velocity amplitude A, velocity phase φ, and blade amplitude A_x for k = 4 with two sensors.

ω (–)	A (m/s)	φ (deg)	A_i (mm)
3.51	0.25	47	0.34
5.46	0.23	61.4	0.21
7.90	0.10	59.4	0.07
9.97	0.05	46.5	0.03

Table 4. Velocity amplitude A, velocity phase φ, and blade amplitude A_x for k = 2 with two sensors.

ω (–)	A (m/s)	φ (deg)	A_i (mm)
3.51	0.24	52.3	0.32
5.46	0.21	91.1	0.19

4. Conclusions

This paper presents a multimode tip-timing algorithm for the vibration analysis of steam and gas turbine rotor blades using the Least Squares Technique. The rotor blade velocity obtained from the measured time of rotor blade arrival is used in the functional of the Least Square Technique, which is a novelty. This method requires only two sensors in the casing for multimode synchronous and asynchronous vibrations.

The presented algorithm requires the number of blade vibration components to be assumed. Therefore, the first thing to do is to use the measured times of blade arrival to determine the number of harmonics. The presented method of determining the number of coupled modes and frequencies in the mode of blade vibrations is also a novelty. The same algorithm can be used with these frequencies to calculate the blade amplitude and phase of each mode.

Acknowledgments

The authors wish to acknowledge NCBiR for the financial support of this work (POIR.04.01.04-00-0116/17).

References

[1]. P. Russhard, Perspective on Tip-Timing, Panel Session, Tip-Timing: State-of-the-art and future developments, in *Proceedings of the ASME TURBO-EXPO,* Denmark, Copenhagen, June 11-15, GT2012-70187, 2012.

[2]. R. Rzadkowski, E. Rokicki, L. Piechowski, R. Szczepanik, Analysis of middle bearing failure in rotor jet engine using tip-timing and tip-clearance technique, *Mechanical Systems and Signal Processing*, Vols. 76-77, 2016, pp. 213-227.

[3]. S. Heath, M. Imregun, An improvement single-parameters tip-timing method for turbomachinery blade vibration measurements using optical laser probes, *International Journal of Mechanical Sciences*, Vol. 38, Issue 10, 1996, pp. 1047-1058.

[4]. C. Ao, B. Qiao, M. Liu, W. Zhu, Y. Zhu, Y. Wang, X. Chen, Non-contact full-field dynamic strain reconstruction of rotating blades under multi-mode vibration, *Mechanical System and Signal Processing*, Vol. 186, 2023, 109840.

(020)

Boosting Robustness and Continual Learning in Spiking Neural Networks through Dendrite Dynamics Optimization

Mingkun Xu, Jianping Xiong , Jing Pei and Lei Deng
Center for Brain-Inspired Computing Research (CBICR), Department of Precision Instrument,
Tsinghua University, Beijing, China
E-mail: {peij,leideng}@mail.tsinghua.edu.cn

Summary: Artificial neural networks (ANNs) have demonstrated impressive performance in various applications, but are limited by their simplified models of neurons and synapses. Recent research has shown that incorporating dendrite functionality into ANNs can improve their expressivity and learning efficiency. However, most of work has focused on rate-based models of ANNs. In this paper, we propose a novel framework for incorporating dendrite functionality in spiking neural networks (SNNs), which are more biologically realistic and interpretable. We evaluate the effect of dendritic processing in SNNs using pyramidal neurons, a common type of neuron in the cerebral cortex. Our results show that the incorporation of dendritic processing in SNNs leads to a significant improvement in robustness and continual learning efficiency, enabling SNNs to approach the performance of biological neural networks with prominent interpretability.

Keywords: Dendrite dynamics, Spiking neural networks, Pyramidal neuron, Continual learning, Robustness.

1. Introduction

Spiking neural networks (SNNs) are a class of neural networks that use spikes, rather than continuous values, to communicate between neurons. SNNs have shown great promise in various applications, such as robotics, signal processing, and pattern recognition. However, one of the major challenges in SNNs is to ensure robustness and continual learning in the face of changing environments and input data.

To address this challenge, we propose a novel approach that integrates dendrite dynamics of pyramidal neurons into SNNs (dubbed DenNet) to boost their robustness and enable continual learning. Dendrites are the input branches of neurons that receive signals from other neurons. The dynamics of dendrites play a crucial role in shaping the response of neurons to input signals and modulating the strength of synaptic connections. Our approach focuses on optimizing the hierarchical dynamics of dendrites and non-monotonic activation function to enhance the adaptability and resilience of SNNs. Specifically, we introduce a set of dendrite dynamics that enable SNNs to adapt to changing input distributions and recover from neuron failures, based on a biologically-inspired framework that incorporates the principles of synaptic plasticity and homeostasis.

We demonstrate the effectiveness of our approach on neuromorphic MNIST (N-MNIST) and Shuffled MNIST (S-MNIST) benchmark datasets [1,2]. Our results show significant improvements in classification accuracy, robustness to perturbation attacks, and continual learning performance compared to state-of-the-art SNNs. We also conduct analysis experiments to investigate the hidden representation and neural activity to obtain better interpretability.

2. Methods

The dendritic extension is functionalized to be able to integrate synaptic inputs not only spatially but also temporally with certain degree of leakage over time. We model the dynamic process by:

$$\tau_d \frac{dV_d^{i,k}(t)}{dt} = -\left(V_d^{i,k}(t) - V_{d\_reset}\right) + \sum_{j=1}^{N_{L-1}^{i,k}} w^{ij,k}(t) s^{ij,k}(t) \tag{1}$$

$$s^{ij,k}(t) = \sum_{t_F^{ij,k}<t} \delta\left(t - t_F^{ij,k}\right) \tag{2}$$

where $w^{ij,k}$ denotes the weight of the synapse connecting pre-neuron j and post-neuron i and harbouring at the kth dendritic branch of post-neuron i, $V^{i,k}_d$ is the membrane potential of the dendritic branch, V_{d_rest} is the resting potential of dendrite that is set to zero in our simulations, τ_d is the time constant of dendritic membrane potential leakage, and $s^{ij,k}(t)$ is the spike train input from pre-neuron j in sequence of firing time $t^{ij,k}_F$ to the kth dendritic branch of post-neuron i (eq. 2). Individual spikes arrived at the kth dendritic branch of post-neuron i within each minimum time step Δt from $N^{i,k}_{L-1}$ pre-neurons in layer $L-1$ are weighted at the corresponding synapses and linearly summed up to provide total input to dendrite and then integrated into the instantaneous dendritic membrane potential $V^{i,k}_d(t)$.

We model the dynamic behaviour of dendrite by refering to the leaky integration behaviour of the neuron cell body. Apparent differences between dendrite and the neuron cell body occur in the ensuing activation or nonlinear transfer process. In pyramidal neurons, dendrite nonlinearly transforms the spatio-temporally integrated input through an activation function, i.e., DA. In contrast to the more familiar neuronal activation that is commonly abstracted as a monotonic function with respect to the integrated input, such as sigmoid and ReLU functions, dendritic activation is simulated by a non-monotonic Gaussian function:

$$V_{DA}^{i,k}(t) = \mathrm{G}\left(V_d^{i,k}(t)\right) = \frac{A}{\sigma\sqrt{2\pi}} e^{\frac{-(V_d^{i,k}(t)-\mu)^2}{2\sigma^2}} \qquad (3)$$

where $V^{i,k}{}_{DA}$ denotes dendritic output after Gaussian activation, μ is position of the peak and is set to the activation threshold (V_{th}) of neuron, σ controls the width and the coefficient A controls the height of the peak. This process mimics the newly found dCaAP [3]. In this way, dendrite responds weakly to small and large inputs, but strongly to moderate ones. Immediately following dendritic processing, leaky integrate-and-fire (LIF) neural processing (eq.4 and 5 where $V^i{}_n$ denotes neuronal membrane potential, τ_n is the time constant of neuronal membrane potential leakage, V_{n_rest} is the resting potential of neuron that is set to zero in our simulations, D^i is the number of dendritic branches of neuron i, and H is Heaviside step function) begins, by integrating dendritic output $V^{i,k}{}_{DA}$ into the instantaneous neuronal membrane potential $V^i{}_n(t)$.

$$\tau_n \frac{dV_n^i(t)}{dt} = -\left(V_n^i(t) - V_{n\_rest}\right) + \sum_{k=1}^{D^i} V_{DA}^{i,k}(t) \qquad (4)$$

$$V_{nAF}^i(t) = H(V_n^i(t) - V_{th}) \qquad (5)$$

When $V^i{}_n(t)$ exceeds a certain threshold V_{th}, neuron emits a spike (activated) and its membrane potential is spontaneously reset to V_{n_rest}, ready to be activated again. The network is trained with spatio-temporal back-propagation algorithm (STBP) via gradient substitution [4].

3. Results

To benchmark the robustness of our proposed DenNet to noise perturbations, we train a three-layer model on intact neuromorphic MNIST (N-MNIST) dataset for 100 epochs and test it using salt & pepper noise (SP) perturbed test samples. In N-MNIST, we accumulated spike trains (5ms) for acceleration and directly input them into SNNs. In each trial, the models are initialized by a uniform initialization and trained by minimizing the mean square error (MSE) loss on the 50000 training samples, and tested the accuracy by using 1000 noisy testing samples. For all SNN models, we set the threshold V_{th} as 0.75, set the soma decay factor k_s as 0.75, and set dendrite decay factor k_d as 0.1 to achieve their favourable performance; The time window T is set as 10 for all models. We adopt the three-layer structure [2312-FC512-FC10] and use Adam optimizer to train. After training process on clean N-MNIST data, we validated the model robustness in the noise experiments. We controlled the noise intensity by control the noise level variable (r), which refers to the $r \times 2e^{-2}$ for the proportion of noise region, ranging from 1~14. All models were pre-trained on the standard training dataset and tested on the noise dataset with the same parameter configurations and the MSE loss, where the results are shown as Fig. 1a. For Cosine distance comparison, we used the averaged membrane potential of the second hidden layer across all timesteps as the learnt representations to calculate the distance between noise corrupted samples and intact ones. We randomly sampled 1000 testing data from N-MNIST and plotted its statistical average distance on Fig. 1b. Besides, we calculated the firing rate of three layers across different noise levels, shown in Fig. 1e.

We utilize the shuffled MNIST (S-MNIST) to conduct continual learning experiments. All tasks are to classify handwritten digits from zeros to nine. For each new task, the image pixels were randomly permuted with the same randomization across all digits in the same task and different randomization over different tasks. We trained each task by ten epochs and used a three-layer spiking neural networks with [784-FC1024-FC10] structure to minimize the MSE by the ADMM. During the training of each task for these three SNN models, we randomly generated a sparse and fixed connection matrix to receive supervision signals, where the sparsity is kept as 3%. For the configuration of SNN models, we set the time window T as 5, set the threshold V_{th} as 0.5, and set the soma decay factor k_s as 0.7. The dendrite decay factor k_d is set to 0.001. After each task is learned, we fixed the weights and randomly took validation task samples from the learned task set to test the capability for exploiting historical information. We trained each model with 50 epochs and 4 trials and presented the results in Fig. 1 c, d. In addition, we visualized the patterns of learnt neural representation in pre-trained models across all tasks in Fig.1f-h. The patterns are calculated by the averaged membrane potentials of layer-2 neurons across all timesteps, then the neurons are clustered by K-means algorithm to reveal the different modes in these SNN models.

The superiority of noise-robustness in DenNet, compared to other models with different dendritic activations, is illustrated in Fig.1a. Furthermore, the comparison of neuron membrane potential similarity between the two test sets reveals an increasing trend from the baseline SNN to DenNet, as depicted in Fig.1b. The enhanced noise resilience of DenNet can be attributed to its Gaussian dendritic activation, which functions as a band-pass filter. Specifically, it effectively filters out noise that disturbs the neuron membrane potential in either direction, thereby preventing overactivity or underactivity of the neuron. This is supported by the findings presented in Fig.1e, where an analysis of neuronal firing patterns in both SNNs, subjected to inputs contaminated by varying levels of noise, suggests that the computational properties of SNNs are closely tied to their code transmission properties, exhibiting adaptivity.

Fig. 1 c,d present a comparison of the fifty-task continual learning performance of multiple SNN models. The results clearly indicate that DenNet consistently outperforms the other models as the number of learned tasks increases, with the simple baseline SNN exhibiting the poorest performance.

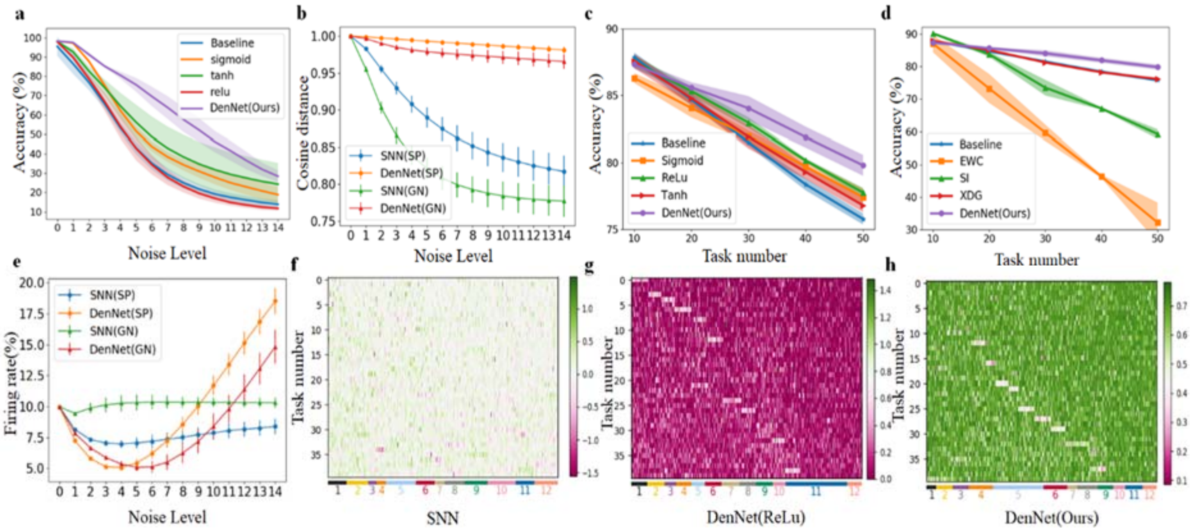


Fig. 1. a. Test accuracies on salt & pepper noise perturbed N-MNIST data as functions of noise level for SNN models with different dendrite activation. The curves (shadows) are mean accuracies (standard deviations) over four test trials. **b.** The Cosine distances with respect to the average layer-2 neuron membrane potential between tests on noise corrupted samples and tests on intact ones for these three SNN models. The error bars are standard deviations over four test trials. **c,d.** Test accuracies averaged over tests on all previously learnt tasks on S-MNIST as functions of the number of tasks for SNN models with different activations and continual learning techniques, respectively. The curves (shadows) are mean accuracies (standard deviations) over four test trials. **e.** The layer-2 neuron firing rates as functions of noise level. **f-h.** Learnt neural representations across all tasks for three SNN models. Neurons are sorted by their cluster membership, indicated by coloured lines at the bottom. K-means clustering method based on task variances is used.

Notably, Fig. 1f-h reveal distinct characteristics between the baseline SNN and the two DenNet models. In the case of the baseline SNN, no evident clustering of neurons is observed, implying that neural resources are shared across different tasks. Additionally, the overall neural activity is relatively low, with most neurons maintaining membrane potentials close to their resting-state values (zero), as depicted in Fig. 1f-h. In contrast, both DenNet and DenNet(ReLu) (where the non-monotonic dendritic activation is replaced by ReLu function) exhibit a degree of self-organization in layer-2 neurons, forming distinct clusters through continual learning. This clustering indicates that neurons become task-specific, resulting in reduced representational overlap and mitigating the catastrophic forgetting problem. While both DenNet models feature a majority of non-clustered neurons serving as a background for each task, they differ in terms of the relative activity levels of clustered neurons compared to the background activity. Specifically, the clusters of neurons in DenNet display higher levels of activity than the remaining task-irrelevant neurons, which are nearly at rest. In contrast, the neuron clusters in DenNet exhibit lower levels of activity than the non-clustered neurons. This observation suggests that these networks are trained to form task-specific neuron clusters characterized by activity contrasts with their respective backgrounds. This allows for reliable propagation and decoding of input spike rates encoding specific tasks in the decision layer (layer 3), effectively filtering out irrelevant information.

4. Conclusions

In this paper, we have proposed a novel approach that optimizes dendrite dynamics in SNNs to boost their robustness and enable continual learning, based on a biologically-inspired framework that incorporates the principles of synaptic plasticity and homeostasis. We have demonstrated the effectiveness of our approach on benchmark datasets, showing significant improvements in classification accuracy, robustness to perturbation attacks, and continual learning performance compared to state-of-the-art SNNs. Our results suggest that optimizing dendrite dynamics is a promising approach to enhance the performance and reliability of SNNs in real-world scenarios.

Acknowledgements

This work was supported by Science and Technology Innovation 2030 - New Generation of Artificial Intelligence, China project (2020AAA0109101) and National Natural Science Foundation of China (No. 62106119, 62276151) and Zhejiang Lab's International Talent Fund for Young Professionals.

References

[1]. G. Orchard, A. Jayawant, G. K. Cohen, N. Thakor, Converting static image datasets to spiking neuromorphic datasets using saccades, *Front. Neurosci*, 9, 2015, 00437.

[2]. S. W. Lee, J. H. Kim, J. Jun, J. W. Ha, B. T. Zhang, Overcoming catastrophic forgetting by incremental moment matching, in *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017),* Long Beach, CA, USA, 2017.

[3]. A. Gidon, T. A. Zolnik, P. Fidzinski, F. Bolduan, A. Papoutsi, P. Poirazi, M. Holtkamp, I. Vida, M. E. Larkum, Dendritic action potentials and computation in human layer 2/3 cortical neurons, *Science,* 367, 2020, pp. 83-87.

[4]. Y. Wu, L. Deng, G. Li, J. Zhu, L. Shi, Spatio-temporal backpropagation for training high-performance spiking neural networks, *Front. Neurosci*., 12, 2018, 00331.

(021)

Stereophonic Acoustic Echo Cancellation with the RLS Algorithm using the Conjugate Gradient Method

I.-D. Fîciu [1], **C. S. Stanciu** [1], **C. Paleologu** [1], **J. Benesty** [2], **C. Elisei-Iliescu** [1],
C. Anghel [1] **and S. Ciochină** [1]
[1] University Politehnica of Bucharest, 313 Splaiul Independenţei, Bucharest, Romania
[2] INRS-EMT, University of Quebec, Montreal, Quebec, H5A 1K6, Canada
E-mails: ionut.ficiu22@gmail.com, cristian@comm.pub.ro, pale@comm.pub.ro, jacob.benesty@inrs.ca,
camelia.elisei@romatsa.ro, canghel@comm.pub.ro, silviu@comm.pub.ro

---

Summary: Stereophonic communication systems employ terminals with two loudspeakers and two microphones in order to create the impression of audio directionality. Such setups can be affected by acoustic echoes corresponding to any loudspeaker-to-microphone pair. The standard approach for stereophonic acoustic echo cancellation (SAEC) framework was simplified in the past with the help of the widely linear (WL) model, which replaces the four corresponding adaptive filters working with real-valued variables with a single adaptive filter working with fewer complex-valued variables. In the WL context, this paper proposes a complex-valued adaptive algorithm based on the combination between the exponentially weighted recursive least-squares (RLS) algorithm and the conjugate gradient (CG) method. The namely WL-RLS-CG replaces the direct solution approaches associated with the classical RLS adaptive systems with solving an auxiliary system of equations based on an iterative method provided by the CG algorithm. Simulations results will be shown in order to demonstrate the performances of the proposed algorithm with respect to another classical solution.

Keywords: Stereophonic echo cancellation, Adaptive filters, RLS, Auxiliary equations, Conjugate gradient.

---

1. Introduction

In recent years, acoustic communication terminals were developed to provide an increased sensation of comfort for all interlocutors. Such is the case of stereophonic endpoints, which are comprised of a pair of loudspeakers, respectively a corresponding pair of microphones [1]. This configuration has the benefit of creating the impression of directionality for the acoustic messages. However, each loudspeaker-to-microphone pair has the potential of creating unwanted acoustic replicas, which might return to the original speaker [1, 2]. For the purpose of cancelling the acoustic echo, four adaptive filters working with real-valued variables must be employed in order to estimate all possible echo generating impulse responses. In [1], the widely linear (WL) model was proposed in order to improve the handling of the stereophonic acoustic echo cancellation (SAEC) by using a single adaptive filter working with a smaller amount of complex-valued variables. In this context, several adaptive algorithms were proposed from the recursive least-squares (RLS) family, which have increased capabilities of decorrelating the input signals. Such is the case of the RLS version based on the matrix inversion lemma (also known as Woodburry's identity), which solves the corresponding system of equations using a direct method and is susceptible to stability issues [1, 3].

Other approaches are designed to iteratively solve an associated auxiliary system of equations. The dichotomous coordinate descent (DCD) technique [3, 4] is a low-complexity inexact line search method which was combined with the RLS in the framework provided by the WL model. Nevertheless, the properties associated with the pairing between the RLS and the DCD limits the possibilities of further development, such as applying variable forms for the classical parameters of the RLS algorithms.

In this contribution, we analyze the adaptive system formed with the mixture between the exponentially weighted RLS adaptive and the conjugate gradient (CG) iterative method [3, 5]. The structure is organized as follows. The theoretical model, respectively the proposed algorithm are presented in Section 2. Section 3 shows several simulation results and compares performances with the RLS algorithm based on Woodburry's identity. Finally, conclusions are drawn in Section 4.

2. WL-RLS-CG Algorithm

We designate the combination between the RLS algorithm and the CG method working within the WL framework as the WL-RLS-CG. The algorithm is presented in Table 1. We denote the discrete time index with *n*, and the length of the four impulse responses corresponding to the unknown echo paths with *L*. The superscript ^{H} denotes the Hermitian operator. The algorithm updates its estimates using the forgetting factor λ, where $0 \ll \lambda < 1.$. In concordance with the WL model [1], the $2L \times 1$ vector $\tilde{\mathbf{x}}(n)$ comprises the last $2L$ complex-valued input samples, and the $2L \times 2L$ estimate of the correlation matrix is represented by $\mathbf{R}_{\tilde{\mathbf{x}}}(n)$ [1, 4]. The small positive valued constant δ and the identity matrix $\mathbf{I}_{2L}$ are employed to initialize $\mathbf{R}_{\tilde{\mathbf{x}}}(n)$ at the start of the simulation. The time-shift nature of the input signal ensures that the update performed for the correlation matrix in step 2 can be replaced by copying the upper-left $(2L-2)\times(2L-2)$ sub-matrix to the lower-right $(2L-2)\times(2L-2)$ sub-matrix, then compute the first column and copy it using the Hermitian operator on the first row. Moreover, the second column, respectively second row, can be easily obtained by again copying and conjugating the values from the first column, respectively first row. More details can be found in [1] and [3]. The highest costs in terms of arithmetic operations are associated with step 5 of the algorithm and have a magnitude proportional to the square of the complex valued filter length 2*L*.

In step 4 of the algorithm the *residual component* $\mathbf{p}_0(n)$ is updated and used in step 5, where the CG algorithm determines the so-called $2L\times1$ *solution vector* $\Delta\tilde{\mathbf{h}}(n)$ and the $2L\times1$ *residual vector* $\mathbf{r}(n)$ [1], [3]. The former is then employed to update the 2*L* adaptive filter's coefficients $\tilde{\mathbf{h}}(n)$ in step 6, while the latter will be necessary in the next filter iteration (i.e., at time index *n*+1) to update the residual component $\mathbf{p}_0(n+1)$. The solution replaces the classical direct approach for updating the adaptive filter's taps with the auxiliary system of normal equations in step 5, which instead determines $\Delta\tilde{\mathbf{h}}(n)$, and subsequently updates $\tilde{\mathbf{h}}(n)$ through the summation in step 6 [3].

Table 1. Exponentially Weighted RLS using CG.

Step	Actions
1.	Initialization: $\tilde{\mathbf{h}}(0)=\mathbf{0};\mathbf{r}(0)=\mathbf{0};\mathbf{R}_{\tilde{\mathbf{x}}}(0)=\delta\mathbf{I}_{2L}$
	For *n* = 1, 2…
2.	$\mathbf{R}_{\tilde{\mathbf{x}}}(n)=\lambda\mathbf{R}_{\tilde{\mathbf{x}}}(n-1)+\tilde{\mathbf{x}}(n)\tilde{\mathbf{x}}^H(n)$
3.	$e(n)=d(n)-\tilde{\mathbf{h}}^H(n-1)\tilde{\mathbf{x}}(n)$
4.	$\mathbf{p}_0(n)=\lambda\mathbf{r}(n-1)+\tilde{\mathbf{x}}(n)e^*(n)$
5.	$\mathbf{R}_{\tilde{\mathbf{x}}}(n)\Delta\mathbf{h}(n)=\mathbf{p}_0(n)\xrightarrow{\text{CG}}\Delta\tilde{\mathbf{h}}(n),\mathbf{r}(n)$
6.	$\tilde{\mathbf{h}}(n)=\tilde{\mathbf{h}}(n-1)+\Delta\tilde{\mathbf{h}}(n)$

The CG algorithm is summarized in Table 2. It is designed to minimize the cost function

$$f\left[\Delta\mathbf{h}(n)\right]=\frac{1}{2}\Delta\mathbf{h}(n)^H\mathbf{R}_{\tilde{\mathbf{x}}}(n)\Delta\mathbf{h}(n)- \\ -\mathbf{p}_0^H(n)\Delta\mathbf{h}(n), \tag{1}$$

and it updates the solution vector $\Delta\tilde{\mathbf{h}}(n)$ in an iterative manner using a variable *step size* α and a direction vector **d**. For each iteration, the direction **d** is chosen to ensure conjugacy with respect to $\mathbf{R}_{\tilde{\mathbf{x}}}(n)$ [5], which involves that

$$\mathbf{d}_i^H\mathbf{R}_{\tilde{\mathbf{x}}}(n)\mathbf{d}_j=0,\ i\neq j, \tag{2}$$

where $i,j=1\ldots N_{\mathrm{u}}$, respectively N_{u} denotes the maximum number of allowed iterations (i.e., the number of steps corresponding the iterative process). In practice, it is expected for the parameter N_{u} to have values in the range $N_u \ll L$ [1, 3].

The CG method requires divisions at steps 1 and 3, and a number of multiplications with an order of magnitude $O(4L^2N_u)$, which is determined by the denominator shown in step 2. The denominator imposes the condition

$$\mathbf{d}_i^H\mathbf{R}_{\tilde{\mathbf{x}}}(n)\mathbf{d}_i\neq0,\ i=1\ldots N_{\mathrm{u}} \tag{3}$$

Despite having an overall complexity comparable to the WL-RLS algorithm based on the matrix inversion lemma [4], the WL-RLS-CG approach has good convergence speeds and leaves room for some possible improvements not attainable using simpler versions of the RLS, such as the WL-RLS-DCD. A good example would be the implementation of the WL-RLS-CG using variable values of the forgetting factor [6].

Table 2. The Conjugate Gradient Algorithm.

Step	Actions at time index *n*
	Initialization: $\Delta\tilde{\mathbf{h}}=\mathbf{0}_{2L\times1};\mathbf{r}=\mathbf{p}_0(n);\rho_0=\mathbf{r}^H\mathbf{r};\ \mathbf{d}=\mathbf{r}$
For *k* = 1, 2…N_{u}	
1.	$if\ k>1\ \rightarrow\mathbf{d}=\mathbf{r}+\dfrac{\rho_{k-1}}{\rho_{k-2}}\mathbf{d}$
2.	$\alpha=\dfrac{\rho_{k-1}}{\mathbf{d}^H\mathbf{R}_{\tilde{\mathbf{x}}}(n)\mathbf{d}}$
3.	$\Delta\tilde{\mathbf{h}}=\Delta\tilde{\mathbf{h}}+\alpha\mathbf{d}$
4.	$\mathbf{r}=\mathbf{r}-\alpha\mathbf{R}_{\tilde{\mathbf{x}}}(n)\mathbf{d}$
5.	$\rho_k=\mathbf{r}^H\mathbf{r}$

3. Simulation Results

Simulations were performed in the context of SAEC using Gaussian noise, respectively speech, as inputs, which were filtered through two acoustic impulse responses in order to mimic spatial diversity. The two channels were also predistorted using the method presented in [1] with the purpose of reducing the correlation between the stereo channels. For the four unknown echo paths we used real measured impulse responses decimated in order obtain the length *L* = 256. For the complex microphone signal *d*(*n*), the echo-to-noise ratio (ENR) was experimentally set to 25 dB.

The performance indicator is the normalized misalignment measured in dB, which is defined in [1]. We compared the WL-RLS-CG method for different values of N_u and we used as a reference the RLS algorithm based on Woodbury's identity, (i.e., the WL-RLS) [1, 2].

In Fig. 1 simulation results are illustrated for the Gaussian noise input signal. A compromise can be noticed for the WL-RLS-CG as the value of N_u is increased. The corresponding convergence speed improves, and the accuracy of the algorithm at steady-state decreases. However, for $N_u > 2$ the performance increase is caped, and the value $N_u = 2$ is sufficient for the WL-RLS-CG to match the convergence speed of the WL-RLS, respectively to have superior precision in the steady-state.

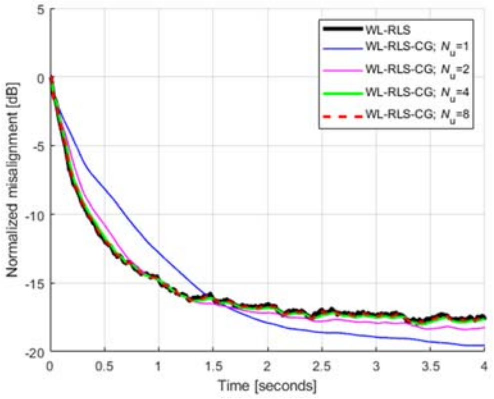


Fig. 1. Normalized misalignment for WL-RLS and WL-RLS-CG with different values of N_u. Input signal is Gaussian noise and the length of the echo paths is $L = 256$.

In Fig. 2 the input signal is a speech sequence. It can be noticed again that increasing the value of N_u leads to improved convergence speeds for the WL-RLS-CG. Moreover, for a value of N_u large enough, the convergence speed is comparable to the WL-RLS, and the performance at steady-state is superior to the one corresponding to the reference algorithm.

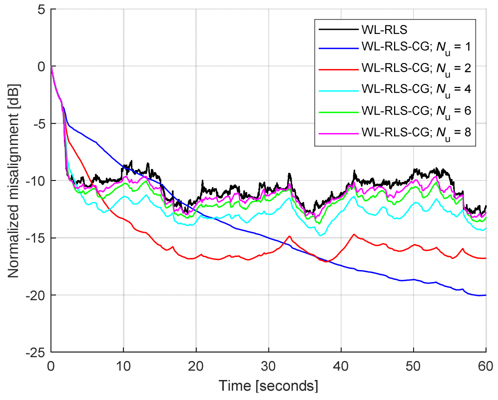


Fig. 2. Normalized misalignment for WL-RLS and WL-RLS-CG with different values of N_u. Input signal is speech and the length of the echo paths is $L = 256$.

Considering that the speech is a highly correlated signal, it can also be noticed that, in comparison to the first scenario, a higher value of N_u is necessary in order to achieve the cap in performance. For practical implementations of SAEC, the parameter N_u should be set to values which ensure sufficient performance when working with difficult signals.

4. Conclusions

Despite having a relatively high arithmetic complexity, the proposed WL-RLS-CG algorithm for SAEC scenarios can match the convergence speed of the WL-RLS method, and it outperforms the latter when working at steady-state. The maximum number of allowed iterations N_u is one of the key parameters, and it influences the overall arithmetic costs associated with the WL-RLS-CG.

When comparing the CG approach to other RLS methods which solve the corresponding auxiliary system of equations (such as the WL-RLS-DCD), the WL-RLS-CG method relies less on the statistics of the input signal. Consequently, the proposed algorithm is more suitable for further developments, such as versions employing variable forgetting factors [6].

Acknowledgements

This work was supported by one grant of the Ministry of Research, Innovation and Digitization, CNCS-UEFISCDI, project PN-III-P4-PCE-2021-0438, within PNCDI III.

References

[1]. C. Stanciu, J. Benesty, C. Paleologu, T. Gänsler, S. Ciochină, A widely linear model for stereophonic acoustic echo cancellation, *Signal Processing*, Vol. 93, Issue 2, 2013, pp. 511-516.

[2]. S. Haykin, Adaptive Filter Theory, *Prentice Hall*, Upper Saddle River, NJ, USA, 2002.

[3]. Y. V. Zakharov, G. P. White, J. Liu, Low-complexity RLS algorithms using dichotomous coordinate descent iterations, *IEEE Trans. Signal Processing*, July 2008, Vol. 56, pp. 3150-3161.

[4]. J. Liu, Y. V. Zakharov, B. Weaver, Architecture and FPGA design of dichotomous coordinate descent algorithms, *IEEE Trans. Circuits and Systems I: Regular Papers*, Vol. 56, Nov. 2009, pp. 2425-2438.

[5]. G. K. Boray, M. D. Srinath, Conjugate gradient techniques for adaptive filtering, *IEEE Trans. Circuits Syst. I: Fundam. Theory Appl.*, Vol. 39, Issue 1, 1992, pp. 1-10.

[6]. C. Stanciu, C. Paleologu, J. Benesty, S. Ciochină, F. Albu, Variable-forgetting factor RLS for stereophonic acoustic echo cancellation with widely linear model, in *Proceedings of the 20th European Signal Processing Conference (EUSIPCO'12)*, 2012, pp. 1960-1964.

(022)

Implementation of SSN in the Evaluation of the Robotic Welding Process of Aircraft Engine Casing Components

A. Burghardt [1], **P. Gierlak** [1], **M. Muszyńska** [1], **K. Kurc** [1], **D. Szybicki** [1], **M. Uliasz** [2]
and T. Muszyński [3]
[1] Rzeszów University of Technology, Department of Applied Mechanics and Robotics,
Faculty of Mechanical Engineering and Aeronautics, al. Powstancow Warszawy 8,
35-959 Rzeszow, Poland
[2] Pratt&Whitney Rzeszow S. A., Hetmanska 120, 35-078, Rzeszow, Poland
[3] Sturm Polska sp.z o.o., Maczka 6, 35-234, Rzeszow, Poland
E-mails: andrzejb@prz.edu.pl, kkurc@prz.edu.pl, pgierlak@prz.edu.pl, magdaw@prz.edu.pl,
dszybicki@prz.edu.pl, marek.uliasz@prattwhitney.com, t.muszynski@sturm-gruppe.com

---

Summary: The paper presents a neural system for verifying the correctness of the implementation of the Intermediate Case welding process of the PW800 engine parts manufactured by Pratt&Whittney. Welding is carried out on a robotic station, whose equipment allows for recording more than a dozen process parameters. In addition, information from quality control (FPI, X-ray) about the type and size of defects is provided. This allows for building a knowledge base about the process, and in the case of neural networks, to prepare learning data. Currently, elements of descriptive statistics are used for process control, which allows process improvement of up to 57 % [1]. The proposed solution using neural networks shows a very good efficiency of 99.7 %.

Keywords: Robot, TIG welding, Robot programming, Process monitoring, Jet engine casing.

---

1. Introduction

The jet engines used in modern passenger aircraft consist of many complex structural assemblies. In order to reduce production costs and weight, and thus engine operating costs, the precision casting technology used to date is often replaced by the implementation of welded complex thin-walled assemblies [2].

Such an approach requires the use of advanced material joining processes, the development of implementation and control technologies. Thin-walled components such as the engine casing are safety-critical components. Each manufactured component is subjected to quality control operations several times in the form of visual inspection, CMM measurements, non-destructive testing, or fluorescence inspection. Nowadays, inspection at various stages of the process is being introduced in aircraft engine production control processes [3-5], with increasing use of machine intelligence, fuzzy logic [6], or artificial neural networks [7].

The component to be welded is large in size and complex in shape. It is the complex shape that prevents the use of modern technologies such as friction welding (FRW), in which metals are joined by mechanical friction. Another variant of this technology is friction stir welding (FSW) [8]. The aforementioned FRW and FSW joining methods enable the welding of aluminum alloys that were previously considered non-weldable [9]. The aforementioned complex shape has also eliminated electron beam hull joining technologies or the use of laser welding. In both of these cases, the beam source is very large. TIG (Tungsten Inert Gas) welding produces an electric arc through a non-fusible tungsten electrode in a shield of inert gas (argon, helium or mixtures thereof). The advantages of TIG welding are: very good joint quality while maintaining welding productivity, welding of workpieces of different thicknesses – less than 1 mm up to approx. 10 mm, the possibility of working in different positions, very high control over the precision of the welding process, the possibility of welding different types of materials and alloys, the formation of a negligible amount of slag. The disadvantages of TIG welding are, the need for shielding when welding in an open area, lower productivity when welding by hand, the quality of the welds is dependent on the welder's ability. The aforementioned advantages led to the choice of the TIG method for joining the components of the intermediate case of the PW814 engine. The disadvantage related to the dependence of quality on the welder's ability was eliminated by using an industrial robot.

The quality of the performance of robotised TIG welded joints depends on the correction of the torch position and the stabilisation of the welding parameters such as current, voltage and shielding gas quantity. Corrections to the position of the robot's TCP movement, which in the case under analysis is related to welding, can be achieved in various ways, e.g. by using vision systems [10, 11], built from a single camera [12], a camera with a laser marker [13, 14] or a 3D measuring system [15].

The quality of the welding process depends not only on the position of the TCP, but also on the welding parameters. In [16], the influence of different parameters on the welding process was studied. The penetration depth, microstructure and hardness measurement of the resulting weld were analysed for mild steel using gas-shielded robotic arc welding. The variables chosen in this study were arc voltage, welding current and welding speed.

An algorithm using a weld width prediction element of the welded joint based on fuzzy logic and deep learning was used for parameters such as welding current, arc length and welding speed [17].

Another article dealing with the optimisation of the TIG welding process, understood as the correct shape of the weld, was included in [18]. Welding current, arc length and torch travel speed were chosen as the input process parameters, while fusion depth, depth-to-width ratio, heat input and the width of the heat affected zone were taken as the output variables for the study.

In work [19], a feed forward, back propagation neural network with Levenberg-Marquardt learning algorithm was used to optimise the quality of mild steel pipe welds. The study showed that the neural network, based on the obtained process parameters, is also able to determine the achieved tensile strength and yield strength. SSN algorithms have also been developed and described in [20, 21], to control the welding process of artillery munitions. One of the models, with back propagation, was used for qualitative classification: compliant, non-compliant, based on process parameters.

The small number of literature studies on detailed analyses of inference and prediction of weld quality based on process parameters is complemented by the issues addressed in this article.

2. Problem Definition

The Intermediate Case of the Pratt & Whitney PW814 engine (Fig. 1), which is shown in Fig. 2, is a thin-walled component consisting of dozens of interconnected components, where each component is made to the specified tolerances.

Fig. 1. Pratt & Whitney PW814 aircraft engine, with indication of intermediate case [1].

The advantage of welded construction is the reduced weight of the assembly. During welding, it is subjected to stresses due to heat shrinkage. This can result in defects in the welds despite ensuring constant process parameters.

Due to the high price and the need to ensure quality and flight safety, manufactured components are subjected to numerous quality control stages using visual inspection methods, CMM measurements, FPI fluorescence inspection and X-ray inspection. If non-conformities are found, an approved repair process is required. The repair process generates additional costs. The part has to be re-transported from the measurement laboratory and re-situated and repositioned on the production bench. It is worth mentioning that the diameter of the intermediate case of the PW814 engine is 1400 mm. A view of the case is shown in Fig. 2.

Fig. 2. Intermediate case component of the PW814 engine [1].

Digitally obtainable process parameters are collected and stored in the robot controller's memory in a structured, designed structure. They are subjected to statistical analysis such as standard deviation, variance, skewness, kurtosis and mean value. In previous research [1], an identifier was determined in the form of a skewness function for the welding voltage waveform, the limit value of which allows the determination of a defect such as partial lack of fusion of the weld or the occurrence of inclusions of foreign material. This solution was tested on a series of 50 production pieces. The success rate for defect identification was 48 % [1]. Test work carried out with elements of descriptive statistics does not allow the detection of all types of defects. In order to improve the results, an artificial neural network was used to classify weld states on the basis of the recorded parameters. The main objective was to obtain the highest possible defect detection rate.

3. Test Stand

The problem described here concerns the TIG welding operation of the intermediate case of the PW814 engine (Fig. 1). 36 welds with a total length of more than 7 (m) are analysed. The welding process is carried out on a robotic production station (Fig. 4), equipped with components such as: ABB IRB2600ID 6-axis industrial robot with ABB IRBP B1000 positioner, FRONIUS MagicWave 5000 welding source. The correct flow of shielding gas, argon, is ensured by Bronkhorst Mass-Stream regulators. Keyence CV-X150FP vision system is installed on the bench, which supervises the torch calibration process and allows parts to be measured prior to welding, allowing the robot's movement to be directed to the actual path of the welding path. Real-time compensation of the distance between the electrode and the workpiece is handled by the ABB WeldGuide III controller. The execution of the main program is managed by the ABB IRC5 controller, which communicates with all components of the workstation. It supervises the movements of the robot and positioner, as well as the operation of other systems.

Fig. 3. Robotic production station at Pratt & Whitney Rzeszów S. A. [1].

A schematic of how the devices on the test stand are connected, using the industry standards EthernetIP and DeviceNet, is shown in the diagram in Fig. 3.

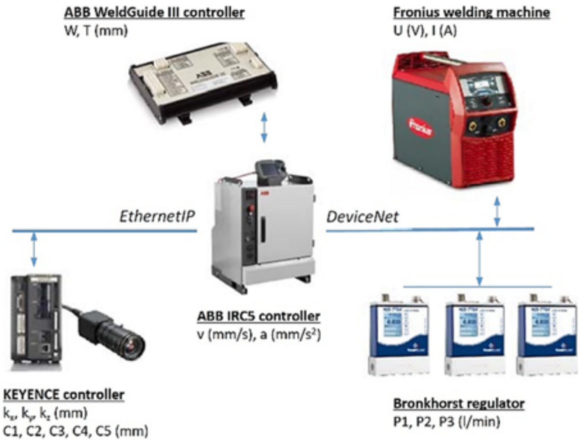


Fig. 4. Schematic diagram of the equipment connection on the test stand [1].

The main objective of the work described is to assess the correctness of the welds based on the analysis of the process parameters. Table 1 shows all the parameters that were taken into account during the analysis.

Table 1. Summary of data recorded.

Group	Parameter	Designation	Unit	Device
Pre-process parameters	TCP calibration parameters	Kx, Ky, Kz, Px, Py, Pz	(mm)	ABB IRC5 Keyence CV-S200M
	Welding trajectory compensation	C1,C2,C3, ΔC (C3-C1)	(mm)	ABB IRC5 Keyence CV-S200M
In-process parameters	Welding voltage	WeldVolt, WgWeldVolt	(V)	Fronius MagicWave 5000
	Welding current	WeldCurr	(A)	Fronius MagicWave 5000
	Shielding gas flow rate	Sg1, Sg2, Sg3	(l/min)	ABB IRC5 Bronkhorst Mass-Stream
	Travel speed	WeldSpeed	(mm/s)	ABB IRC5
	Motion acceleration	WeldAcc	(mm/s^2)	ABB IRC5
	Z-axis position compensation	WgCorr, WgTotCorr	(mm)	ABB IRC5 WeldGuide III

These parameters can be divided into two groups. The first is the tool calibration and weld path compensation values, stored before welding. Each is a single number, where Kx, Ky, Kz and Px, Py, Pz are the coordinates of the TCP system K – camera, P – torch respectively. The values C1, C2, C3 are the weld path compensation values at its three main points, and ΔC is the difference of C3 and C1 compensation, representing the angular deviation of the compensation. The second group is the time waveforms of the welding parameters. This includes the voltage (WeldVolt, WgWeldVolt), current (WeldCurr), speed and acceleration (WeldSpeed, WeldAcc), as well as shielding gas flow parameters (Sg1, Sg2, Sg3) and torch distance compensation parameters (WgCorr, WgTotCorr).

The welding time per weld is approximately 60 seconds and the recording rate is 43 samples per second. The initial condition is an assessment of the effect of the welding process. The welds can be classified as correctly performed or in need of improvement. Their designations are shown in Table 2. k_x k_y k_z.

Table 2. Classification of welds, correct, faulty.

Classification	Designation
Conforming weld	P
Non-Conforming weld	Np
	Nw
	Nb
	Nb$_{70}$
	Nb$_{20}$

A correctly made weld (P) meets the requirements of the standards and is accepted in quality control. Defects arising may completely disqualify the part from further production. In the case of minor defects, the standards allow repair operations to be carried out.

Such defects include burn-through (Np), inclusion of foreign mat. (Nw) and incorrect remelting: along the entire length (Nb), below 70 % (Nb_{70}) or 20 % (Nb_{20}).

4. Solving the Problem Using Neural Networks

Artificial neural networks are a very good tool for pattern recognition and classification. In this study, MATLAB software was used together with elements of the Neural Net Pattern Recognition library. The learning functions available in the software were analysed. The selection of functions depends on the specific case and is carried out experimentally.

4.1. Preparation of Learning Data

One of the main conditions for the successful operation of neural networks is the correct execution of the learning process. In order to carry out the learning process, an input matrix X and an output matrix Y were generated. The structure of the X matrix contains the characteristics (elements of descriptive statistics, Table 3), while the output matrix (T) presented the corresponding performance states shown in Table 4.

Table 3. Elements of the input matrix.

Parameter/Function	Parameter designation	Nr characteristic
Path compensation	C1	1
	C2	2
	C3	3
	ΔC (C3-C1)	4
Standard deviation	WeldCurr	5
	WeldVolt	6
	WgWeldVolt	7
	WgCorr	8
	WgTotCorr	9
Variance	WeldCurr	10
	WeldVolt	11
	WgWeldVolt	12
	WgCorr	13
	WgTotCorr	14
Skewness	WeldCurr	15
	WeldVolt	16
	WgWeldVolt	17
	WgCorr	18
	WgTotCorr	19
Kurtosis	WeldCurr	20
	WeldVolt	21
	WgWeldVolt	22
	WgCorr	23
	WgTotCorr	24
Average value	WeldCurr	25
	WeldVolt	26
	WgWeldVolt	27
	WgCorr	28
	WgTotCorr	29

The learning data were prepared on the basis of data recorded for 50 production parts. In order to reduce the amount of data analysed, a comparison of characteristics was carried out and those whose values did not change or changed slightly were eliminated. The camera and torch TCP point calibration parameters, shielding gas flow values and welding motion speeds and accelerations showed high stability and repeatability for both correctly made welds and those containing detected defects. Finally, the elements of the input learning matrix are the 29 characteristics shown in Table 3.

Response matrices were also prepared for the data sets. Table 5 shows the quantitative distribution of the data sets for the different output states from Table 2.

It can be seen that there is a relatively small number of data sets for invalid states, in the series described, this was about 4 % of the welds, so the data sets containing the Nb and Nc parameters were doubled, copying the original data, Table 5.

Table 4. Generated Outputs (classification types).

Welding result type	P	Nb	Nw	Nb_{20}	Np	Nb_{70}
Number of samples	1726	41	22	9	0	2
Classification type	P	Nb	Nc		–	–

In the series described, there was not a single case of a flaw such as material burn-through, and a 20-70 % lack of fusion occurred in only two welds. This number of cases does not allow for the preparation of learning data for the categories listed, so they were omitted from the further process.

Table 5. Adopted data distribution for each classification type.

Classification type	P	Nb	Nc
Training data	123	73	49
Validation data	21	9	13
Sum	144	82	62

When comparing the matrices for the different classification types, one can see that the amount of learner data was not balanced, which negatively affected the results obtained. Too much data of one type resulted in a lack of sensitivity of the learned network to the other types.

Table 5 shows the number of learning and verification sets for each classification type.

The learning dataset for the correct welding process results (P) was restricted so that it represented 50 % of the total learning set. The target set was obtained by selecting every 10th data set.

The verification data for each classification type were randomly separated from the learning set using a built-in function (*dividerand*). They accounted for 15-20 % of the total data, depending on the classification type.

4.2. Neural Network Training

The process of training the neural network was carried out with the Neural Net Pattern Recognition function in the MATLAB environment, using prepared learning data sets in the form of matrices X and T. A number of attempts were made to train the neural network using different learning algorithms and the number of neurons in hidden layer. The network learning process is described below, adopting the parameters and algorithms for which the best results were obtained.

During testing, the best results were obtained using the Bayesian Regularisation (trainbr) algorithm.

It is a modified Lavenberg-Marquardt algorithm with additional procedures for optimising the number of network parameters in order to create networks with better generalisation properties. The performance of learning at each step is assessed by calculating the Mean Squared Error (mse).

The learning process continues until a target value (Target Value) is reached for one of the parameters, with a maximum number of learning epochs of 1000. Fig. 5 shows a plot of the Mean Squared Error values at each learning epoch.

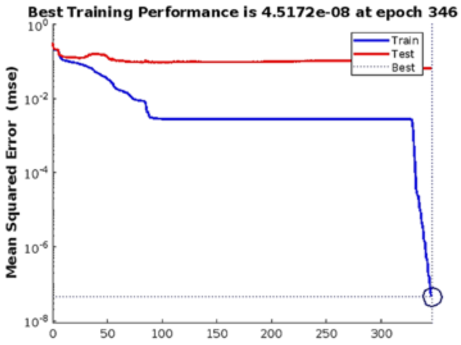
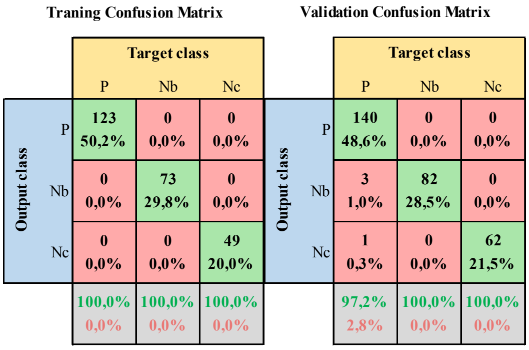


Fig. 5. Graph of the mean squared error value.

The matrix for assigning the data sets to the different classes is shown in Fig. 6, the first illustrating the results for the learning stage and the second for the validation stage.


Traning Confusion Matrix

Output class	Target class P	Target class Nb	Target class Nc
P	123 50,2%	0 0,0%	0 0,0%
Nb	0 0,0%	73 29,8%	0 0,0%
Nc	0 0,0%	0 0,0%	49 20,0%
	100,0% 0,0%	100,0% 0,0%	100,0% 0,0%

Validation Confusion Matrix

Output class	Target class P	Target class Nb	Target class Nc
P	140 48,6%	0 0,0%	0 0,0%
Nb	3 1,0%	82 28,5%	0 0,0%
Nc	1 0,3%	0 0,0%	62 21,5%
	97,2% 2,8%	100,0% 0,0%	100,0% 0,0%

Fig. 6. Assignment matrix of the data sets to the different classes.

The classification made by the network is shown in the rows of the matrix (Fig. 6), while the columns correspond to the actual evaluation. In the case of a concordant correlation of the two values, the examined data set is counted as correctly classified. All these cases are located on the diagonal of the matrix and are marked in green. Incorrectly classified cases are in the cells in red.

The effectiveness of the resulting solution is measured by the level of correlation of the classification determined by the neural network against the data of the known real classification. These are shown as percentages on the horizontal axis (grey cells).

4.3. Presentation/Statement of Results

The Neural Net Pattern Recognition (nprtool) model showed the best results of all those tested in the MATLAB environment. Table 6 summarises the selected trials using different learning algorithms and number of neurons in hidden layer.

Table 6. Table of neural network model learning results for different input parameters.

No	Learning algorithm	Neurons in hidden layer	Match level P	Match level Nb	Match level Nc
1	Bayesian Regularization	30	97,2 %	100,0 %	100,0 %
2	Bayesian Regularization	20	95,8 %	100,0 %	100,0 %
3	Bayesian Regularization	40	95,8 %	100,0 %	100,0 %
4	Levenberg-Marquardt	30	93,1 %	100,0 %	100,0 %
5	Resilient Backpropagation	30	96,5 %	97,6 %	100,0 %
6	Levenberg-Marquardt	40	96,5 %	97,6 %	100,0 %
7	One Step Secant	20	96,5 %	97,6 %	100,0 %
8	Bayesian Regularization	10	95,8 %	97,6 %	100,0 %
9	Bayesian Regularization	50	95,8 %	97,6 %	100,0 %
10	One Step Secant	30	95,8 %	97,6 %	100,0 %
11	Levenberg-Marquardt	50	95,1 %	97,6 %	100,0 %
12	Resilient Backpropagation	50	97,2 %	95,1 %	100,0 %
13	One Step Secant	50	97,2 %	92,7 %	100,0 %
14	Polak-Ribiére Conjugate Gradient	40	96,5 %	92,7 %	100,0 %
15	One Step Secant	10	96,5 %	90,2 %	100,0 %

The data were sorted according to the learning outcomes obtained, i.e. the correlation levels of the classifications.

In the adopted model, the best results were obtained using the Bayesian Regularisation learning algorithm. For this function, very good correlation levels were obtained already for the standard number of hidden layers of 10. As the number of layers was reduced, the correlation level decreased. The highest correlation level was obtained with 30 neurons in hidden layer, where the correlation level for defect classes Nb and Nc was 100 % and for correct indications (P) 97.2 % (Table 6).

Increasing the number of layers further did not produce better results.

Using Levenberg-Marquardt function with 30 neurons in hidden layer resulted also in 100 % defect detect for classes Nb and Nc, but the match level for correct indications (P) was lower in this case – 93,1 %.

For other learning functions shown in Table 6 lower detection values were observed. For other considered cases, the match level was below 90 %.

The study did not achieve 100 % classification for class P (correct). Some were classified as incorrect in classes Nb and Nc. Such cases may represent very small defects that are within the tolerances specified by the standards and were therefore classified as correct.

The resulting solution can be implemented on the production stand. In the first phase, the implemented algorithm should be tested using the new data. Already during testing, the system's responses will identify welds that need to be verified for quality. This will improve the flow of the part through the production line and, in some cases, also eliminate the need for the part to return to the production stand to perform the repair.

5. Conclusions

The analysis carried out showed the possibility of using the acquired process characteristics to predict weld defects occurring in the process of robotic TIG welding of PW814 engine intermediate case fairings.

The conducted research work made it possible to develop a system using SSN to check the correctness of the welds of aircraft parts. Very good defect detection results of 97,2 % were obtained. Currently, verification work is being carried out on the implemented system for controlling the correctness of the implementation of the welding process.

References

[1]. M. Uliasz, A. Ornat, A. Burghardt, M. Muszyńska, D. Szybicki, K. Kurc, Automatic evaluation of the robotic production process for an aircraft jet engine casing, *Applied Sciences*, Vol. 12, 2022, 6443.

[2]. A. Zawadzki, Robotic welding of thin-walled aircraft engine fuselage assemblies, *Engineering Design and Construction*, March, 2015, pp. 28-34.

[3]. D. Szybicki, A. Burghardt, K. Kurc, P. Pietruś, Calibration and verification of an original module measuring turbojet engine blades geometric parameters, *Archive of Mechanical Engineering*, Vol. 66, 2019, pp. 97-109.

[4]. A. Burghardt, K. Kurc, D. Szybicki, M. Muszyńska, T. Szczęch, Monitoring the parameters of the robot-operated quality control process, *Advances in Science and Technology, Research Journal*, Vol. 11, 2017, pp. 232-236.

[5]. A. Burghardt, K. Kurc, D. Szybicki, M. Muszyńska, J. Nawrocki, Software for the robot-operated inspection station for engine guide vanes taking into consideration the geometric variability of parts, *Tehnički Vjesnik*, Vol. 24, 2017, pp. 349-353.

[6]. Z. Hendzel, A. Burghardt, P. Gierlak, M. Szuster, Conventional and fuzzy force control in robotised machining, *Solid State Phenomena*, Vol. 210, 2014, pp. 178-185.

[7]. A. Burghardt, D. Szybicki, K. Kurc, M. Muszyńska, Robotic grinding process of turboprop engine compressor blades with active selection of contact force, *Tehnički Vjesnik*, Vol. 29, 2022, pp. 15-22.

[8]. C. Tierney, C. Higgins, D. Quinn, J. De Backer, C. Allen, A. Örtnäs, A. Murphy, A scalable cost modelling architecture for evaluating the production cost-effectiveness of novel joining techniques for aircraft structures, *Procedia Manufacturing*, Vol. 54, 2021, pp. 7-12.

[9]. P. Mendez, T. Eagar, Welding processes for aeronautics, *Advanced Materials and Processes*, Vol. 159, 2001, pp. 39-43.

[10]. Z. Hou, Y. Xu, R. Xiao, S. Chen, A teaching-free welding method based on laser visual sensing system in robotic GMAW, *The International Journal of Advanced Manufacturing Technology*, Vol. 109, 2020, pp. 1755-1774.

[11]. D. Xu, L. Wang, M. Tan, Image processing and visual control method for arc welding robot, in *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, 2004, pp. 727-732.

[12]. A. Bzymek, M. Fidali, W. Jamrozik, A. Timofiejczuk, Diagnostic vision system for welded joint and welding process assessment, *Problemy Eksploatacji*, Vol. 4, 2008, pp. 39-51.

[13]. B. Xue, B. Chang, G. Peng, Y. Gao, Z. Tian, D. Du, G. Wang, A vision based detection method for narrow butt joints and a robotic seam tracking system, *Sensors*, Vol. 19, 2019, 1144.

[14]. J. Zeng, B. Chang, D. Du, L. Wang, S. Chang, G. Peng, W. Wang, A weld position recognition method based on directional and structured light information fusion in multi-layer/multi-pass welding, *Sensors*, Vol. 18, 2018, 129.

[15]. P. Xu, G. Xu, X. Tang, S. Yao, A visual seam tracking system for robotic arc welding, *The International Journal of Advanced Manufacturing Technology*, Vol. 37, 2008, pp. 70-75.

[16]. I. A. Ibrahim, S. A. Mohamat, A. Amir, A. Ghalib, The Effect of Gas Metal Arc Welding (GMAW) processes on different welding parameters, *Procedia Engineering*, Vol. 41, 2012, pp. 1502-1506.

[17]. M. A. Kesse, E. Buah, H. Handroos, G. K. Ayetor, Development of an artificial intelligence powered TIG welding algorithm for the prediction of bead geometry for TIG welding processes using hybrid deep learning, *Metals*, Vol. 10, 2020, 451.

[18]. J. J. Vora, V. K. Patel, S. Srinivasan, R. Chaudhari, D. Y. Pimenov, K. Giasin, S. Sharma, Optimization of activated tungsten inert gas welding process parameters using heat transfer search algorithm: With experimental validation using case studies, *Metals*, Vol. 11, 2021, 981.

[19]. I. U. Abhulimen, J. I. Achebo, Application of artificial neural network in predicting the weld quality of a tungsten inert gas welded mild steel pipe joint, *International Journal Of Scientific & Technology Research*, Vol. 3, 2014, pp. 277-285.

[20]. A. Aktepe, S. Ersöz, M. Lüy, Backpropagation neural network applications for a welding process control problem, in *Proceedings of the 13th International Conference Engineering Applications of Neural Networks (EANN'12)*, Vol. 13, London, UK, September 20-23, 2012, pp. 172-182.

[21]. A. Aktepe, S. Ersöz, M. Lüy, Welding process optimization with artificial neural network applications, *Neural Network World*, Vol. 6/14, 2014, pp. 655-670.

5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2023),
7-9 June 2023, Tenerife (Canary Islands), Spain

(024)

ConTraGAN – A Conditional Transformer-based Generative Adversarial Network for Zero-Day Network Attack Analysis and Detection

S. A. Gabayre [1], X. Shi [1], S. Dogan [1], Y. Rahulamathavan [1], A. Weightman [2] and G. Cooper [2]
[1] Loughborough University London, Institute for Digital Technologies, London E20 3BS, UK
[2] University of Manchester, School of Engineering, Manchester M13 9PL, UK
Tel.: + 44 20 38180777
E-mail: x.shi@lboro.ac.uk

Summary: The Domain Name Service (DNS) protocol has become a sophisticated tool for malicious actors to bypass network firewalls and Intrusion Detection Systems (IDS) for cybercrimes, such as exfiltrating stolen data through tunnelled DNS traffic. This paper proposes a solution, named as ConTraGAN, to generate unknown zero-day network attack vectors for the purpose of training IDS and detecting malicious traffic. The ConTraGAN encompasses a hybrid conditional transformer-based generative adversarial network model and is trained using exfiltrated data that are tunnelled over DNS traffic. A self-attention mechanism is also built into the ConTraGAN, which serves as a module of providing attention weights to each feature of the captured DNS traffic. The preliminary results show that the proposed network can function as an effective generator of new attack vectors for IDS training and detection.

Keywords: Conditional transformer, Generative adversarial network, Intrusion detection, Zero-Day, Exfiltration.

1. Introduction

There has been an increased interest in research into anomaly detection pertaining to Domain Name Service (DNS) based traffic with deep machine learning in recent years [1-3]. DNS forms an extremely vital function of network communication. No matter how tight an organisation restricts outbound communication, DNS queries and services will always be permitted to external parties [4]. This is mainly because of the function of the DNS protocol which serves as a lookup and registry of the countless web sites and their corresponding IP addresses on the Internet. It is for this reason that the DNS protocol has been become a sophisticated tool for malicious actors with the knowhow to bypass firewall and Intrusion Detection Systems (IDS) when establishing contact with a compromised network for a number of purposes such as exfiltrating stolen data amongst a myriad of other criminal activities. With cyber criminals expanding their exploitative capabilities with increasing sophistication, cybersecurity tools have remained in a cycle of playing catch up with every new Zero-Day attack (i.e. unknown attack) that is exposed.

This paper proposes the use of a variant of Deep Generative Model (DGM), which is trained using exfiltrated data tunnelled over DNS traffic, to offer a solution to the problem of detecting unknown DNS tunnelling attack. One of the advantages of DGM is that it allows researchers to overcome issues pertaining to analysing and learning from big data without the need of supervised learning, hence, to build models that are capable of learning the latent nature of highly dimensional data thereby providing meaningful insights for traffic analysis.

We hypothesise that just like any language, network traffic has semantic representation that can be learned by a Neural Network (NN) [5]. When breaking down the extracted features in a DNS packet, the weighted embeddings of the vector representation present in each feature have hidden context that gives meaning to each packet in a unique way. This context is what we believe can be learned and replicated by a Generative Adversarial Network (GAN). We believe that training a GAN extensively can generate results that are unique in a manner that sheds a light into unknown forms of malicious DNS traffic.

The variant of DGM proposed in this paper, the ConTraGAN, is a Conditional Generative GAN enhanced with Transformers. However, considering the nature of the problem we are trying to solve, we argue that the problem posed by the nature of network traffic shares a similar paradigm to Natural Language Processing (NLP) [5]. As such, a Self-Attention Mechanism (SAM) is also built into the ConTraGAN [5]. The SAM serves as a module responsible for providing attention weights to each feature of a packet, much like in NLP where the SAM learns latent vector embeddings of weighted values representing aspects of words such as common pairing with other words. This self-attention mechanism means that transformers perform best at sequence-to-sequence tasks, such as traffic packets, with strong predictive capabilities.

With this hybrid ConTraGAN model, we are able to synthesise unique data distributions that are close to that of the real data, so that previously unknown abnormal DNS traffic can be generated and used in training IDS systems for predicting and detecting malicious Zero-Day network attacks.

2. Related Work

There are a number of prominent studies that propose novel Machine Learning (ML) driven solutions for cyber threats and anomaly detection in network traffic.

Shoneet al. [6] approach the challenge of network intrusion from an unsupervised deep learning perspective with a Nonsymmetric Deep Autoencoder (NDAE). The authors argue that due to the limitation of the dataset they fitting their model on, the best solution is one that is capable of generalising the latent features inherent in the data. By taking advantage of a network built on two encoder blocks with a random forest classifier, their model is able to improve performance on a variety of known cyber-attacks on already existing network intrusion models.

A survey into GANs in cybersecurity worth noting is [7], which provides a guide into a number of research articles that delve into the applications of GANs in intrusion and malware detection. The paper surveys research done proposing a number of variations of GANs in both learning and replicating network traffic patterns so as to produce highly performant and accurate models. One of the research surveyed is [1], where Chen and Jiang investigated anomaly detection using a variant of Bidirectional GAN (BiGAN) with the KDD-99 cyber-intrusion dataset. Their BiGAN model takes advantage of the vanilla GAN's generator (G) and discriminator (D) features as well as an encoder (E) that learns the representations of the data space, allowing the generative elements of the GAN to increase the complexity of the data that the discriminator scrutinises.

Das et al. [8] developed a model with a two-layered architecture to detect DNS exfiltration attacks. The first layer is a classifier capable of handling high volume stateless features, and the second layer is built on a NN trained on stateful features. Any traffic that seems suspicious and requires further checking is forwarded to the second NN layer for a more detailed analysis, yielding a light-weight performant model.

A Self-Attention Generative Adversarial Network (SAGAN) [9] is used to explore the idea of SAM in image generation tasks in the form of attention masks as an extra channel in a Convolutional NN (CNN). The self-attention module is meant to complement convolutions and helps with applying attention in modelling long range, multi-level dependencies across regions of images that are otherwise missed by CNNs. With this attention mask added as a feature represented by an additional channel, the generator network is able to generate information in images in detail at every location whether these details occur in the foreground or in distant portions of an image. The self-attention driven Transformer architecture proposed in [5] has revolutionised language modelling and NLP. There the authors introduce a SAM that is capable of learning latent vector embeddings of weighted values representing aspects of words such as common pairing with other words. The SAM makes the transformers perform better at sequence-to-sequence tasks with strong predictive capabilities leading to the technology at the heart of chat-bots, sentiment analysis and recommendation systems to name a few. Similarly in network traffic analysis, a SAM could serves as a mode of providing attention weights to each feature of an IP packet as a subset of the packet as a whole.

An Attentional Generative Adversarial Network (AttnGAN) is proposed in [10], which incorporates attention into a multi-stage refinement process to solve the task of detailed image generation that is conditioned on text input. This model handles the process of encoding and deriving attention features from text inputs. The attention weights are fed through layers of network that sequentially maps these attentions to the latent space which the Generator then samples from. The generator then takes on the task of modelling the latent distribution of the sampled noise to match that of the labelled distribution that best represents the attention weights of the input text.

In "TransGAN: Two pure transformers can make one strong GAN, and that can scale up" [11], Jiang et al. takes a transformer-based approach to an image synthesis task, incorporating the SAM and a conditional label that further adds context with respect to the nature of the image. The TransGAN's generator samples latent noise that is fed-forward through a series of transformer encoder blocks with additional sublayers that up-sample the noise to a point where generated image can be passed through the discriminator network. In the discriminator, the image is filtered through a grid self-attention channel that adds positional information to each of the pixels relative to its grid location. Through this process, the discriminator is able to distinguish anomalous details in a loss function that is back propagated to the generator network for learning and recalibration.

All of the studies mentioned above shows that GANs have the flexibility and complexity necessary to provide a number of versatile perspectives into the possibility of automated cybersecurity solutions. It is in this context that our research into GANs and hybrid variants of the DGM are necessary to advance the state-of-the-art in network and information security.

3. Proposed Methodology

The network model in this work is a combination of a Conditional GAN (C-GAN) enhanced with the mechanism of a Transformer in particular the SAM mechanism [5].

3.1. GANs

A GAN is primarily made up of two NNs, a Generator G and a Discriminator D, with the G being that of a forger and the D being the decider between forged information and real information [12].

G works by sampling some noise z from a latent dimension $\mathcal{Z}$ and shapes the probability distribution of the sampled noise p_z such that it is indistinguishable from the probability distribution of the real data p_{data} from the $\mathcal{X}$. G and D are trained at the same time, with the objective of the G is to minimise its log loss with respect to the D's objective of maximising G's loss. This simultaneous minimax relationship of the objectives of the G and D is shown by the objective function $V(G,D)$ in (1), whereby D tries to maximize $V(G,D)$ whilst G tries to minimize $V(G,D)$.

$$\min_G \max_D V(G,D) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(Z)}[\log(1 - D(G(z)))]. \quad (1)$$

3.2. Conditional GAN

C-GAN is a variant of GANs whereby the model's process of generating and discriminating is augmented with a conditional data y. This condition y consists of any arbitrary information relevant to the expected outcome such as a class label if the intended outcome of the GAN is, for example, to produce images of numerical digits from the MNIST dataset with an input of class label [13].

The conditional data y is an input to both the G and D, with the objective function V(G,D), as defined in (2), to minimise $D(G(z|y))$ instead of $D(G(z))$ as is the case with the vanilla GAN.

$$\min_G \max_D V(G,D) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x|y)] + \mathbb{E}_{z\sim p_z(Z)}[\log(1 - D(G(z|y)))]. \quad (2)$$

3.3. Self-attention Mechanism

The SAM is used to solve the problem of sequence-to-sequence translation tasks such as the one we find in language translation [5]. It assigns learnable weights to each word as a subset of a sequence of words in a sentence. In doing so, each word will be a different weighted representation of the average of the whole sentence which provides a contextual value of each word in reference to its position in that sentence.

The SAM maps a query and a set of key-value pairs to an output, whereby the query Q, keys K, values V, and output are all matrix of the weighted sum of the values such that the weights assigned to each value is computed by a compatibility function of the query with its corresponding key. To simplify this, each word is represented by a vector of integers and the target is to compute the matrix multiplication or dot-product of each word as a query against the key and value vectors of the other words in the sentence so that each word is now a weighted sum of its own query, key and value against the sentence as a whole [5].

An attention function creates a rating for which element in a sequence of words is meant to receive the highest attention by taking the query and a key as inputs, and as a function of matrix multiplying between these inputs returns the attention scores as a weight of the query-key pair, as shown in (3) [5].

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad (3)$$

where d_k is the dimension of keys.

3.4. Transformer Block

In the transformer block, SAM is built into a multi-head attention block in an encoder, along with a layer normalisation network, a feed-forward multi-layer perceptron network and a final normalising layer in a stacked architecture (see Fig. 2). This is then repeated for a decoder block which has a similar structure except for an additional masked multi-head attention block that computes additional self-attention weights on the outputs from the encoder block [5].

3.5. Proposed Conditional Transformer-based GAN – ConTraGAN

The ConTraGAN architecture proposed in this research is inspired by the TransGAN as that of [10] but differs in compartmental design. We have chosen an architecture that reflects the nature of the problem we are dealing with, which has to do with self-attention as a mechanism that provides attention weights to each feature of the DNS traffic segment as a subset of the whole IP packet in much the same way as dealing with the challenge of assigning attention to words within a sentence in NLP.

The proposed ConTraGAN network is shown in Fig. 1. Given a vectorised conditional label y, the Generator works by sampling noise z_i, i∈[1, N], from a latent dimension Z = (z_1,…,z_N) and shapes the probability distribution of the sampled noise such that it is indistinguishable from the probability distribution of input sequence x_i from the real data dimension X = (x_1,…,x_N).

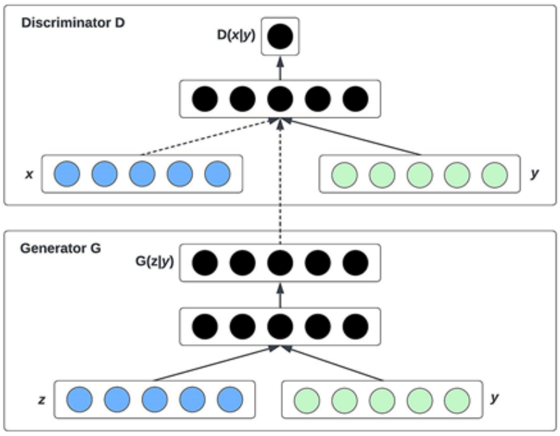


Fig. 1. Structure of a simple C-GAN [12].

The main argument of using the ConTraGAN is that with the components of a Transformer, we are able to better understand the attention each feature of a packet has in respect to the nature of that packet. For example, each DNS packet is a combination of record types with the main difference occurring with packets that have additional information increasing the entropy inherent in that packet [14]. Learning this representation is what we believe a Transformer model's SAM can handle better than any other NN. Furthermore, the sequential nature of this model is also a very strong reason why we believe a Transformer based model is best suited for deployment in an environment that requires analysis of high-volume traffic.

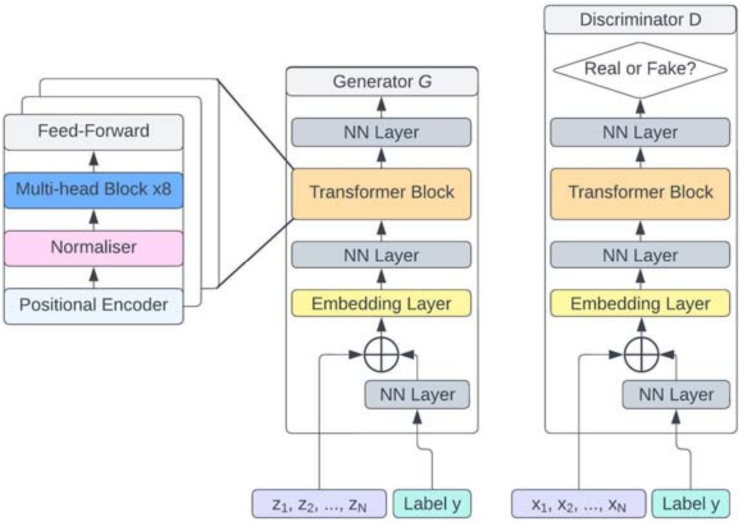


Fig. 2. The proposed network architecture of ConTraGAN

4. Experiments and Results

4.1. Dataset

The dataset used in this study is the CIC-Bell-DNS-EXF-2021 Dataset [15], which is collected with an active DNS data collection methodology using a controlled DNS exfiltration tool. The dataset is composed of more than one million benign, light attack and heavy attack samples with 30 features extracted from the captured DNS packets.

4.2. Model Training

The main idea behind our implementation is to build the models and tune them according to the expected outcomes for our model. Note that the main goal of our model is to be able to learn anomalous aspects in the features we feed into the model. By converting the inputs into byte tensors, we are able to create embedding vectors with greater depth added more towards the Q, K, V matrices. However, with this increase in dimensionality, we must be cautious not to stumble into the issue of potential model collapse in training GANs, as they are notorious for suffering from vanishing gradients when built with a deep architecture. The main elements that are likely to cause this issue are the activation and normalisation of the different layers. Due to the nature of the outputs from the Transformer, adding additional normalisation to the preceding linear layers for both G and D networks could create a situation where the gradients could vanish. To combat this issue, different combinations of dropout optimiser parameter values are experimented. In the end, optimiser Adam is selected with decay rates $\beta 1 = 0.09$ and $\beta 2 = 0.98$, and a learning rate $lr = 0.0001$.

We use Binary Cross Entropy Loss (BCELoss) as the loss function, as we found it yielded better results with a more stable training.

The ConTraGAN is trained with 100,000 samples from the dataset and the batch size is set to 200 for each epoch for a total of 20 epochs. Due to the prevailing hardware limitations and associated model complexity, the above-mentioned settings have proven to offer the most applicable configuration.

4.3. Results

Some of the main challenges faced during the training phase apart from finding the optimal model parameters had a lot to do with the metrics for this type of model. The expected outputs of a GAN normally consist of a generator G and a discriminator D. The output from the G is simply noise that approximates the shape and dimension of the input to the D. Therefore, the best metric to judge the performance of a GAN is the divergence between the output from the G and the input to the D during training. Our G on the other hand gives outputs of vector embeddings that convey the attention weights for the noise inputs that the D discriminates.

The ConTraGAN's loss in each training step is obtained and shown in Fig. 3. The objective of the G is to reduce its loss and increase the D's loss. From the results, it is clear that the loss objective function is approaching a point where both networks converge. We are sure that the proposed model will be producing the results we are anticipating.

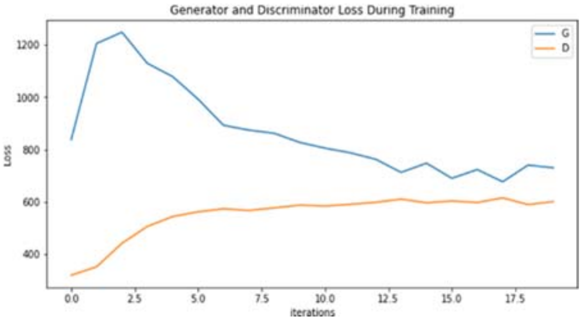


Fig. 3. Training loss of the ConTraGAN.

Fig. 4 adds further weight to the argument raised above. As illustrated, the measure of the distance between the predicted against the expected labels shows that the model would reach an optimum performance whereby the G is producing unique information that is similar to the distribution of our dataset. By learning this unique data, D is capable of distinguishing more anomalous behaviours in unseen data. It is this behaviour we argue that can help identify future Zero-Day based attacks.

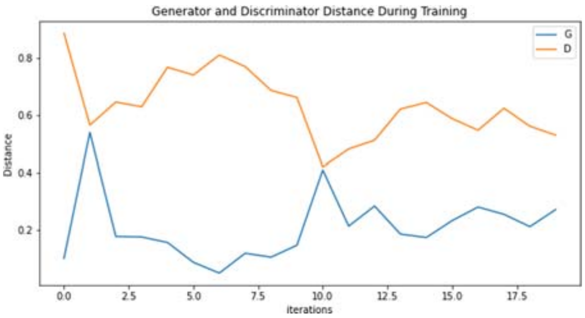


Fig. 4. Distance between G and D of the ConTraGAN.

4.4. Limitations

In our implementation, it would be better if the synthesised DNS extrafiltration data generated by the generator G in the ConTraGAN is fed into an IDS model for the following reasons:

a) To validate the effectiveness of ConTraGAN;
b) To evaluate the IDS performance with unknown attack traffic against known traffic;
c) To compare the performance of ConTraGAN and other GANs, specifically with the vanilla GAN and the C-GAN in the context of abnormal DNS traffic analysis and detection.

However, these have not been completed due to limited computational capacity with the complex large model. In future, we will make a comprehensive evaluation of the ConTraGAN as we have identified above.

5. Conclusion

The ConTraGAN model would reach an optimum performance whereby the generator G is producing unique information that is similar to the distribution of the dataset of real DNS exfiltration attack. By learning this unique data, the discriminator D is capable of distinguishing more anomalous behaviours in unseen data. It is this behaviour that can help identify future Zero-Day network attacks.

Acknowledgements

The authors would like to acknowledge the Engineering and Physical Sciences Research Council (EPSRC), UK for the support of this work (EP/W00366X/1).

References

[1]. H. Chen, L. Jiang, Efficient GAN-based method for cyber-intrusion detection, *arXiv Preprint*, 2019, arXiv:1904.02426.

[2]. M. A. Ferrag, L. Maglaras, S. Moschoyiannis, H. Janicke, Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study, *Journal of Information Security and Applications*, Vol. 50, 2020, 102419.

[3]. S. Sambangi, L. Gondi, A Machine Learning Approach for DDoS (Distributed Denial of Service) Attack Detection Using Multiple Linear Regression, *MDPI Proceedings*, Vol. 63, 2020, 51.

[4]. S. Mahdavifar, A. A. Ghorbani, Application of deep learning to cybersecurity: A survey, *Neurocomputing*, Vol. 347, 2019, pp. 149-176.

[5]. A. Vaswani, et al., Attention Is All You Need. in *Proceedings of the Conference on Neural Information Processing Systems (NIPS'17),* Vol. 31, 2017, pp. 1-11.

[6]. N. Shone, T. N. Ngoc, V. D. Phai, Q. Shi, A Deep Learning Approach to Network Intrusion Detection. *IEEE Transactions on Emerging Topics in Computational Intelligence*, Vol. 2, Issue 1, 2018, pp. 41-51.

[7]. I. K. Dutta, B. Ghosh, A. H. Carlson, M. W. Totaro, M. A. Bayoumi, Generative, Adversarial Networks in Security: A Survey, in *Proceedings of the 11th IEEE Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON'20)*, New York, NY, USA, 2020, pp. 0399-0405.

[8]. A. Das, M. Y. Shen, M. Shashanka, J. Wang, Detection of Exfiltration and Tunneling over DNS, in *Proceedings of the 16th IEEE International Conference on Machine Learning and Applications (ICMLA'17)*, Cancun, Mexico, 2017, pp. 737-742.

[9]. H. Zhang, I. Goodfellow, D. Metaxas, A. Odena, Self-Attention Generative Adversarial Networks, *arXiv Preprint*, 2019, arXiv:1805.08318.

[10]. T. Xu, et al., AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks, *arXiv Preprint,* 2017, arXiv:1711.10485.

[11]. Y. Jiang, S. Chang, Z. Wang, TransGAN: Two Pure Transformers Can Make One Strong GAN, and That

Can Scale Up, *arXiv Preprint*, 2021, arXiv:2102.07074.

[12]. I. Goodfellow, et al., Generative adversarial nets, in Advances in Neural Information Processing Systems, 2014, *Curran Associates, Inc*., pp. 2672-2680.

[13]. M. Mirza, S. Osindero, Conditional Generative Adversarial Nets, *arXiv Preprint*, 2014, arXiv:1411.1784.

[14]. S. Mahdavifar, et al., Lightweight Hybrid Data Exfiltration using DNS based on Machine Learning, in *Proceedings of the 11th International Conference on Communication and Network Security (ICCNS'21)*, 2021, pp. 80-86.

[15]. CIC-Bell-DNS-EXF-2021 Dataset, https://www.unb.ca/cic/datasets/dns-exf-2021.html

(025)

A Novel Intra Prediction Mode using Transformer-based GAN for VVenC

T. Katayama [1], T. Song [1], T. Shimamoto [1] and X. T. Jiang [2]
[1] Tokushima University, Industrial and Social Sciences, Graduate School of Technology, Tokushima, Japan
[2] Shanghai Maritime University, Department of Information Engineering, Shanghai, China
Tel.: + 816567482
E-mail: katayama@ee.tokushima-u.ac.jp

Summary: Video codec efficiency is improved by adding a new methodology. Moreover, AI-IoT environments have great potential to assist video codec. The intra prediction of versatile video coding (VVC) generates prediction pixels by using linear predictions based on some predefined directions. However, the intra prediction method involves a coding efficiency reduction for a high resolution with complexity texture. To solve this problem, we proposed a new intra prediction mode using the transformer-based generative adversarial network (GAN). The proposed new framework consists of a traditional video encoder and an embedded machine learning module including CNN and transformer. The simulation results show that the proposed algorithm can achieve an improvement of 22.44 % BD-rate Y compared to the original VVC algorithm.

Keywords: Versatile video coding (VVC), Intra prediction, Transformer.

---

1. Introduction

Video codec technology is implemented in various consumer devices. The video codec technology enables high compression transmission and reception. Currently, H.266/VVC (Versatile Video Coding) is recommended as a novel video coding standard in July 2020 [1, 2]. VVC inherits a hybrid coding structure that incorporates inter-frame and intra-frame coding tools from the traditional H.265/HEVC [3]. The intra coding efficiency of VVC has been improved by using increased intra-prediction modes. Additionally, an efficient H.266/VVC encoder, which is called Fraunhofer versatile video encoder (VVenC), is implemented [4]. To offer a flexible, fast, and user-friendly video encoding solution for the VVC standard, VVenC supports real-world encoder features such as frame-level rate control and visually optimized encoding. However, intra-prediction can lead to compression loss due to predicted pixels from spatial correlation.

Conventional intra prediction schemes predict pixel values using directional prediction. Generally, a complex texture is encoded with the directional intra prediction using a small prediction unit (PU). Thereby, VVC can achieve a highly efficient encoding for complex textures. However, large PU is not encoded with the directional intra prediction because the directional intra prediction makes a linear prediction. Therefore, the intra prediction of a large PU without using the directional intra prediction is essential for improving the coding efficiency.

Many modern devices are capable of deep learning processing. Deep learning has been utilized to accomplish visual recognition, image generation, and image enhancement with computing devices like GPU and TPU [5-11]. Video coding processing using the neural network have been proposed, for example, fractional interpolation, deblocking, and fast mode decision [12-16].

Previous works have focused on an improvement of the intra prediction. In particular, a previous work proposed progressive spatial recurrent neural network (PS-RNN) which use three spatial recurrent units are stacked sequentially [17]. The prediction method of PS-RNN achieved a variable-block-size configuration and the proposed loss function to improve the rate-distortion performance of the codec. However, encoding complex textures using large Prediction Units (PUs) remains a challenge for PS-RNN. Therefore, a new intra prediction mode for large PU is highly required. Moreover, previous works do not propose an efficient video coding method using a transformer-based generative adversarial model (GAN). For these reasons, our proposed algorithm contributes to the improvement of coding efficiency using transformer-based GAN when encoding the complex texture with large PU.

Our contributions can be summarized as follows:

1) The proposed algorithm using transformer-based GAN is implemented for VVenC;
2) The simulation results show that BD-rate Y is improved by about 22 % in random access mode and about 47 % in low-delay mode.

2. Related Works

2.1. Intra Prediction of VVC

The intra prediction of luma component in VVC supports six PUs (128×128, 64×64, 32×32, 16×16, 8×8, and 4×4). As shown in Fig. 1, the prediction pixel used for PU refers to the encoded neighboring blocks. Each PU has 67 prediction modes, including Planar mode, DC mode, and 65 angular modes, for luma intra prediction. The best mode in 65 intra prediction modes is selected based on the rate-distortion optimization (RDO). On the other hand, the intra prediction of chroma component is encoded with a mode number of a fixed value corresponding to the prediction mode of luma. Therefore, more accurate prediction pixel of luma.

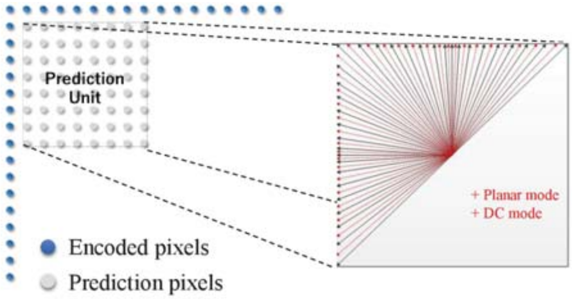


Fig. 1. Intra prediction mode of VVC.

Low delay (LD) mode and random access (RA) mode are adopted for the coding structure of VVC. Both LD mode and RA mode use frames encoded by intra prediction as reconstructed reference frames. Therefore, a highly accurate intra-prediction method can greatly contribute to achieving high compression efficiency.

2.2. Generator of Mask-aware Transformer

Mask-aware transformer for large hole image inpainting (MAT) is an image inpainting method designed to fill missing regions in an image, especially for large missing regions [18]. MAT architecture consists of several components, including a convolutional head, the main Transformer module, a convolutional tail, a style manipulation module, and a Conv-U-Net. These components work together to maximize the advantages of both Transformers and CNNs. The convolutional head and tail (CNN) are responsible for processing the input and output images, while the main transformer module performs the core image generation task. The style manipulation module allows for fine-grained control over the appearance of the generated images when training the model. Finally, the Conv-U-Net is used for upsampling and refinement of the generated images. The five components constitute the inpainting network, which restores the input images with missing regions using a combination of CNN and Transformer modules. Thereby, MAT architecture achieves image generation tasks that require both high quality and high efficiency.

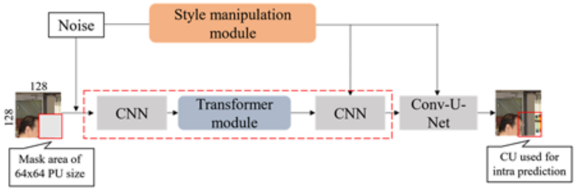


Fig. 2. MAT architecture.

A new transformer block (TB) is proposed in previous work [18]. The proposed method entails the elimination of layer normalization (LN) from TBs and the utilization of fusion learning with feature concatenation, instead of residual learning [19]. As shown in equations 1 and 2, we concatenate the input and output of attention and apply a fully connected (FC) layer. Moreover, $X_{k,l}$ is calculated by using a Multi-Layer Perceptron (MLP) with the outputted FC layer.

$$X'_{k,l} = FC([MCA(X_{k,l}), X_{k,l-1}]), \tag{1}$$

$$X_{k,l} = MLP(X'_{k,l}), \tag{2}$$

where $X_{k,l}$ is represented with the TB of the l-th block in the k-th stage. Specifically, multi-head contextual attention (MCA) of h-th head is formulated as

$$\begin{aligned} MCA_{\square}(X_{k,l}) &= Attention_{\square}(Q,K,V) = \\ &= Softmax\left(\frac{QK^T+M'}{\sqrt{d_k}}\right)V, \end{aligned} \tag{3}$$

where Q, K, V are the query, key, value matrices and $\frac{1}{\sqrt{d_k}}$ is the scaling factor. The M' indicates the region to be masked in Attention.

This approach facilitates the efficient completion of pixels with large holes. However, an optimal encoder structure using MAT does not proposed. Hence, an optimal MAT structure is required for VVenC. We proposed a transformer-based GAN inspired by MAT.

3. Proposed Algorithm

Our proposed algorithms consist of two parts. A machine learning module is implemented for VVenC, which is described in detail. Additionally, the proposed intra prediction of VVenC is improved by extending the number of prediction modes. As shown in Fig. 3, the machine learning module generates prediction pixels for each prediction unit (PU) size. The predicted pixels can be used to VVenC. Our approach aims to improve coding efficiency by the high accuracy prediction of complexity texture with large PU.

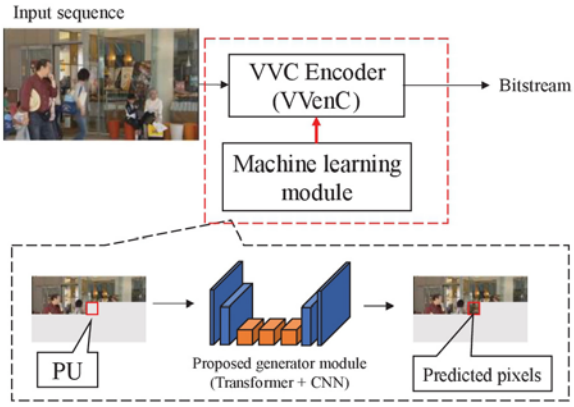


Fig. 3. Overall processing of the proposed algorithm.

3.1. Proposed Generator Module

VVC encoder (VVenC) with the trained machine learning module are synchronized, as shown in Fig. 3. In our approach, the trained machine learning module is achieved by offline training. The trained machine learning module is performed with the current block of 128×128 size as input data. The 64×64 PU which is output by the trained machine learning module is supplied to VVenC as the predicted PU of our proposed intra prediction mode.

As shown in Fig. 4, compared to the conventional generator module based on MAT, our proposed architecture features a reduced number of TBs and CNN layers in the generator module. Specifically, the proposed method reduces the number of CNN layers from 8 to 4 and the number of TBs from 5 to 2. To optimize feature extraction without adding unnecessary complexity, we adjusted the number of CNN layers to avoid excessive layer depth. Moreover, to promote long-range relationships and stabilize learning in images with large missing regions, we included TBs in our architecture. We set the number of TBs to ensure minimum stability of the learning process. This architecture is designed to balance the benefits of both TBs and CNNs, leveraging the strengths of each to achieve high-quality image prediction while maintaining computational efficiency.

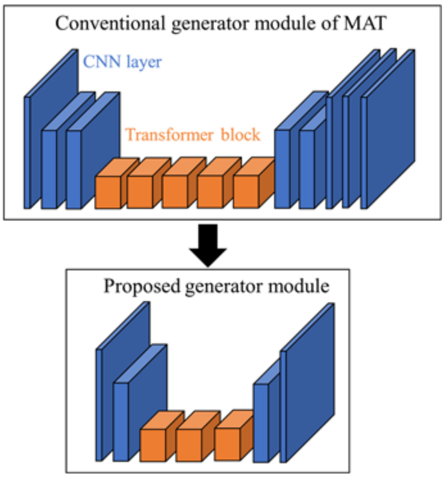


Fig. 4. The proposed generator architecture.

3.2. Implementation of a Machine Learning Module for VVenC

Variable block sizes are supported for VVenC. The size of the prediction unit (PU) can range from 128×128 to 4×4, which is determined by the split strategy of the VVC standard. Due to the limitations of the angular prediction mode, VVC tends to split large blocks into smaller ones when their textures are complex. This requires more bits to encode the prediction mode due to the split. Our proposed generator module generates more accurate prediction pixels for large blocks compared to VVC.

Due to the diverse nature of video content, it is challenging for a single intra prediction mode to handle all conditions. Therefore, the optimal intra prediction mode is determined by RDO calculation, which selects either the original VVC intra prediction modes or the proposed intra prediction mode. The RDO calculation of the proposed intra prediction mode is performed with the predicted pixels of a 64×64 block. Additionally, to support variable PU size, when other PU sizes are encoded, the separated 64×64 block is utilized for the proposed intra prediction. Thereby, the proposed method allows the intra prediction mode from 64×64 PU to 4×4 PU, and can contribute to the improvement of coding efficiency.

4. Simulation Results

All proposed algorithms are implemented in reference software VVenC version 1.6.1 [4]. A coding structure is used with RA mode and LD mode configuration. The Quantization Parameter (QP) is set to 22, 27, 32, and 37. The simulation environment is Intel(R) Core(TM) i7-9700 CPU 4.70 GHz with 8 cores, RAM 32.00 GB, and Windows 10 Home Edition 64-bit. The encoding performance is represented by BD-rate [20]. Additionally, the difference value of bitrate between the unmodified VVenC and the proposed algorithm is represented as bitrate reduction (BR). BR is defined as

$$BR(\%) = \frac{Bitrate_{proposed} - Bitrate_{VVenC}}{Bitrate_{VVenC}}, \tag{4}$$

where $Bitrate_{VVenC}$ is the bitrate of the unmodified VVenC and $Bitrate_{proposed}$ is that of the proposed algorithm.

For training on the proposed generator architecture, we use a batch size of 8 and train the models on 416 blocks of size 128×128. We adopt an Adam optimizer with $\beta_1 = 0$ and $\beta_2 = 0.99$. The initial learning rate is set to 0.001. To prevent overfitting, we perform a step-wise decay of the learning rate by a factor of 0.1 at the iterations of {200, 500, 1000}. We choose the model of the lowest validation loss of the last 200 iterations. batch size is 8. A training process takes about 20 hours on an NVIDIA RTX A6000 GPU.

As shown in Fig. 3, we present a visualization of the prediction results for a 64×64 block using both MAT and the proposed generator module. If the predicted 64×64 block is close to raw data, the performance is good. From the results, the proposed generator module can better utilize global contexts. Moreover, the inference time of the proposed generator module decreased from 1.0 seconds per block (s/b) to 0.7 s/b compared to MAT. Table 1 represents the inference time and encoding time of the proposed algorithm for one frame. From the results in Table 1, the inference time of the proposed generator module is longer than VVenC encoding time.

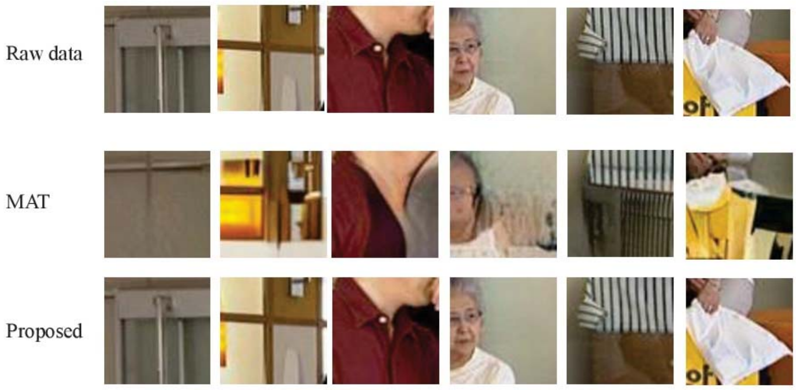


Fig. 5. The comparison of MAT and the proposed generator by subjective evaluation.

Table 1. Inference time and encoding time of the proposed algorithm for one frame.

	Inference time [s]	Encoding time [s]
LD mode	72.8	19.38
RA mode	72.8	11.76

The simulation results are shown in Table 2 and Table 3. Table 2 represents BD-rate comparison results for random access coding of Y, U, and V by our proposed algorithm and the VVenC. Our proposed algorithm improved BD-rate by an average of 22.44 %, 17.10 %, and 16.47 %, which represent Y, U, and V, respectively. Moreover, Table 3 shows that the performance of our algorithm is compared with the VVenC algorithm in low-delay coding. The results given in Table 3 indicate that the proposed algorithm improves BD-rate Y by about 47.34 %. LD mode improves coding efficiency over RA mode. As the proposed method generates images exclusively for intra frames, the difference between the first and subsequent frames increases as the number of frames increases. Therefore, continuous processing of a GOP with LD mode enables more efficient utilization of intra frames. In particular, the coding efficiency is greatly improved with the Y component of BasketballDrill in LD mode.

Table 4 and Table 5 show the BR achieved by the proposed algorithm at each QP. From the results, the proposed algorithm achieves significant bitrate reduction (BR) at higher QP values, indicating that it can contribute greatly when a low bitrate stream is required. This is because the proposed algorithm is designed to improve the coding efficiency of intra frames, which are often referenced to inter coding.

4. Conclusions

In this work, the intra prediction mode using transformer-based GAN is proposed for VVenC. The proposed algorithm can generate more accurate prediction pixels for intra prediction. The simulation results show that the improvement of coding efficiency is achieved. In future work, a more efficient transformer-based GAN model and the complexity reduction method will be discussed.

Table 2. Performance evaluation of the proposed algorithm and VVenC with RA mode.

Resolution	Sequence	BD-rate[%]		
		Y	U	V
ClassC (832×480)	BasketballDrill	-32.23	-18.26	-18.00
	BQMall	-24.52	-22.15	-21.48
	PartyScene	-20.15	-18.49	-16.86
	RaceHorses	-12.86	-9.50	-9.55
	Average	-22.44	-17.10	-16.47

Table 3. Performance evaluation of the proposed algorithm and VVenC with LD mode.

Resolution	Sequence	BD-rate[%]		
		Y	U	V
ClassC (832×480)	BasketballDrill	-60.52	-39.05	-38.89
	BQMall	-51.93	-49.63	-48.99
	PartyScene	-42.63	-39.54	-39.03
	RaceHorses	-34.27	-27.96	-28.77
	Average	-47.34	-39.05	-38.92

Table 4. Evaluation of BR (%) with four QPs (namely 22, 27, 32, 37) in RA mode.

Sequences	QP=22	QP=27	QP=32	QP=37
BasketballDrill	-13.12	-18.57	-21.06	-21.09
BQMall	-5.74	-16.33	-29.94	-38.24
PartyScene	-5.5	-12.79	-23.7	-35.83
RaceHorsesC	-2.43	-9.28	-14.43	-19.28

Table 5. Evaluation of BR (%) with four QPs (namely 22, 27, 32, 37) in LD mode.

Sequences	QP = 22	QP = 27	QP = 32	QP = 37
BasketballDrill	-31.51	-40.17	-43.23	-41.89
BQMall	-17.84	-41.96	-60.74	-66.25
PartyScene	-16.51	-32.96	-50.18	-60.02
RaceHorsesC	-13.47	-27.01	-36.09	-39.78

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers 22K17913 and 20K11790.

References

[1]. J. Brandenburg, A. Wieckowski, T. Hinz, A. Henkel, V. George, I. Zupancic, C. Stoffers, B. Bross, H. Schwarz, D. Marpe, Towards fast and efficient VVC encoding, in *Proceedings of the International Workshop on Multimedia Signal Processing (MMSP'20)*, Tampere, Finland, 21-24 September 2020, pp. 1-6.

[2]. K. Suehring, X. Li, JVET common test conditions and software reference configurations, in *Proceedings of the 8th Meeting*, *JVET-H1010,* Macao, CN, 18-25 Oct. 2017.

[3]. G. J. Sullivan, J.-R. Ohm, W.-J. Han, T. Wiegand, Overview of the high efficiency video coding (HEVC) standards, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 22, Issue 12, 2012, pp. 1649-1668.

[4]. A. Wieckowski, J. Brandenburg, T. Hinz, C. Bartnik, V. George, G. Hege, C. R. Helmrich, A. Henkel, C. Lehmann, C. Stoffers, I. Zupancic, B. Bross, D. Marpe, VVENC: An open and optimized VVC encoder implementation, in *Proceedings of the IEEE International Conference on Multimedia & Expo Workshops (ICMEW'21)*, Shenzhen, China, 5-9 July 2021, pp. 1-2.

[5]. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'16)*, Las Vegas, NV, USA, 27-30 June 2016, pp. 770-778.

[6]. Y. Li, S. Song, Y. Li, J. Liu, Temporal bilinear networks for video action recognition, in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI'19)*, Honolulu, Hawaii, USA, Jan. 27-Feb. 1, 2019.

[7]. S. Song, C. Lan, J. Xing, W. Zeng, J. Liu, Spatio-temporal attention-based LSTM networks for 3D action recognition and detection, *IEEE Transactions on Image Processing*, Vol. 27, Issue 7, 2018, pp. 3459-3471.

[8]. J. Liu, W. Yang, X. Zhang, Z. Guo, Retrieval compensated group structured sparsity for image super-resolution, *IEEE Transactions on Multimedia*, Vol. 19, Issue 2, 2016, pp. 302-316.

[9]. W. Yang, J. Feng, J. Yang, F. Zhao, J. Liu, Z. Guo, S. Yan, Deep edge guided recurrent residual learning for image super-resolution, *IEEE Transactions on Image Processing*, Vol. 26, Issue 12, 2017, pp. 5895-5907.

[10]. W. Yang, S. Xia, J. Liu, Z. Guo, Reference guided deep super resolution via manifold localized external compensation, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 29, Issue 5, 2019, pp. 1270-1283.

[11]. N. P. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, N. Boden, A. Borchers, et al., In-datacenter performance analysis of a tensor processing unit, in *Proceedings of the Annual International Symposium on Computer Architecture (ISCA'17)*, New York, NY, USA, 24-28 June 2017, pp. 1-12.

[12]. S. Xia, W. Yang, Y. Hu, S. Ma, J. Liu, A group variational transformation neural network for fractional interpolation of video coding, in *Proceedings of the Data Compression Conference (DCC'18)*, Snowbird, UT, USA, 27-30 March 2018, pp. 127-136.

[13]. X. Zhang, W. Yang, Y. Hu, J. Liu, DMCNN: Dual-domain multi-scale convolutional neural network for compression artifacts removal, in *Proceedings of the IEEE International Conference on Image Processing (ICIP'18)*, Athens, Greece, 07-10 October 2018, pp. 390-394.

[14]. Z. Liu, X. Yu, S. Chen, D. Wang, CNN oriented fast HEVC intra CU mode decision, in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS'16)*, Montreal, QC, Canada, 22-25 May 2016, pp. 2270-2273.

[15]. J. Li, B. Li, J. Xu, R. Xiong, Intra prediction using fully connected network for video coding, in *Proceedings of the IEEE International Conference on Image Processing (ICIP'17)*, Beijing, China, 17-20 September 2017, pp. 1-5.

[16]. W. Cui, T. Zhang, S. Zhang, F. Jiang, W. Zuo, Z. Wan, D. Zhao, Convolutional neural networks based intra prediction for HEVC, in *Proceedings of the Data Compression Conference (DCC'17)*, Snowbird, UT, USA, 04-07 April 2017.

[17]. Y. Hu, W. Yang, M. Li, J. Liu, Progressive Spatial Recurrent Neural Network for Intra Prediction, *IEEE Transactions on Multimedia*, Vol. 2, Issue 12, 2019, pp. 3024-3037.

[18]. W. Li, Z. Lin, K. Zhou, L. Qi, Y. Wang, J. Jia, MAT: Mask-aware transformer for large hole image inpainting, in *Proceedings of the Computer Vision and Pattern Recognition (CVPR'22)*, New Orleans, LA, USA, 18-24 June 2022, pp. 10748-10758.

[19]. J. L. Ba, J. R. Kiros, G. E. Hinton, Layer normalization, *arXiv Preprint*, 2016, arXiv:1607.06450.

[20]. G. Bjøntegaard, Document VCEG-M33: Calculation of average PSNR differences between RD-curves, in *Proceedings of the ITU-T VCEG Meeting*, Tech. Rep., Austin, Texas, USA, 2001.

(026)

A Two-Phase Regularization Approach to Enhance Generalization and Convergence in Neural Networks

Mingkun Xu, Jing Pei and Lei Deng
Center for Brain-Inspired Computing Research (CBICR), Department of Precision Instrument,
Tsinghua University, Beijing, China
E-mail: {peij,leideng}@mail.tsinghua.edu.cn

Summary: Overfitting and slow convergence are common challenges when training deep neural networks (DNNs). While regularization techniques such as dropout and weight decay can help mitigate these issues, they may not be effective for more complex models and hardware deployment. In this paper, we propose a two-phase regularization approach to improve the generalization and convergence speed of neural networks. Different from canonical dropout techniques that randomly drop a number of neurons or connections during training, our approach consists of a two-phase transition mechanism which can randomly select large weights that are over a threshold and convert them to opposite phase state according to a certain rule before each training batch starts. Specifically, the pre-treated weights will be recovered to their original values and then updated using the corresponding gradients calculated from the pre-treated DNN. This corresponds to an interconversion mechanism between high-impedance state and low-impedance state. We evaluate our approach on CIFAR-10 benchmark dataset, and show that it outperforms other regularization techniques in terms of convergence speed meanwhile maintaining comparable generalization performance, providing a promising way to boost neural network training and applications.

Keywords: Regularization, Deep neural networks, Convergence, Generalization, Two-phase transition.

---

1. Introduction

Deep neural networks (DNNs) have shown remarkable success in various domains such as computer vision, natural language processing, and speech recognition. However, training deep neural networks is a challenging task due to issues such as overfitting and slow convergence. Overfitting occurs when the model learns the training data too well and fails to generalize to new, unseen data. Slow convergence, on the other hand, can lead to long training times and may prevent the model from achieving optimal performance.

Traditionally, a typical DNN has so many parameters (over-parameterized) and limited available data, exhibiting a significant tendency toward overfitting on the training dataset. Approaches to reduce error in generalizing to out-of-sample data points are referred to as regularization methods [1]. Generally speaking, the rationale behind regularization is constraining the complexity of the DNN model by either reducing the number of synaptic connections or reducing values of synaptic weights. Dropout [2, 3] and its variant DropConnect [4] are two of the most widely used regularization methods. As their names suggest, these two methods randomly drop a number of neurons or connections for each batch during training. With this, not all but only a fraction of weights are updated. After the training on a batch of samples, the omitted neurons (and connections attached to it) or omitted connections in the last epoch are recovered. A new set of neurons or connections are randomly selected and omitted in the next training batch. Both are commonly used to address overfitting problems. However, these techniques may not always be sufficient for complex models with large numbers of parameters and hardware deployment. Unlike these two methods, our method adapts the idea of reducing values of synaptic weights to improve generalization performance.

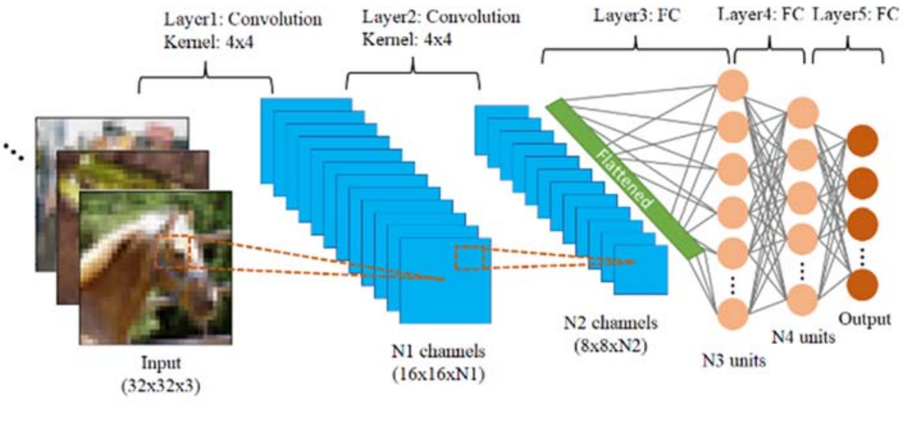


Fig. 1. Schematic illustration of the adopted network structure for image recognition.

In this paper, we propose a two-phase regularization approach to enhance the generalization and convergence of neural networks. The proposed approach, called PNdrop, consists of a two-phase transition mechanism which can randomly select large weights that are over a threshold and convert them to an opposite state according to a certain rule before each training batch starts. Specifically, the pre-treated weights will be recovered to their original values and then updated using the corresponding gradients calculated from the pre-treated DNN. The principle is to utilize the "PNdrop" mechanism to finish flexible transition of weights in forward and backward path, enabling all of the weights can be updated unevenly during one batch of training. This corresponds to an interconversion mechanism between high-impedance state ('negative') and low-impedance state ('positive'), which can discourage the network from overfitting to the training data and encourage the network to explore different solutions and converge faster by perturbing the weights during training using two-phase transition.

2. Methods

In a deep neural network (DNN), large weights can indicate a complex network that is prone to overfitting the training data [5]. To address this issue, we propose a method where, before the start of each training batch, randomly selected weights exceeding a predefined threshold are converted to a high-impedance state based on a specific rule. We adopt an inverse scaling transition rule, which aligns well with the characteristics of PN-transition memristors. Following this pre-treatment step, the standard backpropagation (BP) method is employed to compute gradients for all weights in the DNN. With the exception of the pre-treated weights, the remaining weights are directly updated using the computed gradients. The pre-treated weights are first restored to their original values and then updated using the corresponding gradients calculated from the pre-treated DNN. In the subsequent training batch, a fresh set of large weights is randomly selected and scaled down, followed by the same process of BP calculations and weight updates.

In this study, we evaluate the performance of our proposed PNdrop method in comparison to the Dropout and DropConnect methods on a five-layer convolutional neural network (CNN), as depicted in Figure 1. It is important to note that the PNdrop method is exclusively applied to the fully-connected (FC) layers, similar to Dropout and DropConnect. To assess the effectiveness of our approach, we conduct experiments on the CIFAR-10 dataset, which comprises 50,000 training images and 10,000 testing images of size 32 × 32, distributed across 10 classes. The network architecture employed is denoted as [Input-12C4-12C4-768FC-256FC-64FC-10], where C represents the convolutional layer and FC signifies the fully-connected layer. In our implementation, we incorporate the mechanism of inverse proportionality to model the transition process between large and small values. The PNdrop method is then applied specifically to the FC layers. During optimization, we utilize the adaptive moment estimation (Adam) optimizer with an initial learning rate of 0.0005 and a batch size of 100. The models undergo training for 160 epochs with a retaining proportion of 0.5 for the dropout models. For PNdrop, we randomly select half of the synaptic connections and dynamically apply PN transition based on their actual values during training. During the validation and testing phases, the selected connections from the last iteration are restored. It is noteworthy that the hyperparameters remain consistent across all implemented models in this study. To perform the simulations, we employ TensorFlow 1.15.0 on four RTX 2080Ti GPUs, which enables efficient computation and analysis of the models.

3. Results

Fig.2a and 2b demonstrate the performance of a CNN trained with various regularization techniques, namely Dropout, DropConnect, and PNdrop. Compared to the baseline model without any regularization technique, the CNN models incorporating these regularization techniques exhibit higher validation accuracies and lower loss values. Moreover, the gaps between the training and validation curves are significantly reduced, indicating improved generalization performance through the use of these regularization methods. It is worth noting that while the final validation accuracy and loss values of the CNN implemented with the PNdrop method are similar to those of the CNNs implemented with Dropout and DropConnect, the PNdrop method demonstrates faster convergence. This can be attributed to the fact that all weights are updated in each training iteration in the CNN regularized by PNdrop, whereas in the Dropout or DropConnect approach, connections that are omitted are simply not involved in weight updating.

The mean, standard deviation, and sum of absolute values of weights are compared among CNN models that are regularized using different methods, as illustrated in Fig. 2c-e. It is observed that our PNdrop method leads to predominantly excitatory connections, indicated by a positive mean value. In contrast, the baseline CNN without regularization and CNNs regularized by Dropout and DropConnect exhibit overall inhibitory connections, reflected by a negative mean value. Additionally, our PNdrop method yields the largest standard deviation and sum of absolute values of weights. The weight distributions are also visualized, with the truncated Gaussian distribution used for weight initialization shown in Fig. 3a. Following training, the baseline model without regularization demonstrates a wide weight distribution, with many weights having large values (Fig.3a). In contrast, the weight distributions for Dropout and DropConnect are narrower, centered around zero (Fig. 3b, c). Notably, the weight distribution in the CNN regularized by the PNdrop method is multimodal and even narrower, with several peaks located close to zero but with smaller values (Fig. 3d). These results highlight the distinctive characteristics of our PNdrop method, which enables state-of-the-art performance.

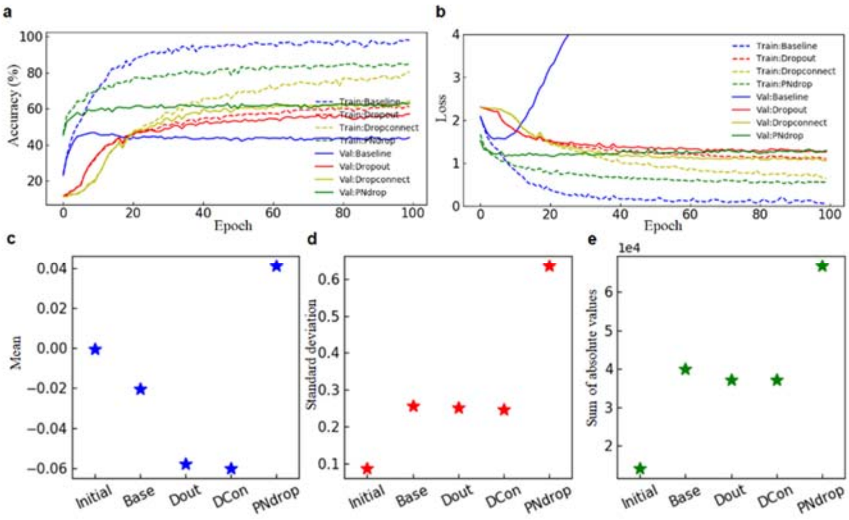


Fig. 2. Performance and characteristics of the CNNs regularized by different methods. Comparison of the **a.** convergence curves and **b.** loss curves obtained from unregularized model and models regularized by different methods. Comparison of the **c.** mean, **d.** stand deviation and **e.** sums of absolute values of synaptic weights among different models after training, where "Initial" denotes the untrained model, "Base" denotes unregularized baseline model, "Dout", "DCon" and "PNdrop" denote models regularized by Dropout, DropConnect and PNdrop methods, respectively.

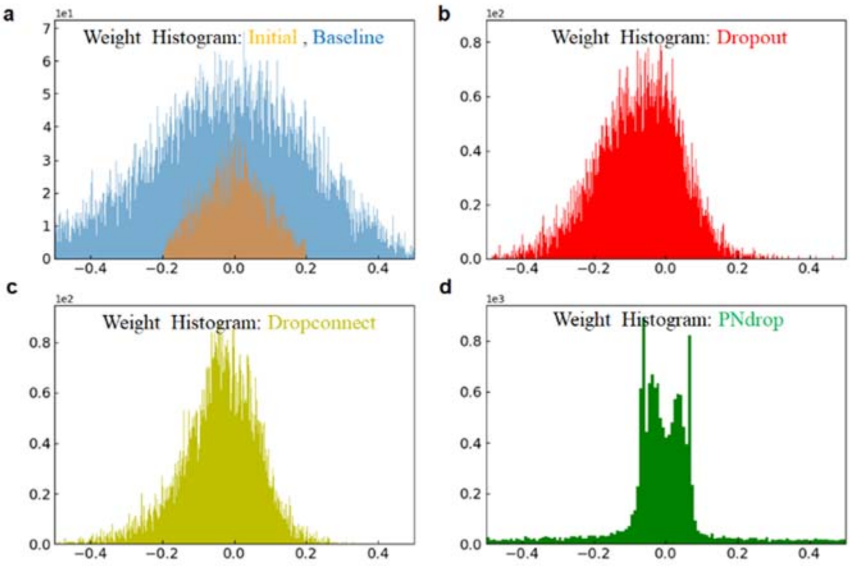


Fig. 3. a. The untrained model and the unregularized baseline model, and models regularized by **b.** Dropout, **c.** DropConnect, and **d.** PNdrop methods, respectively.

4. Conclusions

In this paper, we proposed a two-phase regularization approach to empower networks with enhancing generalization but far superior convergence efficiency than canonical regularization techniques. Our experimental results show that our proposed approach achieves better performance and faster convergence compared to other regularization techniques. Besides, our approach is simple to implement and can be used in various deep learning models, providing a promising way to boost neural network applications.

Acknowledgements

This work was supported by Science and Technology Innovation 2030 - New Generation of Artificial

Intelligence, China project (2020AAA0109101) and National Natural Science Foundation of China (No. 62106119, 62276151) and Zhejiang Lab's International Talent Fund for Young Professionals.

References

[1]. Larsen, Jan, and Lars Kai Hansen, Generalization performance of regularized neural network models, in *Proceedings of IEEE Workshop on Neural Networks for Signal Processing*, IEEE, 1994, pp. 42 - 51.

[2]. Hinton, Geoffrey E., et al., Improving neural networks by preventing co-adaptation of feature detectors, *arXiv preprint arXiv:1207.0580*, 2012.

[3]. Srivastava, Nitish, et al., Dropout: a simple way to prevent neural networks from overfitting, *The Journal of Machine Learning Research*, 15, 56, 2014, pp. 1929-1958.

[4]. Wan, Li, et al., Regularization of neural networks using dropconnect, in *Proceedings of the 30th International Conference on Machine Learning (PMLR),* 28, 3, 2013, pp. 1058-1066.

[5]. Ghojogh, Benyamin, and Mark Crowley, The theory behind overfitting, cross validation, regularization, bagging, and boosting: tutorial, *arXiv preprint arXiv:1905.12787* (2019).

(029)

The Use of a Fuzzy Controller in the Machining of Aircraft Engine Components

M. Muszyńska, A. Burghardt, D. Szybicki, P. Gierlak and K. Kurc
[1] Department of Applied Mechanics and Robotics, Faculty of Mechanical Engineering and Aeronautics,
Rzeszow University of Technology, 35-959 Rzeszów, Poland
Tel.: + 178651377, fax: + 178543116
E-mail: magdaw@prz.edu.pl

Summary: The paper presents the selection of appropriate parameters in the machining of a blade edge through a proposed proprietary solution based on fuzzy logic systems. Thanks to special ATOS Professional software, a CAD model of a blade was compared with an obtained three-dimensional image. This allowed for the creation of a quality control report in the form of a colour map of deviations at specific measurement points. For data obtained in this way, a fuzzy model was proposed in order to determine the dependence of the amount of the material layer removed on the tool clamping force. The entire process of creating a fuzzy controller was presented as well as a graphical interpretation of the model surface.

Keywords: Aircraft turbine blade, Diagnostics, Fuzzy logic systems, Robot.

1. Introduction

The blade is one of the basic elements of rotating machines, it is responsible for its effective and failure-free operation. Damage to the blade may lead to machine failure, and in special cases (the tearing off of a fragment or the entire blade) to complete destruction of the rotating machine, which in turn leads to catastrophic accidents. Blades are manufactured using many methods by means of very advanced monocrystalline casting methods, they are produced on CNC machines with the EDM method. This method is used most often; however, it turns out that it is sensitive to a number of factors that cause changes of process parameters. Unfortunately, this leads to a lack of repeatability of the shape of the blades (Fig. 1).

Fig. 1. Blade printed by DMLS method.

In the figure it is easy to see the existing defects as well as an excess of material on the surface of the blade. The mentioned uniqueness of the blade shape requires the use of a manual process of grinding and finishing polishing. Therefore, the possibility of automating the process of grinding blades deserves attention. The problem of grinding and polishing the blades using the Yamaha ZII robot was analyzed in the article [1], the focus was on the automation of such a process, which can lead to a significant cost reduction and quality improvement. This article presents the implementation of the SMART robotic system for automatic grinding and polishing of blade blades. On the other hand, a similar approach using the ABB IRB4400_45 robotic station is described in publication [2]. This publication introduces a novel robotic belt grinding method that focuses primarily on system calibration and force control to improve grinding performance. The general theory is described and the experimental results are presented. However, this article focuses on keeping the clamping force constant and we want this force to be variable depending on the amount of material removed. The authors of publications [3-9] have been dealing with such problems for years. These articles describe in detail the robotic machining processes and diagnostics of the mentioned blades. Article [3, 4] presents a robotic control station for measuring blade wall thickness using an ABB IRB 1600 robot, equipped with an Atos Core 3D scanner and connected to the Atos Professional software package. Paper [5] presents a robotic test stand that uses ultrasonic transmission tomography (UTT) to control the parameters of stator blades. The article [9] presents the subject of verification of the module being part of a robotic workstation for machining turbojet engine components. The task of the module was to measure the geometrical parameters of the compressor blades of a turbojet engine. Noteworthy are also articles on how to remove the allowance on the blade (excess material). The work [10, 11] focuses on presenting a platform for modeling and simulating a robotic belt grinding system. A model of material removal during grinding was developed and presented, which can be adapted to workpieces with complex shapes. The problem of selecting the clamping force and speed of the robot is discussed here. In turn, the article [12] proposes an adaptive modeling method based on statistical machine learning. It involves building an initial support vector regression model using grinding data as training samples. Such a trained model is modified according to the in-situ measurement data. The robot control parameters can then be calculated by modeling the grinding process. However, the selection of these parameters is not adaptive in real time.

The blade referred to in this article was made by chemical EDM. This process is very unstable, therefore the processing of such a blade is difficult. Requires the removal of sometimes less and sometimes more material during grinding. This requires the use of a variable clamping force, unlike in the case of blades made on a CNC machine, where machining is much simpler. For this purpose, in this article, a fuzzy model was proposed for the selection of the force collecting excess material

2. Data Collection System

The robotic station includes an IRB 140 robot, a grinding tool and an IRB 1600 robot with a 3D scanning head installed. The measuring system works with ATOS Professional software and communicates with a robot controller (IRC5) using the TCP/IP protocol. The RobotWare robot controller software controls two robots and is additionally equipped with a force control option. The described robotic station (Fig. 2) was used for research work related to the creation of an intelligent system of cutting force selection in order to remove the appropriate layer of material (allowance).

Fig. 2. Robotic station with IRB 140 and IRB 1600 robots, a grinding tool and a 3D scanning head.

Details of the blade machining process on the proposed stand are described in articles [3-7]. Information about the shape of the machined workpiece was obtained thanks to the use of the Atos Core 3D scanner [13]. During scanning, a three-dimensional image of the blade was obtained (Fig. 3b), which was compared with the reference CAD model (Fig. 3a). As a result, an image was obtained with visible allowances on the surface of the blade (Fig. 3c). The generated deviations are presented in three colours. The green colour means that the dimension is within the tolerance, the yellow colour is on the border of the tolerance, and the red colour is outside the tolerance.

(a) (b) (c)

Fig. 3. Aircraft engine blade: a) CAD model, b) image obtained with a scanner, c) final result of matching the CAD model and the scan.

The next step was to generate a quality control report (Fig. 4) using the ATOS Professional V 7.5 SR2 program.

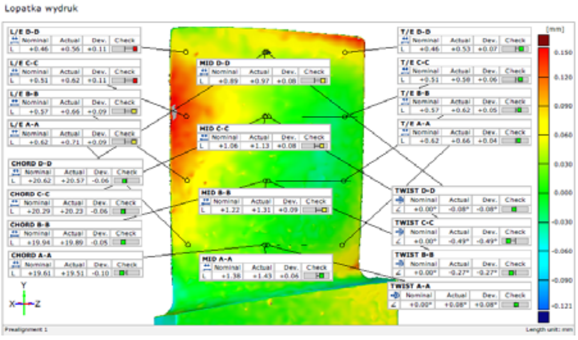


Fig. 4. Quality control report.

Analysis of the geometric accuracy of the blade can be performed in the form of detailed deviations at selected points on the blade surface (Figs. 5, 6), as well as with the use of Analyze Blades. Fig. 5 shows a report presenting the thickness of the blade at specified measuring points.

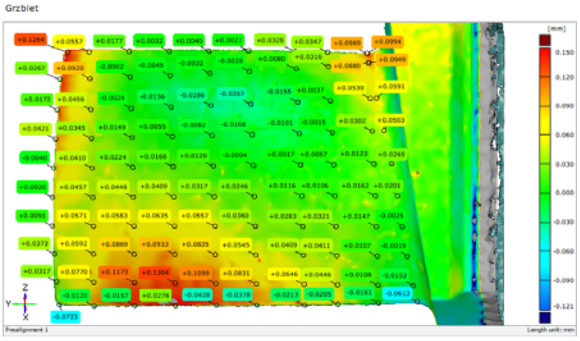


Fig. 5. Colour map of deviations with measurement points for the blade edge.

Excess material can be seen on the blade, as well as some cavities.

The applied data collection system allowed us to obtain information on the relationship between the cutting force and the amount of material removed in the process of blade grinding. This information concerned 80 measurement points. The data obtained in this way was sent to MATLAB/Simulink software.

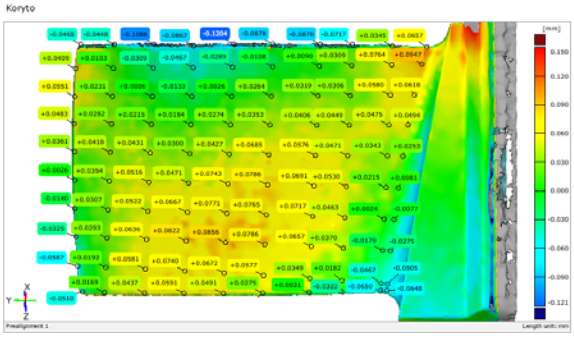


Fig. 6. Colour map of deviations with measurement points for the blade trough.

3. Fuzzy Process Controller

On the basis of the obtained measurements, a fuzzy model was proposed which defines the cutting force that should be applied to remove the appropriate amount of allowance at a given measuring point. A fuzzy model of the Mamdani type, fuzzy implication type (MIN) and a sharpening method type (CA) were adopted. 80 regularly spaced measurement points were adopted (Fig. 7), where measurements were made to determine the value of the allowance. Three clamping forces were used to build the fuzzy model: 9[N], 5[N] and 3[N]. The values of the proposed cutting forces result from the fact that forces less than 2[N] are not used because with them there is no possibility of force control in the robot control system and forces exceeding the value of 8 [N] damage the blade surfaces. The proposed fuzzy model consists of three input signals x_1, y_1, N and one output signal (cutting force). Where x_1, y_1 are the coordinates of the location of the measurement points on the blade surface, N is the amount of material removed.

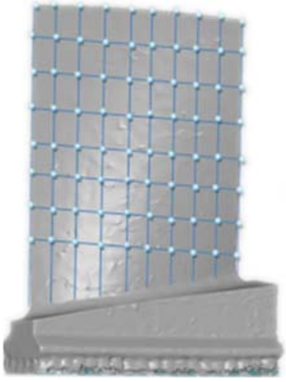
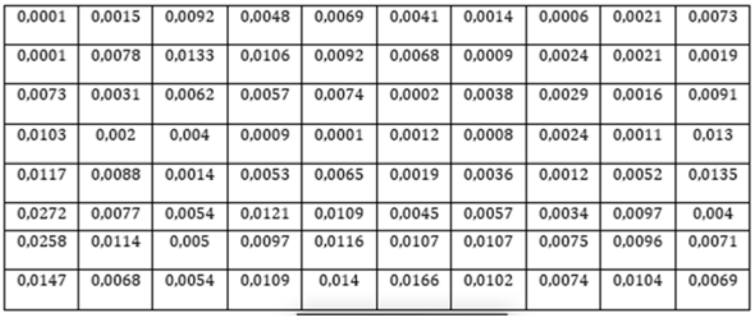


Fig. 7. Arrangement of measurement points on the machined blade.

The values of the material removed for a cutting force equal to 5[N] are presented in Table 1.

Table 1. Obtained measurements for a clamping force of 5[N] at 80 measuring points.



0,0001	0,0015	0,0092	0,0048	0,0069	0,0041	0,0014	0,0006	0,0021	0,0073
0,0001	0,0078	0,0133	0,0106	0,0092	0,0068	0,0009	0,0024	0,0021	0,0019
0,0073	0,0031	0,0062	0,0057	0,0074	0,0002	0,0038	0,0029	0,0016	0,0091
0,0103	0,002	0,004	0,0009	0,0001	0,0012	0,0008	0,0024	0,0011	0,013
0,0117	0,0088	0,0014	0,0053	0,0065	0,0019	0,0036	0,0012	0,0052	0,0135
0,0272	0,0077	0,0054	0,0121	0,0109	0,0045	0,0057	0,0034	0,0097	0,004
0,0258	0,0114	0,005	0,0097	0,0116	0,0107	0,0107	0,0075	0,0096	0,0071
0,0147	0,0068	0,0054	0,0109	0,014	0,0166	0,0102	0,0074	0,0104	0,0069

Each input signal has been divided into an appropriate number of fuzzy sets. An even division of the input space by fuzzy sets was assumed. Fuzzy sets in the form of triangular sets were adopted. The input space for the signal x_1 was divided into 10 fuzzy sets from A_1 to A_{10}. For the signal y_1 8 fuzzy sets from B_1 to B_8 were assumed. This distribution results from the division of the blade into appropriate fragments (Fig. 8). In the case of the N signal (material removed), the space was divided into 60 fuzzy sets in the range from 0 to 0.006. These values are related to the amount of allowance removed by the tool (sets from n_1 to n_{60}).

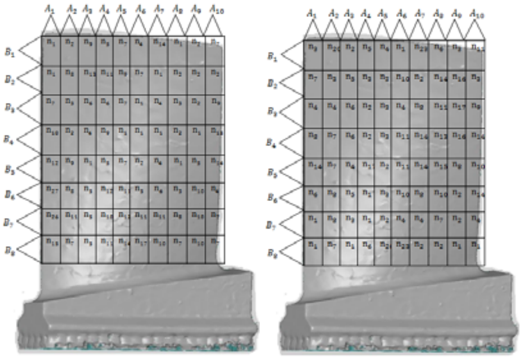
For the output signal, 3 fuzzy sets were assumed. These are triangular sets with centres of 3, 5, 9 with an accepted tolerance of 0.1 mm. The sharp values x_1^*, x_2^*, N^* (Fig. 8) were introduced in the fuzzy block (Fig. 9), where they are blurred, i.e. the degree of belonging to individual fuzzy sets is calculated A_i, B_j, N_k.

In the case of the implementation of fuzzy inference, it is required to assess the degree of fulfilment of particular rules. The rule will not be activated when the degree of fulfilment of the rule's condition is equal to zero. This is also related to the fact that this rule will not be involved in the application process. The higher the degree of fulfilment of the condition, the higher the membership of a given rule. In the proposed fuzzy model, the condition consists of three simple conditions connected by logical conjunction I [15], for example:

$$\mathrm{IF}\ (x_1 = A_1)\ \mathrm{I}\ (y_1 = B_2)\ \mathrm{I}\ (N = n_{20}), \tag{1}$$

then for the values of input signals $x_1 = x_1^*$, $y_1 = y_1^*, N = N^*$ the degree to which it is true is calculated as the degree of membership of the relationship R [14]:

$$\begin{gathered}\mu_R(x_1^*, x_2^*, N^*) = \mu_{A_1 \cap B_2 \cap n_2} = \\ = (x_1^*, x_2^*, N^*) = T(\mu_{A_1}(x_1^*), \mu_{B_2}(x_2^*), \mu_{n_{20}}(N^*))\end{gathered} \tag{2}$$



(a) (b)

Fig. 8. Division of the input space x_1, y_1, N with a clamping force equal to a) 5[N], b) 3[N].

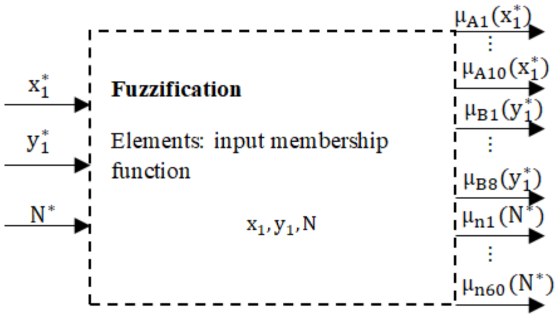


Fig. 9. Fuzzy model fuzzification block proposed in the blade machining process.

In the next stage, inference (Fig. 10) of the proposed fuzzy model was made.

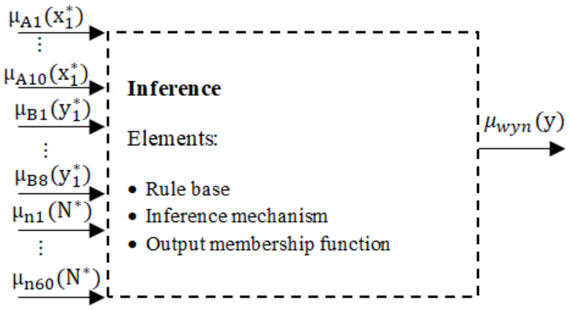


Fig. 10. Inference block for the fuzzy model proposed in the blade machining process.

By means of the input degrees of membership $\mu_{Ai}(x_1^*), \mu_{Bj}(x_2^*), \mu_{nk}(N^*)$ the resulting membership function $\mu_{wyn}(y)$ of the model output is calculated. This block specifies the rule base. In this case, it consists of 240 rules. This is due to the fact that the blade has 80 measurement points, and the tests are carried out with the participation of 3 forces. A fragment of the rule base for a force equal to 5[N] is as follows:

If x_1 is A_1 and y_1 is B_1 and N is n_1 then Force is 5
If x_1 is A_1 and y_1 is B_2 and N is n_1 then Force is 5
If x_1 is A_1 and y_1 is B_3 and N is n_7 then Force is 5
If x_1 is A_1 and y_1 is B_4 and N is n_{10} then Force is 5
If x_1 is A_1 and y_1 is B_5 and N is n_{12} then Force is 5
If x_1 is A_1 and y_1 is B_6 and N is n_{27} then Force is 5
If x_1 is A_1 and y_1 is B_7 and N is n_{26} then Force is 5
⋮
If x_1 is A_7 and y_1 is B_7 and N is n_{11} then Force is 5

The model described above was designed in MATLAB/Simulink with the use of the "Fuzzy Logic Toolbox" library. Thanks to this approach, the surface of the fuzzy model was obtained (Figs. 11-13).

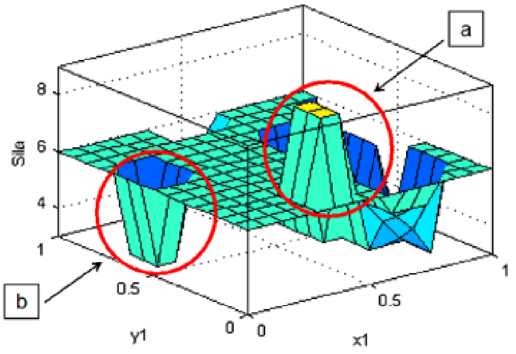


Fig. 11. Graphical interpretation of the surface of the fuzzy model for the allowance 0.014 [mm].

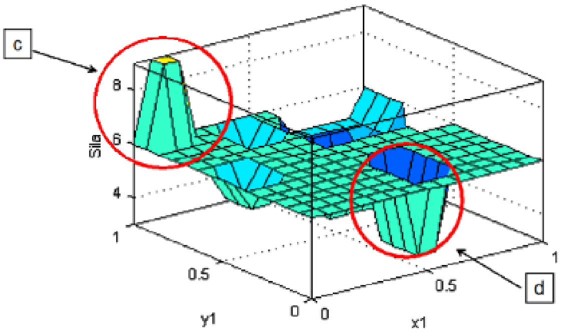


Fig. 12. Graphical interpretation of the surface of the fuzzy model for the allowance 0.01 [mm].

Interpreting the obtained result, it can be easily noticed that to remove an allowance of 0.014 [mm], a force of 6[N] should be used; however, in the area marked as (a) this force should be much greater, and within (b) significantly smaller. In this case, to remove an allowance of 0.01 [mm], a force of 6[N] should also be used, and within the place marked (c), a force of

9[N] should be used. For area d, a smaller force of 3[N] should be used.

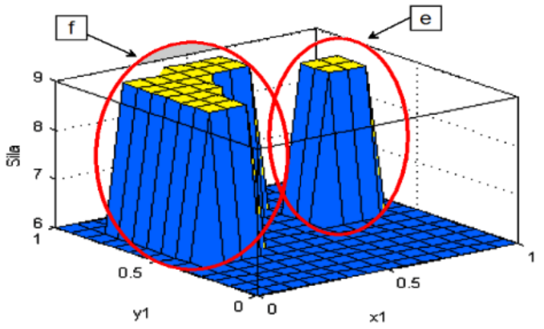


Fig. 13. Graphical interpretation of the surface of the fuzzy model for the allowance 0.04 [mm].

For the allowance of 0.04 [mm], according to the obtained result, a force of 6 [N] should be used, and also in the marked places (f, e) a force of 9[N]. The results obtained in this way are sent from the MATLAB model to the robot that machines the workpiece.

4. Conclusions

The article deals with the development and application of an intelligent controller in order to generate the value of the tool pressure forces depending on the value of the material allowance that occurs as a result of the technological process used in the production of the detail. A solution based on fuzzy logic systems has been proposed. The proposed solution made it possible to generate information allowing for the adoption of the value of the detail-tool contact force in such a way as to ensure the grinding process with the imposed requirements. The proposed model, based on modern methods of artificial intelligence, at measurement points where there was a large allowance, generate a cutting force greater than at points where the allowance was small.

The main driving force behind this type of work, in addition to minimizing human labor costs, is the improvement of safety conditions and increasing the repeatability of the product.

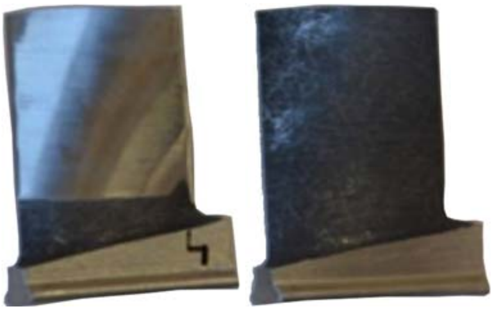


Fig. 14. View of a blade before and after grinding.

The results presented in the work showed that the methods based on intelligent systems are universal, very effective methods that can be used in the selection of parameters of robotic machining processes, which was demonstrated on the example of grinding aircraft engine blades. The proposed fuzzy system application is an example of the possibility of their implementation. and can serve as a model and broaden the recipient's knowledge, thereby influencing the awareness of management staff and engineers, which in the future may result in other applications improving the broadly understood quality of the Polish economy.

The next step will be to take into account the speed of the blade's feed relative to the tool, which also affects the amount of material removed.

References

[1]. H. Huang, Z. Gong, X. Chen, Smart robotic system for 3D profile turbine vane airfoil repair, *The International Journal of Advanced Manufacturing Technology*, Vol. 21, Issue 4, 2003, pp. 275-283.

[2]. Y. Q. Sun, D. J. Giblin, K. Kazerounian, Accurate robotic belt grinding of workpieces with complex geometrie using relative calibration techniques, *Robotics and Computer-Integrated Manufacturing*, Vol. 25, Issue 1, 2009, pp. 204-210.

[3]. A. Burghardt, K. Kurc, D. Szybicki, M. Muszyńska, T. Szczęch. Robot-operated inspection of aircraft engine turbine rotor guide vane segment geometry, *Tehnički Vjesnik*, Vol. 24, Issue 2, 2017, pp. 345-348.

[4]. A. Burghardt, K. Kurc, D. Szybicki, M. Muszyńska, J. Nawrocki, Software for the robot-operated inspection station for engine guide vanes taking into consideration the geometric variability of parts, *Tehnički Vjesnik*, Vol. 24, Issue 2, 2017, pp. 349-353.

[5]. A. Burghardt, K. Kurc, D. Szybicki, M. Muszyńska, J. Nawrocki. Robot-operated quality control station based on the UTT method, *Open Engineering*, Vol. 7, Issue 1, 2017, pp. 37-42.

[6]. P. Gierlak, A. Burghardt, D. Szybicki, M. Szuster, M. Muszyńska, On-line manipulator tool condition monitoring based on vibration analysis, *Mechanical Systems and Signal Processing*, Vol. 89, 2017, pp. 14-26.

[7]. A. Burghardt, K. Kurc, D. Szybicki, M. Muszyńska, T. Szczęch, Monitoring the parameters of the robot-operated quality control process, *Advances in Science and Technology-Research Journal*, Vol. 11, Issue 1, 2017, pp. 232-236.

[8]. A. Burghardt, D. Szybicki, K. Kurc, M. Muszyńska, Robotic grinding process of turboprop engine compressor blades with active selection of contact force, *Tehnički Vjesnik,* Vol. 29, Issue 1, 2022, pp. 15-22.

[9]. D. Szybicki, A. Burghardt, K. Kurc, P. Pietruś, Calibration and verification of an original module measuring turbojet engine blades geometric parameters, *Archive of Mechanical Engineering,* Vol. 66, Issue 1, 2019, pp. 97-109.

[10]. S. Wu, K. Kazerounian, Z. Gan, Y. Sun, A simulation platform for optimal selection of robotic belt grinding system parameters, *The International Journal of Advanced Manufacturing Technology*, Vol. 64, 2013, pp. 447-458.

[11]. Y. Song, W. Liang, Y. Yang, A method for grinding removal control of a robot belt grinding system, *Journal of Intelligent Manufacturing*, Vol. 23, 2012, pp. 1903-1913.

[12]. S. Yixu, L. Hongbo, Y. Zehong, An adaptive modeling method for a robot belt grinding process, *IEEE/ASME Transactions on Mechatronics*, Vol. 17, Issue 2, 2011, pp. 309-317.

[13]. O. Yilmaz, N. Gindy, J. Gao, A repair and overhaul methodology for aeroengine components, *Robotics and Computer-Integrated Manufacturing*, Vol. 26, Issue 2, 2010, pp. 190-201.

[14]. R. R. Yager, L. A. Zadeh, An Introduction to Fuzzy Logic Applications in Intelligent Systems, *Springer Science & Business Media*, 2012.

[15]. A. Ornat, M. Uliasz, G. Bomba, A. Burghardt, K. Kurc, D. Szybicki, Robotised geometric inspection of thin-walled aerospace casings, *Sensors,* Vol. 22, 2022, 3457.

5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2023),
7-9 June 2023, Tenerife (Canary Islands), Spain

(031)

Active Learning-based Online Coupling of Sawmill Simulators and Their Surrogate Model: The Effect of Sampling Bias on Concept Drift Detection

S. Chabanet, P. Thomas and H. Bril El-Haouzi
CRAN, Université de Lorraine, CNRS, Epinal, F-88000, France
E-mail: sylvain.chabanet@univ-lorraine.fr

Summary: Coupling numerical models with different computational costs and fidelity levels is a promising solution for developing efficient industrial digital twins able to process fast data streams. Such a coupling strategy between a high fidelity but computationally intensive simulation model and its machine learning-based surrogate model in the context of the sawmill industry had been proposed in a previous publication [1]. The strategy proposed is inspired by active learning and based on a measure of prediction confidence. It relies, however on the assumption that the input data stream models make predictions upon is stationary. Concept drifts are, however, frequent in an industrial context. It is, therefore, necessary to integrate into the proposed strategy a mechanism to detect such drifts. The present article evaluates the usage of several drift detection methods in conjunction with the previously proposed coupling strategy and highlights the issue caused by sampling bias. In particular, it can increase the time needed to detect the drift. Four datasets are used for evaluation. The first contains sawing simulation results, while the three others are benchmark datasets from the literature.

Keywords: Active learning, Sampling bias, Drift detection, Surrogate models.

1. Introduction

The coordination of digital models performing the same prediction task but varying in terms of fidelity level and computational requirement is an important step toward the integration of digital sobriety into operational digital twins [2]. While such twins might integrate high-fidelity simulation models, these tend, indeed, to be too computationally intensive to be used online over streams of data continuously collected from the physical twin. For example, [3] reports on the development of an industrial boiler digital twin integrating a multi-physics simulation model that requires five to seven days of computation to 1164 processors. A less extreme example is given by sawing simulators used in the forest-product industry which may need several minutes to simulate the sawing of a single log. This is, however, too slow for short-term decision problems that can involve several thousand such simulations. A common solution to lower the computational cost of such simulation models is to use surrogate models based on machine learning algorithms [4]. Such surrogate models, however, are only approximations of the original simulation model and cannot be expected to generalize as well, especially to scenarios never seen before. In the context of digital twins, it, therefore, appears necessary to retain the original simulation model to make predictions for a selected fraction of the data collected from the physical twin.

Such a strategy to couple a simulation model and its surrogate by actively selecting which model to use to predict labels for which data points collected from a physical twin has been proposed by [1]. This strategy is inspired by the theory of stream-based active learning (AL) [5] which studies how to actively sample data points from unlabeled data streams. In classic AL settings, these data points, and only these, are labeled and used to train ML models. The initial objective of such AL methods is to train ML models under restricted labeling budgets. Here, data points are selected from a data stream based on an estimation of a level of uncertainty in the surrogate prediction in order to minimize the prediction error of the coupled models. If this uncertainty level is too high, the simulation model is used instead of the ML model.

The strategy introduced by [1], however, relies on the assumption that the stream of data collected from the physical twin is stationary. This assumption ensures, in particular, that a specific simulation budget is targeted and that the performance of the surrogate model does not deteriorate. This assumption, however, might not be respected in industrial contexts, due, for example, to tool wear and changes in the characteristics of the raw material. Detecting concept drift in the input dataset stream would, therefore, be a first step toward the adaptation of the surrogate model.

This article discusses the integration of such a concept drift detection module in the strategy proposed by [1] and evaluates several drift detection strategies in this context. It additionally discusses and highlights a limitation of the proposed coupling strategy, which like many active learning-based methods causes sampling bias. This means that the statistical distribution of the data sampled to be labeled by the simulation model is not representative of the real distribution of the data [6]. It is known to cause various problems such as complicating the comparison and selection of ML models or undersampling important areas of the parameter space which leads in some cases to poorly trained ML models. In the setting presented in this study, it can also, in some cases, impair drift detection.

The remainder of this article is organized as follows. Section 2 presents the context of this study and reviews previous works on coupling sawmill simulation models and their surrogate models. Section 3 then overviews the literature on concept drift detection and discusses the integration of these methods into the coupling strategy previously proposed by [1]. Section 4 presents experimental results evaluating several drift detection methods and highlights the problem caused by sampling bias. Section 5 concludes this paper.

2. Context

Many authors have studied the usage of sawing simulation models to support production planning and control in the forest-product industry [7]. These simulation models can be used to predict what set of lumbers would be obtained from sawing a specific log. Many authors, however, mention the important time taken by such simulations which limits their usage to mid and long-term production planning problems. In order to limit the computation time of these simulations and make possible their usage for short-term planning problems, [8], among others, proposed to use surrogate models of these simulators, that is, machine learning algorithms trained on past simulation results to perform the same prediction as the simulation model. Such surrogate models, however, remain only approximations of the original model, trained on relatively few data, and cannot be expected to generalize well to data inputs too different from the ones used to train them.

Recently, [1] proposed to couple both the simulation model and its surrogate model to benefit from their respective advantages as a step toward the development of a sawmill digital shadow able to support operational production planning. The objective is to decide for every data item transmitted by a stream whether the more precise simulation model should be used to replace the surrogate model prediction while targeting a user-defined simulation budget *b*.

The general workflow of this strategy is presented in Fig. 1. Consider a data stream transmitting new data items at random intervals. It is, here, modeled as a Poisson process. Similarly, the time required by the simulation model to predict the label of a data item is modeled as a random variable. It is assumed that any number of simulations can run in parallel. However, to limit computation costs, a budget corresponding to the average portion of data points sent to the simulator is targeted.

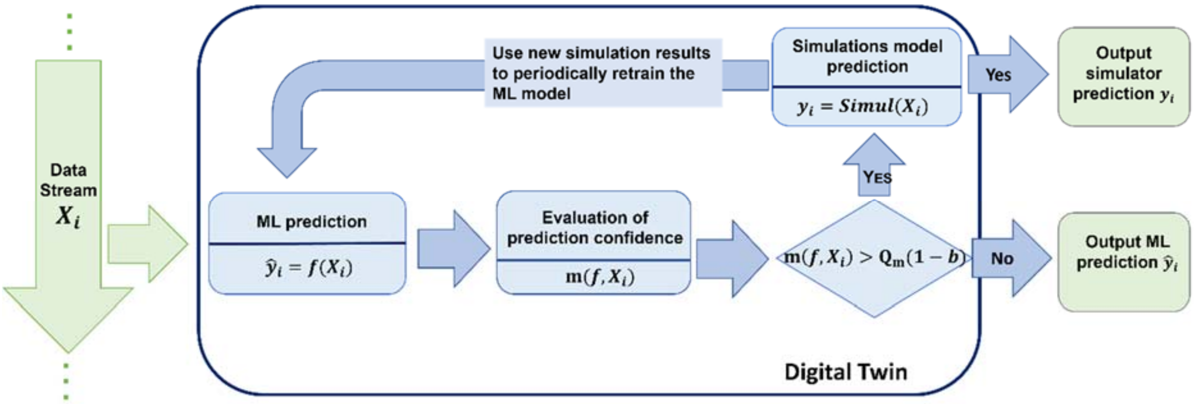


Fig. 1. Simplified framework of the coupling strategy proposed by [1].

For every data item X_i transmitted by the stream, a first prediction $\hat{y}_i$ is made by the machine learning model f. In this work, the machine learning models used are random forests, due to their good performances as sawmill simulators metamodels [7, 8]. A measure of uncertainty of the prediction is then computed. It has to be stressed out, however, at this stage the real label y_i of X_i is unavailable. The measure of uncertainty used here is, therefore, computed as the variance of the predictions of the trees in the forest:

$$\mathrm{m}(\mathrm{X}, \boldsymbol{f}) = \frac{1}{\mathrm{B}}\sum_{\mathrm{i}=1}^{\mathrm{B}}\left(f_i(X) - f(X)\right)^2$$

Here, $\boldsymbol{B}$ is the number of trees in the forest. The $\boldsymbol{f_j}$ are the infividual trees in the forest f. The quantile of this measure with respect to its overall distribution is then estimated from past values observed from the stream and compared with the threshold $1 - b$. This ensures that, as long as the stream remains stationary, the average portion of the stream that is sampled will tend toward b.

This strategy, however, relies on the assumption that the stream remains stationary to respect the simulation budget and ensure that past simulation results used to update the surrogate model remain representative of the prediction task.

3. Drift Detection

Many types of drifts have been identified in the related literature. In particular, they can be classified into real and virtual drift [11]. Real drift refers to the relationship between the inputs X_i and their labels y_i changing with time. Virtual concept drift refers to changes in the statistical distribution of the inputs X_i.

Drift detection methods often rely on monitoring univariate or multivariate streams updated with new data inputs. Different such streams can be used. Many authors proposed methods monitoring the prediction loss $l(y_i, \hat{y}_i)$. For example, [12] introduces a method based on control charts, often use in statistical process control, to monitor the error rate of neural networks. Similarly, the Page-Hinckley algorithm [13] is often used to monitor increases in these errors. Monitoring this error stream has the advantage that alerts can be raised only when concept drift results in a decrease in the model performance. It can also detect both real and virtual drift. However, it requires the obtention of the real targets y_i which, in practice, might not be obtained immediately, if at all. Other authors such as [14] propose to monitor the input feature stream, i.e., the stream of the X_i. More precisely, they propose a method based on the Kolmogorov-Smirnov statistical test between a sliding window of the stream and a fixed reference window. These data have the advantage of always being available. However, this stream is, more often than not, multivariate, which increases the risk of false alarms. Similarly, drift that does not impact the ML model performance can be detected and trigger useless updates of the model. In addition, purely real drift cannot be detected. Lastly, [15], for example, propose to use a stream composed of uncertainty measures of the predictions made by the ML model. More precisely, they use uncertainties associated with the predictions of Bayesian neural networks. Once again, this stream does not rely on the potentially unavailable targets. Additionally [15] argue that as long as the uncertainty measures are sufficiently correlated with the prediction loss $l(y_i, \hat{y}_i)$, detection will focus on drifts lowering the performance of the ML model. However, once again, it cannot detect purely real drifts.

These three types of streams can be monitored to detect drift in the previously described coupling strategy. The streams of the input features, X_i, and uncertainty measures, $\mathrm{m}(\mathrm{X}, \boldsymbol{f})$, are naturally part on the framework. The real targets y_i, however, are only available for the fraction of the inputs for which the simulation model is used, and only after a set amount of time corresponding to the time needed for the simulation to run.

While any of these streams could be monitored to detect concept drift, depending on the weight given to their respective advantages and disadvantages, the present article focuses on the stream of the prediction errors. While it is only partially available, it is the only type of stream allowing the detection of purely real drift and checking if the drift being detected effectively results in an increased prediction error.

4. Experiments

The performances of three drift detection procedures are evaluated to detect purely virtual drift from the partially measured stream of prediction losses. The first method is the control chart method (CC), introduced by [12]. The second is the method based on the Kolmogorov-Smirnov test used, for example, in [14]. This method depends on a parameter α corresponding to the risk of the test. The value of α used here was set to 0.01. Lastly, the Page-Hinckley method [13] is used. This method depends on two parameters, the so-called delta factor, and a detection threshold. Considering the sensitivity of the method to these parameters, whose scale might change depending on the dataset considered, they were all fixed here as multiple of the standard deviation σ of the errors observed on each dataset. More precisely, the delta factor was set, here, to 0.5σ for all experiments and the value of the threshold h was set to 10σ.

2.1. Datasets

Four datasets are used for the experiments presented in this paper.

The first dataset (SS) originates from the Canadian forest product industry. It contains information over 2219 real wood logs, including an estimate of the set of lumber sawed in it, simulated by the software Optitek. This set of lumber is modeled as a vector of integer of dimension 47 because up to 47 types of lumber can be sawed at the modeled sawmill. Every wood log is described by a vector of six know-how features commonly used in the industry. These features are the length, volume, curvature, diameters at both extremities, and shrinking of the log.

The three other datasets are from the UCI machine learning repository. They were selected because they cover regression tasks based on multivariate inputs and contain all more than 2000 points.

The first UCI dataset is the wine quality (WQ) dataset [16]. The task associated with this dataset is the prediction of wine qualities based on their physicochemical properties.

The second UCI dataset is the combined cycle power plant (CCPP) dataset [17]. Its aim is the prediction of a power plant production based on meteorological and physical data.

The last UCI dataset is the superconductivity data (SD) dataset [18] whose goal is to predict the critical temperature of superconductors based on their chemical formula.

None of these datasets naturally contains concept drift which allows controlling the type and time of drifts being artificially introduced during experiments.

Table 1. Detection rate of drift detection strategies for every dataset.

	SS		WQ		CCPP		SD	
	AL	RD	AL	RD	AL	RD	AL	RD
Kolmogorov-Smirnov	**1.0**	**1.0**	0.48	0.88	**0.99**	0.98	0.99	**1.0**
Page-Hinkley	0.87	0.91	0.54	0.84	**0.99**	0.95	0.97	0.95
Control-Chart	0.69	0.81	**1.0**	**1.0**	0.98	**0.99**	**1.0**	**1.0**

More precisely, the datasets were used to generate data drift composed of two stationary sections separated by an abrupt purely virtual drift. These streams are composed of a random ordering of the data points paired with random inter-arrival times sampled following an exponential law of rate $\lambda = 1$ to generate Poisson processes. In the second half of the stream, the targets of all data points were permutated at random to erase the relation between the input and output and create a purely real drift. Since the ML models will not be retrained during these experiments, there is no need to keep a structural relationship between the inputs and outputs past the drift.

2.1. Results

For every dataset and drift detection method, streams were generated following the previously described procedure and used as input to the coupling strategy described in Fig. 1. In addition, the partial error flow was used for drift detection. The time of the first alarm raised after the time of drift, if it exists, is considered as the time of detection. This was repeated 100 times for every dataset and drift detection method. To highlight the effect of the sampling bias on drift detection, this procedure was also repeated 100 more times but replacing the measure of prediction uncertainty used to select which samples to use the simulation model on by a random measure, so that the stream sampling process becomes independent of characteristics of the data points transmitted by the stream.

Table 2 presents the drift detection rate of every strategy on the four datasets. The drift detection rate corresponds, here, to the portion of the 100 streams that lead to a drift being detected. In this table, AL refers to the experiments using the measure of uncertainty $\mathrm{m}(.,\boldsymbol{f})$ to sample the stream, while RD refers to the experiments where this measure is replaced by randomly generated values. These rates are close to one, except for the Page-Hinkley and control chart detection methods on the SS dataset, and for the Kolmogorov-Smirnov and Pages-Hinckley methods on the WQ dataset. These two datasets are the shortest used for evaluation and might be too short for drift to be efficiently detected. Interestingly, however, these low detection rates are mostly observed for the AL stream sampling method which is a first indication that the sampling bias lower the performances of these drift detection methods on some datasets.

The median and interquartile range of the drift detection times for all drift detection methods and datasets are presented in Table 2. Medians are used due to the high standard deviations observed during experiments, making average estimates unstable. These experiments show larger detection times for the AL-based methods on the SS and WQ datasets. In particular, one-sided Brunner-Munzel comparison tests [19] were systematically run to evaluate the propension of the AL-based methods detection time at being higher than the baseline random method. These tests show significant differences at the 5 % level for all detection methods on the SS and WQ datatsets, except for the control-chart-based method on the WQ dataset. The test is also significant for the Page-Hinkley method on the SD dataset. This demonstrate that sampling bias can impair drift detection. It depends greatly, however, on the dataset and exact detection method used. The Page-Hinkley method, however, maintains rather low detection times even when affected by sampling bias.

Table 2. Medians and interquartile ranges of the drift detection time for every drift detection strategy and dataset.

	SS		WQ		CCPP		SD	
	AL	RD	AL	RD	AL	RD	AL	RD
Kolmogorov-Smirnov	241 (80)	224 (55)	963 (977)	587 (463)	209 (65)	206 (59)	289 (112)	270 (71)
Page-Hinkley	90 (108)	**49 (40)**	557 (758)	**180 (258)**	**31 (38)**	37 (42)	74 (48)	**59 (41)**
Control-Chart	111 (112)	66 (66)	402 (396)	313 (402)	331 (647)	289 (537)	151 (394)	325 (579)

5. Conclusion

This paper discusses the integration of a concept drift detection module in a previously proposed coupling strategy between a simulation model and its ML surrogate model. Several types of streams which can be monitored to detect drift are identified and their advantages and disadvantages are listed. Several drift detection methods are also tested on one of these drifts. Results point to the strong effect of the sampling bias induced by the coupling strategy which tends to slow down significantly drift detection.

Table 3. Summarized characteristics of the four datasets used for evaluation.

Dataset name	Number of points	Number of features	Number of outputs
Sawmill Simulations (SS)	2219	6	47
Wine Quality (WQ)	4898	11	1
Combined Cycle Power Plant (CCPP)	9568	4	1
Superconductivity Data (SD)	21263	81	1

Future works will, therefore, have to find ways to mitigate this problem, either by investigating drift detection methods less sensitive to sampling bias or using statistical methods to correct the bias induced.

In addition, a single type of drift is studied here: abrupt real drift. The impact of sampling bias on other types of drift, and in particular of virtual drift, remain to be studied.

Acknowledgments

The authors gratefully acknowledge the financial support of the ANR-20-THIA-0010-01 Project LOR-AI (Lorraine intelligence artificielle) and région Grand EST.

We are also extremely grateful to FPInnovation who gathered and processed the dataset we are working with.

References

[1]. S. Chabanet, H. B. El Haouzi, Thomas, Toward a sawmill digital shadow based on coupled simulation and supervised learning models, in Service Oriented, Holonic and Multi-Agent Manufacturing Systems for Industry of the Future, *Springer*, Cham, 2023, pp. 59-70.

[2]. N. Julien, M. A. Hamzaoui, Integrating lean data and digital sobriety in digital twins through dynamic accuracy management, in Service Oriented, Holonic and Multi-Agent Manufacturing Systems for Industry of the Future, *Springer*, Cham, 2023, pp. 107-117.

[3]. J. P. Spinti, P. J. Smith, S. T. Smith, Atikokan Digital Twin: Machine learning in a biomass energy system, *Applied Energy*, Vol. 310, Mar. 2022, 118436.

[4]. S. Razavi, B. A. Tolson, D. H. Burn, Review of surrogate modeling in water resources, *Water Resources Research*, Vol. 48, Issue 7, 2012, W07401.

[5]. P. Kumar, A. Gupta, Active learning query strategies for classification, regression, and clustering: A survey, *J. Comput. Sci. Technol.*, Vol. 35, Issue 4, Jul. 2020, pp. 913-945.

[6]. R. Krishnan, A. Sinha, N. Ahuja, M. Subedar, O. Tickoo, R. Iyer, Mitigating sampling bias and improving robustness in active learning, *arXiv Preprint*, Sep. 13 2021, arXiv:2109.06321.

[7]. S. Chabanet, H. Bril El-Haouzi, M. Morin, J. Gaudreault, P. Thomas, Toward digital twins for sawmill production planning and control: Benefits, opportunities, and challenges, *International Journal of Production Research*, Vol. 61, Issue 7, Apr. 2023, pp. 2190-2213.

[8]. M. Morin, *et al.*, Machine learning-based models of sawmills for better wood allocation planning, *International Journal of Production Economics*, Vol. 222, Apr. 2020, 107508.

[9]. M. Morin, F. Paradis, A. Rolland, J. Wery, F. Laviolette, F. Laviolette, Machine learning-based metamodels for sawing simulation, in *Proceedings of the Winter Simulation Conference (WSC'15)*, Dec. 2015, pp. 2160-2171.

[10]. S. Wager, T. Hastie, B. Efron, Confidence intervals for random forests: The Jackknife and the infinitesimal jackknife, *J Mach Learn Res*, Vol. 15, Issue 1, Jan. 2014, pp. 1625-1651.

[11]. G. I. Webb, R. Hyde, H. Cao, H. L. Nguyen, F. Petitjean, Characterizing concept drift, *Data Min. Knowl. Disc.*, Vol. 30, Issue 4, Jul. 2016, pp. 964-994.

[12]. M. Noyel, P. Thomas, A. Thomas, P. Charpentier, Reconfiguration process for neuronal classification models: Application to a quality monitoring problem, *Computers in Industry*, Vol. 83, Dec. 2016, pp. 78-91.

[13]. D. V. Hinkley, Inference about the change-point from cumulative sum tests, *Biometrika*, Vol. 58, Issue 3, Dec. 1971, pp. 509-523.

[14]. R. Dos, M. Denis, P. Flach, S. Matwin, G. Batista, Fast unsupervised online drift detection using incremental Kolmogorov-Smirnov test, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1545-1554.

[15]. L. Baier, T. Schlör, J. Schöffer, N. Kühl, Detecting concept drift with neural network model uncertainty, *arXiv Preprint*, Sep. 23 2022, arXiv:2107.01873.

[16]. P. Cortez, A. Cerdeira, F. Almeida, T. Matos, J. Reis, Modeling wine preferences by data mining from physicochemical properties, *Decision Support Systems*, Vol. 47, Issue 4, Nov. 2009, pp. 547-553.

[17]. P. Tüfekci, Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods, *International Journal of Electrical Power & Energy Systems*, Vol. 60, Sep. 2014, pp. 126-140.

[18]. K. Hamidieh, A data-driven statistical model for predicting the critical temperature of a superconductor, *Computational Materials Science*, Vol. 154, Nov. 2018, pp. 346-354.

[19]. K. Neubert, E. Brunner, A studentized permutation test for the non-parametric Behrens-Fisher problem, *Computational Statistics & Data Analysis*, Vol. 51, Issue 10, Jun. 2007, pp. 5192-5204.

(032)

How Science and Technology Limit the Performance of AI Networks

János Végh
Kalimános Bt, 4032 Debrecen, Hungary
Tel.: + 36203331364
E-mail: Vegh.Janos@gmail.com

Summary: Artificial Intelligence (AI) systems face huge hype and unrealistic expectations today. Those are based theoretically on an outdated computing paradigm; the cooperation of many-processor systems uses decades-old technical implementations, and scaling of their computing performance is expected to follow the "weak scaling" fallacy. Biological information processing is purely understood mainly because of making wrong parallels with electronic information processing, abusing notions of information, entropy, etc. Mathematical methods and electronic implementations are developed for those wrong models. Consequently, AI systems above the "toy" level face severe computing performance limitations. Their low computing and energetic efficiency do not enable reaching their ambitious goals and make their development not sustainable. Furthermore, the notion of intelligence is loosely defined in biology, and its artificial counterpart imitates it using inappropriate design principles. The generalized time-aware computing model explains why scientific and technological principles place obstacles on the road towards developing AI systems and suggests changes in design, technology, and mathematical methods for preparing better AI systems.

Keywords: Artificial intelligence, Neural networks, Performance scaling, Technological limitations.

1. Introduction

Instead of Artificial Intelligence (AI), R. P. Feynman preferred using the wording 'advanced applications' decades ago [1]. Recently, M. Jordan explained [2] why today's artificial intelligence systems are not actually intelligent. Despite this misnomer, the proliferation of AI systems is impressive in all fields (science, economy, society, etc.), with frequent and sound success stories and (rarely published) project failures, depending on the size and type of the task. No doubt, those systems are helpful and successful for specific tasks. However, their range of applicability has limitations, and their excessive utilization leads to sustainability issues.

Computing theory originates from von Neumann's classic report [3]. However, it has been forgotten that with changing technology, the omissions he made to model his immediate goal, the vacuum tube computer, should have been revisited. His omissions are no longer valid for our present technical systems with GHz clock frequencies, stadium-sized supercomputers, brain-imitating artificial networks, and biological networks [4]. The consequences are, among others, the excessive heat production by modern processors, the experienced slowdown of supercomputing performance development [5], the low computing performance of larger artificial networks [6], and the failures of simulating biological networks in size comparable to (at least parts of) the brain [7].

2. Origins of Limitations

Timing relations of the vacuum tube computer allowed to omit data transmission time (usec order) aside from data processing time (msec order). For today, the timing relations changed to their exact opposite: within the processors, even the large-scale miniaturization could not keep pace with the ever-quicker operating elements, and in computing systems, the length of buses (and so: data transmission time) remained the same for decades while the operating time decreased by a factor of million. This stealthy development of technology resulted in that today, the timing relations of technical computing are much closer to those of biological systems than to the abstract computing model (modeled for vacuum tubes), for which von Neumann told [3] to be 'unsound' to apply his simplified paradigm to. At the same time, the proliferation of computing technology irrationally increased computational demand: before introducing GPUs, it doubled every two years; before AI, in every quarter of a year; nowadays, in every two months [8]. It is not sustainable and limits the utility of AI [9].

The general computing model that works for any processing and transfer times ratios has been elaborated only recently [4]. Introducing a finite data transmission speed, similar to the finite interaction speed of classic physics, resulted in the so-called 'modern physics' changing classic computing theory to time-aware computing theory.

2.1. The Finite Speed of Data Transfer

In the case of chained operations, data must be transferred to the input section of the processor before processing can begin; furthermore, the result must only be transferred from its output section once data processing is finished. Data transfer and data processing mutually block each other, leading to an inherent performance loss [4]. The primary reason for the loss is science: the finite propagation speed of electromagnetic waves makes the transfer times non-negligible. In addition, technology can also contribute to the loss of using inappropriate designs. The loss heavily depends on the processor's workload, requiring the introduction of ’empirical computing efficiency’ [5]. So, one must distinguish the nominal and payload performance of computing systems.

Specifically, AI networks were inspired by biological networks in the sense that many relatively simple computing units are working together, sharing their computing results: they need to communicate permanently. Communication is implemented through Input/Output (I/O) instructions; this is why "AI,... it's the most disruptive workload from an I/O pattern perspective” [10], so computing efficiency is especially bad for an AI-type workload [6]. The communication need can even collapse large, heavily communicating systems [11].

The synchronous operating mode is easy to implement, but it proves to be effective only for slow and small systems when the dispersion of the timings is small [3]. The timing delays caused by the different signal path temporal lengths consume a large proportion of energy, even inside processors.

For AI networks, it has also been confirmed experimentally [12] that their computing performance scaling stalls at a couple of dozens of neurons due to the above reasons. It has also been found in other applications [13] that even a large number of processors cannot provide more payload performance than a few dozen of processors. The same reason is that large supercomputers do not use all their available cores [5] when running the HPCG benchmark that imitates real-life tasks. Due to similar reasons, the brain simulator SpiNNaker can use only 1 % of its hardware for the task it was designed for [15].

2.2. Effect of Serial Bus

Conventional computer architecture is built around a bus (mistakenly called “von Neumann architecture”), and data are moved to relatively large distances between its components. This organization alone constraints the components to wait for each other, but the real bottleneck is the single-processor point of view of the design: in large systems, bus arbitration takes most of the operating time and both the bus and the processors operate very much below their nominal performance.

The logical (mathematical or biological) network connection is shown in Fig. 1A: the messages to and from the hidden layer are transferred simultaneously. The technical implementation using a technical bus is shown in Fig. 1B. The seemingly little difference is that technical neurons must communicate through a (usually high-speed) serial bus has enormous consequences.

Given that for communication bus arbitration must take place, the bus cannot deliver the two messages to and from neurons in the hidden layer at the same time. As Fig. 2A displays, between some processing at the neurons, the arguments must be transferred to the next neuron through the bus one after the other. The high-speed bus is not loaded in most part of the computing time. Still, is causes delays until it delivers the messages to/from the hidden layer. For a large number of neurons, it may happen that the total length of messages issued by the neurons at the same time exceeds the available bus bandwidth (see Fig. 2B), leading to further delay. For a detailed discussion see [6].

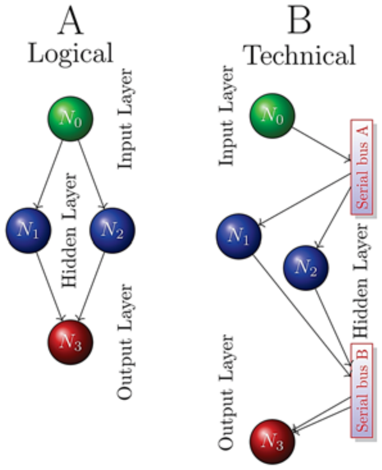


Fig. 1. In a technical implementation, a bus delivers neuronal signals one at a time.

In a general case, *n* input nodes issue messages, each one *m* messages (queuing#1) to nodes in the first hidden layer, i.e., altogether *nm* messages. If one uses a commonly used shared bus to transfer messages, these *nm* messages must be queued (queuing#2). Also, every single node in the hidden layer receives (and processes) *m* input messages (queuing#3). Between hidden layers, the same queuing is repeated (maybe several times) with *mm* messages, and finally, *km* messages are sent to the output nodes. That is, the number of messages grows up quickly as the size of the network grows. What is worse, the average delivery time grows with the number of nodes/messages. Moreover, to make sure that all messages needed for the computation are delivered, one must calculate with the worst case, the possible longest time. For details see [7]. Using different buses between the layers (as in Fig. 1) may mitigate the issue.

The order of delivering messages is an internal business of the bus, so we cannot tell for sure whether a particular message is delivered until all messages are delivered. That is, we can make the next calculation step only if all nodes received all their needed inputs and performed their calculation. Given that there are no guaranties that the input messages will be delivered in the needed order (as the computation or the hardware architecture would require) to the nodes, some computations must be repeatedly performed, so some relaxation time must be added. Either we accept that the final result might be wrong or if some intermediate nodes receive their input message the change has to pass the chain of neurons and may induce other late changes. In vast systems, the feedback in the learning process involves results partially based on undefined inputs: the feedback is calculated later (before computation made with it), furthermore the calculated and the (maybe correct) feedback may be neglected because it was delayed. This effect also means that the computed feedback, based maybe on undefined inputs, reaches the previous layer's neurons faster.

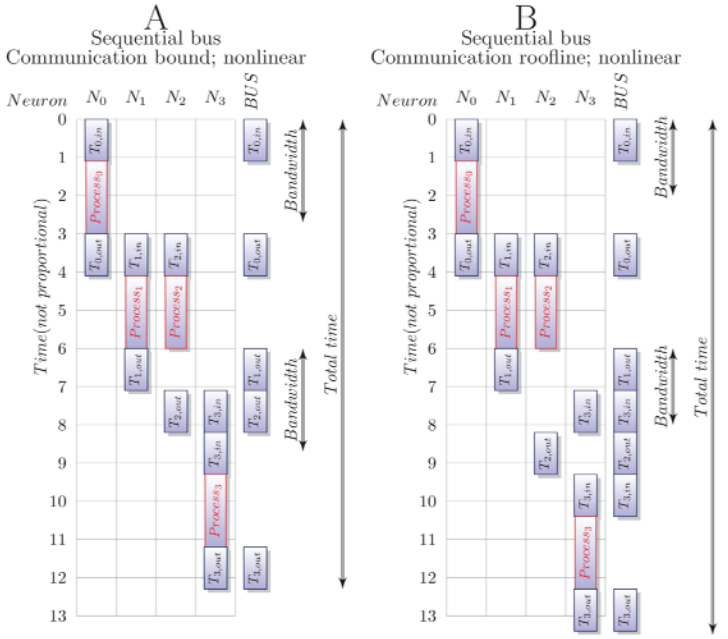


Fig. 2. Using a shared bus (the need to arbitrate the bus) enormously increases data transfer time, makes signal timing non-aligned. The bandwidth of the bus may introduce another bottleneck.

The delay exists – a but usually remains obscured – also in small systems. In large systems in can grow up tremendously. One bad option is to wait until all messages arrive, the other one is to reject part of them to achieve a seemingly reasonable computing time. In the case of supercomputer Frontier, it takes 40 minutes to deliver all results from the 7 million fellow cores to the initiator core, see Fig. 1 in [14]. In the case of brain simulation by SpiNNaker, “spikes are processed as they come in and are dropped if the receiving process is busy over several delivery cycles” [15]. If you do not take either option, after terminating the operation, the rest of messages will arrive from different buffers and you may think that ”Lack of Sleep Could Be a Problem for AIs” [16].

Anyhow, the delayed message delivery due to bus arbitration of the serial bus remains a huge problem, especially when training AANs. ”Yet the task of training such networks remains a challenging optimization problem. Several related problems arise: very long training time (several weeks on modern computers, for some problems), the potential for over-fitting (whereby the learned function is too specific to the training data and generalizes poorly to unseen data), and more technically, the vanishing gradient problem" [17]. Performing the operation faster is dangerous: “The immediate effect of activating fewer units is that propagating information through the network will be faster, both at training as well as at test time. …” A natural consequence is that (see their Fig. 5): “As $lambda_s$ increases, the running time decreases, but so does performance” [17].

2.3. Time Dependence of Result

Given that the training has two characteristic time constants: how long is the operating time of the network (until all signals are relaxed) and how much time passes between showing adjacent elements of the training set. Larger AI networks have considerably high relaxation time; the larger and deeper the network, the longer. If not enough time is given to the network to relax (to reach its consistent state), the incomplete computation may result in time-dependent result.

Introducing the spatio-temporal behavior of ANNs, even in a simple form, using separated time and space contributions (i.e., not connected in the way proposed in [4]) to describe them, significantly improved the efficacy of video analysis [18]. The role of time (mismatching) is confirmed directly, via making investigations in the time domain. "The CNN models are more sensitive to low-frequency channels than high-frequency channels" [19]. The reason is clearly that during the fixed time frame of the video, the feedback can follow the slow changes with less difficulty compared to the faster changes.

3. Relation to Biological Networks

Although it is commonly accepted that "Brain computes" [20], and in this sense, AI networks have close relations to biological networks they were inspired by; it is hard to interpret their notions in the respective other field [21]. The wrong parallels with electronic circuits covered for a long time the real nature of biological computing that has been recognized only recently. The electronic computer performs synchronous operations with a well-defined number of input arguments and delivers a mathematically well-defined result. Both the input arguments and the result are digital quantities. Brain performs asynchronous and mixed analog/digital computing with loosely defined arguments and delivers a loosely defined result. Both input and output operands are timings. The technical models assume that a shared high speed serial bus can be equivalent with the many low speed private connections and that the parallelized sequential operation of technology is equivalent with the truly parallel biological operation. These vital differences need to reinterpret also the notion of information and its communication [22]. The study also demonstrated that the commonly used merit number in neuroscience does not measure transmitted information and it does not correspond to the information as interpreted in technical communication. As the abstract model of neural operation [23] explains, it is not an easy task to map biological computation to technical one.

4. Conclusions

AI networks are constructed following the classic computing paradigm and their architecture and operation only slight resemble the biological ones. Given that the performance scaling of AI systems is strongly nonlinear, the efficiency decays strongly as the size of system increases and their computing performance saturates at a low level, due to science and the recent technology solutions.

References

[1]. R. P. Feynman, Feynman Lectures on Computation, *CRC Press*, 2018

[2]. M. I. Jordan, Stop Calling Everything AI, https://spectrum.ieee.org/stop-calling-everything-ai-machinelearning-pioneer-says

[3]. J. von Neumann, First draft of a report on the EDVAC, *IEEE Annals of the History of Computing*, Vol. 15, Issue 4, 1993, pp. 27-75.

[4]. J. Végh, Revising the Classic Computing Paradigm and Its Technological Implementations, *Informatics*, Vol. 8, Issue 4, 2021, 71

[5]. J. Végh, Finally, how many efficiencies the supercomputers have?, *J. Supercomputing*, Vol. 76, Issue 12, 2020, pp. 9430-9455.

[6]. J. Végh, Which scaling rule applies to large artificial neural networks, *Neural Computing and Applications*, Vol. 33, 2021, pp. 16847-16864.

[7]. J. Végh, How Amdahl's Law limits performance of large artificial neural networks, *Brain Informatics*, Vol. 6, Issue 4, 2019, 4.

[8]. A. Mehonic, A. J. Kenyon, Brain-inspired computing needs a master plan, *Nature*, Vol 604, Issue 7905, 2022, pp. 255-260.

[9]. What is the environmental footprint of artificial intelligence?, OECD, https://www.oecd-events.org/cop27/session/f174ec37-5145-ed11-819a-00224880a4d8

[10]. Cray Revamps ClusterStor For The Exascale Era, https://www.nextplatform.com/2019/10/30/cray-revamps-clusterstor-for-the-exascale-era/

[11]. S. Moradi, R. Manohar, The impact of on-chip communication on memory technologies for neuromorphic systems, *Journal of Physics D: Applied Physics*, Vol. 52, Issue 1, 2018, 014003.

[12]. J. Keuper, F. J. Pfreundt, Distributed Training of Deep Neural Networks: Theoretical and Practical Limits of Parallel Scalability, in *Proceedings of the 2nd Workshop on Machine Learning in HPC Environments*, 2016, pp. 1469-1476.

[13]. G. D'Angelo, S. Rampone, Towards a HPC-oriented parallel implementation of a learning algorithm for bioinformatics applications, *BMC Bioinformatics*, Vol. 15, 2014, S2.

[14]. Top500: Exascale Is Officially Here with Debut of Frontier, https://www.hpcwire.com/2022/05/30/top500-exascale-is-officially-here-with-debut-of-frontier/

[15]. S. J. van Albada, et al., Performance Comparison of the Digital Neuromorphic Hardware SpiNNaker and the Neural Network Simulation Software NEST for a Full-Scale Cortical Microcircuit Model, *Frontiers in Neuroscience*, Vol. 12, 2018, 291.

[16]. Lack of Sleep Could Be a Problem for Ais, https://www.scientificamerican.com/article/lack-of-sleep-could-be-a-problem-for-ais/

[17]. E. Bengio, P.-L. Bacon, J. Pineau, D. Precu, Conditional Computation in Neural Networks for faster models, https://arxiv.org/pdf/1511.06297

[18]. S. Xie, et al, Rethinking Spatiotemporal Feature Learning: Speed-Accuracy Trade-offs in Video Classification, in *Proceedings of the European Conference on Computer Vision (ECCV'18)*, 2018, pp. 318-335.

[19]. K. Xu, et al., Learning in the Frequency Domain*, arXiv Preprint*, 2020, arXiv:2002.12416

[20]. C. Koch, Biophysics of Computation, *Oxford University Press*, 1999.

[21]. K. Kar, S. Kornblith, E. Fedorenko, Interpretability of artificial neural network models in artificial intelligence versus neuroscience, *Nature Machine Intelligence*, Vol. 4, 2022, pp. 1065-1067.

[22]. J. Végh, Á. J. Berki, Towards Generalizing the Information Theory for Neural Communication, *Entropy*, Vol. 24, Issue 8, 2022, 1086.

[23]. J. Végh, Á. J. Berki, On the Role of Speed in Technological and Biological Information Transfer for Computations, *Acta Biotheoretica*, Vol. 70, Issue 4, 2022, 26.

(033)

A Hybrid System Containing a 3D Scanner and a Laser Tracker Dedicated to Robot Programming

D. Szybicki, M. Muszyńska, A. Burghardt, P. Gierlak and K. Kurc
Department of Applied Mechanics and Robotics, Faculty of Mechanical Engineering and Aeronautics,
Rzeszów University of Technology, al. Powstancow Warszawy 8, 35-959 Rzeszow, Poland
Tel.: + 48178651843
E-mails: dszybicki@prz.edu.pl, magdaw@prz.edu.pl, andrzejb@prz.edu.pl, pgierlak@prz.edu.pl,
kkurc@prz.edu.pl

Summary: The paper presents the concept and implementation of a robotic hybrid system for combining 3D scanner data using structured light and an absolute laser tracker. The paper discusses the advantages and disadvantages of both measurement systems and the benefits of combining data in the hybrid system. An example of generating a robot program in a reverse engineering task was shown to demonstrate the operation of the developed system. The selected component was measured using a 3D scanner and then its position relative to the industrial robot was measured using a laser tracker. The correlation data obtained from the two types of scanners were used to program the machining paths of the workpiece, which is a copy of the scanned workpiece.

Keywords: 3D scanners, Absolute trackers, Reverse engineering, Robotic machining, Light scanners, Data correlation.

---

1. Introduction

Various types of sensory solutions used in robotic stations are available on the market today. Equipped with sensors, the robot knows exactly where the object is and can dynamically modify positions. This means that instead of getting each workpiece to be positioned perfectly in the same place as the previous one, it is possible to equip the robot with a sense of vision, for example, and let it decide how to move based on what it sees. In addition to using sensory systems to correct paths, they can perform recognition or classification functions for objects. The solutions offered on the market using cameras and structured light, 2D laser scanners, 3D laser scanners and image recognition systems are characterized by different operating principles, the existence of advantages in certain aspects and disadvantages in others. The development of individual measurement methods is carried out in many research centers and concerns laser profilometers [1], measurement and machining with 3D scanners [2, 3], absolute laser trackers [3] or contact measurements [4]. A laser tracker is a tool that enables various types of geometric measurements in three-dimensional space. It can be used for automated position control and also, for example, for 3D scanning of objects. Its most important components are a laser interferometer and an absolute rangefinder. Trackers in robotics, due to their accuracy, are used in testing the accuracy and repeatability of robotic manipulators and for their calibration. The authors of the paper [5] used a laser tracker to improve the absolute accuracy of an ABB IRB1600 industrial robot. They developed an advanced calibration model significantly reducing position errors. Similar work was conducted by the authors of the paper [6], who focused on building calibration systems that can be easily adapted to any type of robot. The topic of compensation using a real-time tracker is addressed in papers [7] and [8]. In the paper [7], the authors described the idea of compensating manufacturing systems by using external metrology systems to compensate machine tools and robots in real time. And in the article [8], they used direct feedback from the position and orientation of an end effector with a Leica laser tracker to develop an algorithm enabled to move along paths at significant speeds. There are many types of 3D scanners. One common solution is a device that uses so-called structured light. Structured light scanners display a pattern of light on the object being scanned. Selection patterns are tracked simultaneously by one camera or multiple cameras. The light pattern is created using light modulators. A 3D optical scanner using structured light obtains geometric data from an existing physical object. This data is used to obtain first a point cloud and then a virtual 3D model of the scanned object. The use of a scanner of this type is presented in the article [9] as a measuring device in the process of robotic pre-inspection of a thin-walled casting of a vane. The article [10] presents an analysis of the deformation of a welded joint by means of optical 3D scanning using structured light. It is very common that a given sensory solution does not offer satisfactory parameters in the implemented solution. Then it is necessary to build a hybrid system combining data from several systems. An example of the design of such a hybrid system is shown in the article [11]. The paper presents a hybrid contact-optical coordinate measuring system, which was designed for measuring and analyzing the geometry of objects. The developed solution combines the accuracy of contact measurement in the identification of measurement points and the speed of optical full-field scanning methods of structured light.

Another example of the design of a hybrid system combining data from lidar and thermal sensors is shown in the work [12]. As part of their work, the authors managed to obtain a 3D thermal model consisting of a point cloud describing the geometry of the building along with thermal data at each point. The literature review did not find solutions combining data from the laser tracker and 3D structural light scanner, which shows the validity of the work carried out. We believe that the synergy effect obtained by combining several solutions allows for advanced research and implementation work.

2. Hybrid System Concept

As part of the work on the design and programming of robotic stations, a hybrid measurement system was developed to automatically obtain robot path points. The basic component of the station is an ABB IRB2400 industrial robot with a Schunk tool changing system. An Atos Core 135 3D scanner or a 2.2 kW electro-spindle is mounted on the robot as required. The electro-spindle has an automatic HSK toolholder exchange system and a cutting tool box. An absolute laser tracker works with the robotic system. Thanks to the developed software characterized in the paper [14], the tracker sends the measured parameters to the robot controller. Using the algorithm developed in the paper [13], the coordinates of the points measured by the laser tracker are automatically converted to a position in the robot's base frame (Fig. 1).

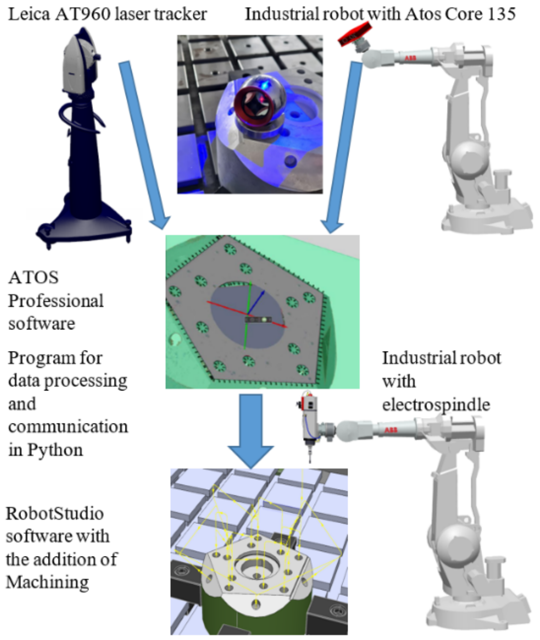


Fig. 1. System concept.

The realization of the algorithm for the operation of the developed system begins with scanning the workpiece with the Atos Core 3D scanner. Scanning is performed with the use of a robot, which moves the scanner to the selected measurement position (Fig. 2a). The advantage of the 3D scanner is the large amount of data acquired during a single measurement. The scanner used for the study registers 5 million pixels during a single measurement. The scanner's software combines the measurements taken into the point cloud shown in Fig. 2b.

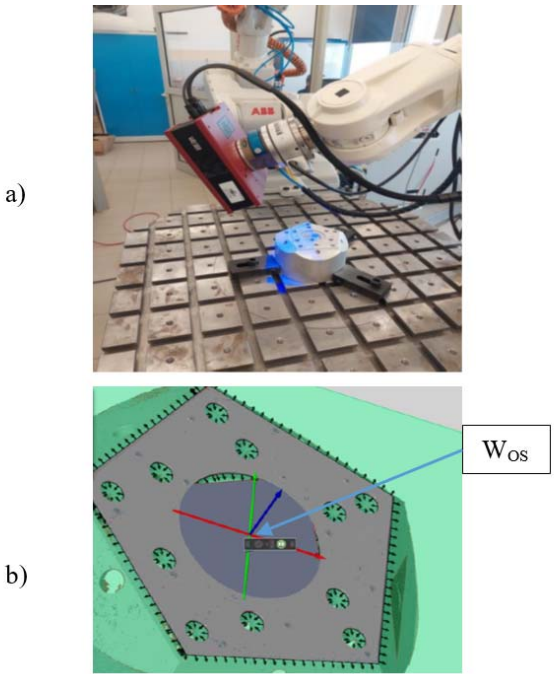


Fig. 2. Structured light scanner measurement.

The scanner software allows comparing the obtained point cloud with the CAD model. The disadvantage of the solution with the scanner is the lack of positioning of the obtained data relative to the robot system. It is primarily an inspection system. The system with a 3D scanner is also characterized by lower accuracy compared to an absolute laser tracker and sensitivity to parameters (light reflections) of the surface.

The accuracy obtained with the 3D structured light scanner depends on several factors but according to the data shown in the paper [15] it is about 80 µm and requires the use of a reflection leveling agent.

The 3D scanning is followed by measurement using an absolute laser tracker (Fig. 3). The advantage of the tracker is the high accuracy of up to 50 µm at a distance of 20 m and the ability to convert coordinates in the robot's base frame.

The disadvantage of the tracker is that the device records measurements one at a time, measuring the entire surface is time-consuming and measuring holes requires changing the retroreflector bases. In the case of the workpiece shown in Fig. 3, measurements were made of 3 points on the upper plane of the workpiece, 5 points marking the central hole and 2 points on one of the edges. The measurements made it possible to determine the coordinate system associated with the detail. The position of this system and therefore of the workpiece was defined relative to the robot's base frame (Fig. 4).

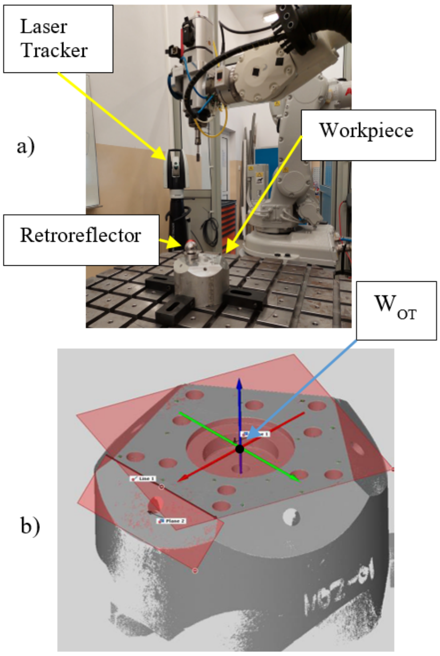


Fig. 3. Laser tracker measurement.

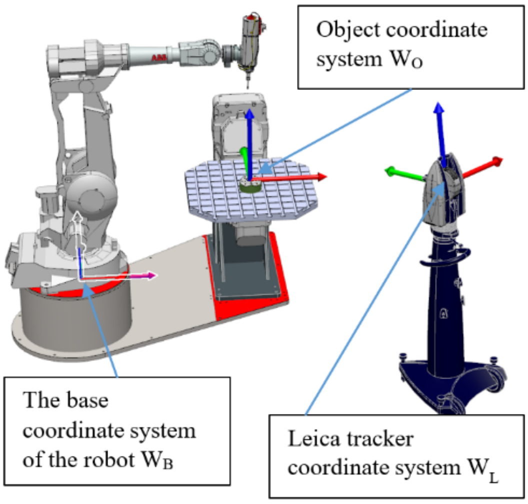


Fig. 4. System concept.

After receiving data from both measurement systems, it was necessary to build a hybrid system combining data from both sensors and positioning the received objects relative to the robot. It was decided to use the software supplied with the Atos Professional scanner, but it is possible to implement the developed algorithm in any programming tool such as Matlab software. Atos Professional software has the ability to create scripts in Python and add libraries of this language. The idea of the script's operation is based on defining relative to the same geometric objects (e.g. planes, circles, edges) the W_{OT} and W_{OS} coordinate systems and superimposing them on each other (the resulting W_O system shown in Fig. 4). The first step of the algorithm written in Python is to import from a text file the coordinates of the points measured with the laser tracker. Using the software described in the article [14], the points measured with the laser tracker are defined relative to the robot's base coordinate system, labeled W_B in Fig. 4. Next, the script defines planes, circles and intersection edges on the imported points. This is a standard feature of Atos Professional software based on the determination of geometric objects from the indicated points possible in many software tools. Once the geometric objects are determined, a new W_{OT} coordinate system is defined (Fig. 3b) at the center of the circle on the upper plane of the workpiece with a direction perpendicular to the edges of intersection of the planes. The point cloud measured by the 3D scanner is then imported into Atos Professional software. In this case, a W_{OS} coordinate system is defined (Fig. 2b) on the same geometric objects. It should be noted that the point cloud measured by the scanner consists of millions of points and the individual scans are combined thanks to the so-called reference points pasted on the object. The next step in the operation of the algorithm is the translation of the point cloud with the W_{OS} coordinate system into the W_{OT} coordinate system. The coordinate system obtained after superimposing the two layouts was called W_O. Since the W_{OT} coordinate system was defined relative to the W_B robot base coordinate system the same is true for the W_O coordinate system. The developed script allows to recalculate the position of all imported points, and thus objects, relative to any coordinate system in this case the W_B robot base coordinate system.

The tracker data was then transferred to Atos Professional software, where a developed Python script combined the point cloud with the data obtained from the laser tracker. The result of the script was the precise positioning of the point cloud relative to the robot's base coordinate system. The next step in the script's operation was the automatic determination of points located on the edges of the workpiece. Dedicated Atos software solutions and elements of the Python language allow the point cloud to determine points at the intersection of planes and holes, edges, etc., with a given density. The obtained data in the form of points in the robot's coordinate system were automatically transferred to RobotStudio software (Fig. 5).

There, on the basis of the received data, robot paths were generated using the additive intended for machining, realizing the machining. The existing milling cutters were used as machining tools to ensure that the desired shape was achieved.

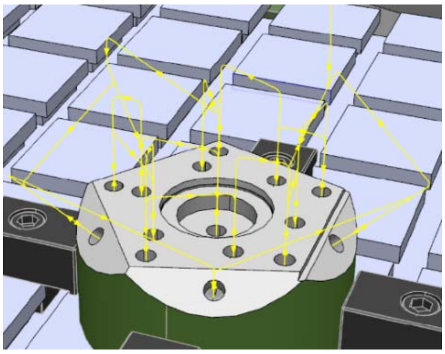


Fig. 5. Robot paths from the machining program.

The machining program was generated automatically, the input element was a cylinder with the diameter of the scanned part (no machining of the side surfaces was needed). The actual position of the cylinder on the machining table was determined using a laser tracker. The robot realized the programmed path with satisfactory results.

3. Conclusions

The paper presents the idea and an example of operation of a robotic hybrid system using a 3D scanner and an absolute laser tracker. The advantage of the developed solution is the ability to combine the large amount of data acquired from the structured-light scanner with the range and accuracy of the laser tracker. The disadvantage of the developed solution is the cost of the system, since its components are relatively expensive. The cost of the laser tracker is about 220000 Euro and the 3D structured light scanner about 95000 Euro. In the case of the developed solution, both of these devices were available for other research and it was decided to build a hybrid system. The software tools associated with the 3D scanner allow to compare the obtained point cloud with CAD models, automatically determine the elements of the object structure such as planes, holes, edges, etc. Solutions developed as a result of previous work allow to automatically convert the position of points and entire coordinate systems determined with the tracker to points in the base frame of an industrial robot. The developed system can be widely used in robotics to acquire data on the geometry of elements and their positions in a robotic system. As for the error that the developed hybrid solution generates, it is necessary to rely on the paper [13], where, due to the robot's accuracy error, the error of the system with a tracker was determined as 0.4 mm. If the robot is eliminated from the system, the coordinates of the points can be determined relative to the W_L tracker coordinate system (Fig. 4). Such a hybrid system for metrology applications consisting only of a tracker and a 3D scanner will be much more precise and its error will be close to the accuracy of the less precise of the system components, the 3D scanner. The study of such a system will be the subject of further work.

References

[1]. A. Ornat, M. Uliasz, G. Bomba, A. Burghardt, K. Kurc, D. Szybicki, Robotised geometric inspection of thin-walled aerospace casings, *Sensors*, Vol. 22, Issue 9, 2022, 3457.

[2]. K. Kurc, A. Burghardt, P. Gierlak, M. Muszyńska, D. Szybicki, A. Ornat, M. Uliasz, Application of a 3D scanner in robotic measurement of aviation components, *Electronics*, Vol. 11, Issue 19, 2022, 3216.

[3]. A. Burghardt, D. Szybicki, K. Kurc, M. Muszyńska, Robotic grinding process of turboprop engine compressor blades with active selection of contact force, *Tehnički Vjesnik*, Vol. 29, Issue 1, 2022, pp. 15-22.

[4]. D. Szybicki, A. Burghardt, K. Kurc, P. Pietruś, Calibration and verification of an original module measuring turbojet engine blades geometric parameters, *Archive of Mechanical Engineering*, Vol. 66, Issue 1, 2019, pp. 97-109.

[5]. A. Nubiola, I. A. Bonev, Absolute calibration of an ABB IRB 1600 robot using a laser tracker, *Robotics and Computer-Integrated Manufacturing*, Vol. 29, Issue 1, 2013, pp. 236-245.

[6]. L. S. Ginani, J. M. S. Motta, Theoretical and practical aspects of robot calibration with experimental verification, *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Vol. 33, Issue 1, 2011, pp. 15-21.

[7]. Z. Wang, P. G. Maropoulos, Real-time laser tracker compensation of a 3-axis positioning system-dynamic accuracy characterization, *The International Journal of Advanced Manufacturing Technology*, Vol. 84, Issue 5, 2016, pp. 1413-1420.

[8]. S. Droll, Real time path correction of industrial robots with direct end-effector feedback from a laser tracker, *SAE International Journal of Aerospace*, Vol. 7, Issue 2, 2014, 222.

[9]. A. Burghardt, K. Kurc, D. Szybicki, M. Muszyńska, J. Nawrocki, Software for the robot-operated inspection station for engine guide vanes taking into consideration the geometric variability of parts, *Tehnički Vjesnik*, Vol. 24, Suppl. 2, 2017, pp. 349-353.

[10]. J. Urminsky, M. Marônek, M. Jáňa, L. Morovič, Analysis of weld joint deformations by optical 3D scanning, *Acta Polytechnica*, Vol. 56, Issue 1, 2016, pp. 76-80.

[11]. J. Sładek, P. M. Błaszczyk, M. Kupiec, R. Sitnik, The hybrid contact–optical coordinate measuring system, *Measurement*, Vol. 44, Issue 3, 2011, pp. 503-510.

[12]. R. Sitnik, J. Sładek, M. Kupiec, P. M. Błaszczyk, M. Kujawińska, New concept of fast hybrid contact and no-contact measurement for automotive industry, *Proceedings of SPIE*, Vol. 6198, 2006, pp. 17-24.

[13]. D. Szybicki, P. Obal, K. Kurc, P. Gierlak, Programming of industrial robots using a laser tracker, *Sensors*, Vol. 22, Issue 17, 2022, 6464.

[14]. D. Szybicki, P. Obal, P. Penar, K. Kurc, M. Muszyńska, A. Burghardt, Development of a dedicated application for robots to communicate with a laser tracker, *Electronics*, Vol. 11, Issue 20, 2022, 3405.

[15]. R. Mendricky, J. Sobotka, Accuracy comparison of the optical 3D scanner and CT scanner, *Manufacturing Technology*, Vol. 20, Issue 6, 2020, pp. 791-801.

(035)

Ongoing Experiments with the DetObj Prototype for Vision Substitution

G. Bologna, L. Pfeiffer, D. Gonzalez, Q. Leblanc and J.-M. Boutay
University of Applied Sciences and Arts of Western Switzerland (HES-SO)
Rue de la Prairie 4, 1202 Geneva (Hepia), Switzerland
E-mail: Guido.Bologna@hesge.ch

Summary: Blindness is likely to double in the next fifteen years due to ageing. Our prototype for vision substitution has a first module denoted as "sensory" which transmits colors to the user through spatialized sounds of musical instruments. As color is not always sufficient to identify an object, we added an object recognition module by means of deep connectionist models. The originality of our system consists in its local/global modes that the user can select according to her needs. The local modes are defined in relation to the center of the image with respect to object recognition, color sonification and distance. The global modes are: the enumeration of the objects on the whole image, and the generation of a legend with respect to the frontal scene. The validation of our prototype was based on several experiments, the most difficult of which was to determine whether a user is able to move and to find several objects.

Keywords: Vision substitution, Sonification, Object recognition, Image captioning.

---

1. Introduction

Blindness is likely to double in the next fifteen years due to ageing. On the whole, blind people are able to move around with the help of traditional mobility aids (i.e. white canes and guide dogs). However, the limitations of these tools become apparent in many everyday situations, creating a strong need to seek help from others. As a result, exploring new environments is particularly challenging.

Retinal implants are intended to restore some of the functional vision lost after photoreceptor damage, the most common cause of blindness being, for example, retinitis pigmentosa and macular degeneration. Unfortunately, clinical trials show that these neuroprostheses still suffer from very limited resolution. As a result, basic visual tasks remain difficult or impossible, such as identifying objects and navigating unfamiliar environments.

A total substitution of vision by another sense would be impossible to achieve, as the bandwidth of touch or hearing would be insufficient. We propose a mobility aid prototype that encodes colored pixels with spatialized musical instrument sounds to represent and emphasize the color and location of visual features in the near environment. As color is not always sufficient to identify an object, we propose to use deep connectionist models to recognize them.

The proposed system highlights several novel aspects. First, its architecture with sensorial and recognition modules mimic the visual system by providing the user with the essential cues of vision (local, and global). In our prototype, the user can select local/global modes according to her needs. The local modes are defined in relation to the center of the image: object recognition; color sonification; distance measurement. The global modes are represented by the enumeration of the objects on the whole image and the generation of the legend of the frontal scene. We are currently validating our prototype with several gradual experiments, the most difficult of which is to determine whether a user is able to move and to find objects in a room.

2. Related Work

Jiang et al. wanted to represent the visual world in an auditory world allowing blind people to know what objects are around them and where [1]. They presented a system consisting of a camera to capture video whose stream is redirected to a server for image recognition using the Yolo detection model [2].

In [3], the objective was to detect a number of specific object classes inside a house. Following the improvement of deep convolutional neural networks, they tested the *RetinaNet* network. They obtained a mean average precision (mAP) of 84.61 %.

A transformer-based approach is the Shifted WINdow transformer (SWIN) [4]. This method used the concept of hierarchical feature maps. This allows the reproduction of the comportment of CNNs without convolution. The maps are divided into groups of patches and the attention of the transformer is computed over a group of patches instead of the map entirely. This allows a more fine-grained feature evaluation and search. SWIN2 achieved 64.2 % mAP on the COCO dataset.

Recently, a novel architecture was proposed for the image captioning problem. The GIT (Generative Image-to-text Transformer) performs image captioning and question answering [5]. It is also based on the encoder/decoder paradigm. It is composed of only one encoder and one decoder. The encoder is a SWIN transformer (see above) and the decoder is a simple transformer aiming at decoding and predicting the text from the feature maps extracted by the encoder.

3. Description of the DetObj Prototype

Our prototype consists of an Intel RealSense depth camera (D455) providing colored images with depth between 0.4 m to 6 m, a laptop and a wireless keypad that allows a user to select five different modes:

1. Color and distance sonification;
2. Identification of an object in relation to the center of the captured image;
3. Object enumeration (whole image);
4. Frontal scene captioning;
5. Distance with respect to the center of the image.

Fig. 1 depicts a blindfolded person using the prototype with the 3D camera attached on ski goggles. The wireless touchpad is not visible and a laptop is in the backpack.

Fig. 1. A person using our prototype of vision substitution without earphones.

The first mode is based on the sonification of colors and distance [6, 7]. In practice, the sonified part of an image is a row of 25 pixels in the center of an image frame. Only a single row is considered, as encoding multiple rows would require the use of 3D spatialization, instead of simple 2D spatialization. Two-dimensional spatialization is achieved by convolving monophonic instrument sounds with filters encompassing typical lateral cues, such as interaural delay and interaural intensity difference. Distance coding is achieved by the duration of the sounds with linear coding from 0.1 to 0.3 s, for distance ranging from 0.5 m to 6 m. The sonified 25 pixels are converted into the HSL color system, with hue (H) represented by the timbre of several instruments and saturation (S) rendered by pitch. For more details, see [6]. Note that brightness was not taken into account in this version of the prototype.

For the second mode we use the Yolo deep network [2]. The model slices the image into a grid. For each cell of this grid, different detection boxes are predicted with a confidence score. The probability of presence of each object class within each box is also evaluated to determine whether a class falls into a box and thus determine the position of the object. If an object is identified in the center of an image frame, its name is transmitted by voice synthesis, as well as the distance. If no object is identified, a distance is simply given.

With the third mode, the aim is to provide the user with a global view of the front scene (photographic mode). The basic idea is to divide the captured image into four sub-images: top-left; top-right; bottom-left; and bottom-right. For each quadrant, Yolo seeks to detect objects which are then communicated to the user.

The fourth mode is an alternative to the third. It uses a Transformer architecture (GIT) able to generate a caption from a picture. Typically, with a few words it globally describes the frontal scene.

The last mode provides the user with the average distance to a square of size 3×3 in the center of the image. For instance, if the user is 2.4 m from a wall then the prototypes says: “Two point four”.

4. Experiments

Five blindfolded participants performed several experiments of increasing difficulty. The simplest was to walk towards a wall and stop in time without impact. It was repeated five times with the fifth mode (distance mode). Note that for each participant and before each trial, the distance was randomly chosen between 2.5 and 5.5 meters. In addition, before each trial, each person was guided by a supervisor who made her or him do partial rotations while walking so that their sense of orientation was disturbed. Table 1 presents the results, with the columns showing for each person the average distance to the wall and the average distance at which a participant felt they had to stop before entering a collision. Overall, there was no collisions with the wall.

Table 1. Average results of the first experiment consisting of walking towards a wall and stopping in time without impact (standard deviations are shown in brackets). Participants used the prototype with the fifth mode.

Participant	Start (m)	End (m)
P1	4.2 (1.5)	0.8 (0.1)
P2	3.1 (0.7)	0.6 (0.0)
P3	4.0 (0.5)	0.6 (0.1)
P4	3.8 (0.8)	0.6 (0.1)
P5	3.8 (0.7)	0.8 (0.4)

The second experiment was similar to the first, but using the first mode (colors and distance mode). With each participant, we carried out a very short five-minute training phase consisting of becoming familiar with the variation in the duration of the sounds as a function of the distance from the wall. It should be noted that our aim was not to train participants in the associations between colors and sounds. Overall, in this second experiment, impacts with the wall were absent; Table 2 illustrates the results.

In the third experiment, participants stood two meters away from a table on which an umbrella, a backpack and a notebook were placed. They had to move forward and touch the three objects. Participants had only to use the second mode (object recognition). With this mode, if an object is not identified in the central part of an image, the distance is still provided to the user. All the participants performed the experiment three times and were successful. Table 3 depicts the results; from left to right are given in seconds the average time to walk and touch the first object, then the time to touch the second object, and finally the third. The last column represents the average time for each participant to complete the experiment (which is repeated three times). In addition, the last row signifies the average time for all the participants. Roughly, it took about a minute to identify an object and touch it.

Table 2. Average results of the second experiment consisting of walking towards a wall and stopping in time without impact. Participants used the prototype with the first mode.

Participant	Start (m)	End (m)
P1	3.9 (0.6)	0.5 (0.1)
P2	3.8 (1.0)	0.4 (0.1)
P3	3.7 (0.3)	0.4 (0.1)
P4	3.7 (0.7)	0.6 (0.1)
P5	4.5 (0.6)	0.5 (0.1)

Table 3. Average results of the third experiment consisting of moving, identifying three objects and touching them.

Participant	First (s)	Second (s)	Third (s)	Avg Time (min.)
P1	47.7 (15.6)	24.7 (8.7)	96.3 (101.9)	2.8 (1.4)
P2	92.0 (75.2)	49.7 (31.9)	63.0 (45.9)	3.4 (0.3)
P3	34.7 (5.0)	109.7 (100.3)	31.0 (28.2)	2.9 (1.2)
P4	88.0 (53.7)	60.0 (30.8)	75.7 (39.1)	3.7 (1.9)
P5	66.7 (18.6)	46.3 (33.3)	37.0 (28.8)	2.5 (0.4)
AVG	65.8	58.1	60.6	3.1

The last experiment was the most complicated. We allowed the participants to use the touchpad to select different modes. The participants were in a room of approximately 25 square meters in which they had to look for several objects. They had to pay attention to the walls and touch two out of three identified objects among backpack, laptop and umbrella. Note that other objects were present, but were not relevant to this experiment and therefore acted as distractors (chairs, tables, windows, etc.). Fig. 2 illustrates a person looking for two objects in a room.

Fig. 2. Fourth experiment: a person looking for two objects in a room.

Before starting the experiment, we explained to the participants a strategy that can be used to search for objects. For example, in order to gain a global perception of the frontal scene, we advised them to use mode three (object enumeration). For simplicity, we decided not to use the other alternative mode, i.e. mode four (scene captioning). In mode three, the distances are not provided. Therefore, we suggested users of the prototype to start moving and exploring the room by selecting mode two, with the option to switch to mode three.

Modes in which a synthesized voice communicates an object and possibly a distance do not allow for a smooth exploration of the environment, as the speech time is about three seconds in mode two and even longer in mode three. Therefore, a user had to move the head very slowly. On the contrary, the color and distance sonification mode is very reactive and in this case, the head can be moved quickly. We suggested that participants use this mode, especially when they are about 1.5 meters away from an object that has been identified. Table 4 presents the results of this experiment that was repeated three times for each participant. The columns show for each person the average time to grasp an object in seconds and the last column depicts the average time required to complete the experiment in minutes.

Table 4. Results of the fourth experiment. Participants moved around a room to find and grab two objects. For each participant it was repeated three times, with columns indicating the average time to grasp an object. The last column depicts the average time of the experiment.

Participant	First (s)	Second (s)	Avg Time (min.)
P1	67.3 (10.3)	254.3 (105.5)	5.4 (1.7)
P2	122.3 (55.8)	373.0 (174.7)	8.2 (2.3)
P3	168.3 (146.5)	175.0 (95.5)	5.7 (2.0)
P4	173.0 (197.3)	97.3 (16.2)	4.5 (3.5)
P5	87.0 (19.1)	94.3 (55.9)	3.0 (1.2)
AVG	123.6	198.8	5.4

Overall, this experiment was successful with an average time of 5.4 minutes. It was interesting to note that the participants quickly felt the need to have a global perception with the use of the third functionality of the prototype. Thus, participants often switched from a global mode of perception to a local mode (central identification of an object). Occasionally, some of them used color sonification, especially when an object was very close and no longer identified.

5. Conclusions

We proposed a prototype for visual substitution encompassing local and global functionalities. We were able to carry out several experiments of increasing difficulty. The basic scientific question was whether it is possible to identify objects and to direct oneself to pick them up. Thanks to the experiments carried out by five people, the answer to this question is positive. New questions then arose, such as whether better object recognition accuracy could speed up the completion times. Finally, it is questionable whether the search for objects could be made faster by taking longer sessions of learning. In particular, users of the prototype could take advantage of the color sonification mode, which is very responsive and useful for determining the location of objects relative to the background. The accomplished experiments are very promising and encourage us to continue with more complex experiments in which the perception of the close environment is crucial.

Acknowledgements

This work on vision substitution was carried out with funding from the University of Applied Sciences and Arts of Western Switzerland (HES-SO) and the Engineering and Architecture domain.

References

[1]. R. Jiang, Q. Lin, S. Qu, Let Blind People See: Real-Time Visual Recognition with Results Converted to 3D Audio, Report No. 218, *Standard University*, Stanford, USA, 2016.

[2]. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, *arXiv Preprint*, 9 may 2016, arXiv.1506.02640.

[3]. M. Afif, R. Ayachi, Y. Said, E. Pissaloux, M. Atri, An evaluation of RetinaNet on indoor object detection for blind and visually impaired persons assistance navigation, *Neural Process. Lett.*, Vol. 51, Issue 3, June 2020, pp. 2265-2279.

[4]. Z. Liu, Y. Lin, Y. Cao, et al., Swin transformer: Hierarchical vision transformer using shifted windows, https://arxiv.org/abs/2103.14030

[5]. J. Wang, Z. Yang, X. Hu, et al., GIT: A Generative image-to-text transformer for vision and language, https://arxiv.org/abs/2205.14100

[6]. G. Bologna, B. Deville, JD. Gomez, T. Pun, Toward local and global perception modules for vision substitution, *Neurocomputing*, Vol. 74, Issue 8, 2011, pp. 1182-1190.

[7]. J. D. Gomez, G. Bologna, T. Pun, See ColOr: An extended sensory substitution device for the visually impaired, *Journal of Assistive Technologies*, Vol. 8, Issue 2, 2014, pp. 77-94.

(037)

Extraction of Characteristic Ranges for Unbalance and Misalignment Conditions in Rotating Machines from Cepstrum Coefficients

A. D. Rincon-Quintero [1], C. L. Sandoval-Rodriguez [1], O. Lengerke [1],
J. G. Maradey-Lazaro [2] and O. A. Acosta-Cardenas[1]
[1] Unidades Tecnológicas de Santander, Calle de los estudiantes 9-82, 680005, Bucaramanga, Colombia
[2] Universidad Autonoma de Bucaramanga, Av. 42 #48 – 11, 680002, Bucaramanga, Colombia
Tel.: (+57) 607 6917700
E-mail: arincon@correo.uts.edu.co

Summary: The present investigation used the mean value for the extraction of characteristic ranges in order to contribute to the classification of faults in rotating machines by applying the Cepstrum coefficients. The classification of faults in rotating machines will allow, in the near future, to be used as a decisive tool in the preventive maintenance of machines. The methodology used is of the descriptive and correlational type, the method is deductive and the technique is experimental. The process carried out for the execution of this research requires two stages. The first stage is to obtain samples which are made from the induction of faults in a vibration bank. The second stage is the elaboration of an algorithm in Matlab for the extraction of ranges from the mean value. The determination of ranges from the mean value allowed to establish that failures due to unbalance represent 88 % efficiency in the extraction of ranges and failures due to misalignment represent 33.33 %.

Keywords: Cepstrum coefficients, Fault detection, Mechanical vibrations, Rotary machines, Testing bench.

---

1. Introduction

Rotating machines during their useful life present different states of operation determined by the bathtub curve, which are; adaptation stage, functioning stage and aging stage [1]. The industry's interest in making the machine remain in the operating stage encouraged it to create different maintenance plans, among the most prominent; pre-fault, corrective and failure. The study of the causes by which a machine can fail is preponderant for the industry; According to Ogun and Jackson [2], it is of great importance in order to study them, through control and monitoring of vibrations [3].

Different investigations focused on the knowledge and detection of the causes of common failures in rotating machines allowed to find that these generally are; unbalance, misalignment, mechanical play, shaft bending and wear, according to [4-6].

For this reason, in this investigation two important failures in rotating machines are studied based on the pre-fault maintenance plan, angular misalignment and unbalance, looking for identifiers of these two states or operating conditions, using the cepstrum transform applied to vibration records, mechanics obtained by sensing on a test bench [7-9].

2. Methodology

In this investigation, a vibration bench was used (Fig. 1), to simulate unbalance and misalignment faults in rotating machines. The vibration bench consists of a 3340 RPM and ½ HP Weg Motor, a flexible coupling, two sliding supports for the angular misalignment test, and two flywheels for the unbalance test [10-13].

For the acquisition of the data of the unbalance and misalignment faults, the Labview program, a National Instruments data acquisition card and a Dytran brand piezoelectric sensor were obtained [14-16].

Fig. 1. Vibration test bench.

For the acquisition of the data of the unbalance and misalignment faults, the Labview program, a National Instruments data acquisition card and a Dytran brand piezoelectric sensor were obtained [14-16].

The proposed methodology for the extraction of characteristic ranges from the mean value is described in the different sections of Fig. 2, where the type of investigation, the method used and the technique necessary for the development of the investigation are explained [17, 18].

In the Fig. 3 shows the Fourier spectrum for a misalignment test with an elevation located at approximately 120 Hz, corresponding to the second harmonic of the fundamental. Additionally, in the

Fig. 4 shows the spectrum envelope (Fig. 3) obtained from the Ceptrum coefficients.

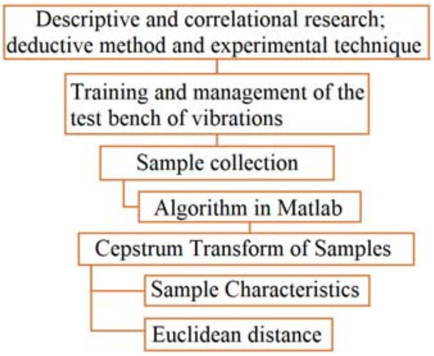


Fig. 2. Methodology applied in research.

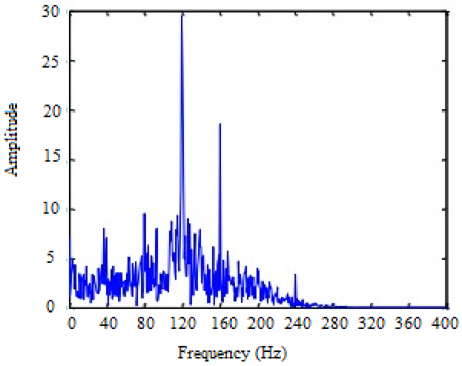


Fig. 3. Fourier spectrum.

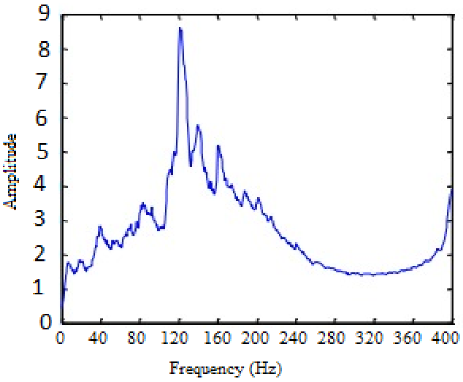


Fig. 4. Spectrum envelope from Ceptrum coefficients.

3. Results

The ranges obtained in this research allow a broad study of the classification based on the mean value, since it presents an efficiency of 88 % in the estimation of ranges for the variables of the unbalanced group with respect to the GCAB reference group. Regarding the variables of the misaligned group, it only represents 33.33 % effectiveness compared to the GCAB reference group, because it only allows estimating ranges for the variable of 1.5 degrees. For the variables of 0.5 and 1.0 degrees, an estimation of the range extraction cannot be made because there is a conflict between these variables. To solve the existing problem between the variables of 0.5 and 1.0 degrees of the misalignment group, a review of the cepstrum coefficients could be carried out in order to establish a relationship between variables corresponding to the failure groups, in order to determine a coefficient in which the variables can be differentiated from each other, especially the conflicting variables of the misaligned group.

Regarding the imbalance, in the Fig. 5 shows the maximum peak at 60 Hz, detecting an increase in the fundamental component in which the machine operating at 3600 RPM is oscillating, obtaining for all the tests an important excitation, corresponding to the spectrum envelope.

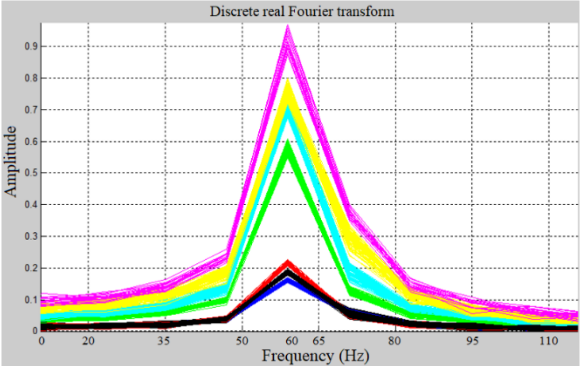


Fig. 5. Fourier spectrum.

4. Discussion

The samples obtained from the acquisition system should have been studied by means of algorithms in Matlab, in order to extract the largest number of samples corresponding to the same variable; that are grouped in a specific range with respect to frequency amplitude.

The visual method applied to the Cepstrum transform allowed to establish that the real and complex Cepstrum transforms, in the spectra of the Cuefrencia and of the envelope; they did not allow the determination of defined Cepstrum coefficients and differences for unbalance and misalignment failures in rotating machines. The real Cepstrum transform on the Mel scale allowed to find defined Cepstrum coefficients and differences in the failures studied in rotating machines by means of the visual method.

Coefficients on the Mel scale for failures in rotating machines can be found up to the fourth Cepstrum coefficient. The Euclidean distance allowed finding differences between the Cepstrum coefficients on the Mel scale according to the coefficient studied for unbalance and misalignment failures in rotating machines.

The characteristics of the Cepstrum coefficients on the Mel scale were determined through statistical analysis taking into account the variables of mean value, standard deviation, coefficient of variation, root mean square, kurtosis, asymmetry, and shape factor.

The statistical variables of mean value and root mean square allow finding differences according to the Cepstrum coefficient studied for each one of the unbalance and misalignment failures in rotating machines.

The coefficient of variation allowed us to determine that the variables studied for unbalance and misalignment failures do not have a very high standard deviation.

The shape factor made it possible to establish that all the variables studied for unbalance and misalignment failures according to the Cepstrum coefficient generally have a defined shape that is the same for all samples of the same variable.

5. Conclusions

The extraction of characteristic ranges from cepstrum coefficients by means of the average value allows to establish ranges for unbalance and misalignment, which could help in the precise classification of failures in rotating machines. The percentage of error found for the analyzed variables allows validating the method based on the determination of ranges from the total mean value. Finally, the similarity of some variables for the ranges determined by average value made their classification difficult, so it is necessary to find other coefficients in order to differentiate variables with similar ranges.

References

[1]. C. L. Sandoval-Rodriguez, E. A. Correa-Quintana, B. E. Tarazona-Romero, A. D. Rincon-Quintero, J. G. Maradey-Lazaro, Characterization of mechanical vibrations in a metal structure using the transform Cepstrum, *Period. Eng. Nat. Sci.*, Vol. 9, Issue 4, Oct. 2021, 767.

[2]. P. S. Ogun, M. R. Jackson, Active vibration control and real-time cutter path modification in rotary wood planing, *Mechatronics*, Vol. 46, 2017, pp. 21-31.

[3]. C. L. Sandoval-Rodriguez, J. G. Ascanio-Villabona, C. G. Cardenas-Arias, A. D. Rincon-Quintero, B. E. Tarazona-Romero, Characterization of the mechanical vibration signals associated with unbalance and misalignment in rotating machines, using the cepstrum transformation and the principal component analysis, *IOP Conference Series: Materials Science and Engineering*, Vol. 844, Issue 1, 2020, 12057.

[4]. Á. Miklós, Z. Szabó, Simulation and experimental validation of the dynamical model of a dual-rotor vibrotactor, *J. Sound Vib.*, Vol. 334, 2015, pp. 98-107.

[5]. M. Zou, P. Fang, Y. Hou, H. Peng, Investigation on multiple-frequency synchronization experiment of vibration system with dual-rotor actuation, *Mech. Syst. Signal Process.*, Vol. 164, 2022, 108261.

[6]. C. G. Cardenas-Arias, C. L. Sandoval-Rodriguez, A. D. Rincon-Quintero, D. Galvan, H. Tellez, Data Collector Design for Vibration Analysis by Raspberry Pi 3B Embedded System Means for Industrial Applications, *Journal of Physics: Conference Series*, 2022, Vol. 2224, Issue 1, 12032.

[7]. C. E. Garcia, M. R. Camana, I. Koo, Machine learning-based scheme for multi-class fault detection in turbine engine disks, *ICT Express*, Vol. 7, Issue 1, 2021, pp. 15-22.

[8]. C. L. Sandoval-Rodriguez, C. L. Higuera, J. G. Ascanio-Villabona, A. D. Rincon-Quintero, J. G. Maradey-Lazaro, Flow control strategies using classical regulatory technique and advanced H2 technique in an irrigation emulation pilot plant, in Recent Advances in Electrical Engineering, Electronics and Energy: Proceedings of the CIT 2021, Vol. 1, *Springer*, 2022, pp. 56-72.

[9]. T. Grüner, *et al.*, Evaluation of machine learning for sensorless detection and classification of faults in electromechanical drive systems, *Procedia Comput. Sci.*, Vol. 176, 2020, pp. 1586-1595.

[10]. T. Rooker, J. Stammers, K. Worden, G. Potts, K. Kerrigan, N. Dervilis, Error motion trajectory-driven diagnostics of kinematic and non-kinematic machine tool faults, *Mech. Syst. Signal Process.*, Vol. 164, August 2021, 2022, 108271.

[11]. C. L. Sandoval-Rodriguez, E. Y. Veslin-Diaz, B. E. Tarazona-Romero, J. G. Ascanio-Villabona, C. G. Cardenas-Arias, C. A. Angulo-Julio, Electromechanical Hand Prototype for the Simulation of the Opening and Closing Movement, *IOP Conference Series: Materials Science and Engineering*, Vol. 1154, Issue 1, 2021, 12035.

[12]. A. Theissler, J. Pérez-Velázquez, M. Kettelgerdes, G. Elger, Predictive maintenance enabled by machine learning: Use cases and challenges in the automotive industry, *Reliab. Eng. Syst. Saf.*, Vol. 215, 2021, 107864.

[13]. C. L. Sandoval-Rodriguez, A. C. Pita-Mejia, R. Villamizar-Mejia, B. E. Tarazona-Romero, O. Lengerke, Model to Relationship the Speed of Hand Movements with the SEMG Signals from the Forearm, *Journal of Physics: Conference Series*, Vol. 2224, Issue 1, 2022, 12094.

[14]. M. Richtarik, M. Gavlas, M. Drbúl, A. Czán, M. Sága, Influence of machine tool vibrations on geometric specifications of balls, *Transp. Res. Procedia*, Vol. 55, 2021, pp. 568-575.

[15]. C. L. Sandoval-Rodriguez, R. Mejia-Villamizar, B. E. Tarazona-Romero, A. D. Rincon-Quintero, A. J. Rodriguez-Nieves, Relationship between force signal and superficial electromyographic signals associated to hand movements, *Period. Eng. Nat. Sci.*, Vol. 11, Issue 1, 2023, pp. 64-73.

[16]. B. E. Tarazona-Romero, C. L. Sandoval-Rodriguez, C. G. Arias-Cardenas, J. G. Ascanio-Villabona, J. J. Valencia, Detection of structural alterations in metal bodies: An approximation using Fourier transform and principal component analysis (PCA), *Sci. Tech.*, Vol. 25, Issue 2, 2020, pp. 255-260.

[17]. S.-A. Korteland, T. Heimovaara, Quantitative inverse modelling of a cylindrical object in the laboratory using ERT: An error analysis, *J. Appl. Geophys.*, Vol. 114, 2015, pp. 101-115.

[18]. A. Ainapure, X. Li, J. Singh, Q. Yang, J. Lee, Enhancing intelligent cross-domain fault diagnosis performance on rotating machines with noisy health labels, *Procedia Manuf.*, Vol. 48, 2020, pp. 940-946.

(038)

Pulse Signals Standing out Based on Stochastic Resonance Effect

O. Kharchenko [1] **and Z. Kovacheva** [1,2]
[1] Information modeling, Institute of mathematics and informatics, Bulgarian academy of sciences,
Acad. Georgi Bonchev Str., Block 8, Sofia, Bulgaria
[2] University of Mining and Geology, ul. "Prof. Boyan Kamenov", 1700, Sofia, Bulgaria
E-mails: kharchenko@ math.bas.bg, zkovacheva@math.bas.bg

Summary: Reliability of message transmission is very important to the operation of telecommunications and radar systems. This is due to an increase in the amount of interference and an increase in the requirements for the quality of signal reproduction. In modern conditions, the development of techniques to avoid or reduce interference comes to the fore. The article considers a paradoxical method of signal extraction. It uses noise to stand out the signal. This method is the effect of stochastic resonance. A sequence of video pulses with a passive pause is considered. The possibility of standing out of these signals is shown using the effect of the stochastic resonance. Standing out of radio signals using linear frequency modulated and phase-code-manipulated signals, often used in the location, is also considered. The noise dispersion value after processing by the considered method is obtained. In addition, the possibility of amplifying the signal using the effect of the stochastic resonance is shown. A comparison is made between the nonlinear filtering of the pulse signals based on the stochastic resonance effect and the linear filtering. The numerical calculation showed a higher value of the output signal-to-noise ratio in the case of non-linear filtering.

Keywords: Stochastic resonance, Signal-to-noise ratio, Signal standing out, Noise dispersion.

---

1. Introduction

In a number of applications of radio engineering and radar, it is necessary to solve the problem of detecting signals in a complex interference environment. Receiving radio signals in the presence of interference creates one of the most difficult problems of signal reception – ensuring the noise immunity. This is due to an increase in the amount of interference and an increase in the requirements for the quality of signal reproduction. Therefore, increasing the noise immunity of radio receivers seems to be one of the promising and practically possible methods for increasing the range of telecommunication and other radio engineering systems. Optimal filters and correction codes are created for the noise suppression. Signal detection is held by the accumulation usage. Signal receiving issues are considered based on the theory of statistical solutions. Systems based on discrete and digital signals are increasingly used [1-3].

A discovery made in the 1980s provides a new method for solving this problem. This discovery is the phenomenon of stochastic resonance (SR). It enables to highlight weak signals against the background of powerful noise [4-9].

In this case, the output parameters of a nonlinear system, such as gain and signal-to-noise ratio (SNR), have a clearly defined maximum under certain conditions.

The concept of SR was introduced by physicists who investigated the occurrence of ice ages. For the first time this phenomenon was observed by Kramer during his investigation of the Brownian motion of a particle.

Currently, this phenomenon is known as fundamental and it takes place in various sciences: biology, physics, chemistry, medicine, sociology, etc.

SR is described by the equation [7]:

$$dy/dt = y(t) - y^3(t) + x(t) + n(t),$$

where $x(t)$ is an input sinusoidal signal; $n(t)$ is an input white Gaussian noise; $y(t)$ is an output signal of the non-linear system with SR effect.

For an SR system driven by a sinusoidal signal and white Gaussian noise, the output signal-to-noise ratio (SNR) have maximum at the defined input noise.

Stochastic resonator is a nonlinear system with SR effect.

2. Rectangular Pulse Standing out Applying SR

The rectangular pulse signal model is widely used in information and coding theory [10]. All modern telecommunication systems are currently moving from analog signals and the corresponding methods for their processing to digital signals, which requires the development of appropriate theoretical methods and models.

Consider a sequence of video pulses with a passive pause (Fig. 1, green line). Signal-noise mixture is illustrated by the black line. The input noise dispersion is 1 p.u.

The result of the numerical solution of the SR equation illustrates the efficient standing out of impulse signals from a mixture with high intensity noise (Fig. 1, red line). The output noise dispersion is 0.05 p.u. A significant decrease in the noise level at the output of the stochastic resonator can be noted even visually.

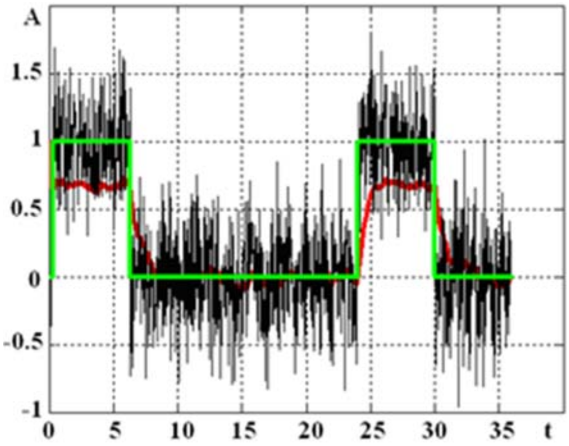


Fig. 1. A rectangular pulse signal standing out from an additive mixture of signal and Gaussian noise.

But in this case, there is no signal amplification.

A comparison is made between the nonlinear filtering of the rectangular pulse signal based on the SR effect and the linear filtering. Comparison with Butterworth filter is made on the basis of rectangular coefficient.

The rectangularity coefficient k_r is defined as the duration (ΔT) at a given level γ to duration $(\Delta T_{0.7})$ at 0.707 level of the maximum value ratio

$$k_r = \Delta T_\gamma / \Delta T_{0.7}$$

Usually, the conditional value of γ is equal to one of the certain values: 0.1; 0.01; 0.001. Obviously, for an ideal rectangular momentum $k_r = 1$. For our calculation $\gamma = 0.1$ is used.

Numerical calculations have shown that the quality of standing out, characterized by the rectangular coefficient, in the case of the nonlinear filter is higher than that of the Butterworth filter.

3. Radio Signals Standing out Applying SR

The use of complex signals in radar provides high resolution in terms of range and speed and can significantly increase the information content of radio systems. Linear frequency modulated (LFM) and phase-code-manipulated (FCM) signals are the most commonly used, as they are the easiest to generate and process and provide the required quality parameters of radars.

The main advantages of those radio signals are the long-distance detection and high probability of detection, reliability of detection of complex targets, and suppression of the narrowband interference.

A feature of LFM is the change in time of the instantaneous frequency according to a linear law [11]:

$$\omega(t) = \omega_0 + \beta t,$$

where ω_0 is the carrier frequency; β is the coefficient of dimension rad/s2, equal to the rate of change of the instantaneous frequency.

The frequency deviation over a time equal to the duration of the pulse τ_i is equal to $\Delta\omega = \beta\tau_i$.

The LFM use is possible only under this condition:

$$m = \Delta f \tau_i = \frac{\Delta\omega\tau_i}{2\pi} >> 1$$

The coefficient m is called the modulation parameter of the LFM.

When $m >> 1$, the value of the modulation parameter is almost equal to the value of the signal base.

The full phase of the signal has the form

$$\psi(t) = \int \omega(t)dt = \omega_0(t) + \frac{\beta t^2}{2}$$

The LFM radio pulse is described by the following mathematical model:

$$s(t) = \begin{cases} 0, & \mathrm{t} < -\tau_\mathrm{i}/2 \\ A\cos\left(\omega_0 t + \dfrac{\beta t^2}{2}\right) & -\tau_\mathrm{i}/2 \le \mathrm{t} \le \tau_\mathrm{i}/2 \\ 0, & \mathrm{t} > \tau_\mathrm{i}/2 \end{cases}$$

In Fig. 2, an example of LFM extraction using SR effect is given. We observe signal amplification and noise dispersion reduction at the output. Input signal is shown by the green line, signal-noise mixture is illustrated by the black line, and output signal – by the red line. The input noise dispersion is 1 p.u., the output noise dispersion is 0.4 p.u. The result illustrates the efficient extraction of LFM from a mixture with high intensity noise. In this case, the signal is amplified, therefore, the power gain is greater than 1.

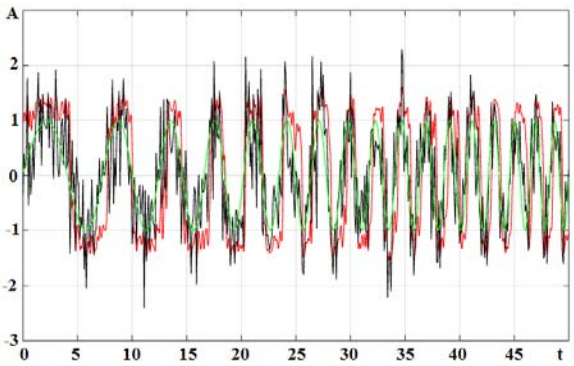


Fig. 2. LFM signal standing out from an additive mixture of signal and Gaussian noise.

FCM differ from conventional radio pulses in that the initial phase of their high-frequency filling changes at certain times by a certain amount. When emitting such signals, an increase in the noise immunity of reception and secrecy is provided [12].

The mathematical model of FCM is the following:

$$s(t) = \begin{cases} A\sin(2\pi f_0 t + \phi_i), \ (i-1)\Delta < t < i\Delta \\ 0, \ \mathrm{t} < 0 \text{ и } t > i\Delta \end{cases},$$

where $i = 1, \ldots, N$; $\Delta = T/N$ is the time interval into which the radio pulse of duration T is divided, the number of intervals is N; ϕ_i the phase change law.

Ordinary, signals with the Barker code are used as modulating signals, since in their autocorrelation functions the smallest level of side lobes is implemented, what provides noise immunity for communication, radar, and so on.

Barker code is known only for. $N = 2, 3, 4, 5, 7, 11, 13$. Barker signals with an odd value N are the only signals which are used in radar.

The main properties of autocorrelation function (ACF) of Barker signals are:

− The maximum level of the side lobes modulo does not exceed $1/N$ of the maximum value of the central lobe.

− The width of all ACF petals is the same and is equal to Δ.

− The ACF complex envelope is not negative at $N = 5, 13$. And it is not positive at $\mathrm{N} = 3, 7, 11$, beside the interval $-\Delta < \tau < \Delta$.

− The number of lateral lobes on each side of the central lobe is $(N-1)/2$. So the total number of petals is N.

− The height of the main lobe of the ACF complex envelope of the Barker signal is $A^2 N\Delta$, the amplitude of the side lobes is equal to $A^2\Delta$.

− The width of the main lobe is N times less than the duration of the Barker signal at 0.5 of the maximum ACF value.

Barker Code of length 13 has the form: +1, +1, +1, +1, +1, − 1, − 1, +1, +1, − 1, +1, − 1, +1.

We apply the effect of SR to Barker Code of length 13 (Fig. 3). Input signal is shown by the green line, signal-noise mixture is illustrated by the black line, and output signal – by the red line; the input noise dispersion is 1 p.u.; the output noise dispersion is 0.4 p.u.

The result illustrates the effective standing out of FCM from a mixture with high intensity noise. In this case, the signal is amplified, hence the power gain is greater than 1.

A comparison is made of the nonlinear filtering of radio signals based on the SR effect and the linear filtering. A matched filter is considered as a linear filter.

Matched filter generates an output to maximize the output peak power ratio to mean noise power within its frequency response. It is the optimal linear filter used to increase the SNR in the existence of additive stochastic noise.

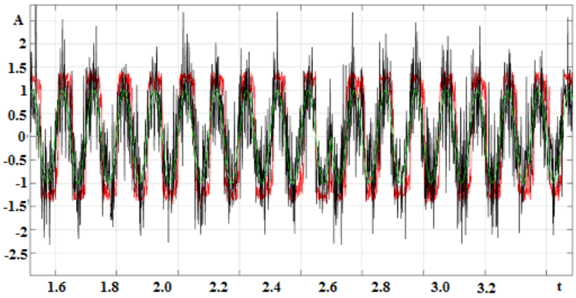


Fig. 3. FCM signal standing out from an additive mixture of signal and Gaussian noise.

These types of filters are generally used in radar, where a known signal is transmitted out and the reflected signal can be compared with the transmitted signal.

The output SNR of the matched filter is determined

$$SNR = 2E_S(t_0)/W_0,$$

where $E_S(t_0)$ is the signal energy at the time t_0; W_0 is spectral density of noise power.

The output SNR is determined for each filter. Calculations have shown that the output SNR of the nonlinear filter based on the SR effect is also higher than that of the matched filter with different input SNRs.

Comparison of the results of processing the LFM signal using a stochastic resonator and a matched filter showed that the stochastic resonator provides better signal standing out (Fig. 4). Parameters of the LFM signal in relative units are: $f = 0.13$, $\beta = 4$, $\tau_i = 10$.

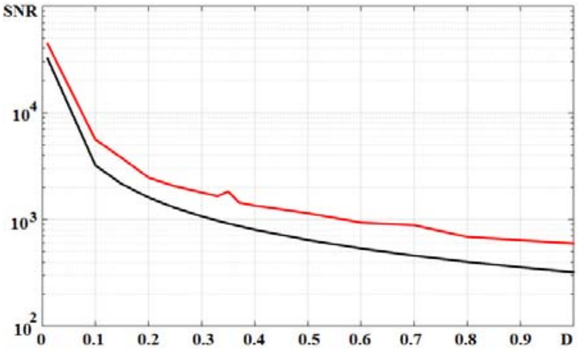


Fig. 4. Output SNR of the LFM stochastic resonator (red line) and output SNR of the matched filter (black line) on the input noise dispersion.

The SNR curve at the output of the stochastic resonator on the dispersion of the input noise has a local maximum with noise dispersion $D = 0.35$.

Comparison of the results of processing the FCM signal using a stochastic resonator and a matched filter showed that the stochastic resonator provides better signal standing out (Fig. 5). Parameters of the FCM signal in relative units are $f = 1$, $N = 13$, $\tau_i = 65$.

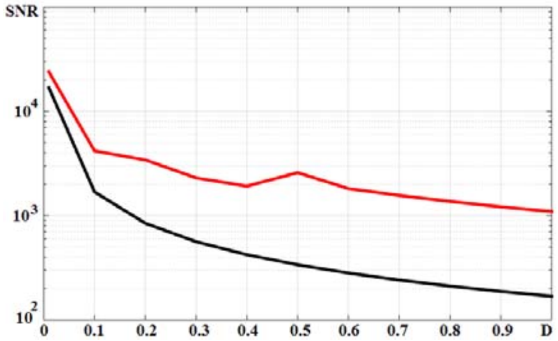


Fig. 5. Output SNR of the FCM stochastic resonator (red line) and output SNR of the matched filter (black line) on the input noise dispersion.

In this case the output SNR curve at the of the stochastic resonator on the dispersion of the input noise has a local maximum with noise dispersion $D = 0.5$.

4. Conclusions

This paper discusses the application of the SR effect to pulse signals. Signals are considered against the background of noise.

The numerical analysis of the rectangular pulse signal shows that the output noise dispersion is 0.05 p.u., when the input noise dispersion is 1 p.u.

The numerical analysis of the LFM shows that the output noise dispersion is 0.4 p.u., when the input noise dispersion is 1 p.u. and power gain is greater.

Thus, the article shows that the SR effect can significantly reduce the noise level in the case of pulse signals.

Acknowledgements

This work was supported by a grant from The Simons Foundation International (grant no. 992227, IMI-BAS).

References

[1]. R. Hanus, D. Mazur, Ch. Kreischer, Methods and Techniques of Signal Processing in Physical Measurements, *Springer*, 2019.

[2]. Y. Shirman, Radio electronic Systems: Fundamentals and Theory. Reference Book (Yu. Shirman, Y. Losev, D. Lakhvi's, Eds.), *ZAO "Makis"*, 2007, in Russian.

[3]. Y. Shirman, Radio electronic Systems: Fundamentals of Construction and Theory. Directory (Y. Shirman, S. Baghdasaryan, D. Lakhvi's, Eds.), *Radiogenic*, 2007, in Russian.

[4]. L. Tan, J. Jiang, Digital Signal Processing Fundamentals and Applications, *Elsevier*, 2018.

[5]. M. Jerome, A. Rama Kalyan, Brief Survey of Stochastic Resonance and Its Application to Control, in *Proceedings of the Third International Conference on Advances in Control and Optimization of Dynamical Systems (ACODS'14)*, Kanpur, India, 2014, pp. 313-321.

[6]. L. Gemmation, P. Hangi, P. Jung, F. Marchesani, Stochastic Resonance: A remarkable idea that changed our perception of noise, *The European Physical Journal*, Vol. 69, Issue 1, 2009, pp. 1-3.

[7]. O. Kharchenko, Simulation of the Stochastic Resonance Effect in a Nonlinear Device, *Global Journal of Researches in Engineering – F*, Vol. 15, Issue 7, 2015, pp.19-23.

[8]. P. Hangi, P. Jung, Stochastic resonance, *Reviews of Modern Physics*, Vol. 70, Issue 1, January 1998, pp. 224-283.

[9]. Z. Lai, Y. Leng, Weak-signal detection based on the stochastic resonance of bistable Duffing oscillator and its application in incipient fault diagnosis, *Mechanical Systems and Signal Processing*, Vol. 8, 2016, pp. 60-74.

[10]. B. Sklar, Digital Communication, Fundamental and Application, 2nd Ed., *University of California,* Los Angeles, 2001.

[11]. L. Xu, D. Feng, X. Wang, Matched-filter properties of linear-frequency-modulation radar signal reflected from a phase-switched screen, *IET Radar, Solar and Navigation*, Vol. 10, Issue 2, pp. 227-434.

[12]. R M. Davis, R. L. Fante, R. P. Perry, Phase-coded waveforms for radar, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 43, Issue 1, January 2007, pp. 401-408.

(039)

New Moment Functions for Signal and Image Analysis

Barmak Honarvar Shakibaei Asli
Centre for Life-cycle Engineering and Management, School of Aerospace, Transport and Manufacturing,
Cranfield University, Cranfield, Bedfordshire MK43 0AL, United Kingdom
Tel.: + 44 (0) 1234 754599
E-mail: barmak@cranfield.ac.uk

Summary: Unlike the existing moment functions which were defined in a traditional style, new moment functions with respect to the orthogonal polynomials' degree are defined, and it is shown how to reconstruct an image from its transformed continuous function. This technique is applied to the orthogonal Legendre, Chebyshev (of the first kind), and Hermite polynomials. The idea is about considering the order of polynomials as the independent variable to play a role in the signal or image functions during the summation operation. Switching the polynomial's order with their corresponding independent variable will give us a new moment transformation by having the same polynomials as moment kernels. A unique reconstruction of a signal or an image is given using integration over the specific polynomial's interval. A series of experiments are performed to validate the performance of the new moment function with respect to the signal/image reconstruction scheme.

Keywords: Orthogonal polynomials, Moment functions, Legendre moments, Chebyshev moments, Hermite moments, Signal reconstruction.

---

1. Introduction

More than sixty years ago, Hu [1] introduced geometric and complex moments (GMs and CMs) as image descriptors for visual pattern recognition by the theory of moment invariants. He proposed seven moments that can be used to characterize the shape of an object in an image. Since Teague [2] introduced the concept of orthogonal moments and proposed Zernike moments and Legendre moments, many new achievements in the field of moment functions have been presented. Examples of applications of Legendre moments include 3D image recognition [3], medical image compression [4], and object classification [5]. The basis of the image moments could be divided into two parts: non-orthogonal and orthogonal moments. GMs and CMs are examples of non-orthogonal moments while Zernike, Chebyshev, Hermite, Katcho and Meixner moments can be considered as orthogonal moments. Apart from this classification, we can divide the moment functions in terms of their continuity or discontinuity (discrete nature). There are continuous moments such as Zernike [6], pseudo-Zernike [7], Chebyshev [8], and Legendre [2] while Tchebichef [9, 10], Krawtchouk [11-13], and Racah moments [14] are examples of the discrete moments.

In this paper, we propose some new moment functions based on continuous orthogonal polynomials and focus on their orders (polynomial degree) instead of the independent variable. It shows us after more than six decades, the same proposed moment functions by Hu and Teague could be used from a different point of view. The experimental results confirm the reconstruction ability of these proposed continuous moments functions using integration instead of summation.

The rest of the paper is organised as follows. Section 2 presents the new form of three continuous moment functions (Legendre, Chebyshev of the first kind, and Hermite) unlike their ordinary transformation. Two sets of experiments (1D and 2D signal analysis) perform in Section 3 to validate the reconstruction ability of the new moment functions based on the theoretical framework proposed in the previous section. Concluding marks and future works are discussed in Section 4.

2. Continuous Moments

Many continuous moments such as Legendre, Hermite, Laguerre, Zernike, and Chebyshev (of the first and the second kinds) need to be mapped into image pixel values regions. For example, the region of the definition of the Legendre polynomials (LPs) is the interior of $[-1,1]$, therefore, a square image with the size of $N \times N$ should be mapped to this region. In this paper, we focus on this problem (digitalization) by revisiting the continuous moments from another perspective.

2.1. New Legendre Polynomials (LPs)

The explicit expression of the ordinary LPs in terms of simple monomials is given by

$$P_n(x) = 2^n \sum_{k=0}^{n} \binom{n}{k} \binom{\frac{n+k-1}{2}}{n} x^k \tag{1}$$

The graphs of the first five LPs are shown in Fig. 1 (left-hand side). This figure illustrates the variation of the LPs with respect to the continuous variable, x for different values of discrete order, n. If we plot the same LPs with respect to the discrete variable, n for different selected values of the continuous variable, $x \in [-1,1]$, the graphs can be found in Fig. 1 (right-hand side). From Fig. 1, it can be seen that the amplitude of the LPs always changes between -1 and 1.

2.2. New Legendre Moments (LMs)

The ordinary 1D LMs can be defined as:

$$L_n = \frac{2n+1}{2} \int_{-1}^{1} P_n(x) f(x) dx, \qquad (2)$$

where $P_n(x)$ is the LPs of the order n and $f(x)$ is the 1D signal with a length of N. It is possible to recover the approximated version of the original signal from its LMs, L_n, by using the following summation:

$$\tilde{f}(x) = \sum_{n=0}^{N-1} P_n(x) L_n, \qquad (3)$$

By revisiting Eqs. (2) and (3) and changing the independent variables from x to n, we define a new Legendre moment function as follows:

$$F_P(u) = \sum_{n=0}^{N-1} P_n(u) f(n), \qquad (4)$$

$$f(n) = \frac{2n+1}{2} \int_{-1}^{1} F_P(u) P_n(u) du \qquad (5)$$

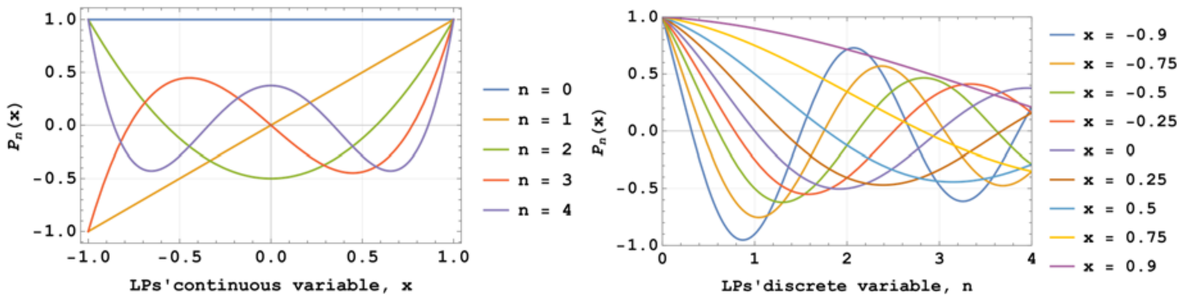


Fig. 1. The plots of the LPs with respect to the: (Left) continuous variable, x for different values of discrete order, $n = 0, 1, 2, 3$ and 4. (Right) discrete variable, n for selected values of the continuous variable, $x = 0, \pm 0.25, \pm 0.5, \pm 0.75$, and ± 0.9.

This new definition of the moment function and its reconstructed version (inverse transform) use the order of the LPs as a discrete variable to generate a continuous moment function as $F_P(u)$. The transformed variable, u can be varied from -1 to 1. By using this new moment, no need to map the image pixels to a small area of $(x, y) \in [-1,1] \times [-1,1]$. In other words, the approximation error rising from this mapping could be neglected. Eq. (4) can be extended in 2D form by having 2D summations and 2D integrations with discrete values of (n, m) and continuous transformed values of (u, v).

2.3. New Chebyshev Polynomials (CPs)

The trigonometric definition of the ordinary CPs of the first kind is given by

$$T_n(x) = \cos(n \cos^{-1} x); \; |x| \le 1 \qquad (6)$$

The graphs of the first five CPs of the first kind are shown in Fig. 2 (left-hand side). This figure illustrates the variation of the CPs with respect to the continuous variable, x for different values of discrete order, n. If we plot the same CPs with respect to the discrete variable, n for different selected values of the continuous variable, $x \in [-1,1]$, the graphs can be found in Fig. 2 (right-hand side).

2.4. New Chebyshev Moments (CMs)

Similar to the LMs, we can define the 1D CMs of the first kind as follows:

$$F_T(u) = \frac{1}{2} f(0) + \sum_{n=1}^{N-1} T_n(u) f(n), \qquad (7)$$

$$f(n) = \frac{2}{\pi} \int_{-1}^{1} \frac{F_T(u)}{\sqrt{1-u^2}} T_n(u) du, \qquad (8)$$

where $T_n(x)$ is the CPs of the order n and $f(n)$ is the 1D signal with a length of N. The integration process for signal reconstruction in (8) is identical to the proposed LMs in terms of the integration ranges (from -1 to 1). The above equations support the definition of the new CMs to avoid image mapping to the small area of $(x, y) \in [-1,1] \times [-1,1]$. Note that the summation in Eq. (7) does not start from 0 and the first term of the signal should appear in the form of $1/2 \; f(0)$.

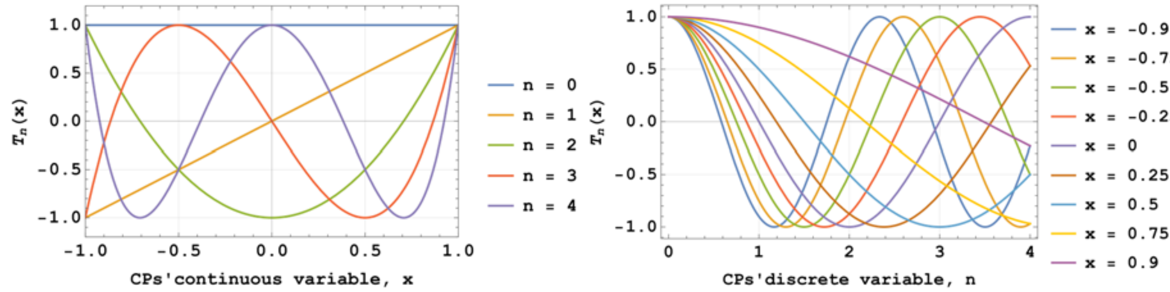


Fig. 2. The plots of the CPs with respect to the: (Left) continuous variable, x for different values of discrete order, $n = 0, 1, 2, 3$ and 4. (Right) discrete variable, n for selected values of the continuous variable, $x = 0, \pm 0.25, \pm 0.5, \pm 0.75$, and ± 0.9.

2.5. New Hermite Polynomials (HPs)

The Hermite polynomials of degree n are given by Rodrigue's formula as:

$$H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n}\left(e^{-x^2}\right) \qquad (9)$$

Unlike the LPs and CPs whose orthogonality ranges are between -1 and 1, the HPs' orthogonality can be defined in a wide range of $(-\infty, +\infty)$.

The graphs of the first five HPs are shown in Fig. 3 (left-hand side). This figure illustrates the variation of the HPs with respect to the continuous variable, x for different values of discrete order, n, while the right-hand side of this figure represents the plots of the same HPs with respect to the discrete variable, n for different selected values of the continuous variable, $x \in [-1,1]$. Note that the values of x could be varied by all real numbers but we only used the same values of x as the LPs and CPs.

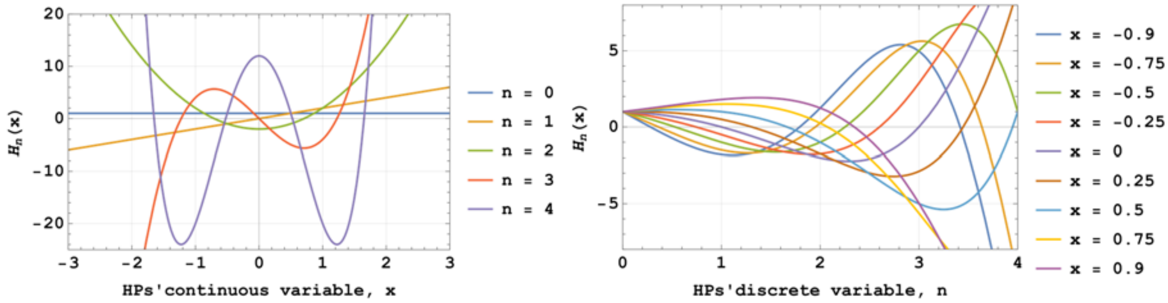


Fig. 3. The plots of the HPs with respect to the: (Left) continuous variable, x for different values of discrete order, $n = 0, 1, 2, 3$ and 4. (Right) discrete variable, n for selected values of the continuous variable, $x = 0, \pm 0.25, \pm 0.5, \pm 0.75$, and ± 0.9.

2.6. New Hermite Moments (HMs)

Equations (10) and (11) define the new Hermite transformation for a 1D signal, $f(n)$. As can be seen from (11), the integration limit is the whole range of the real numbers. Moreover, the exponential term, e^{-u^2}, plays a role as a weighting function to make sure that the recovered signal is reconstructed precisely.

$$F_H(u) = \sum_{n=0}^{N-1} H_n(u) f(n), \qquad (10)$$

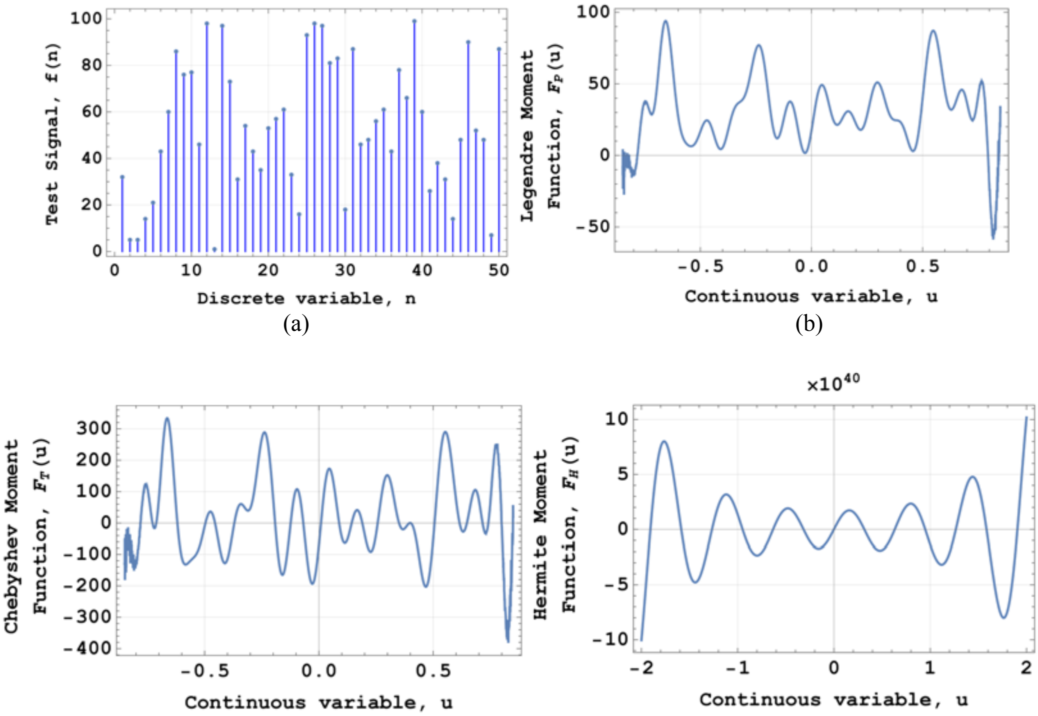
$$f(n) = \frac{1}{2^n n! \sqrt{\pi}} \int_{-\infty}^{+\infty} e^{-u^2} F_H(u) H_n(u) du \qquad (11)$$

In the case of the HMs, the image mapping issue might not be crucial, but mapping an image with the exact size of $N \times M$ is not easy to be compatible in any range of integer numbers. Therefore, the proposed HMs would help us to transform any 1D signal/2D image into a continuous form of moment functions.

3. Experimental Results

We executed three sets of experiments to validate the performance of the proposed moment functions in terms of their reconstruction abilities. The first experiment focused on a random 1D discrete signal with a length of 50 and a maximum amplitude of 100 which is shown in Fig. 4(a). Fig. 4(b) illustrates the new LMs function of the 1D signal. If we use the full integration ranges in (4) from -1 to 1, the reconstructed signal is perfect with no error and the original signal could be recovered. Fig. 4(c) and (d) show the proposed CMs and HMs functions of the same 1D signal, respectively. As can be seen from Fig. 4(d), the amplitude of the HM function is higher than the LMs and CMs since they perform in a wider range and not only in the unit square. For all proposed continuous moments, if we use full integration limits to recover the original signal, the reconstruction error would be zero (perfect reconstruction). Unlike the ordinary moment theory, the moments are not numbers and they can be considered as a continuous function ($F_P(u), F_T(u)$ and $F_H(u)$).

The second experiment proves the ability of the proposed moment functions for 2D discrete signals as images. Fig. 5(a) shows a binary image of size 9×9 and Fig. 5(b) illustrates its LMs function plot in the 3D format in terms of (u, v). Fig. 5(c) and (d) illustrate the same 3D plots for CMs and HMs with respect to values u and v.



(a) (b) (c) (d)

Fig. 4. (a) 1D Random test signal, (b) plot of the new LMs function, (c) plot of the new CMs function, and (d) plot of the new HMs function.

The third experiment is about reconstructing the binary image from its LM, CM, and HM functions using integration. As we mentioned earlier, if the integration limits are complete (it means $[-1,1]$ for LMs and CMs and $(-\infty, +\infty)$ for HMs), the reconstruction error would be zero. Fig. 6 shows the image reconstruction for the binary image with incomplete integration limits. The incomplete reconstructions can be seen in Fig. 6. The first row of this figure is about the LMs reconstruction. The integration limits taken in this experiment are $[-0.99,0.99]$ and $[-0.96,0.96]$, respectively [see Figs. 6 (a) and (b)]. The second row shows the reconstruction of the same binary image from its CMs functions. For this experiment, we used the same integration limits as $[-0.96,0.96]$ shown in Fig. 6(c) and $[-0.99,0.99]$ demonstrated in Fig. 6(d). Finally, the last row represents the image reconstruction results of the binary image from its HMs functions. Fig. 6(e) is generated based on the 2D version of (11) with integration limits of $[-0.96,0.96]$, while Fig. 6(f) shows the reconstructed image with integration limits of $[-0.99,0.99]$. As a simple comparison between these proposed continuous moment functions, we measured the reconstruction accuracy by the normalized mean square reconstruction error (NMSRE) defined as [15]

$$NMSRE = \sqrt{\frac{\sum_{n=0}^{N-1}\sum_{m=0}^{M-1}\left(f(n,m)-\tilde{f}(n,m)\right)^2}{\sum_{n=0}^{N-1}\sum_{m=0}^{M-1} f^2(n,m)}}, \qquad (12)$$

where $f(n,m)$ represents the original image and $\tilde{f}(n,m)$ is its reconstructed version. The comparison of the reconstruction errors is presented in Table 1. It can be seen that the NMSRE for the proposed CMs is much better than the proposed HMs and LMs for the same integration limits.

4. Conclusions

This paper concentrated on a new style of orthogonal moments by considering the polynomial degree as the independent variable of moment functions. Examples of three polynomials (LPs, CPs, and HPs) are applied for their moment function definition. Three experiments using 1D signals and images as 2D signals have been performed to validate the theoretical framework of the proposed method in terms of signal/image reconstruction. This work could be extended and generalized for any type of continuous orthogonal polynomials and will use the disadvantage of the image mapping into a unit square/circle which is time consumable and imprecise.

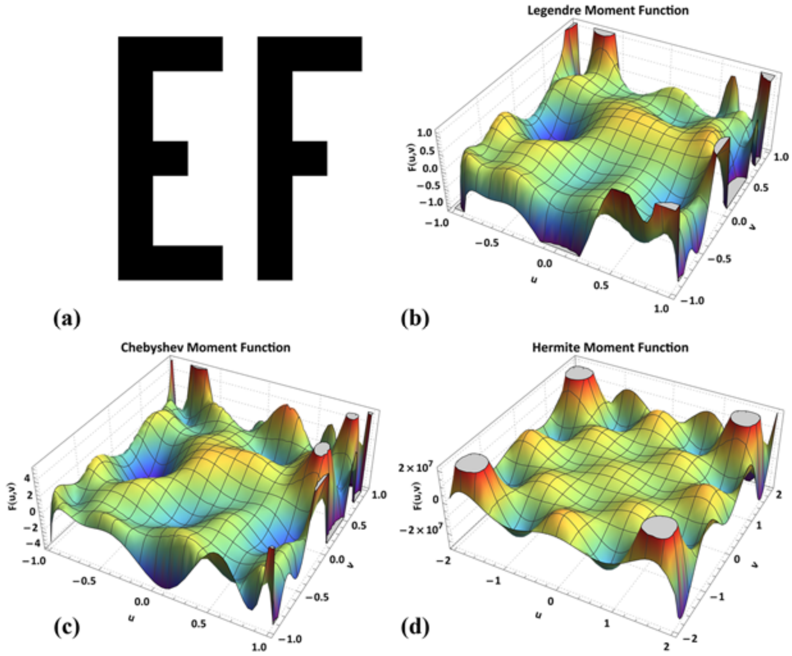


Fig. 5. (a) Binary image, (b) 3D plot of the new LMs function, (c) 3D plot of the new CMs function, and (d) 3D plot of the new HMs function.

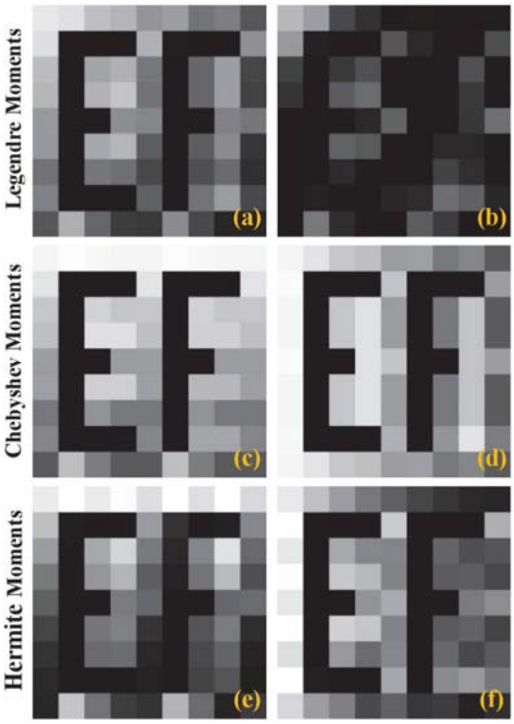


Fig. 6. Reconstructed images from different moment functions: (a) LMs with integration limit $[-0.99, 0.99]$, (b) LMs with integration limit $[-0.96, 0.96]$, (c) CMs with integration limit $[-0.96, 0.96]$, (d) CMs with integration limit $[-0.99, 0.99]$, (e) HMs with integration limit $[-0.96, 0.96]$, and (f) HMs with integration limit $[-0.99, 0.99]$.

Table 1. Reconstruction error for different moment functions.

Moment Functions	**Integration Limits**	
	$[-0.96, 0.96]$	$[-0.99, 0.99]$
LMs	0.9607	0.9233
CMs	$\underline{\mathbf{0.7582}}$	$\underline{\mathbf{0.6771}}$
HMs	0.8604	0.8011

References

[1]. M. K. Hu, Visual pattern recognition by moment invariants, *IRE Transactions on Information Theory*, Vol. 8, Issue 2, 1962, pp. 179-187.

[2]. M. R. Teague, 1980. Image analysis via the general theory of moments, *JOSA*, Vol. 70, Issue 8, 1980, pp. 920-930.

[3]. O. El Ogri, H. Karmouni, M. Sayyouri, H. Qjidaa, 2021. 3D image recognition using new set of fractional-order Legendre moments and deep neural networks, *Signal Processing: Image Communication*, Vol. 98, 2021, 116410.

[4]. K. M. Hosny, A. M. Khalid, E. R. Mohamed, Efficient compression of volumetric medical images using Legendre moments and differential evolution, *Soft Computing*, Vol. 24, 2020, pp. 409-427.

[5]. T. Arif, Z. Shaaban, L. Krekor, S. Baba, Object classification via geometrical, Zernike and Legendre moments, *Journal of Theoretical & Applied Information Technology*, Vol. 6, Issue 3, 2009, pp. 31-37.

[6]. B. H. Shakibaei, R. Paramesran, Recursive formula to compute Zernike radial polynomials, *Optics Letters*, Vol. 38, Issue 14, 2013, pp. 2487-2489.

[7]. C. W. Chong, P. Raveendran, R. Mukundan, The scale invariants of pseudo-Zernike moments, *Pattern Analysis & Applications*, Vol. 6, 2003, pp. 176-184.

[8]. J. Flusser, B. Zitova, T. Suk, Moments and Moment Invariants in Pattern Recognition, *John Wiley & Sons*, 2009.

[9]. B. H. S. Asli, R. Paramesran, C. L. Lim, The fast recursive computation of Tchebichef moment and its inverse transform based on Z-transform, *Digital Signal Processing*, Vol. 23, Issue 5, 2013, pp. 1738-1746.

[10]. R. Mukundan, S. H. Ong, P. A. Lee, Image analysis by Tchebichef moments, *IEEE Transactions on Image Processing*, Vol. 10, Issue 9, 2001, pp. 1357-1364.

[11]. P. T. Yap, R. Paramesran, S. H. Ong, Image analysis by Krawtchouk moments, *IEEE Transactions on Image Processing*, Vol. 12, Issue 11, 2003, pp. 1367-1377.

[12]. B. H. S. Asli, J. Flusser, Fast computation of Krawtchouk moments, *Information Sciences*, Vol. 288, 2014, pp. 73-86.

[13]. B. H. S. Asli, M. H. Rezaei, Four-term recurrence for fast Krawtchouk moments using Clenshaw algorithm, *Electronics*, Vol. 12, Issue 8, 2023, 1834.

[14]. H. Zhu, H. Shu, J. Liang, L. Luo, J. L. Coatrieux, Image analysis by discrete orthogonal Racah moments, *Signal Processing*, Vol. 87, Issue 4, 2007, pp. 687-708.

[15]. B. H. S. Asli, J. Flusser, New discrete orthogonal moments for signal analysis, *Signal Processing*, Vol. 141, 2017, pp. 57-73.

(041)

Wavelet Coefficients and Autoregressive Reflection Coefficients for Hand Movement Detection

Mohammad Nur Hossain Khan [1], Anik Baul [2], Utpal Mozumder [3], Gobinda Chandra Sarker [4], Jannatul Ferdaous Progga [5] and Ahmed Abdelgawad [2]
[1] Bangladesh University of Engineering and Technology, Electrical and Electronic Engineering, Dhaka, Bangladesh
[2] Central Michigan University, Electrical and Electronic Engineering, Michigan, United States
[3] University of Alaska Fairbanks, Alaska, United States
[4] Mymensingh Engineering College, Mymensing, Bangladesh
[5] North Dakota State University, Fargo, North Dakota, United States
Tel.: +17745190452
E-mail: nurhossain2301@gmail.com

Summary: Signals from surface electromyography (sEMG) play a vital role in action detection. The detection of fundamental hand motions from sEMG signals is presented in this work using an effective and efficient pattern recognition method. Raw EMG data is subjected to wavelet analysis in order to extract wavelet coefficients. Following that, the feature space is constructed by extracting reflection coefficients during the feature extraction stage. The technique is validated using a publicly accessible dataset, and K-nearest neighbor (KNN) is utilized to categorize six fundamental hand movements. Our experiment indicates that this approach achieved highly satisfactory performance and can be used in analyzing sEMG signals.

Keywords: Surface electromyogram, Wavelet coefficients, Reflection coefficients, Autoregressive model parameter.

---

1. Introduction

Electromyography (EMG) refers to the collective electric signal from muscles controlled by the nervous system and produced during muscle contraction. The EMG signal has a variety of clinical and biomedical applications. For example, it is used as a diagnostics tool for identifying neuromuscular diseases to detect Parkinson's disease and motor unit disorders [1, 2]. Many prosthetic devices such as prosthetic hands or arms can be controlled by EMG signals [3, 4]. In comparison to brain signal-controlled prosthetic arm, the myoelectric prosthetic arm is more convenient, which uses the EMG signals from the patient's available organ and controls the movement of the prosthetic arm [5].

For the identification of neuromuscular tasks from sEMG signals, different approaches have been suggested. In [6], the sEMG signal is first decomposed using the empirical mode decomposition (EMD), and then time-domain features such as zero crossing, waveform length, Willison amplitude, and integrated EMG are extracted from the EMD decomposed Signals for different hand movement classification with the help of the linear discriminant analysis. Frequency domain analysis is also performed to classify sEMG signal [8-10]. For example, in [10], spectral collaborative representation is utilized for hand gesture recognition. Considering the complex nature of the sEMG signal, various decomposition techniques are also employed on sEMG data prior to feature extraction. Kakoty et al. [11] utilizes wavelet decomposition on the entire frame and linear kernel support vector machine (SVM) classifier to differentiate six wrist motions. In [12], along with wavelet decomposition, fractal analysis is used to classify hand movements. Deep learning and neural network are also trending in analyzing EMG signal [13, 14]. In general, it is observed that due to noise-like characteristics of sEMG signals, satisfactory classification performance cannot be achieved for similar types of neuromuscular actions. In order to overcome this problem, in many cases, a very high feature dimension is used. An efficient scheme of extracting features from decomposed sub-frame of sEMG signal, which offers very satisfactory classification performance with low feature dimension, is still in great demand.

In this paper, a Wavelet domain autoregressive (AR) modeling of sEMG signal is introduced to classify various types of hand movements. We propose using AR reflection coefficients as features that offer significantly discriminative features with a very low feature dimension. In the wavelet domain, both approximate and detail coefficients are utilized. Moreover, features obtained from two different channels are also concatenated to obtain better classification performance. For the purpose of classification, the K nearest neighbor classifier is used, and the performance of the proposed scheme is tested extensively on a standard sEMG dataset. The structure of the paper is as follows: Section 2 describes the proposed method, which is divided into three subsections, Section 3 provides experimental results and analysis, and finally, conclusions are provided in Section 4.

2. Proposed Method

The proposed method consists of four steps: Pre-processing of data, wavelet decomposition of pre-processed data, feature extraction using autoregressive modeling, and classification. In the following subsections, each process is described in detail.

2.1. Pre-processing

Considering the real-life application, a decision on the type of hand movements is required from a small duration of a given EMG data. As a result, each long-duration raw EMG signal trial is divided into short-duration sub-frames for feature extraction and classification purposes. One advantage of dealing with a short duration of EMG signal is to expect fewer random fluctuations in that sub-frame. Moreover, such a sub-frame-based analysis helps in obtaining consistent feature characteristics. Therefore, inconsistency in a small portion of the test frame will have a less significant effect on the overall performance of classification. The neural firing in EMG data changes throughout the whole trial, and the applied stress levels also vary from the beginning to the end of action, which also differs for different actions. Too large a subframe would have no additional advantage over the whole frame, resulting in non-stationary data with variable mean and inconsistent features. Hence a reasonable sub-frame duration is chosen in the proposed method. A percentage of overlap is chosen, which allows getting more training data. Different hand movements involve different rates of neural firing; for example, movement like tip requires a smaller number of neural firing compared to the movement type hook for a certain time duration. As a result, it is expected that there will be significant variation in spectral distribution. However, if raw data are considered to extract such variation, it may not provide satisfactory time and frequency resolution. As an alternate, in this paper, we propose to use discrete wavelet transform (DWT). Instead of dealing with raw data for feature extraction, we develop an efficient feature extraction scheme that utilizes wavelet decomposed signals.

Next, the wavelet decomposed EMG signals may be used for feature extraction. However, the objective is to capture the spectro-temporal variation to differentiate various types of hand movements. In [15], for needle EMG data the MUAPs were extracted from the wavelet decomposed EMG signal. However, for the surface EMG data, due to the presence of noise, precise determination of MUAP is very difficult and strongly depends on the acquired data. Hence, MUAP based feature extraction for the decomposed EMG data is not adopted in the case of the surface EMG signal. One possibility is to use the statistical features of the wavelet coefficients as features. In this case, it is difficult to select the optimal statistical features to obtain the best accuracy in training and testing. Therefore, instead of conventional non- parametric spectral analysis, in this paper, we propose to use Autoregressive modeling of the decomposed wavelet EMG signal and utilize the model parameters.

2.2. Feature Extraction

A feature extraction scheme based on the autoregressive (AR) model is proposed to perform the basic hand movement classification. The output of a liner time invariant (LTI) AR system can model the DWT coefficients of a sub-frame of an EMG signal [16], which can be expressed as

$$y(n) = -\sum_{m=1}^{R} a_m y(n-m) + w(n), \qquad (1)$$

where $w(n)$ is white Gaussian noise having a zero mean and a non-zero variance that can be depicted as σ^2, $[a_m]$. These are considered as the AR parameters or linear prediction coefficients. AR parameters can be calculated using Yule-Walker equations as

$$\begin{aligned} r_y(l) &= -\sum_{m=-1}^{R} a_m y(l-m) + \sigma_w^2\,\delta(l), l \geq 0 \\ &= r_y(-l), l < 0 \end{aligned} \qquad (2)$$

here, $r_y(l)$ is the l-th lag of auto-correlation function (ACF) of y(m). This ACF can be expressed as

$$r_y(l) = -\frac{1}{N}\sum_{n=0}^{N-1-l} y(n)y(n+1), l \geq 0 \qquad (3)$$

Although AR parameters have been widely used as features in signal processing, it suffers from some major drawbacks. AR parameters have a large variation in parameters value without having any boundary. Therefore, in this paper, we propose to employ reflection coefficient for feature extraction as an alternative to AR parameters. The *i-th* reflection coefficient (b_i) calculates the correlation between current output $y(n)$ and $y(n-i)$ that is i sample past output, which filter the intermediary observations $y(n-1)$ to $y(n-i+1)$. The reflection coefficients can be determined by utilizing Levinson-Durbin recursion formulas as follows [16]:

$$b_i = \frac{r_y(i)+\Sigma_{J=1}^{i-1} c_J^{i-1} r_y(i-j)}{h^{i-1}}, i = 1,2,\dots,p, \qquad (4)$$

where c_j^i at the i-th iteration can be obtained as

$$\begin{aligned} c_j^i &= c_j^{i-1} - b_i c_{i-j}^{i-1}, 1 \leq j \leq (i-1), \\ c_i^i &= b_i, \end{aligned} \qquad (5)$$

here, h^i is the residual energy after *i-th* iteration that can be calculated as

$$h^i = (1 - b_i^2)h^{i-1}, \qquad (6)$$

initially, $h_0 = r_y(0)$, $c_0 = 0$ and $i = 1, 2, \dots, M$. $r(y)$ is an auto-correlation estimation of y(n).

As one of the advantages, the reflection coefficient can be computed directly from auto-correlation values by employing the above recursive formula, whereas AR parameters calculation involves inverse matrix computation, which can be computationally costly and inefficient. So increasing the number of reflection coefficients from P to $P+1$ will provide only one new coefficient where the rest of the coefficients will remain unchanged. Another shortcoming of AR parameters is having no limit for the value that can be a problem in feature- based classification. In contrast, for a stable AR system, the value of the reflection coefficient is bounded, which is ≤ 1. sEMG signals may contain various types of noises, such as lead noises and environmental noise. These signals can also get corrupted from procedural mistakes. Reflection coefficients suffer from less variation in effects of these external noises than AR parameters. Hence, using reflection coefficients as features acquired from wavelet coefficients of an sEMG sub- frame to perform signal classification is a much better option rather than using AR parameters. Considering the size of feature vectors, without further experimentation, in this paper, 1st eight reflection coefficients from detail and approximate coefficients of decomposed sEMG sub-frame is obtained for classification purpose.

2.3. Classification

In a real-life application involving EMG data, a minimal number of trials are generally available from a particular subject, and in that case, it is a challenging task to obtain satisfactory training. Here the classification accuracy is tested for each subject separately. Considering its simplicity and consistent performance, in the proposed method, (KNN) classifier is employed for movement classification. The KNN scheme is a simple and efficient method used in pattern recognition. The classifier in question considers a distance function that is computed between the feature set of the test data and k neighboring patterns in the training data set. In the proposed method, classification is performed using only the reflection coefficients features obtained from wavelet decomposed EMG sub-frames.

3. Results and Discussion

In this work, the proposed method involves wavelet de- composition and AR modeling of decomposed sEMG signals. After extracting features in the form of reflection coefficients, classification is done by the KNN classifier. In this section performance of the proposed method along with the database description is presented.

3.1. Database Description

A publicly available surface EMG databases are used in this work [6]. This database is available at the dataset archive (http://archive.ics.uci.edu/ml/datasets/sEMG+ for+ Basic+Hand+movements). The databases contain six different movements shown in Fig. 1 and described below:

a) Spherical (spher): to hold spherical instrument;
b) Tip: to hold small instrument;
c) Palmar (palm): to grasp with palm when face an object;
d) Lateral (lat): to hold objects that are thin and flat;
e) Cylindrical (cyl): to hold cylindrical instruments;
f) Hook: to support something with a heavy load.

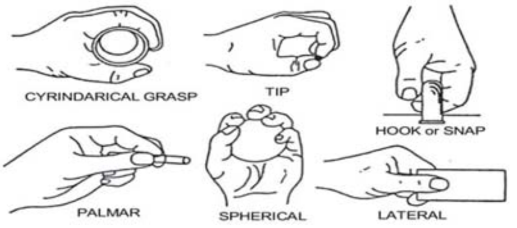


Fig. 1. Illustration of hand gesture.

Data are collected from five subjects (two males and three females) of the age group approximately (20 to 22-year-old). Every subject performs each of the six movements thirty times for six seconds. Therefore, a total of one hundred and eighty-six-second long two-channel EMG signals are recorded for each subject. The data are collected at a rate of 500 samples/sec, which results in 3000 samples in one six-second trial.

3.2. Goodness of Feature

The quality of the proposed feature is investigated by means of class separability. Standard goodness of feature measures Bhattacharyya Distance (BD) and Geometrical Separability Index (GSI) are utilized to investigate the feature quality. For data clusters, BD is computed as [17]

$$BD = \frac{1}{8}(\mu_2 - \mu_1)^T \left[\frac{1}{2}(\delta_1 + \delta_2)\right]^{-1} (\mu_2 - \mu_1) + + \frac{1}{2} ln(\frac{\det\left(\frac{\delta_1+\delta_2}{2}\right)}{\sqrt{\det(\delta_1)} * \sqrt{\det(\delta_2)}}, \tag{7}$$

here, δ_i and μ_i are covariance matrix and mean vector of $i-th$ cluster. If the distance is larger, it is easier to separate the class.

The geometrical separability index (GSI) that is also known as Thornton's separability index, which provides the measure of the separability of two classes in the nearest neighbor sense [17] and is defined as

$$GSI = \frac{\sum_{i=1}^{N}(f(x_i)+f(x_i^t+1))}{N}, \tag{8}$$

where x^t is the nearest neighbors of x and N denotes the number of data points. Higher values of GSI represent better feature quality.

Average Bhattacharyya distances and average GSI for five subjects are shown in Table 1. From the average distances and GSI of five subjects, it can be easily deduced that male 2 performs with highest separability and female 2 performs with lowest separability. Average BD for male 2 is 29.17 and GSI is 0.99 which is higher than other movements. Therefore, classification result is better for this subject than other four subjects. Same reasoning can be concluded between separability index and accuracy (Table 3 for all models in the proposed method.

Table 1. Average Bhattacharyya Distance and GSI for all Subjects.

Subject	Bhattacharyya Distance	GSI
Female 1	16.61	0.95
Female 2	8.52	0.93
Female 3	21.55	0.98
Male 1	19.71	0.97
Male 2	29.17	0.99

Confusion matrix for both channel combined result of five subjects is shown in Table 2. From the confusion matrix, it is clear that most of the misattributed results are coming from hand movements requiring lower energy like tip, palmer and lateral movements.

Table 2. Confusion Matrix for all Subjects.

Female 1	Cylin-drical	Hook	Spherical	Palmer	Lateral	Tip
Cylindrical	217	12	21	0	0	0
Hook	3	223	3	0	0	0
Spherical	20	5	216	0	0	0
Palmer	0	0	0	230	26	11
Lateral	0	0	0	4	210	0
Tip	0	0	0	6	4	229

Female 2	Cylin-drical	Hook	Spherical	Palmer	Lateral	Tip
Cylindrical	231	12	3	6	0	0
Hook	2	224	0	6	3	0
Spherical	6	0	237	0	0	0
Palmer	1	4	0	208	4	13
Lateral	0	0	0	7	226	25
Tip	0	0	0	13	7	202

Female 3	Cylin-drical	Hook	Spherical	Palmer	Lateral	Tip
Cylindrical	235	0	3	1	0	0
Hook	0	240	0	0	0	0
Spherical	4	0	237	0	0	0
Palmer	1	0	0	227	12	7
Lateral	0	0	0	11	228	7
Tip	0	0	0	1	1	226

Male 1	Cylin-drical	Hook	Spherical	Palmer	Lateral	Tip
Cylindrical	239	0	2	0	2	1
Hook	0	240	2	0	0	0
Spherical	1	0	236	0	0	0
Palmer	0	0	0	230	5	5
Lateral	0	0	0	1	226	4
Tip	0	0	0	9	7	230

Table 2. Continued.

Male 2	Cylin-drical	Hook	Spherical	Palmer	Lateral	Tip
Cylindrical	238	1	0	0	0	0
Hook	0	238	0	0	1	0
Spherical	0	0	240	0	0	0
Palmer	0	0	0	240	0	0
Lateral	0	1	0	0	239	0
Tip	2	0	0	0	0	240

Table 3. Comparison of Proposed Method with Existing Methods.

Subject	Classification Accuracy(%)			
	[6]	[18]	[19]	Proposed
Female 1	87.25	93.04	88.15	94.17
Female 2	88.05	86.66	84.04	93.40
Female 3	85.53	97	90.61	97.08
Male 1	90.42	99.23	91.22	98.19
Male 2	94.80	97.66	96.83	99.58
Average	89.21	94.72	90.17	96.48

It can be also concluded that a little portion of cylindrical, hook or spherical class is misclassified but fewer of them should be classified as tip, lateral and palmer class which can be viewed from confusion matrix of female 3. In Fig. 2 4th reflection coefficient values of both channels for female 2 are plotted in the box plot to show how it varies for all movement of five subjects when the AR model order is 8. It shows a clear distinction between high and low force movement which can be easily distinguished but there are also some overlaps mostly among lower energy movements and some among high energy movements. These overlaps are causing misclassification and lowering accuracy.

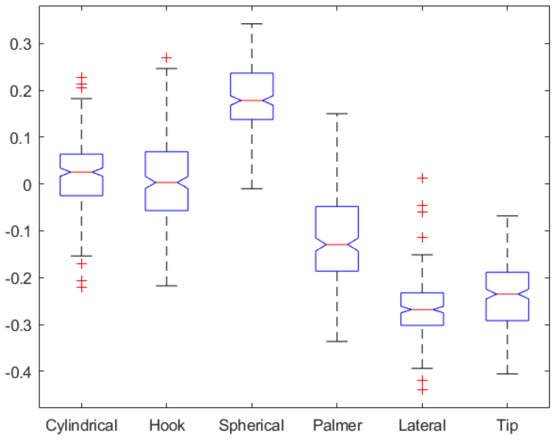


Fig. 2. Box plot of 4th reflection coefficient for Female 2 channel 1.

3.3. Classification Result

The performance of the proposed method described above is investigated by simulation in terms of classification accuracy. After feature extraction, the sub-frame features are fed into the classifier. In this work, the data from each subject are not mixed with data from any other subject. Cross-validation approach is used here in order to estimate the performance. The proposed method is tested with ten-fold cross-validation. In 10 fold cross-validation, 90 % of sub-frames are selected for training and another 10 % for test. The whole process is repeated ten times, and the accuracy is recorded.

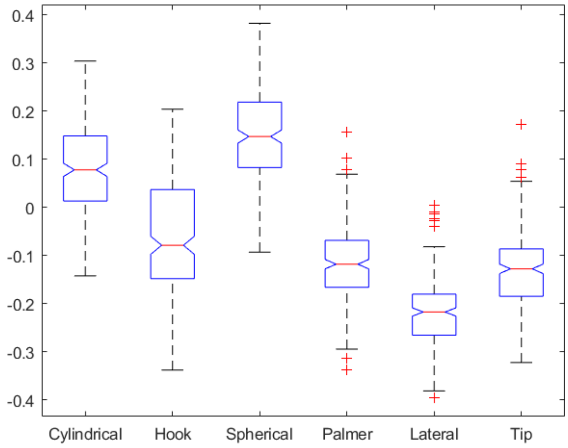


Fig. 3. Box plot of 4th reflection coefficient for Female 2 channel 2.

To our knowledge, the proposed method performs better than the existing state-of-the-art method. In 2017, Akben [18] used energy value and concordance correlation to classify six hand movements, and the classification accuracy is 94.72 %. Khan et al. [19] use singular value decomposition to classify six kinds of arm movements with 90.17 % accuracy. Sapsanis et al. [6] used EMD with an LDA classifier to differentiate six hand movements, and the average classification accuracy is 89.21 %. By analyzing the results of table III it can be inferred that the classification accuracy of the proposed method is better than previous methods.

4. Conclusions

Basic hand action movements can be classified by different types of conventional methods, which can sometimes be tedious and computationally difficult. The outcome of this research is a robust feature extraction technique, which can be implemented for the classification of various hand movements with the help of non-invasive sEMG signals. In this paper, a simplified method is presented, which helps in enhancing the correct extraction of feature values and thus appropriate classification. The proposed method performs satisfactorily in classifying six hand movements. In this work, instead of extracting the feature considering the entire frame, overlapping sub-frames of the raw signals are created, and then wavelet coefficients are extracted using DWT. After that, reflection coefficients are computed from each of these decomposed sub- frames. Finally, these reflection coefficients extracted from both channels are subjected to classification. This method gives better performance than the existing method to distinguish hand movements. In our future work, we look forward to working with a larger database and also focusing on improving accuracy by developing more efficient classifying techniques.

References

[1]. M. W. Flood, B. R. Jensen, A. S. Malling, M. M. Lowery, Increased EMG intermuscular coherence and reduced signal complexity in Parkinson's disease, *Clinical Neurophysiology*, Vol. 130, Issue 2, 2019, pp. 259-269.

[2]. M. Chen, X. Zhang, P. Zhou, A Novel validation approach for high-density surface EMG decomposition in motor neuron disease, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 26, Issue 6, 2018, pp. 1161-1168.

[3]. D. Karabulut, F. Ortes, Y. Ziya Arslan, M. A. Adli, Comparative evaluation of EMG signal features for myoelectric controlled human arm prosthetics, *Biocybernetics and Biomedical Engineering*, Vol. 37, Issue 2, 2017, pp. 326-335.

[4]. K. Xu, W. Guo, L. Hua, X. Sheng, X. Zhu, A prosthetic arm based on EMG pattern recognition, in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO'16)*, 2016, pp. 1179-1184.

[5]. K. Kiguchi, Y. Hayashi, A study of EMG and EEG during perception-assist with an upper-limb power-assist robot, in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'12)*, 2012, pp. 2711-2716.

[6]. C. Sapsanis, G. Georgoulas, A. Tzes, D. Lymberopoulos, Improving EMG based classification of basic hand movements using EMD, in *Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC'13)*, 2013, pp. 5754-5757.

[7]. D. Karabulut, F. Ortes, Y. Z. Arslan, M. A. Adli, Comparative evaluation of EMG signal features for myoelectric controlled human arm prosthetics, *Biocybernetics and Biomed. Eng.*, Vol. 37, 2017, pp. 326-335.

[8]. X. Zhai, B. Jelfs, R. H. M. Chan, C. Tin, Short latency hand movement classification based on surface EMG spectrogram with PCA, in *Proceedings of the 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC'16)*, 2016, pp. 327-330.

[9]. X. Zhai, B. Jelfs, R. H. M. Chan, C. Tin, Self-recalibrating surface EMG pattern recognition for neuroprosthesis control based on convolutional neural network, *Frontiers in Neuroscience*, Vol. 11, Issue 379, 2017, pp. 1-11.

[10]. A. Boyali, N. Hashimoto, Spectral collaborative representation- based classification for hand gestures recognition on electromyography signals, *Biome. Signal Processing and Control*, Vol. 24, 2016, pp. 11- 18.

[11]. N. M. Kakoty, A. Saikia, S. M. Hazarika, Exploring a family of wavelet transforms for EMG based grasp recognition, *Signal Image Video Process*, Vol. 9, Issue 3, 2015, pp. 553-559.

[12]. L. Ge, L. j. Ge, J. Hu, Feature extraction and classification of hand movements surface

electromyogram signals based on multi-method integration, *Neural Process. Letter*, Vol. 49, 2019, pp. 1179-1188.

[13]. A. Phinyomark, E. Scheme, EMG pattern recognition in the era of big data and deep learning, *Big Data and Cognitive Computing*, Vol. 2, Issue 3, 21.

[14]. A. Gharehbaghi, M. Linden, A deep machine learning method for classifying cyclic time series of biological signals using time- growing neural network, *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 29, Issue 9, Sept. 2018, pp. 4102-4115.

[15]. A. B. M. S. U. Doulah, S. A. Fattah, W. P. Zhu, M. O. Ahmad, Wavelet domain feature extraction scheme based on dominant motor unit action potential of EMG signal for neuromuscular disease classification, *IEEE Transactions on Biomedical Circuits and Systems*, Vol. 8, Issue 2, 2014, pp. 155-164.

[16]. S. A. Fattah, K. Fatima, C. Shahnaz, An approach for classifying alcoholic and non-alcoholic persons based on time domain features extracted from EEG signals, in *Proceedings of the IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE'15)*, 2015.

[17]. A. Bhattacharjee, S. Saha, S. A. Fattah, W.-P. Zhu, M. O. Ahmad, Sleep apnea detection based on Rician modeling of feature variation in multiband EEG signal, *IEEE Journal of Biomedical and Health Informatics*, Vol. 23, Issue 3, May 2019, pp. 1066-1074.

[18]. S. B. Akben, Low-cost and easy-to-use grasp classification, using a simple 2-channel surface electromyography (sEMG), *Biomedical Research*, Vol. 28, Issue 2, 2017, pp. 577-582.

[19]. M. N. Hossain Khan, S. A. Fattah, Surface EMG based basic hand movement detection using singular value decomposition, in *Proceedings of the IEEE International Conference on Biomedical Engineering, Computer and Information Technology for Health (BECITHCON'19)*, 2019, pp. 97-102.

(042)

Classification of Time Series as Images using Deep Convolutional Neural Networks: Application to Glitches in Gravitational Wave Data

Shuzhu Jin [1], **Soumya D. Mohanty** [2], **Qunying Xie** [1, 3], **Hanzhi Wang** [3] **and Xue-Hao Zhang** [3]
[1] School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China
[2] The University of Texas Rio Grande Valley, One West University Blvd, Brownsville, TX 78526, USA
[3] Institute of Theoretical Physics & Research Center of Gravitation, Lanzhou University,
Lanzhou 730000, China
Tel.: + 19568826680
E-mail: soumya.mohanty@utrgv.edu

Summary: The classification of frequently occurring terrestrial-origin transient signals, called *glitches*, in the time series data from gravitational wave detectors is important for mitigating their adverse effects on searches for rare and valuable astrophysical signals. While formally a time series classification (TSC) problem, recent successes in glitch classification have mostly been achieved through image classification methods using their time-frequency images. Using transfer learning with the VGG16 deep convolutional neural network for image classification, we compare the efficacy of several types of image representations for classifying simulated glitches. We find the novel result that training the network with 2D plots of the noisy glitch time series provides better classification accuracy than their time-frequency images. For both the 2D time series plots and the time-frequency images, we find that using a training set that resembles the test set in its distribution of the signal to noise ratios of the glitches results in better performance.

Keywords: Time series classification, CNN, Gravitational waves.

1. Introduction

With the detection of the first signal from two merging black holes in 2015 [1] by the LIGO [2] detectors, the birth of gravitational wave (GW) astronomy has opened a new window for observing the Universe. LIGO has been joined by Virgo [3] and KAGRA [4], creating a worldwide network of GW detectors that is collectively more sensitive than the individual detectors. So far, about 90 astrophysical GW signals have been detected from the mergers of binary systems consisting of compact objects such as neutron stars and black holes.

Data analysis plays a crucial role in GW astronomy because signals of astrophysical origins are weak compared to the noise in the time series output of current GW detectors and appear rarely (about once per week). An additional challenge here is the high rate of diverse types of transient terrestrial-origin signals, called *glitches*, that can mimic GW signals and, hence, elevate the false alarm rate of GW searches. During the first observation run (O1), the average rate of glitches in LIGO data was about 10 per minute [5], while in observing run O3b, it reached up to 1 per minute [6]. (These rates correspond to a certain minimum strength of the glitches.) The mitigation of glitches, therefore, is of foremost importance in the constant quest to improve GW search sensitivity.

GW detectors undergo a planned sequence of upgrades that incorporate innovative technologies emerging from ongoing research and development efforts. In addition, they are extraordinarily complex and large-scale instruments that respond to the terrestrial environment and operating conditions within an observing run in ways that are not always well-understood. Due to these reasons, the population properties of glitches are dynamic, with changes in the rates of the prevalent types of glitches (including disappearance in some cases) and the emergence of new ones.

The classification of glitches plays a critical role in helping experimentalists identify their origins and eliminate them, although this is not always successful. For the glitches that remain in the data, classification guides the post-processing steps that must be carried out to mitigate their statistical effects. Therefore, the development of robust and automated classification algorithms for glitches is a key ingredient in the success of GW astronomy.

Many AI and machine learning methods have been investigated for the glitch classification task. Although formally a time series classification (TSC) problem [7, 8], the most successful methods to date for automated glitch classification (e.g., [9-13]) have used convolutional neural networks (CNNs) [14] trained on their constant Q-transform [15] (CQT) time-frequency images. To aid the training needed for such methods, the Gravity Spy [5] project has released a dataset of ≈ 8000 glitches that have been divided into about 20 manually labeled classes based on their CQT images.

A shortcoming of using CQT images is that they suffer from the uncertainty relation governing time-frequency resolution and, hence, can mask the details of the glitch waveforms. This is especially the case for short duration and broadband glitches for which the CQT images can be quite similar. In addition, CQT images discard the phase information in the time series. This means that, for example, two glitches are indistinguishable if their waveforms are negatives of each other.

In this paper, we propose a novel approach to glitch classification that uses the 2D plots of noisy glitch time series as the glitch representation instead of their CQT images. The motivation to use 2D plots arises from two reasons. First, using the time series directly overcomes the distinguishability problem mentioned above and can also potentially use more details in the glitch waveform that are masked in a CQT. Secondly, prior work has shown that TSC methods that directly use the time series as the input do not perform as well as image classifiers in the context of glitches. We find the remarkable result that not only does the idea of using 2D plots of time series work for image classification but that a CNN trained on 2D plots offers better classification performance on short and broadband glitches than CQT images. Our findings may have wider implications for the TSC problem in general.

The rest of the paper is organized as follows. In Section 2, we present the steps followed in setting up the classification method and the simulated data sets. Section 3 describes the setup of the different experiments we have carried out. Section 4 presents our main results derived from these experiments. Section 5 discusses an interesting situation where using 2D plots and CQT images for training leads to significantly different results. Our conclusions are presented in Section 6.

2. Methodology

We have employed VGG16 [16] for image classification in our study. VGG16 is a powerful deep convolutional neural network with 16 layers of convolutional and fully connected layers that can learn complex and abstract features from input images, giving it strong representational power. This model has been pre-trained on the large-scale ImageNet dataset, allowing it to learn a wide range of features that can be transferred to other computer vision tasks through transfer learning. Moreover, the use of a straightforward design, employing a stack of small 3×3 convolutional filters, makes VGG16 easy to implement and train faster than more complex architectures.

To adapt the network to our classification tasks, we removed the fully connected layers at the top of the network and created a new fully connected head model to train on our simulated dataset. To avoid overfitting and ensure effective transfer learning, we froze the weights of the base model layers up to the 5th convolutional block. This allowed only the custom layers to be trained during the training process, ensuring that the network learned task-specific features while retaining the general features learned from ImageNet.

During training, we employed the adaptive moment estimation (Adam) optimizer with the categorical cross-entropy loss function to optimize the model's parameters since, in our prior experience, it had a faster convergence rate than stochastic gradient descent. We trained the model over 10 epochs, using the training data to update the model's weights and biases in each epoch, and the validation set to evaluate the model's performance. Through this training process, we could achieve a balance between model accuracy and generalization ability, ensuring that the model would perform well on new, unseen data.

As mentioned earlier, the Gravity Spy database of labeled glitches uses CQT images. Since our goal is to compare different glitch representations, using class labels based on the CQT representation alone is not appropriate. We decided, therefore, to simulate glitches that resemble the ones in the Gravity Spy database but with a well-defined class structure that is independent of any particular representation. To create a simulated glitch $s(t)$, we use a superposition of Gaussian functions

$$s(t) = \sum_{i=1}^{M} A_i e^{-(t-\tau_i)^2/\sigma_i^2},$$

where $\{(A_i, \tau_i, \sigma_i)\}$ and M are the parameters of a glitch. We then pick one real glitch $s^{(C)}(t)$ from a given Gravity Spy class C and adjust the parameters until we get a good visual fit between $s(t)$ and $s^{(C)}(t)$. This creates a simulated glitch that we denote as $s_0^{(C)}(t)$. Next, we draw the parameters (except M) randomly from narrow ranges that are centered on their respective values in $s_0^{(C)}(t)$. Each draw of the parameter values leads to a corresponding simulated glitch $s_j^{(C)}(t)$, $j > 0$. The simulated glitch $s_j^{(C)}(t)$ is retained only if its squared Euclidean distance

$$\sum_{k=0}^{N-1} \left(s_j^{(C)}(t_k) - s_0^{(C)}(t_k)\right)^2,$$

from $s_0^{(C)}(t)$, where $\{t_0, t_1, \dots, t_{N-1}\}$ are the sample times of the glitch time series, is smaller than a preset threshold. Repeated drawings are made in the above manner until a preset number of simulated glitches are acquired. By construction, such a set of simulated glitches forms a well-defined cluster or class under the Euclidean metric while also resembling the real glitch $s^{(C)}(t)$ used as the seed.

Following the procedure above, we simulated an ensemble of glitch time series, each 2.0 sec long and sampled at 4096 Hz with the glitch signal $s_j^{(C)}(t)$ occupying the central T sec, where $T \leq 0.04$ sec. To generate different classes of glitches, we took representative glitches, $s^{(C)}(t)$, from three of the most frequently occurring Gravity Spy classes called *Blip*, *Koi Fish*, and *Tomte*. Similarly, we generated 6 additional clusters with arbitrarily chosen centroids (also modeled as superpositions of Gaussians).

Any classification method applied to GW data must deal with the constant presence of noise. Therefore, to make our tests realistic, we simulated noisy glitch time series by adding pseudo-random white Gaussian noise (WGN) with unit variance to the entire glitch time series. The overall strength of the glitch relative to the noise is quantified by its signal to noise ratio (SNR) defined as the Euclidean norm

$$\left[\sum_{k=0}^{N-1}\left(s_j^{(C)}(t_k)\right)^2\right]^{\frac{1}{2}}$$

By varying the SNR, we created glitches with various levels of strength, allowing us to test our algorithm's ability to classify glitches in the presence of noise.

3. Experiments

We considered the following types of glitch time series for the training set. (1) **Noisy**: glitch added to WGN and SNR drawn from a uniform distribution over $[10, 100]$. (2) **Noise-less**: glitch SNR fixed at 60 and no WGN added. (3) **Denoised**: time series obtained by passing an SNR = 60 glitch added to WGN through an adaptive spline fitting method called SHAPES [17].

For the test set, we used noisy glitch time series having a uniform distribution in SNR over (1) a **broad** ([10, 100]) and (2) a **narrow** ([5, 20]) range. The test dataset presented to VGG16 always has the same image representation as the corresponding training data.

For each type of time series, we compared two image representations for training VGG16: (1) CQT computed with the full 2.0 sec, and (2) the 2D plot of a 0.2 sec section of the time series centered on the glitch. We also used a multi-view approach (2D+CQT) in which the two types of images are paired with their common class label in training. The Y-axis range of the 2D plots were set automatically by the graphing software in all cases. The training and testing datasets had 600 and 450 images, respectively, of each type of time series for each of the 9 classes.

Fig. 1, left panel, illustrates the noisy and denoised time series. The noise-less time series is an idealization that serves as a benchmark. For real data, the denoised time series is the closest that one can get to the noise-less case. The right panel shows the CQT image of the noisy glitch.

4. Results

Tables 1 and 2 summarize our results. Each numerical entry corresponds to a given combination, labeled by the column and row headings, of the type of time series and image representation used for the training data. The numerical value is the accuracy of classification given by the fraction of all correctly classified test images.

We see that training with 2D plots of the noisy glitch time series outperforms all other combinations although CQT is not significantly behind. For the case of the broad SNR range test, an accuracy of 0.9552 is achieved when using 2D plots compared to 0.9162 with CQT images. For the narrow SNR range test, in which most of the glitches are weak to moderate in strength, the accuracy drops significantly for both types of training data but 2D plots still outperforms CQT with accuracies of 0.7324 and 0.7065, respectively. In all cases, training with 2D+CQT does not confer an advantage over 2D plots alone. It always seems to perform in between 2D plots and CQT.

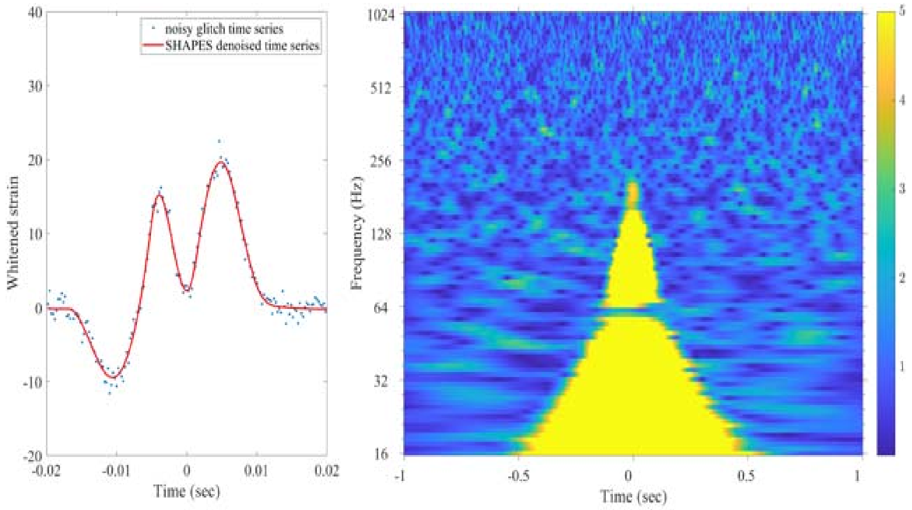


Fig. 1. (Left) Noisy (dots) and denoised (red) glitch time series example. (Right) CQT of the noisy time series.

Surprisingly, training with denoised or noise-less glitches, both having fixed SNR, is significantly less effective than the variable SNR noisy glitches. However, it is interesting to note that training with the denoised time series is comparable in performance to the ideal but unrealizable case of noise-less glitches. This suggests that denoising does not lose any information about a glitch that is critical to its classification. As expected, all methods perform worse for the narrow SNR range test set where the glitches are weaker on average.

Table 1. Classification accuracies: broad SNR range for the test set.

$\mathrm{SNR} \in [10,100]$		**Type of training time series**		
		Noise-less	**Noisy**	**Denoised**
Image type	**2D plot**	0.7573	0.9552	0.7287
	CQT	0.8382	0.9162	0.8225
	2D+CQT	0.7725	0.9315	0.7531

Table 2. Classification accuracies: narrow SNR range for the test set.

$\mathrm{SNR} \in [5,20]$		**Type of training time series**		
		Noise-less	**Noisy**	**Denoised**
Image type	**2D plot**	0.5843	0.7324	0.5329
	CQT	0.6007	0.7065	0.5934
	2D+CQT	0.5820	0.7216	0.5411

To further investigate the influence of different SNRs, we carried out additional experiments in which we both trained and tested the network using fixed SNRs of 20, 40, and 60. The results are shown in Table 3, where each sub-table corresponds to one value of SNR. We see from SNR = 20 case, that classification using 2D plots performs significantly better than CQT. The difference in performance diminishes, but never disappears, as one goes to higher SNR values. (In these experiments, we do not retain the noise-less glitches case.)

Table 3. Classification accuracies: fixed SNR range.

$\mathrm{SNR} = 20$		**Type of time series**	
		Noisy	**Denoised**
Image type	**2D plot**	0.9058	0.6339
	CQT	0.8380	0.7171
	2D+CQT	0.8587	0.6642
$\mathrm{SNR} = 40$		**Type of time series**	
		Noisy	**Denoised**
Image type	**2D plot**	0.9831	0.7543
	CQT	0.9505	0.8376
	2D+CQT	0.9593	0.7928
$\mathrm{SNR} = 60$		**Type of time series**	
		Noisy	**Denoised**
Image type	**2D plot**	0.9879	0.7902
	CQT	0.9568	0.8544
	2D+CQT	0.9623	0.8215

5. The Case of Flipped Glitches

While our results show that training VGG16 with 2D time series plots and CQT images lead to comparable results in most cases, there is an interesting situation where the two could have starkly different performances. This is the case where one takes a glitch $s_j^{(C)}(t)$ belonging to some class C and simply flips its sign: $s_j^{(C)}(t) \rightarrow -s_j^{(C)}(t)$. According to the Euclidean metric, these two glitches are clearly far apart, yet the CQT will be identical for both and VGG16 trained on CQT images will put them in the same class. However, when trained on the 2D plots (and using a different class label for the flipped glitches) VGG16 will be able to distinguish them clearly.

The above expectation is confirmed by our experiments. We considered the same case as that in column 2 of table 1 but included flipped glitches in our training and testing set without labeling them differently. We found that the classification accuracy with the 2D plot training set fell to 0.7183 while the accuracy for training with CQT images remained high at 0.8914. However, if the flipped glitches were labeled as a separate class in both training and testing, the accuracy for 2D plot training went up to 0.8795, while that for CQT images dropped significantly to 0.5167. (Some drop in the classification accuracy for both is expected because of the increase in the number of classes from 9 to 18.)

Although the above situation is a hypothetical one, it prompted us to check if situations like this can occur for real glitches. Consider Fig. 2, which shows the time series of the first 50 glitches belonging to the Tomte class taken from the dataset distributed with the `glitschen` package [18] on GitHub (in the file named *L1O3a_Tomtes_10-128.npy*). Indeed, this class that was constructed using CQT images does have a few glitch time series that appear quite distinct from the bulk but that have been labeled as belonging to the same class.

To isolate the anomalous cases further, we computed the pair-wise Euclidean distances for all ≈ 2000 glitches in the dataset above and found the pair with the largest and smallest distances. The largest Euclidean distance occurs between the 1st and the 4th glitch, and the shortest one is between the 20th and the 21st glitch. The 4 glitches are shown in Fig. 3 and the CQT images of the 1st and 4th are shown in Fig. 4.

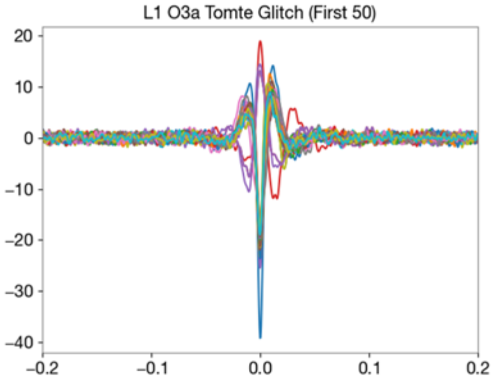


Fig. 2. Time series plots of 50 Tomte glitches from the glitschen dataset.

It is clear from Fig. 3 that, apart from scaling of the amplitude, the 1st and 4th glitches are flipped versions of each other while the 20th and 21st are nearly the same, agree well with the 1st glitch in terms of the placement and relative signs of the peaks, and also agree well with the bulk of the glitches shown in Fig. 2. This proves that CQT will classify a quite different looking time series in the same class even if it is well outside it in terms of Euclidean distance. This is also evident in the CQT plots in Fig. 4 where the two images look very similar.

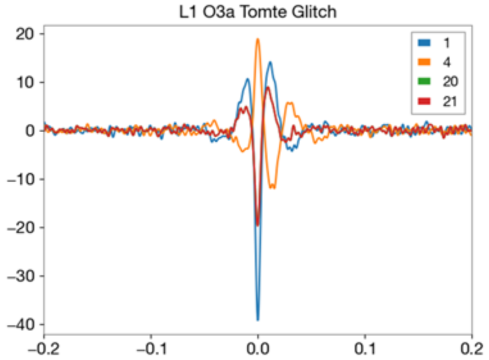


Fig. 3. Glitch numbers {1,4,20,21} from the glitschen database.

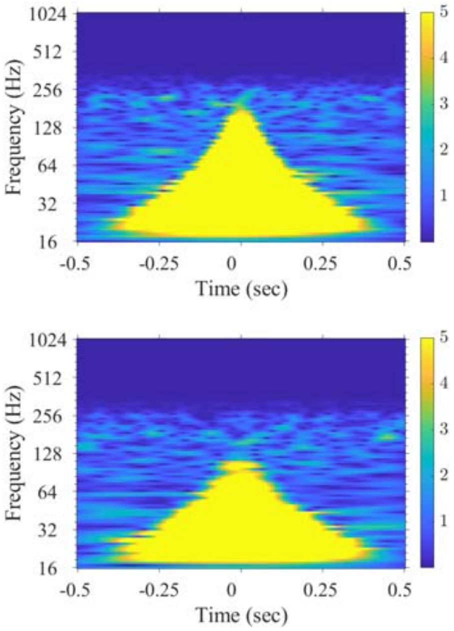


Fig. 4. CQT of the 1st (top) and 4th (bottom) Tomte glitches from the glitschen database.

6. Conclusions

We conclude that a 2D plot of time series as the image representation for training deep convolutional networks offers a promising approach to glitch classification. It is, therefore, proved that the VGG16 network can not only recognize the shape of the glitch, such as how many peaks, the distance between each peak, and the trend of the peaks, but can also recognize the phase information within it. As shown in an experiment with flipped glitches, the latter information is lost in generating CQT images from time series.

As we have discovered, the situation of flipped glitches is not a hypothetical one and it appears in the real data also although we have not yet quantified its frequency of occurrence. This suggests that the use of 2D plots as a glitch representation could have real advantages over CQT images for the classification of glitches and, perhaps, for the general TSC problem also.

Acknowledgements

Jin, Xie, Wang, and Zhang are supported by the National Key Research and Development Program of China grant: 2021YFC2203003. S.D.M is supported by U.S. National Science Foundation (NSF) grant PHY-2207935.

References

[1]. B. P. Abbott, et al., Observation of gravitational waves from a binary black hole merger, *Physical Review Letters*, Vol. 116, 2016, 061102.

[2]. J. Aasi, B. P. Abbott, R. Abbott, et al., Advanced LIGO, *Classical and Quantum Gravity*, Vol. 32, Issue 7, 2015, 074001.

[3]. F. Acernese, M. Agathos, K. Agatsuma, et al., Advanced Virgo: A second-generation interferometric gravitational wave detector, *Classical and Quantum Gravity*, Vol. 32, Issue 2, 2014, 024001.

[4]. Y. Aso, Y. Michimura, K. Somiya, et al., Interferometer design of the KAGRA gravitational wave detector, *Physical Review D*, Vol. 88, Issue 4, 2013, 043007.

[5]. M. Zevin, S. Coughlin, S. Bahaadini, et al., Gravity Spy: integrating advanced LIGO detector characterization, machine learning, and citizen science, *Classical and Quantum Gravity*, Vol. 34, Issue 6, 2017, 064003.

[6]. R. Abbott, T. D. Abbott, F. Acernese, et al., GWTC-3: compact binary coalescences observed by LIGO and Virgo during the second part of the third observing run, *arXiv Preprint*, 2021, arXiv:2111.03606.

[7]. H. Ismail Fawaz, G. Forestier, J. Weber, et al,. Deep learning for time series classification: a review, *Data Mining and Knowledge Discovery*, Vol. 33, Issue 4, 2019, pp. 917-963.

[8]. S. Mukherjee, R. Obaid, B. Matkarimov, Classification of glitch waveforms in gravitational wave detector characterization, *Journal of Physics: Conference Series*, Vol. 243, Issue 1, 2010, 012006.

[9]. R. Biswas, L. Blackburn, J. Cao, et al., Application of machine learning algorithms to the study of noise artifacts in gravitational-wave data, *Physical Review D*, Vol. 88, Issue 6, 2013, 062003.

[10]. J. Powell, D. Trifiro, E. Cuoco, et al., Classification methods for noise transients in advanced gravitational-wave detectors, *Classical and Quantum Gravity*, Vol. 32, Issue 21, 2015, 215012.

[11]. N. Mukund, S. Abraham, S. Kandhasamy, et al., Transient classification in LIGO data using difference boosting neural network, *Physical Review D*, Vol. 95, Issue 10, 2017, 104059.

[12]. V. Boudart, Convolutional neural network to distinguish glitches from minute-long gravitational

wave transients, *Physical Review D*, Vol. 107, Issue 2, 2023, 024007.

[13]. T. S. Fernandes, S. J. Vieira, A. Onofre, et al., Convolutional Neural Networks for the classification of glitches in gravitational-wave data streams, *arXiv Preprint*, 2023, arXiv:2303.13917.

[14]. K. O'Shea, R. Nash, An introduction to convolutional neural networks, *arXiv Preprint*, 2015, arXiv:1511.08458.

[15]. J. C. Brown, Calculation of a constant Q spectral transform, *The Journal of the Acoustical Society of America*, Vol. 89, Issue 1, 1991, pp. 425-434.

[16]. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, *arXiv Preprint*, 2014, arXiv:1409.1556.

[17]. S. D. Mohanty, E. Fahnestock, Adaptive spline fitting with particle swarm optimization, *Computational Statistics*, Vol. 36, Issue 1, 2021, pp. 155-191.

[18]. J. D. Merritt, B. Farr, R. Hur, et al., Transient glitch mitigation in Advanced LIGO data, *Physical Review D*, Vol. 104, Issue 10, 2021, 102004.

(044)

Data Augmentation of Tumour Histopathological Images using Generative Adversarial Networks

Jose Luis Ruiz-Casado [1], **Miguel A. Molina-Cabello** [1,2], **Enrique Domínguez** [1,2]
and Rafael M. Luque-Baena [1,2]
[1] Department of Computer Languages and Computer Science. University of Malaga, Malaga, Spain
[2] Instituto de Investigación Biomédica de Málaga y Plataforma en Nanomedicina-IBIMA
Plataforma BIONAND, Malaga, Spain
E-mails: joseruizcasado@uma.es, {miguelangel, enriqued, rmluque}@lcc.uma.es

---

Summary: Breast cancer is currently the second most common cancer in the world, affecting mainly women. Histopathological image analysis is used to determine the degree of malignancy of the tumour. The rise of deep learning in recent years has led to the application of this technique to the analysis of these images. The main problem is that the available datasets are often unbalanced, i.e. there are classes with more images available than others. As a result, models trained on these sets have poorer generalisability. One solution to this problem would be to downsample the class with the most images in order to have a set with the same number of images for each class. This solution is not recommended when the set size is small, as it can lead to poor model performance. Traditionally, techniques known as data augmentation are used. These techniques apply a series of simple transformations to the images, such as translation and rotation, to achieve greater variability in the dataset. However, a class of architectures called Generative Adversary Networks (GAN) has emerged that are capable of generating images from a relatively limited training set.

Keywords: Breast cancer, Histopathological images, Deep learning, Generative adversarial networks, Data augmentation.

---

1. Introduction

Early detection of breast cancer can significantly reduce mortality, and non-invasive techniques such as mammography and ultrasound are often used for this purpose. However, to accurately determine the degree of malignancy of a tumour, histopathological image analysis must be used [1]. Histopathological images (see Fig. 1) are obtained by digitising a tissue sample taken by biopsy, an invasive technique. Manual analysis of such images is tedious and time consuming. In these situations, a Computer Aided Diagnosis CAD system helps in the decision-making process [2]. With the current development in technology and informatics, applications and tools have been developed to process and analyse histopathological images. These tools have also evolved and today deep learning models are used to assist in image analysis. For a deep learning neural network to work successfully, it needs a sufficiently large dataset so that the network itself can identify the key features of each class represented in the dataset. This can be a problem because the tests used to obtain such images are expensive and can be harmful to patients, so it is difficult to find a sufficiently large image bank. There is also an imbalance in these datasets in favour of the negative classes.

2. Related Work

Since Generative Adversarial Networks were proposed in [3], this architecture has been applied in a variety of fields. GANs are based on the use of two different models, the generator and the discriminator, which compete in a kind of zero-sum game. The training of such architectures is based on the competition between the discriminator and the generator, so that the goals for each model are opposite. The discriminator receives an image as input and must indicate whether the image belongs to the example dataset or has been generated by the generator. The generator has to synthesise an image that is as close as possible to the images in the example set from a noise vector given as input.

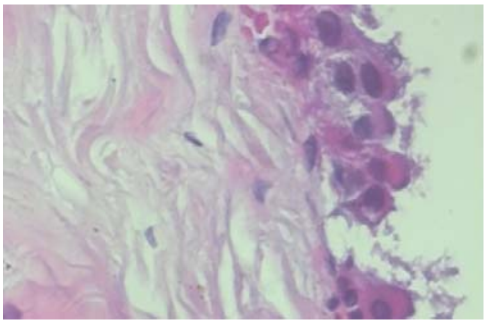


Fig. 1. Example of histopathological image.

To solve the problems of stability and variability during the training of a GAN, Tero Karras proposed an evolution in the architecture of this type of network [4]. The basic idea he proposed is to progressively increase the quality of the generated images. To avoid the generator getting stuck on examples that are too similar, they propose to calculate the standard deviation of all pixels in the generated mini-batch, which allows the discriminator to identify a generated image by the pixel-level standard deviation in the given batch. To avoid divergence by feature magnitude, ProGAN uses pixel-level normalisation, which is less memory intensive than batch-level normalisation. Finally, they use an equalised learning rate that allows the weights to perform update steps at different scales if necessary.

Generative networks require the use of specific loss functions, which allow the special performance of these networks to be calculated. The Wasserstein loss function [5] uses the Wasserstein distance to relate the quality of the images produced to improve stability during training. For this loss function to work correctly, the Lipschitz constraint must be satisfied. When a gradient penalty is used to achieve compliance with this restriction, it is known as a Wassertein loss function with gradient penalty [6].

In order to properly evaluate a generative model, it is essential to be able to compare the generated images with the set of example images. This comparison must be quantifiable in order to be able to compare it with other generative models and to know which one offers better performance. The standard metric for evaluating generative models is the Frechet Inception Distance (FID) [7]. This metric calculates the distance between the feature vectors of the generated images and the example images. It summarises how similar the two groups are in terms of computer vision statistics.

3. Proposal

The image set used was obtained from Laboratório Visão Robótica e Imagem UFPR, consisting of 588 benign tumour images and 1233 tumour images [8]. Starting from this dataset, several GAN architectures are trained to increase and balance the number of images available in both classes.

The GAN architecture consists of two models with conflicting training objectives, so it is necessary to choose a loss function that satisfies this need. The loss function is:

$$\begin{aligned} \mathrm{L} = {} & E_{\tilde{x} \sim P_g}[D(\tilde{x})] - E_{\tilde{x} \sim P_r}[D(x)] + \\ & + \lambda E_{\tilde{x} \sim P_{\hat{x}}}[(||\nabla_x D(\hat{x})||_2 - 1)^2], \end{aligned} \tag{1}$$

where $\tilde{x}$ is the image generated by the generator, x is the real image, P_r is the data distribution, P_g is the distribution of the model or the distribution of the generated images, $P_{\hat{x}}$ is the distribution obtained by sampling uniformly along a straight line defined between pairs of points from the distributions.

The architecture to evaluate the impact of image enhancement consists of four convolutional blocks. In each block, a convolution with a kernel size of 5×5 is applied, followed by a batch normalisation and the use of *Leaky RELU* as the activation function. After these convolution blocks, a flattening layer is applied, followed by a dropout layer using *Softmax* as the activation function. The cross-entropy loss function is used as the loss function for this model. This function is very suitable for classification problems where you have output layers that produce probabilities, such as the *Softmax* activation function. The following equation is used to calculate it.

$$\mathrm{L} = -(y \log p + (1 - y) \log(1 - y)), \tag{2}$$

where y is a binary indicator of whether the label c is the correct label for the observation o, p is the probability that the observation o belongs to the class c.

For the training of both the generative and convolutional networks, Adam is used as the optimiser. This choice is due to the fact that this optimiser has shown better results than others and has a shorter execution time. It should also be noted that it uses fewer hyperparameters.

4. Experiments

In order to be able to quantify the quality of the results produced by generative architectures, it is necessary to define appropriate performance metrics. In this case, the FID is used for such a comparison. The lower this metric is, the more similar the generated results are to the images of the dataset.

For the convolutional model, the accuracy (*acc*) of the model on the test set, i.e. the percentage of well classified examples (see Equation (3)), is used first. It is not appropriate to assess model capability using a single metric, let alone an unbalanced set. A confusion matrix is used to improve understanding of model performance. This matrix shows true positives (*TP*), true negatives (*TN*), false positives (*FP*) and false negatives (*FN*). These measures allow us to see whether the model is able to extract representative features for each class, or whether it is unable to generalise correctly. The true classes of the classified examples are given in each row of the confusion matrix, while the predicted classes are given in the columns. The numbers in each cell indicate the percentage of examples in the row class that were predicted as the column class.

$$\mathrm{Acc} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

Starting from the original dataset, several GAN architectures are trained. The aim is to increase and balance the number of images available in both classes. The different architectures are evaluated using the FID metric and visual evaluation of the results obtained, with the aim of identifying the most suitable architecture.

To test the impact of using these generative architectures, a convolutional network will be used. This network will be trained on the original set and then on the set generated by GAN. The results obtained will be compared using metrics such as accuracy and confusion matrix.

To obtain good set balancing, it was decided to train two generative models, one for each class, using ProGAN architecture. The same hyperparameter settings are used to train both models. A depth value (d) has to be chosen so that the output size is calculated as $2^{(d+1)}$ to determine the size of the generated images. Considering the size of the images, it was decided to use a depth value of 7, so that the generated images are 256×256 pixels in size. This output size allows the generated images not to be distorted and helps to use a larger batch size. Each depth level is trained for 42 epochs.

An appropriate batch size must be selected for each depth level, taking into account the size of the output. It must be taken into account that the size of the selected image is equal to the size of the output at each level, so as not to produce an erroneous operation in the discriminator, so that the consistency of the model is adequate. Table 1 shows the selected value at each depth level.

Table 1. Batch size adjustment by depth level.

Depth	1	2	3	4	5	6	7
Batch Size	64	64	32	16	4	2	1
Image Size	4	8	16	32	64	128	256

If a new block is added when the depth level is changed, it is not added directly, but is faded in with the previous block. To control this block fade, a fade percentage variable is used to specify what percentage of the new block is used for the output. The value of this variable is set to 0.5.

A parameter external to the ProGAN architecture, but which speeds up training, is the number of threads used to load the training set. In this case, 4 loading threads have been defined.

Two convolution models are trained, one with the original dataset and one with the balanced dataset. The convolutional models are cross-validated using K-fold. This allows a better comparison between unbalanced and balanced data sets. This technique consists of dividing the training set into subsets (folds), so that each subset is used for both training and testing. The value chosen for K is 5.

The number of training epochs chosen for this model is 15. This choice is based on empirical experience after using a larger number of epochs, as beyond this number of epochs the model begins to show overfitting. Overfitting is the situation where the model is unable to generalise the characteristics of the training set, resulting in performance metrics obtained on the test set being much lower than those obtained during training. A batch size of 16 was chosen due to the limitations of the GPU and the size of the images during the training phase.

5. Results

Fig. 2 shows on the left the images generated by ProGAN at the end of training for the class *malignant*. On the right of this figure, we can see 4 real images of the same class. At first sight we can see that there is a high similarity. This is confirmed after calculating the FID, which gives an FID of 135. The value is not too high, indicating that the generator was able to identify the main features of the training set.

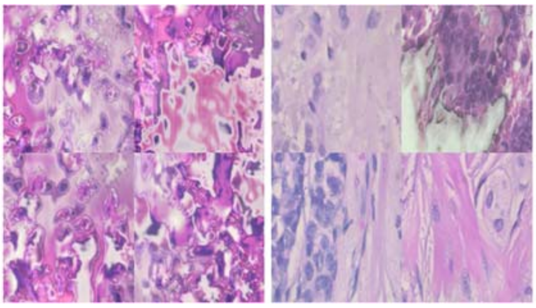


Fig. 2. Comparison of generated and real images of the *malignant* class.

The results obtained for the *benign* class can be seen in the Fig. 3. The results obtained for this class are clearly worse visually. The FID value obtained for this class is 286, which is significantly higher. This may be largely due to the small number of images available for this class. Although the generative model for the *malignant* class is clearly worse, the FID value indicates that there is some similarity between the generated images and the original ones.

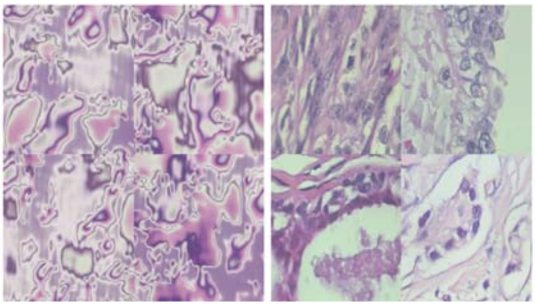


Fig. 3. Comparison of generated and real images of the *benign* class.

When training the convolutional model on the original dataset, it can be seen that from the first epoch the accuracy of the model is around 0.8. During the training phase, little variation in the results was observed, which could indicate that the model is not learning anything from the training data. The loss values obtained also show an irregular behaviour. When comparing these metrics in the different folds, too much variation is observed, indicating that there is a high dependence on the images used for training.

The confusion matrix of this model (see Table 2) shows that most of the predictions belong to the malignant class. This effect is caused by the imbalance in the training set and is evidence of the problem that the model suffers during training. What happens is that the model learns the distribution of the data rather than the representative features of each class. This means that when the model is confronted with many more images of the malignant class, it is hardly penalised for always giving that class as a prediction.

Table 2. Confusion matrix of the convolutional model that was trained on the original dataset.

Real\Predict	Benign	Malignant
Benign	0.19	0.81
Malignant	0.053	0.95

When using the balanced data set with GAN, an initial improvement in the accuracy of the convolutional model is observed. The evolution of this metric over the different epochs shows a stability of the learning of the model. Although there are some folds where the training is not stable, the overall performance is better than that observed with the original dataset.

In the confusion matrix of the model trained on the balanced set (see Table 3), we can see how the model manages to make mostly correct predictions. The model even manages to correctly identify the representative features of the *benign* class, whereas the generative model of this class gave worse metrics.

Table 3. Confusion matrix of the convolutional model that was trained on the balanced dataset.

Real\Predict	Benign	Malignant
Benign	0.9	0.1
Malignant	0.17	0.83

To complete the above results, we add the Table 4 with the results obtained by the two convolutional models on the test set. For the test set, the same is true: the model trained on the balanced set performs better (higher average accuracy) and is more robust (lower standard deviation of accuracy).

Table 4. Results obtained in the test set**.**

Model	Average acc	Acc std	Average loss	Loss std
Original set	73.65	8.89	1.76	0.07
Balanced set	89.01	6.51	1.60	0.06

6. Conclusions

The research line focuses on the use of generative networks (GANs) to improve histopathological image-based breast cancer diagnosis systems. To achieve this goal, a generative model has been developed and evaluated to understand its performance. Once the generative model has been selected, its impact on balancing the set of available images has been studied by looking at the improvement it produces when training a convolutional model.

The results obtained for the ProGAN generative networks are satisfactory, with an FID of 286 for the *benign* class and an FID of 135 for the *malignant* class. By matching the original image set with these generative models, the convolutional model goes from an average accuracy of 73.65 on the test set to an average accuracy of 89.01. These metrics are already a clear improvement, but looking at the confusion matrices shows the improvement: the convolutional model trained on the initial set only predicts as malignant, whereas using the balanced set produces a higher rate of true positives for both classes.

Following the publication of the ProGAN architecture, many improvements have been made around it, and architectures have also been published that follow the philosophy but approximate the solution with different implementations. The main future line of this work is the study of modern architectures that will allow the improvement of the results obtained.

Acknowledgments

This work is partially supported by the Autonomous Government of Andalusia (Spain) under project UMA20-FEDERJA-108, project name Detection, characterization and prognosis value of the non-obstructive coronary disease with deep learning. It is also partially supported by the University of Malaga under grant. It includes funds from the European Regional Development Fund (ERDF). It is also partially supported by the University of Málaga (Spain) under grants B1-2019_01, project name Anomaly detection on roads by moving cameras; B1-2019_02, project name Self-Organizing Neural Systems for Non-Stationary Environments; B1-2021_20, project name Detection of coronary stenosis using deep learning applied to coronary angiography, B4-2022, project name lntelligent Clinical Decision Support System for Non-Obstructive Coronary Artery Disease in Coronarographies, and B1-2022_14, project name Detección de trayectorias anómalas de vehículos en cámaras de tráfico. The authors thankfully acknowledge the computer resources, technical expertise and assistance provided by the SCBI (Supercomputing and Bioinformatics) center of the University of Málaga. They also gratefully acknowledge the support of NVIDIA Corporation with the donation of a RTX A6000 GPU with 48 Gb. The authors also thankfully acknowledge the grant of the Universidad de Málaga and the Instituto de Investigación Biomédica de Málaga y Plataforma en Nanomedicina-IBIMA Plataforma BIONAND.

References

[1]. P. Baran, et al., High-resolution x-ray phase-contrast 3-D imaging of breast tissue specimens as a possible adjunct to histopathology, *IEEE Transactions on Medical Imaging*, Vol. 37, 2018, Issue 12, pp. 2642-2650.

[2]. C. Stoean, et al., How much and where to use manual guidance in the computational detection of contours for histopathological images?, *Soft Computing*, Vol. 23, 2019, pp. 3707-3722.

[3]. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, Y. Bengio, Generative adversarial networks, *Communications of the ACM*, Vol. 63, Issue 11, 2020, pp. 139-144.

[4]. T. Karras, T. Aila, S. Laine, J. Lehtinen, Progressive growing of GANs for improved quality, stability, and variation, *arXiv Preprint*, 2017, arXiv:1710.10196.

[5]. M. Arjovsky, S. Chintala, L. Bottou, Wasserstein generative adversarial networks, in *Proceedings of the International Conference on Machine Learning*, 2017, pp. 214-223.

[6]. I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, A. C. Courville, Improved training of Wasserstein GANs, in Advances in Neural Information Processing Systems, Vol. 30, *Curran Associates, Inc.*, 2017.

[7]. M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, GANs trained by a two time-scale update rule converge to a local Nash equilibrium, in Advances in Neural Information Processing Systems, Vol. 30, *Curran Associates, Inc.*, 2017.

[8]. F. A. Spanhol, et al., A dataset for breast cancer histopathological image classification, *IEEE Transactions on Biomedical Engineering*, 2015, Vol. 63, Issue 7, pp. 1455-1462.

(045)

Detection of the Start of the Gait Cycle by Artificial Intelligence

D. Teran-Pineda and E. Dominguez
University of Malaga, Department of Computer Languages and Computer Science,
Bulevar Louis Pasteur, 35, 29071 Málaga, Spain
E-mails: dfteran@uma.es, enriqued@uma.es

Summary: Human behavior plays an essential role in research health; thus, human information has been converted crucial to medicine, neurology, biomedical, and rehabilitation. Computer science, digital signal processing, and mathematics methods have helped improve the detection accuracy of these human activities acquired by different sensors. In this paper, a methodology with a neural network model was developed to detect the start and end of human gait. The output layer uses a sigmoid activation function; also, binary cross-entropy is applied as a loss function. A slider windows technique has been used to select part of the data to increase the accuracy of the Heel Strike event, which is the start and end of the gait cycle. Our proposed method was evaluated in an open database from the APDM OPAL system used by other researchers. Our approach attains an accuracy rate of 97.0 % in neural network classifiers using slider windows of 50 sizes.

Keywords: Human gait, Event gait, Neural network, Slide windows.

1. Introduction

One of the most common and natural activities in daily human life is walking; the ability to walk involves the concerted of the brain, jitters, and muscles and is necessary to understand the movements of humans (kinematics) and the forces(kinetics) that are applied in mortal joints [1], likewise, the tendency to assess the status, treatment, and recuperation of the cases are examined by physiotherapists, orthopedists, and neurologists [2]. In [3], human gait can be challenging to diagnose; also, the motion analysis is essential in neurology, rehabilitation, and physical therapy to detect disorders and monitor different neurological symptoms in gait human [4].

To study normal and abnormal human gait in [5] explain many parameters, which include the range and pattern of joint motion, floor reaction forces, timing of foot/floor contact, gait speeds, cadences, stride lengths, swing-to-stance phase relations, resting joint angles, isometric torques, and phasic muscular activities. Moreover, the swing-to-stance phase parameters have three important events in one gait cycle: the Mid-Swing (MS), Heel-Strike (HS), and Toe-Off (TO) events.

Gait human studies have helped to improve community health and safety and also have increased the survey in gait training systems, which integrates mechanical, sensing technology, electronic and robotic technologies; there are three crucial field research, (1) gait measurement and intention recognition, (2) gait evaluation, and (3) gait training control strategy [3]. Therefore, knowing and measuring the human gait for its clinical applications and preventing future diseases is essential.

Rehabilitation research has made WS sensors essential in quantifying gait rehabilitation evaluation. These results are often obtained using multiple sensors [6] or other research uses few sensors to improve ergonomics and noise reduction [7].

The main objective of this research is to design a model based on neural networks using inertial sensors so that the model automatically learns the detection of the start and end of walking – HS. In this work, the database of research [8] is used; in addition, to train the model, the start and end of the march were manually marked, and the slide windows method was used to enter the data into the model. The output layer uses a sigmoid activation function to obtain a binary output; also, binary cross-entropy is applied as a loss function to quantify the similarity between the output model and label events.

2. Methodology

2.1. Signal Feature

The database contains signals from two wearable IMUs sensors, and they were collected to signal to detect the kinematics at a sampling rate of 25 Hz. The sensors were implemented in their two legs, and each subject covered 20 m four times. In addition, each IMU provides three axial accelerometers and three axial angular velocities for each leg, forming a vector $N_s = 6$ for each sample and a binary label corresponding to the HS event (Y_s).

Consider a full sample collect of size N_t in N_s data snapshots; the data matrix of the range collected has $N_t x N_s$ and the label vector $N_t x L_s$. To a range r, the matrix $N_t x N_s$ is then partitioned with sliding windows of size $M_t x M_s$, where this results in $L_t L_s$ sub-matrices stacked into a Matrix T. The L_s parameter is given by the length N_s and $L_s = r + t$, t increase in each time.

$$T_j = [\{F\}]_j, \quad (1)$$

where F is the matrix transform in vector of size 1×(Nsxr) in each partition and j represents the number of partitions with sliding windows of total samples.

The vector label Y_s after the sliding windows is formed for size $1xL_s$.

2.2. NN for Heel-Strike (HS) Detection

We applied the measurement matrix T_j to build an NN model for the instant identification of HS event. However, each instance of F_j is marked with an HS event, and this data is applied to train an NN. The dataset was split, one group to training and another to validate.

The model input is $F \epsilon R^{1x(N_s x\, r)}$, the output Ys is then sent to a fully connected layer W to produce the layer output Z and then is applied the sigmoid function to predict the result $Predict = \sigma(Z)$.

3. Experiment

The performance of our model was tested in three groups of the dataset (five elderly health, five stroke patients and five Parkinson's disease). The sensor's IMUs respond on the sagittal plane. The database contains information on the left and right leg; therefore, the model is trained by mixing both data.

The epoch was selected 100 times for the experiment, and the dropout rate was determined at 0.2 to avoid overfitting.

The signal is normalized with a mean of zero and deviation because each person has different walking patterns and speeds. The experiment was developed by $r = 50$, and the result is listed in Table 1.

Table 1. Testing result accuracy.

No.	Elderly health	Stroke Patients	Parkinson's disease
Accuracy	0.97	0.93	0.89
Validation	0.95	0.91	0.88

4. Conclusions

This research proposes detecting HS to reduce force sensors that are usually cast on the soles of the feet to see gait phases and the kinematic curve. The model has a precision of 97 % for Elderly health, 93 % for stroke patients and 0.89 % for Parkinson's disease; however, the results of the confusion matrix still indicate false positives in the detection of the step, which leads to predicting HS in the wrong places.

A commercial IMU system developed the database called OPAL IMU system, which helps the model have good precision because it is not necessary to filter the signal. For this reason, we propose to use this methodology in our system to replace force sensors.

References

[1]. C. Senanayake, S. Senanayake, Human assisted tools for gait analysis and intelligent gait phase detection, in *Proceedings of the Innovative Technologies in Intelligent Systems and Industrial Applications Conference (CITISIA'09)*, 2009, pp. 230-235.

[2]. C. Prakash, R. Kumar, N. Mittal, Recent developments in human gait research: Parameters, approaches, applications, machine learning techniques, datasets and challenges, *Artificial Intelligence Review*, Vol. 49, Issue 1, 2018, pp. 1-40.

[3]. Y. Song, J. Zhang, L. Cao, M. Sangeux, On discovering the correlated relationship between static and dynamic data in clinical gait analysis, in *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2013, pp. 563-578.

[4]. O. Dostál, A. Procházka, O. Vyšata, O. Ťupa, P. Cejnar, M. Vališ, Recognition of motion patterns using accelerometers for ataxic gait assessment, *Neural Computing and Applications*, Vol. 33, Issue 7, 2021, pp. 2207-2215.

[5]. M. Donath, Human gait pattern recognition for evaluation, diagnosis and control, PhD Thesis, *Massachusetts Institute of Technology*, 1978.

[6]. C. Caramia, D. Torricelli, M. Schmid, A. Munoz-Gonzalez, J. Gonzalez-Vargas, F. Grandas, J. L. Pons, Imu-based classification of Parkinson's disease from gait: A sensitivity analysis on sensor location and feature selection, *IEEE Journal of Biomedical and Health Informatics*, Vol. 22, Issue 6, 2018, pp. 1765-1774.

[7]. L. Wang, Y. Sun, Q. Li, T. Liu, J. Yi, Two shank-mounted Imus-based gait analysis and classification for neurological disease patients, *IEEE Robotics and Automation Letters*, Vol. 5, Issue 2, 2020, pp. 1970-1976.

[8]. F.-C. Wang, Y.-C. Li, T.-Y. Kuo, S.-F. Chen, C.-H. Lin, Real-time detection of gait events by recurrent neural networks, *IEEE Access*, Vol. 9, 2021, pp. 134849-134857.

(046)

Inversion Artifacts on Time-lapse Electrical Resistivity Tomography Data

A. Hojat [1,2], **L. Zanzi** [2] **and M. H. Loke** [3]
[1] Department of Mining Engineering, Shahid Bahonar University of Kerman, Kerman, Iran
[2] Dipartimento di Ingegneria Civile e Ambientale, Politecnico di Milano, Milan, Italy
[3] Geotomo Software Sdn Bhd, Penang, Malaysia
Tel.: + 390223994271
E-mails: ahojat@uk.ac.ir, azadeh.hojat@polimi.it, luigi.zanzi@polimi.it, drmhloke@gmail.com

---

Summary: In this paper we discuss the importance of optimizing the inversion strategy for Electrical Resistivity Tomography (ERT) data acquired by ERT monitoring systems installed in sites with complex geology. In March 2022, a permanent ERT monitoring system was installed on a critical slope above an important high-speed train tunnel in Italy. In order to correctly interpret the monitoring data and define thresholds of alarm, ERT inverted images should be as clean as possible from inversion artifacts. We compared singular versus simultaneous inversions of time-lapse data. The complex geology of the study site made time-lapse inversion mandatory. Besides, forward modeling simulations of synthetic time-lapse data were necessary to define the optimal inversion parameters. The results showed the importance of optimizing the time-lapse inversion parameters, mainly the time-lapse damping factor, to produce reliable images and to correctly interpret the results.

Keywords: Tomography, Ill-conditioning, Inversion artifacts, Time-lapse inversion, Time-lapse damping factor, Long-term monitoring, Synthetic models.

---

1. Introduction

In recent years, geophysical monitoring methodologies have continued to be integrated with other technologies to develop early-warning alarm systems for hydrogeological risks such as rockfalls, landslides, and failures of earthen embankments [e.g., 1-3]. Real-time information about the internal conditions of the monitored sites are provided from geophysical monitoring systems.

Electrical Resistivity Tomography (ERT) method is one of the most flexible geophysical techniques which is widely used to determine the distribution of the electrical resistivity in the subsurface to solve a variety of engineering, hydrogeological and environmental problems [e.g., 2-5]. In recent decade, ERT method is continuously integrated with other technologies used to monitor hydrogeological risks because ERT method is very effective to detect the saturation degree of the subsurface material as well as underlining cavities and seepage zones [3]. Each data point in an ERT survey is acquired by passing a current (I) into the ground through two electrodes and measuring the potential difference at two potential electrodes (ΔV). Ohm's law combined with the conservation of current govern the relationship between the electrical resistivity, the injected current and the electrical potential [4]. The apparent resistivity (ρ_a) is obtained from Equation (1)

$$\rho_a = K\frac{\Delta V}{I}, \qquad (1)$$

where K is the geometric factor and depends on the geometric configuration of the four electrodes.

Multi-electrode systems were developed a couple of decades ago and resulted in wider and more efficient applications of two-dimensional (2D) and three-dimensional (3D) ERT surveys. The internal switching circuitry controlled by a programmable microcomputer or microprocessor within the resistivity meter automatically selects four electrodes for each measurement point [4]. In 2D ERT surveys, the purpose is to obtain a subsurface model which illustrates the resistivity changes in the vertical direction as well as in the horizontal direction along the survey line, assuming that the resistivity does not change in the direction perpendicular to the survey line. Considering that the geological structures are 3D in nature, a fully 3D resistivity survey should in theory give the most accurate results. However, 3D surveys have not still reached the level of common use like 2D surveys [6].

2. ERT Monitoring

In ERT monitoring (or time-lapse) projects, measurements along the same 2D survey line or 3D survey grid are repeated at different times. The purpose is to measure the resistivity changes in both space and time [4]. The general-purpose commercial resistivity meters available in the market are normally expensive for monitoring purposes and are not adapted for permanent installations in remote areas. Usually, a customized resistivity meter is developed to be permanently installed in a monitoring site. Such systems have the advantage of being equipped with compatible hardware and software parts to ensure remote control and data transmission to a central office. Moreover, system costs can be considerably reduced through customizing the measurement parameters for each specific application in a certain site [7]. Fig. 1 illustrates an example of the components of a permanent ERT monitoring system for permanent installations. The resistivity meter and the system electronics are usually hosted by one or more external boxes. The system is connected to multi-electrode cables that are usually requested to be deployed below the surface in a shallow trench (20-50 cm deep) along the survey line [8]. The cables are connected to the electrodes that, in order to reduce the contact resistance for permanent installations, are normally plate electrodes. The ERT monitoring station is accompanied with meteorological stations that measure parameters like rainfall, air temperature and soil temperature that are crucial for long-term analysis of ERT data. Solar panels are commonly assembled with the system for energy supply.

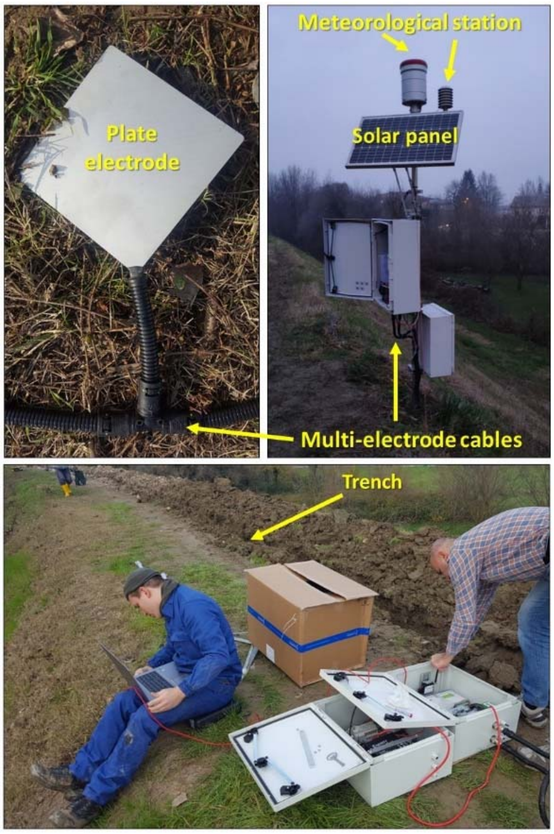


Fig. 1. Elements of an ERT monitoring system for permanent installations. Multi-electrode cables are connected to plate electrodes and are permanently buried inside a trench. The resistivity meter and other electronical elements are hosted in external boxes. The ERT station is accompanied by a meteorological station. All systems are power-supplied by a solar panel.

In this paper we discuss optimizing the time-lapse inversion strategy and parameters for a permanent 2D ERT monitoring system installed on the critical part of a slope located over a high-speed train tunnel in Italy. The ERT monitoring system applies the Wenner electrode configuration (i.e., the spacing between successive electrodes is always the same) with the unit electrode spacing of 3 *m*. The system can support two 24 multi-electrode cables that are connected to a total number of 48 stainless-steel plate electrodes.

3. Time-lapse Inversion of ERT Data

The values measured in an ERT survey are not the true resistivity but the apparent resistivity which is a complex function of the true resistivity distribution unless the subsurface is homogeneous. An inversion procedure must be applied to determine the distribution of the real resistivity in the subsurface from the measured apparent resistivity values, i.e., to find a model that gives a response similar to the actual measured data [6]. One main challenge of geophysical inverse problems is their inherent ill-posedness, i. e., different model parameters might produce almost the same experimental observations [9, 10]. An automatic inversion program uses a finite-element or a finite-difference modelling subroutine to calculate the apparent resistivity values, and using a non-linear smoothness-constrained least-squares optimization technique, the resistivity of the model blocks is calculated [11]. It should be noted that many ERT surveys are measured in areas with significant topography, and the finite-difference method is less efficient in modelling topography compared with the finite-element method [12].

The time-lapse smoothness-constrained least-squares optimization method is described by Equation (2) [13]

$$\begin{aligned}&[\boldsymbol{J}_{\boldsymbol{k}}^{\boldsymbol{T}}\boldsymbol{R}_{\boldsymbol{d}}\boldsymbol{J}_{\boldsymbol{k}}+(\lambda\boldsymbol{W}^{\boldsymbol{T}}\boldsymbol{R}_{\boldsymbol{m}}\boldsymbol{W}+\alpha\boldsymbol{M}^{\boldsymbol{T}}\boldsymbol{R}_{\boldsymbol{t}}\boldsymbol{M})]\boldsymbol{\Delta r}_{\boldsymbol{k}}=\\&=\boldsymbol{J}_{\boldsymbol{k}}^{\boldsymbol{T}}\boldsymbol{R}_{\boldsymbol{d}}\boldsymbol{g}_{\boldsymbol{k}}-(\lambda\boldsymbol{W}^{\boldsymbol{T}}\boldsymbol{R}_{\boldsymbol{m}}\boldsymbol{W}+\alpha\boldsymbol{M}^{\boldsymbol{T}}\boldsymbol{R}_{\boldsymbol{t}}\boldsymbol{M})\boldsymbol{r}_{\boldsymbol{k-1}},\end{aligned} \tag{2}$$

where ***J*** is the Jacobian matrix that contains the partial derivatives of the apparent resistivity values with respect to the model resistivity values (***r***), ***W*** and ***M*** are respectively the spatial roughness filter matrix and the temporal roughness filter, *k* is the iteration number, ***g*** is the data misfit vector, $\boldsymbol{R_d}$, $\boldsymbol{R_m}$, and $\boldsymbol{R_t}$ are weighting matrices used by the L1-norm inversion method [12, 13], *λ* is the spatial damping factor vector while *α* is the temporal damping factor that controls the relative importance given to minimizing the difference between models at different times. In this section, we demonstrate how we could overcome the problem of significant inversion artifacts by using a large value of the time-lapse damping factor.

Core samples taken from the study site approved a quite complex subsurface geology. In a very simplified model, the site is mainly characterized as a layered earth with the following layers from surface to depth: 0.5 *m* of a thin superficial clayey layer with the resistivity of 30 *Am*; about 4 *m* of dry sand and gravel with the resistivity of 7000 *Am*; another thin clay layer with the resistivity of 30 *Am*; a fourth layer with the resistivity of 40 *Am* down to about 10.5 *m*, and the deepest layer that is below the water table for the full length of the ERT profile and has an average resistivity of 8 *Am*. An important characteristic of the study site is the presence of an anomalous central zone with considerably higher porosity and permeability with the resistivity of about 750 *Am* in dry conditions. Fig. 2 illustrates simplified models of the study site constructed based on the coring information. Model 2 in Fig. 2 simulates an expected situation after a rainfall event when a rapid decrease of 50 % happens in the permeable central zone of Model 1 and a slower decrease of 10 % happens in less-permeable lateral zones. Model 3 simulates the return to the initial model after the rainfall was stopped for a few days.

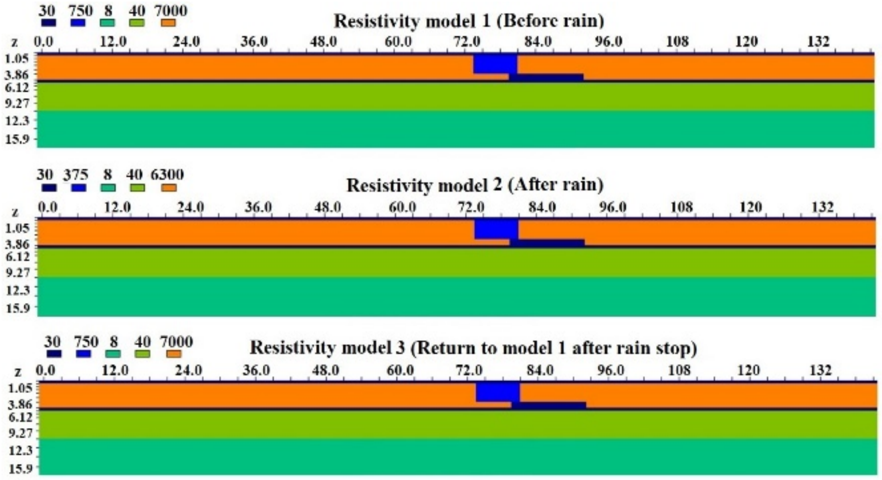


Fig. 2. Time-lapse resistivity models defined based on coring data obtained from the study site.

Preliminary inversions of the ERT monitoring data performed with the Res2dinvx64 inversion software [11] showed that rainfalls always produce a remarkable decrease of resistivity in the higher-permeable central zone due to water infiltration but an unexpected increase in the two lateral zones while we still expect to observe a decrease in the resistivity. An inverse situation was observed when analyzing the inversion results for the data measured after no-rain periods compared to those measured after rainfalls. In this situation we expect to observe increases in resistivity values along the profile. We could observe the increase in resistivity of the central zone but we observed decreases in resistivity values of the lateral zones. This problem was also observed when inverting the synthetic data calculated for the simplified models of the study site shown in Fig. 2. Synthetic data were generated for these simplified models by using the Res2dmod forward modelling software [11]. Fig. 3 illustrates the results of inverting the calculated synthetic resistivity pseudosections in Res2dinvx64 software [11]. The inconsistent behavior of different zones is clearly observed and we know that the observed changes do not reflect a real response of the subsurface. Such unexpected increases in resistivity values of lateral zones after rainfall events and decreases in resistivity values of lateral zones after dry periods are inversion artifacts caused by ill-conditioning affecting the non-linear tomographic problem.

We tried to optimize the inversion parameters by testing different regularization strategies but any inversion independently performed on singular tomographic sections was constantly affected by the artifacts, although smoothed. Therefore, we decided to move to a time-lapse inversion approach where many tomographic sections measured at successive time intervals are simultaneously inverted. To optimize the inversion parameters and remove artifacts as much as possible under controlled conditions, we continued the tests with the synthetic data generated for the geological models of the study site (Fig. 2). The synthetic resistivity pseudosections calculated by Res2dmod forward modeling software were inverted using the time-lapse inversion in Res2dinvx64 inversion software [11]. We tried to change different parameters and it was observed that the time-lapse damping factor was significantly controlling the results. Fig. 4 shows an example of a preliminary time-lapse inversion test using the standard L2-norm method [14] and selecting the time-lapse damping factor equal to 1 which means that equal weight was given to reducing the difference between the models at different times and the individual model roughness [11]. The artifacts are still present in the results shown in Fig. 4 and we still had to optimize the time-lapse inversion parameters.

In order to optimize the time-lapse inversion parameters, numerous tests were performed exploring different parameters and mainly the time-lapse damping factors in the 3D domain directions, i.e., horizontal distance, depth and time. It was observed that the complexity of the study site mainly demands a high time-lapse damping factor to force the different time models to be more similar. Fig. 5 illustrates the results of time-lapse inversion of the calculated synthetic resistivity pseudosections for the models introduced in Fig. 2. For the results obtained in Fig. 5, a temporal damping factor equal to 5 was used and the data were inverted using the L1-norm method. The results are now reasonable and real resistivity changes are observed.

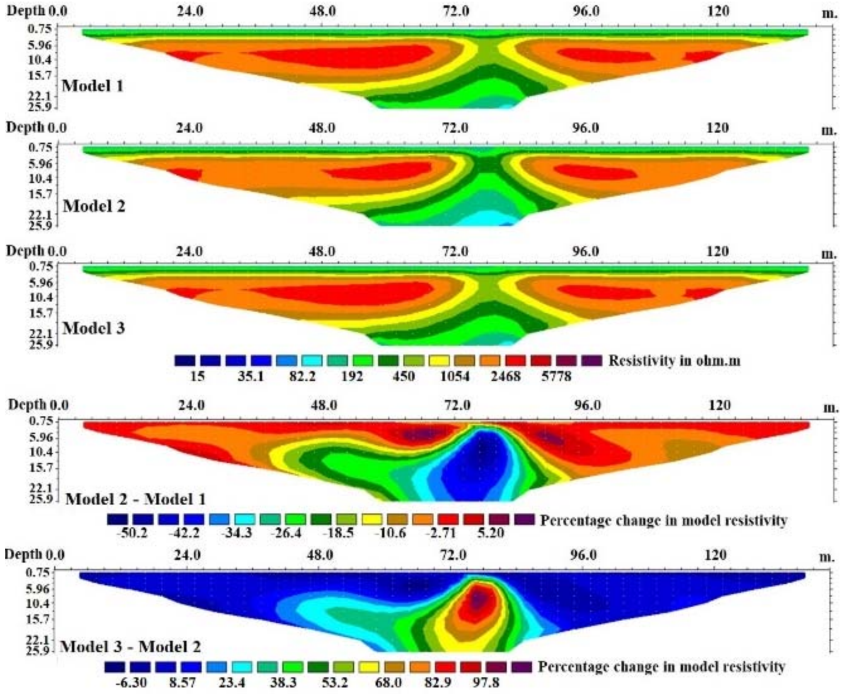


Fig. 3. Inversion results for the synthetic data calculated for the models defined in Fig. 2 and comparing changes in resistivity values between different models. Opposite changes observed in resistivity values of the lateral zones are due to inversion artifacts.

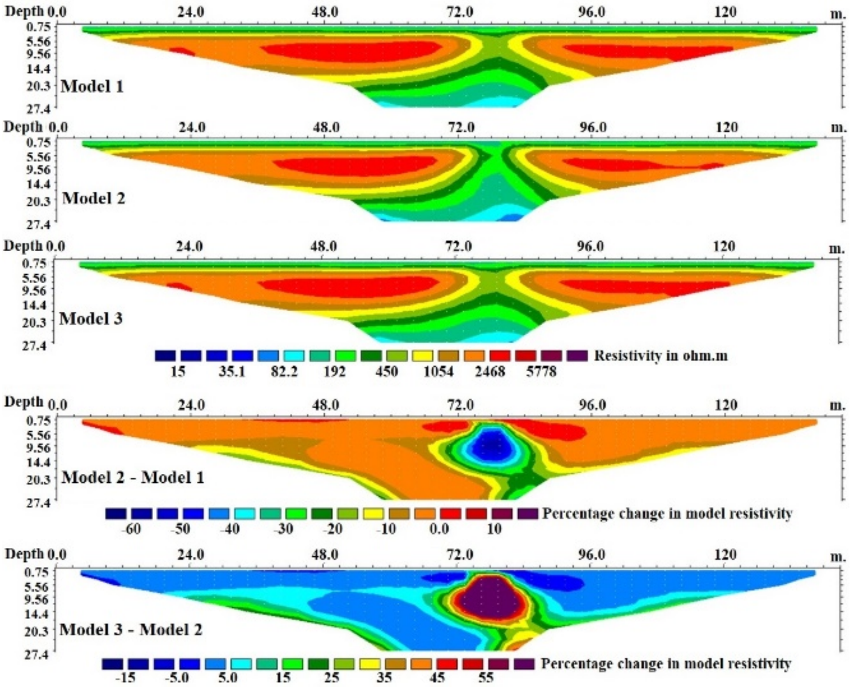


Fig. 4. Inversion results for the synthetic data calculated for the models defined in Fig. 2 using time-lapse inversion with the time-lapse damping factor equal to 1 and using the standard L2-norm method.

The optimized inversion parameters were then used to invert the data measured by the monitoring system and the data were inverted using the L2-norm method. When applied to real data, this resulted in meaningful inversion results for more than 95 % of a 14-month dataset acquired from the day of installing the system, with very minimal artifacts on the remaining 5 %.

3. Conclusions

Several studies are reported where minor differences are observed between individual or time-lapse inversions of long-term ERT monitoring data. On the contrary, we discussed a complex geology for which not only time-lapse inversions are necessary, but also lots of tests were required to optimize the inversion parameters to fight with the ill-conditioning of the inverse problem and to prevent artifacts. The results were highly sensitive to the time-lapse damping factor and the proper calibration of this factor was fundamental. We also observed the benefit of minimizing the proper norm to obtain a model closer to the sharp boundaries of the simplified synthetic model (better reconstructed by using the L1-norm method) or to the smooth boundaries of the real model (better described by using the L2-norm method).

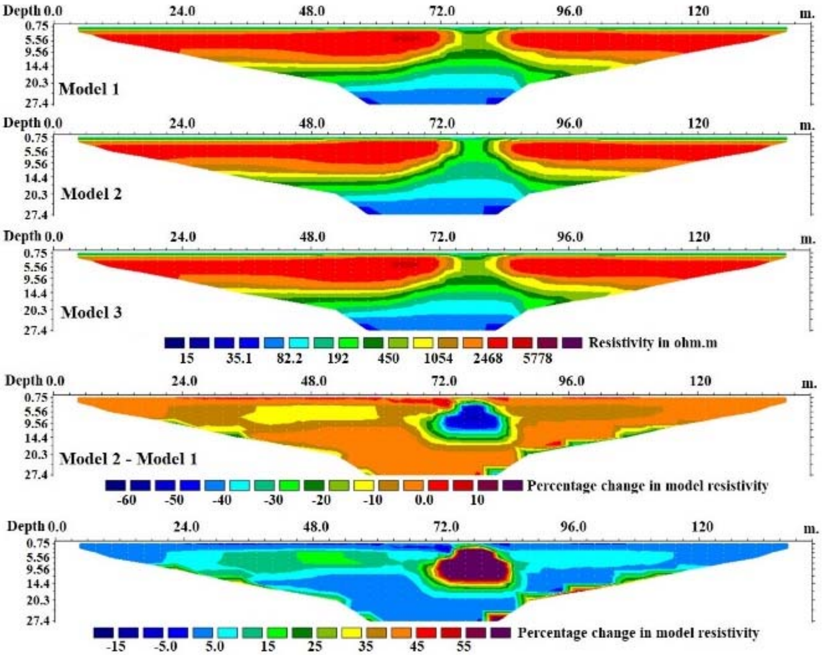


Fig. 5. Inversion results for the synthetic data calculated for the models defined in Fig. 2 using time-lapse inversion with the time-lapse damping factor equal to 5 and using the L1-norm method.

Acknowledgements

We would like to thank Federica Brambilla and Regina Bianchi, MSc students at Politecnico di Milano, who joined the research team working on time-lapse inversion of the annual data from the monitoring system. ERT monitoring systems in Italy are developed by technical support of LSI Lastem s.r.l. with scientific support of Politecnico di Milano. Inversion of ERT data in Res2dinvx64 was generated using Seequent Software (Copyright © Seequent Systems, Incorporated).

References

[1]. Z. Zhang, D. Arosio, A. Hojat, L. Zanzi, Tomographic experiments for defining the 3D velocity model of an unstable rock slope to support microseismic event interpretation, *Geosciences*, Vol. 10, Issue 327, 2020, pp. 1-23.

[2]. G. Tresoldi, A. Hojat, L. Zanzi, G.RE.T.A. installations for real-time monitoring of irrigation dams and canals, *Procedia Environmental Science, Engineering and Management*, Vol. 7, Issue 2, 2020, pp. 271-276.

[3]. A. Hojat, M. Ferrario, D. Arosio, M. Brunero, V. I. Ivanov, L. Longoni, A. Madaschi, M. Papini, G. Tresoldi, L. Zanzi, Laboratory studies using electrical resistivity tomography and fiber optic techniques to detect seepage zones in river embankments, *Geosciences*, Vol. 11, Issue 69, 2021, pp. 1-15.

[4]. M. H. Loke, J. E. Chambers, D. F. Rucker, O. Kuras, P. B. Wilkinson, Recent developments in the direct-current geoelectrical imaging method, *Journal of Applied Geophysics*, Vol. 95, 2013, pp. 135-156.

[5]. M. Moradipour, H. Ranjbar, A. Hojat, S. Karimi-Nasab, S. Daneshpajouh, Laboratory and field measurements of electrical resistivity to study heap leaching pad no. 3 at Sarcheshmeh copper mine, in *Proceedings of the 22nd European Meeting of Environmental and Engineering Geophysics*, Barcelona, Spain, 4-8 September 2016, pp. 1-5.

[6]. M. H. Loke, Tutorial: 2-D and 3-D Electrical Imaging Surveys, http://geotomosoft.com

[7]. A. Hojat, D. Arosio, L. Longoni, M. Papini, G. Tresoldi, L. Zanzi, Installation and validation of a customized resistivity system for permanent monitoring of a river embankment, in *Proceedings of the EAGE-GSM 2nd Asia Pacific Meeting on Near Surface Geoscience & Engineering (NSGE'19)*, Kuala Lumpur, Malaysia, 24-25 April 2019, pp. 1-5.

[8]. A. Hojat, G. Tresoldi, L. Zanzi, Correcting the effect of the soil covering buried electrodes for permanent electrical resistivity tomography monitoring systems, in *Proceedings of the 4th Asia Pacific Meeting on Near Surface Geoscience & Engineering (NSGE'21)*, Ho Chi Minh, Vietnam, 30 November – 2 December 2021, pp. 1-5.

[9]. M. Aleardi, A. Vinciguerra, A. Hojat, A convolutional neural network approach to electrical resistivity tomography, *Journal of Applied Geophysics*, Vol. 193, 2021, 104434.

[10]. M. Aleardi, A. Vinciguerra, A. Hojat, A geostatistical Markov Chain Monte Carlo inversion algorithm for electric resistivity tomography, *Near Surface Geophysics*, Vol. 19, Issue 1, 2021 pp. 7-26.

[11]. Geotomosoft Web Portal, http://geotomosoft.com, www.aarhusgeosoftware.dk

[12]. M. H. Loke, P. B. Wilkinson, O. Kuras, P. I. Meldrum, D. F. Rucker, The use of a semi-structured finite-element mesh in 3-D resistivity inversion, *Geophysical Prospecting*, Vol. 70, Issue 9, 2021, pp. 1580-1601.

[13]. M. H. Loke, P. B. Wilkinson, T. Dahlin, J. E. Chambers, S. Uhlemann, T. Dijkstra, Time-lapse 4-D resistivity imaging inversion with positivity constraints, in *Proceedings of the 24th European Meeting of Environmental and Engineering Geophysics*, Porto, Portugal, 9-12 September 2018, pp. 1-5.

[14]. M. H. Loke, I. Acworth, T. Dahlin, A comparison of smooth and blocky inversion methods in 2D electrical imaging surveys, *Exploration Geophysics*, Vol. 34, 2003, pp. 182-187.

(047)

Information Extraction from Electricity Invoices through Named Entity Recognition with Transformers

A. Salgado and J. Sánchez
University of Las Palmas de Gran Canaria, Computer Science Department,
35017 Las Palmas de Gran Canaria, Spain
E-mail: {agustin.salgado, jsanchez}@ulpgc.es

Summary: This article describes a method for automatically extracting information from electricity invoices. This type of documents contains rich information about the billing of each supply point and data about the customer, the contract, or the electricity company. In this work, we train a neural network to classify the input data among eighty-six different labels. We use the IDSEM dataset that contains 75.000 electricity invoices of the Spanish electricity market in PDF format. Each document is converted into text format and the classification is carried out through a named entity recognition (NER) process. The underlying neural network used in the process is a Transformer. The results demonstrate that the proposed method correctly classifies the majority of the labels with high accuracy. Furthermore, the method exhibits robustness in handling invoices with different layouts and contents, highlighting its versatility and reliability.

Keywords: Machine learning, Natural Language Processing (NLP), Named entity recognition, Transformer, Electricity invoices.

1. Introduction

Electricity invoices are complex documents that contain a large variety of information about the electricity consumption, the billing, the customer, or the electricity company. Many companies need to incorporate the data into their information systems, but this is not an easy task. Information extraction from semi-structured documents plays a significant role in processing this type of invoices fast and reliably.

This task is important for both customers and utility companies. The goal is to automatically extract the information that provides better customer service and optimize operations.

The main impediment to develop new models for extracting information from these bills is the lack of high-quality datasets. In this respect, the recently proposed IDSEM [14] dataset was built to solve this problem, containing many electricity invoices in PDF format of the Spanish market. This is an interesting dataset that has many samples, with many different layouts, where the contents were randomly generated to avoid data protection issues. There are some other datasets about electricity, such as [1, 7 or 16], but these do not completely serve for our purposes since they only contain information about household electricity consumption.

One of the challenges in information extraction from electricity bills is the diversity of formats and layouts, given by the large number of companies that operate in the market. This makes it necessary to rely on deep learning techniques for the development of a general-purpose system.

In this work, we propose a new method for automatically extracting information from electricity invoices. We use the IDSEM database that contains 75.000 invoices with eighty-six different labels. It has a total of nine templates from different electricity companies, each one with its own layout and contents.

We develop a pipeline to process the information, where the bills are first converted into text, and then tokenized and passed through a named entity recognition (NER) system to detect the entities. The NER relies on a Transformer [18], which can detect most of the entities with high accuracy. Since the entities may appear multiple times in various parts of the documents, we fuse the information to obtain a unique value per label. The NER system is an important component of many NLP applications, and it is typically used for different tasks, such as information extraction, question answering or text classification.

Transformers are a powerful type of neural network architecture that has been very successful in NLP. They are particularly effective at capturing long-range dependencies and contextual relationships in text data. However, the text used in invoices is more structured and contains diverse data, that mixes different types of amounts (for example, bill prices and electricity consumption), with textual information. Therefore, there is little contextual information in some cases.

We review some of the most important techniques for this problem in Section 2. Section 3 details the features of the IDSEM dataset and the method that we have designed for recognizing the contents of the invoices. The experimental results are given in Section 4, with a focus on the outputs of the different steps of the pipeline and a study on the performance of the method. Finally, some concluding remarks in Section 5.

2. Related Work

The field of information extraction from semi-structured documents has witnessed significant advancements during the last years. Extracting relevant data from invoices or receipts poses a unique challenge due to the variability in formats and layouts.

Many effective methods have been developed to tackle this problem. Some of them are based on template matching [2, 12], which involves creating predefined templates that capture the structure and key fields of the documents. These templates serve as reference points for information extraction. However, these types of methods are highly dependent on accurately matching document layouts to predefined templates and may not handle variations well.

Rule-based extraction methods (SmartFix [3, 4, 8], Intellix [15]), on the other hand, rely on defining a set of rules or patterns to identify and extract relevant information from the documents. These rules can be based on regular expressions, keywords, or positional patterns. While rule-based methods are straight forward to implement, they can be brittle and require regular maintenance to accommodate document variations.

Traditional machine learning approaches have also been employed to extract information from invoices. Supervised learning involves training a model on annotated data to recognize patterns and extract relevant information. Unsupervised learning, on the other hand, leverages clustering algorithms to identify similar document structures and extract information accordingly.

Natural Language Processing (NLP) techniques treat documents as text and include several tasks, such as named entity recognition [5], part-of-speech tagging, and semantic parsing, which are utilized to extract structured information from unstructured text. These techniques enable the identification of entities, relationships, and key attributes within the document text.

Deep Learning models [9] have shown promising results for information extraction in recent years. Convolutional Neural Networks (CNNs) [6, 19], Recurrent Neural Networks (RNNs) [10, 13], or Transformer-based architectures [20] can be applied for tasks like document layout analysis, character recognition, and sequence labeling.

Hybrid approaches combine multiple methods, such as rule-based and machine learning techniques, to achieve more accurate and robust extraction. For example, rule-based methods can be used for initial extraction, followed by machine learning models to handle variations, and improve extraction accuracy.

The effectiveness of these methods depends on the availability of high-quality training data, domain-specific knowledge, and the complexity of the documents being processed. Additionally, ongoing model evaluation and adaptation are crucial to maintain accurate extraction performance as document formats evolve.

3. Material and Methods

3.1. Electricity Invoices Dataset

We use the IDSEM [14] dataset that contains electricity invoices in PDF format. This recently published database is organized in two directories with 30.000 files for training and 45.000 for testing. Each bill is defined by eighty-six different labels that are stored in JSON files. In the training set, there are two PDF files: one for the invoice and another with annotations around the labels to facilitate information extraction during the training process. The label codes are placed at the beginning and end of the corresponding values, such as #J5 191.32 #J5, where J5 represents the code for the "total price".

For each invoice, there is a corresponding JSON file that contains the value of each tag embedded in the PDF. This JSON file is used to verify the accuracy of the information extracted from the original bill.

Fig. 1 displays a fragment of an invoice from the IDSEM database and Fig. 2 shows the same fragment with label annotations.

MORGAN STANLEY CAPITAL GROUP ESPAÑA, S.L.
CIF B84710292
Domicilio social: C/ PRINCIPE DE VERGARA, Nº 187, 28002 - Madrid

DATOS DE LA FACTURA DE ELECTRICIDAD

IMPORTE FACTURA: 184,40 €
Nº Factura: L9563906771 emitida el 13 de enero de 2010
Periodo de consumo: 12 de noviembre de 2009 a 11 de enero de 2010
Fecha de cargo: 16 de enero de 2010
Referencia del contrato de suministro: 1544388405822

RESUMEN

Por potencia contratada		27,98 €
Por energía consumida		133,02 €
Impuesto electricidad		8,23 €
Alquiler equipos medida y control		3,02 €
IVA	7% s/169,23 €	11,85 €
	10% s/3,02 €	0,30 €
TOTAL IMPORTE FACTURA:		184,40 €

Celeste Zetina Quesada

Calle Hospital

25577 Lladorre

Fig. 1. A fragment of an invoice of the IDSEM dataset.

#C1 MORGAN STANLEY CAPITAL GROUP ESPAÑA, S.L. #C1
CIF #C2 B84710292 #C2
Domicilio social: #C3 C/ PRINCIPE DE VERGARA, Nº 187, 28002 - Madrid #C3

DATOS DE LA FACTURA DE ELECTRICIDAD

IMPORTE FACTURA: #J5 184,40 #J5 €
Nº Factura: #F1 L9563906771 #F1 emitida el #F3 13 de enero de 2010 #F3
Periodo de consumo: #F4 12 de noviembre de 2009 #F4 a #F5 11 de enero de 2010 #F5
Fecha de cargo: #G3 16 de enero de 2010 #G3
Referencia del contrato de suministro: #E9 1544388405822 #E9

RESUMEN

Por potencia contratada		#J1 27,98 #J1 €
Por energía consumida		#J2 133,02 #J2 €
Impuesto electricidad		#N2 8,23 #N2 €
Alquiler equipos medida y control		#M4 3,02 #M4 €
IVA	#N6 7 #N6% s/#N7 169,23 #N7 €	#N8 11,85 #N8 €
	#N4 10 #N4% s/#M4 3,02 #M4 €	#N5 0,30 #N5 €
TOTAL IMPORTE FACTURA:		#J5 184,40 #J5 €

#A1 Celeste Zetina Quesada #A1

#A3 Calle Hospital #A3

#A4 25577 #A4 #A5 Lladorre #A5

Fig. 2. A fragment of an invoice with annotations of the IDSEM dataset. The annotations allow labeling the relevant information of the invoices. The code of the labels surrounds the value in the invoice.

Table 1 shows the file resulting from converting the invoice in Fig. 1 to text format. It presents an extract from the file in Spacy [17] format. Each invoice is described by the text extracted from the PDF along with the entities identified by their start and end positions in the text. Additionally, in Table 1 we provide a list of the various labels identified in the invoice.

We have selected a subset of 5.000 invoices corresponding to five different marketers. A total of 4.000 invoices are used for training and 1.000 for validation. We convert all the invoices from PDF to text and identify the position of the labels in the text. We use the `pdf2text` application for extracting the text from the PDF file. This extraction is done line by line, sometimes causing label information to span multiple lines or overlap with multiple tokens on the same line. We detect the start and end positions of each label using the labels annotations identified in the text, i.e. `(354, 360, 'J5')` for the label 'J5', total price. With this information, we build the entity section of the Spacy file. In Fig. 3, we show the pipeline for converting the IDSEM invoices to Spacy format.

Table 1. An electricity invoice in text format corresponding to the bill shown in Fig. 1. This is the format of the SpaCy library, with the position of entities and their codes at the end. At the bottom, we enumerate the most relevant labels present in the invoice.

```
[('DATOS DE LA FACTURA DE ELECTRICIDAD\n'
'IMPORTE FACTURA: 184,40 €\n'
'N° Factura: L9563906771 emitida el 13 de enero de
2010\n'
'Periodo de consumo: 12 de noviembre de 2009 a #F5
11 de enero de\n'
'Fecha de cargo: 16 de enero de 2010\n'
'Referencia del contrato de suministro:
1544388405822\n'
...
'RESUMEN MADRID #C5\n'
'Por potencia contratada 27,98 €'
'Celeste Zetina Quesada\n'
'Por energía consumida 133,02 €\n'
' 8,23 €\n'
'Calle Hospital\n'
'Impuesto electricidad\n'
'Alquiler equipos medida y control 3,02 € 25577'
'Lladorre\n'
'IVA 11,85 €\n'
' 7 % s/169,23 €\n'
' 0,30 €\n'
' 10 % s/3,02 €\n'
'TOTAL IMPORTE FACTURA: 184,40 €\n'
'\x0c',
{'entities': [(354, 360, 'J5'), (451, 462, 'F1'),
     (474, 493, 'F3'), (590, 613, 'F4'),(1111,
    1130,'G3'), (1246, 1259, 'E9'), (1657, 1662,
   'J1'), (1670, 1692, 'A1'), (1749, 1755, 'J2'),
   (1815, 1819, 'N2'), (1900, 1914, 'A3'), (2008,
   2012, 'M4'), (2021, 2026, 'A4'), (2027, 2035,
   'A5'), (2093, 2098, 'N8'), (2130, 2131, 'N6'),
   (2135, 2141, 'N7'), (2202, 2206, 'N5'), (2240,
      2242, 'N4'), (2246, 2250, 'M4'), (2310,
                  2316,'J5')]})]
```

A1/B1, Customer's name	
A3/B3, Customer's address	N6, Reduced tax rate
A4/B4, Postal code	N8, Tax price
A5/B5, Customer's city	E9, Reference supply contract
J1, Electricity power price	F1, Invoice number
J2, Energy consumed price	F2, Invoice reference
J5, Total price	F3, Invoice release date
M4, Equipment rental price	F4, Start billing date
N2, Electricity tax price	F5, End billing date
N4, Normal tax rate	G3, Payment date
N5, Reduced tax price	

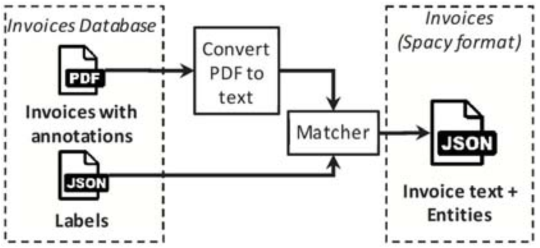


Fig 3. Pipeline for converting the IDSEM invoice to Spacy format, text + entities. In Fig. 1, we have an example of the input data and Table 1, an example of output data.

3.2. Configuration of the NER Model

Named Entity Recognition is a key task in NLP that involves identifying and classifying entities within the text. The NER pipeline typically consists of several steps, including tokenization, part-of-speech tagging, and entity recognition.

The Transformer architecture is based on the self-attention mechanism, which allows the network to attend to different parts of the input sequence with varying degrees of importance. Its architecture consists of an encoder and a decoder, which are connected by multi-headed self-attention and feedforward layers.

We design a system that automatically extracts the relevant information using a NER process. In our method, we choose a Transformer model for training the NER. The system is implemented with the SpaCy [17] library, which is a standard library for NLP in Python.

Fig. 4 shows the pipeline for the training process, which includes the NER component and the input data. The named entity recognizer takes electricity invoices and predicts sentences and their labels in context. The training data includes the text of invoices, the positions of the entities they contain, and the entity label codes. It is necessary to have another entity that do not correspond to any label.

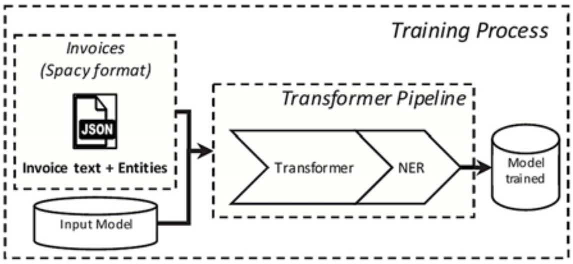


Fig. 4. Transformer pipeline for the training process.

The training data helps the model to specify the information that we want it to predict. Since the model can detect multiple values for the same entity, we fuse the information by checking the format of the label and some statistics of the extracted data. Our trained model can recognize entities in similar contexts, independently of the format employed in each bill and the organization of the entities in the document.

4. Results

To evaluate our model's accuracy, we compare its predictions with invoices not used during training. We calculate the accuracy of our model with standard performance metrics [11], such as Precision, Recall and F1-score. The overall accuracy for all the labels and invoices in the test set is 99.3 %.

Table 2 shows the results for many principal labels, like the customer's name (A1), the total price (J5), the equipment rental price (M4), or the taxes (N2, N5, N8). The precision, recall, and F1-score metrics are presented for each label. There are many labels that obtain an F1-score of 100 %. The F1-score of the N2 label, Electricity tax price, was the lowest one with 41 %. Its recall is low, which indicates many false negatives in this case. These results show that the proposed method achieves high accuracy for most of the labels.

Table 2. Precision, Recall and F1-score obtained for some labels of the invoice of Fig. 1.

Label	Precision	Recall	F1-score
A1/B1, Customer's name	100 %	100 %	100 %
A3/B3, Customer's address	100 %	100 %	100 %
A4/B4, Postal code	100 %	100 %	100 %
A5/B5, Customer's city	100 %	100 %	100 %
C1, Marketer's name	32 %	100 %	48 %
C5, Marketer's city	99 %	99 %	99 %
CD, Customer's support ph.	58 %	100 %	74 %
J1, Electricity power price	94 %	100 %	97 %
J2, Energy consumed price	100 %	100 %	100 %
J5, Total price	100 %	100 %	100 %
M4, Equipment rental price	100 %	100 %	100 %
N2, Electricity tax price	100 %	26 %	41 %
N4, Normal tax rate	100 %	100 %	100 %
N5, Reduced tax price	100 %	100 %	100 %
N6, Reduced tax rate	100 %	100 %	100 %
N8, Tax price	77 %	100 %	87 %
F1, Invoice number	100 %	100 %	100 %
F3, Invoice release date	100 %	100 %	100 %
F4, Start billing date	100 %	100 %	100 %
F5, End billing date	100 %	100 %	100 %

Fig. 9 displays two confusion matrices corresponding to the labels detailed in Table 2. The diagonal values in the matrices indicate the number of correctly identified labels, while the off-diagonal values represent the misclassified labels. The matrices demonstrate high accuracy of the proposed method. The first matrix shows the labels with the highest value of the "Precision" while the second matrix shows the less precise labels.

To verify the correct functioning of the proposed method, Fig. 5 shows an example of an electricity bill used in the tests.

The visualizations produced by `displacy` are informative and provided valuable insights into the model's performance. These visualizations allowed us to identify the entity types that the model performed well on and those that it struggled with. For instance, the model has high precision and recall for identifying the marketer and customer information.

DATOS DE LA FACTURA

IMPORTE FACTURA: 191,32 €
Nº factura: ID8475477607
Referencia: 724759220605/7202
Periodo de consumo: 09/12/2015 a 07/02/2016
Fecha de cargo: 12 de febrero de 2016

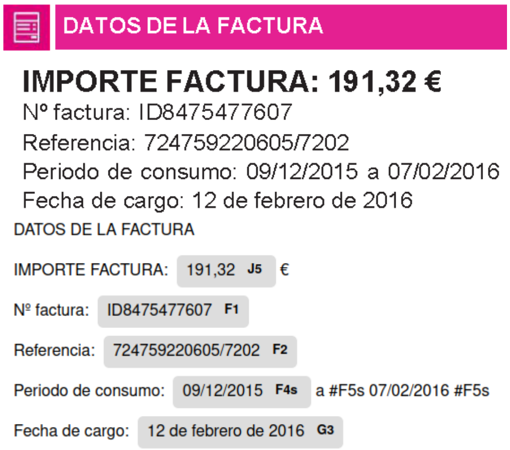
Carina Villaseñor Talavera
GC-1
50512 Torrellas
Zaragoza

FACTURA RESUMEN

Por potencia contratada	31,70 €
Por energia consumida	135,55 €
Impuesto electricidad	8,55 €
Alquiler equipos de medida y control	3,01 €
IGIC reducido (7%)	12,31 €
IGIC normal (6,5%)	0,20 €
TOTAL IMPORTE FACTURA	191,32 €

Fig. 5. A fragment of an invoice of the IDSEM dataset.

Figs. 6, 7 and 8, depict some fragments of the invoice shown in Fig. 5. They highlight named entities and their labels in a text. The figures show that the proposed method successfully identified all the relevant information from the invoice, demonstrating the effectiveness of the proposed method.


DATOS DE LA FACTURA

IMPORTE FACTURA: 191,32 €
Nº factura: ID8475477607
Referencia: 724759220605/7202
Periodo de consumo: 09/12/2015 a 07/02/2016
Fecha de cargo: 12 de febrero de 2016

DATOS DE LA FACTURA

IMPORTE FACTURA: 191,32 J5 €

Nº factura: ID8475477607 F1

Referencia: 724759220605/7202 F2

Periodo de consumo: 09/12/2015 F4s a #F5s 07/02/2016 #F5s

Fecha de cargo: 12 de febrero de 2016 G3

Fig. 6. Fragment of the summary of the invoice shown in Fig. 5 and the named entities identified by our method.


EDP ESPAÑA, S.A
Cif: A33473752
PLAZA DE LA GESTA, Nº2 33007 - OVIEDO

Carina Villaseñor Talavera
GC-1
50512 Torrellas
Zaragoza

EDP ESPAÑA, S.A C1 Carina Villaseñor Talavera A1

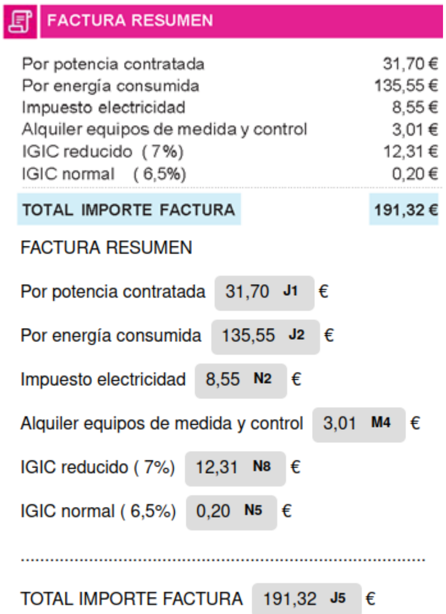
Cif: A33473752 C2 GC-1 A6

PLAZA DE LA GESTA, Nº2 C3 33007 C4 - OVIEDO C5

50512 A4 Torrellas A5

Zaragoza A6

Fig 7. Fragment of the electricity company and the customer of the invoice shown in Fig. 5 and the named entities identified by our method.



FACTURA RESUMEN

Por potencia contratada	31,70 €
Por energía consumida	135,55 €
Impuesto electricidad	8,55 €
Alquiler equipos de medida y control	3,01 €
IGIC reducido (7%)	12,31 €
IGIC normal (6,5%)	0,20 €
TOTAL IMPORTE FACTURA	191,32 €

FACTURA RESUMEN

Por potencia contratada 31,70 J1 €

Por energía consumida 135,55 J2 €

Impuesto electricidad 8,55 N2 €

Alquiler equipos de medida y control 3,01 M4 €

IGIC reducido (7%) 12,31 N8 €

IGIC normal (6,5%) 0,20 N5 €

……………………………………………………………………………

TOTAL IMPORTE FACTURA 191,32 J5 €

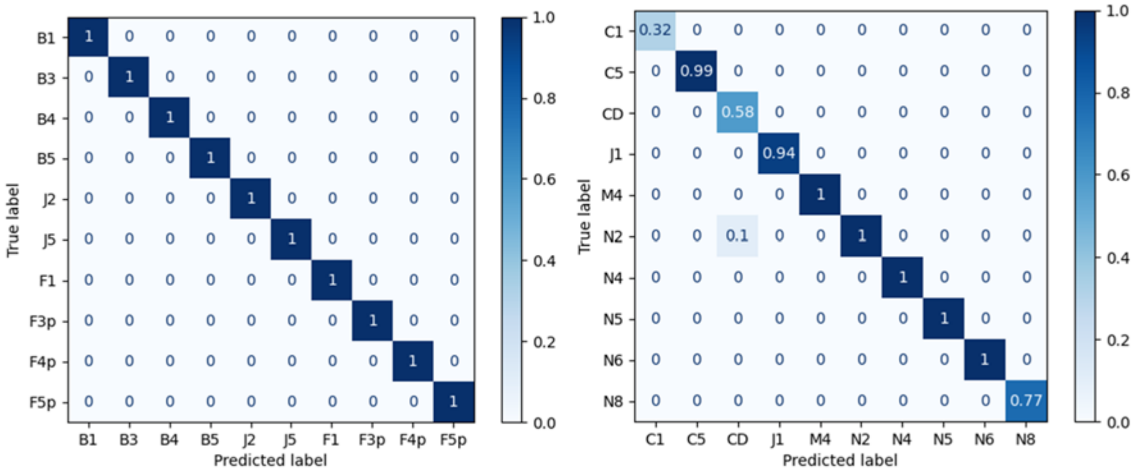
Fig 8. Fragment of the breakdown of the invoice shown in Fig. 5 and the named entities identified by our method.

5. Conclusion

We proposed a method for automatically extracting information from electricity invoices. This method relied on text data and is based on an NLP pipeline to process the information. The main module of the system is based on a NER process, supported by a Transformer neural network, that allows detecting the named entities. The results showed that the proposed system can extract the main labels with high accuracy. The visualizations shown allowed us to gain a deeper insight on the model's strengths and weaknesses. In future works, we will explore other neural network architectures to integrate visual information with text data.

Acknowledgements

This work was partly supported by the Servicio Canario de Salud, Gobierno de Canarias, under Contract F2022/03.



True label \ Predicted label	B1	B3	B4	B5	J2	J5	F1	F3p	F4p	F5p
B1	1	0	0	0	0	0	0	0	0	0
B3	0	1	0	0	0	0	0	0	0	0
B4	0	0	1	0	0	0	0	0	0	0
B5	0	0	0	1	0	0	0	0	0	0
J2	0	0	0	0	1	0	0	0	0	0
J5	0	0	0	0	0	1	0	0	0	0
F1	0	0	0	0	0	0	1	0	0	0
F3p	0	0	0	0	0	0	0	1	0	0
F4p	0	0	0	0	0	0	0	0	1	0
F5p	0	0	0	0	0	0	0	0	0	1

True label \ Predicted label	C1	C5	CD	J1	M4	N2	N4	N5	N6	N8
C1	0.32	0	0	0	0	0	0	0	0	0
C5	0	0.99	0	0	0	0	0	0	0	0
CD	0	0	0.58	0	0	0	0	0	0	0
J1	0	0	0	0.94	0	0	0	0	0	0
M4	0	0	0	0	1	0	0	0	0	0
N2	0	0	0.1	0	0	1	0	0	0	0
N4	0	0	0	0	0	0	1	0	0	0
N5	0	0	0	0	0	0	0	1	0	0
N6	0	0	0	0	0	0	0	0	1	0
N8	0	0	0	0	0	0	0	0	0	0.77

Fig. 9. Confusion matrices of the labels shown in Table 2.

References

[1]. J. Chavat, S. Nesmachnow, J. Graneri, G. Alvez, ECD-UY, detailed household electricity consumption dataset of Uruguay, *Scientific Data*, Vol. 9, 2022, 21.

[2]. V. P. d'Andecy, E. Hartmann, M. Rusiñol, Field extraction by hybrid incremental and a-priori structural templates, in *Proceedings of the 13th IAPR International Workshop on Document Analysis Systems (DAS'18)*, 2018, pp. 251-256.

[3]. A. R. Dengel, B. Klein, Smartfix: A requirements-driven system for document analysis and understanding, in Document Analysis Systems V (D. Lopresti, J. Hu, R. Kashi, Eds.), *Springer*, 2002, pp. 433-444.

[4]. D. Esser, D. Schuster, K. Muthmann, M. Berger, A. Schill, Automatic indexing of scanned documents: A layout-based approach, *Proceedings of SPIE*, Vol. 8297, 2012, pp. 118-125.

[5]. G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, C. Dyer, Neural architectures for named entity recognition, *arXiv Preprint*, 2016, arXiv:1603.01360.

[6]. X. Liu, F. Gao, Q. Zhang, H. Zhao, Graph convolution for multimodal information extraction from visually rich documents, *arXiv Preprint*, 2019, arXiv:1903.11279.

[7]. S. Makonin, B. Ellert, I. Bajic, F. Popowich, Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014, *Scientific Data*, Vol. 3, 2016, 160037.

[8]. I. Muslea, Extraction patterns for information extraction tasks: A survey, in *Proceedings of the AAAI Workshop on Machine Learning for Information Extraction*, Vol. 2, 1999.

[9]. R. B. Palm, F. Laws, O. Winther, Attend, copy, parse end-to-end information extraction from documents, in *Proceedings of the International Conference on*

Document Analysis and Recognition (ICDAR'19), 2019, pp. 329-336.

[10]. R. B. Palm, O. Winther, F. Laws, Cloudscan – A configuration-free invoice analysis system using recurrent neural networks, in *Proceedings of the 14th IAPR International Conference on Document Analysis and Recognition (ICDAR'17)*, Vol. 1, 2017, pp. 406-413.

[11]. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, et al., Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research*, Vol. 12, 2011, pp. 2825-2830.

[12]. M. Rusiñol, T. Benkhelfallah, V. P. d'Andecy, Field extraction from administrative documents by incremental structural templates, in *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR'13)*, 2013, pp. 1100-1104.

[13]. C. Sage, A. Aussem, H. Elghazel, V. Eglin, J. Espinas, Recurrent neural network approach for table field extraction in business documents, in *Proceedings of the International Conference on Document Analysis and Recognition (ICDAR'19)*, 2019, pp. 1308-1313.

[14]. J. Sánchez, A. Salgado, A. García, N. Monzón, IDSEM, an invoices database of the Spanish electricity market, *Scientific Data*, Vol. 9, 2022, 786.

[15]. D. Schuster, K. Muthmann, D. Esser, A. Schill, M. Berger, C. Weidling, K. Aliyev, A. Hofmeier, Intellix-end-user trained information extraction for document archiving, in *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR'13),* 2013, pp. 101-105.

[16]. C. Shin, E. Lee, J. Han, J. Yim, W. Rhee, H. Lee, The ENERTALK dataset, 15 Hz electricity consumption data from 22 houses in Korea, *Scientific Data*, Vol. 6, 2019, 193.

[17]. SpaCy: Industrial Strength Natural Language Processing in Python, https://spacy.io/

[18]. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in Advances in Neural Information Processing Systems, Vol. 30, *Curran Associates, Inc.*, 2017.

[19]. Y. Xu, M. Li, L. Cui, S. Huang, F. Wei, M. Zhou, Layoutlm: Pre-training of text and layout for document image understanding, in *Proceedings of the Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'20)*, New York, NY, USA, 2020, pp. 1192-1200.

[20]. X. Zhao, E. Niu, Z. Wu, X. Wang, Cutie: Learning to understand documents with convolutional universal text information extractor, *arXiv Preprint*, 2019, arXiv:1903.12363.

(048)

Autolabel: Improving Petri Dish Automatic Labels with AI Algorithms

V. Quevit [1,2,3], J.-M. Laferté [2], A.-J. Fougères [2], H. Djelal [4], J.-L. Dillenseger [1] and E. Jalenques [3]
[1] Univ Rennes, Inserm, LTSI – UMR 1099, F-35000 Rennes, France
[2] ECAM Rennes, Louis de Broglie, Campus de Ker Lann, Bruz, Rennes 35091, France
[3] Interscience, 30 Chem. du Bois des Arpents, 78860 Saint-Nom-la-Bretèche
[4] UniLaSalle Rennes, Campus de Ker Lann, Av. Robert Schuman, 35170 Bruz
E-mail: victorien.quevit@ecam-rennes.fr

Summary: The present research aims at improving the accuracy of labels on Petri dish images containing Colony Forming Units using Artificial Intelligence algorithms. Indeed, the labeling methods proposed by classical computer vision software such as ScanStation for example, are prone to errors and the manual correction of these errors is a difficult task. We propose a methodology based on AI models. At first, a YOLO model is trained on the existing labels given by ScanStation. The bounding boxes provided by ScanStation and YOLO are then binarized using the OTSU algorithm to generate semantic labels that are used to train a U-Net. Then, a Xception model is trained to classify all the segments generated by the U-Net as either outlier or colony. For new data, the trained U-Net and Xception models are used to improve the labeling. The results indicate that the proposed approach improves the accuracy of the labeling process without human correction.

Keywords: Deep learning, Convolutional neural network, Yolo, U-Net, Xception, Colony forming units, Petri dish labeling.

1. Introduction

Microbial contamination is a pervasive phenomenon that poses a significant threat to human health worldwide. Microorganisms, including bacteria, viruses, fungi and parasites, can cause a wide range of infections, diseases and even death. According to the World Health Organization (WHO) [1], approximately 10 % of the world's population, or about 600 million people, become ill each year from eating contaminated food, resulting in 420000 deaths and 33 million lost years of healthy life.

The best way to prevent a microbial infection is to test samples for the presence of bacteria and other harmful microorganisms. However, their small size does not make them directly visible in a sample, and expensive equipment is often required to accurately detect and quantify these microorganisms.

Petri dishes provide a cheap and simple alternative to this problem. When a sample is placed on a Petri dish containing a nutrient-rich medium, each bacterial cell will begin to grow and divide, forming a colony or 'Colony Forming Unit' (CFU) as shown in Fig. 1. A CFU is essentially a visible cluster of identical bacteria that have emerged from a single bacterial cell in the original sample. By counting the number of CFUs on a Petri dish, we can estimate the number of bacteria present in the original sample.

Fig. 1. Growth of a CFU at h+0, h+6, h+12, h+18 and h+24 after incubation (from left to right).

The number of CFU is usually between 30 and 300 in a single Petri dish, usually 90 mm in diameter (Fig. 2). Counting CFU's in Petri dishes is an essential process for various industries, especially food but also cosmetics and pharmaceuticals. Accurate CFU counting helps determine the safety and quality of these industrial products for consumption or use, making it a crucial step in the manufacturing process.

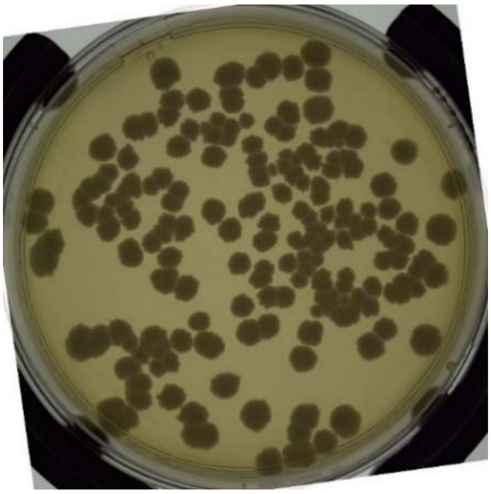


Fig. 2. Image of a Petri dish 24 hours later incubation.

To evaluate automatic counting software, or to develop a supervised detection model, it is therefore essential to accurately label the images of Petri dishes containing CFUs, i.e., to mark with bounding boxes or masks all CFUs present in the dish.

The traditional labeling method involves manual counting. However, counting CFUs is a time-consuming and laborious process that requires skilled technicians with a keen eye for detail. In addition, visual interpretation can be subjective, as evidenced by the non-reproducibility of accurate labeling between two microbiologists, especially on dishes with many CFUs or clusters. Furthermore, the counting process is affected by various factors such as lighting conditions, colony shape and size, and observer experience. This can lead to variability and errors in labeling, affecting the accuracy and reliability of the analysis.

To overcome this challenge, artificial intelligence (AI) algorithms have been explored in recent years [2, 3]. However, these algorithms require a large, properly labeled dataset. In this study, we propose an AI-based approach called 'Autolabel' to improve the accuracy of labeling images of Petri dishes containing CFUs. Specifically, we use a combination of the YOLO [4] (You Only Look Once), an object detection algorithm that uses a single neural network to simultaneously predict bounding boxes, U-Net [5], a convolutional neural network architecture commonly used for semantic segmentation tasks, and Xception [6], an advanced neural network architecture designed for efficient and accurate image classification, to generate more accurate and reliable labels.

The methodology of our approach, which consists of three major components: learning strategy (2.1), inference (2.2), and training and test datasets (2.3), is described in detail in Section 2 of this paper. The results of our study are presented in Section 3, while a conclusion on the effectiveness of our approach is drawn in Section 4. Finally, the perspectives of this research as well as possible improvement are discussed in Section 5.

2. Methodology

2.1. Learning Strategy

CFUs labels that are given by professional tools such as ScanStation (an incubator manufactured by Interscience equipped with a computer vision program that analyzes the box throughout its incubation process, allowing to identify growing CFUs), are not perfect and contains some errors, mainly on the edge of the box or in difficult cases such as a high density of CFUs, clusters, low contrasted CFUs or abnormal shaped CFUs. To solve this problem, we proposed the following process, as describe in the diagram in Fig. 3.

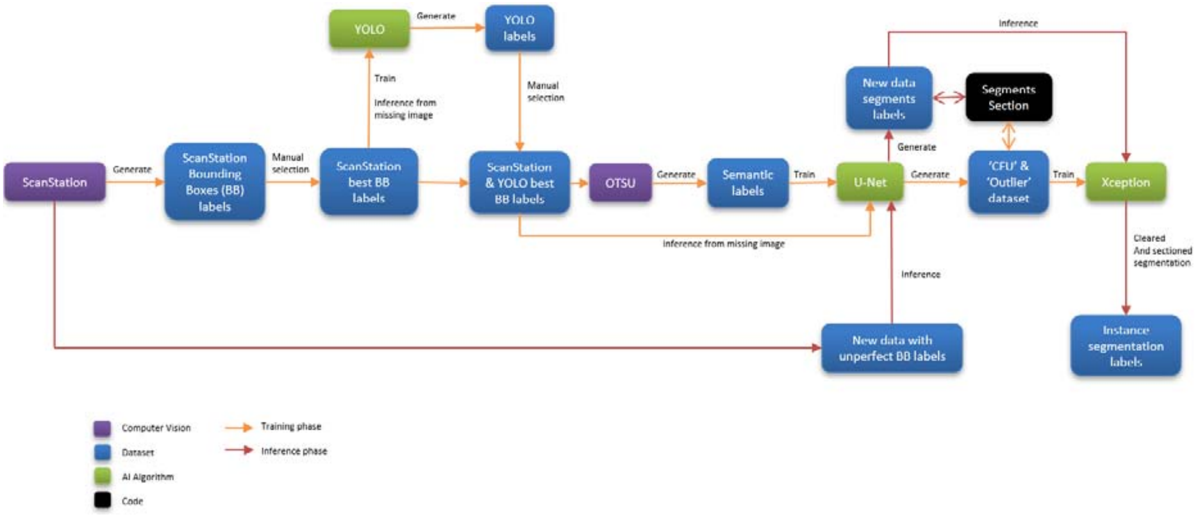


Fig. 3. Diagram representing the Autolabel process. The purple boxes encompass Computer Vision algorithms, the green boxes signify AI algorithms, the black box represents a basic program, and the blue boxes indicate datasets or labels. The orange pathway symbolizes the training phase, or learning strategy, while the red pathway corresponds to the inference process.

1) The deep learning algorithm YOLO was used to relabel the Petri dish images that ScanStation failed to label well. By manually selecting the best Petri dish images (labeled by ScanStation) to train YOLO, and then using the trained model to label the remaining Petri dish images (Fig. 4), a large number of bounding boxes were generated.

2) In order to obtain semantic labels, the OTSU thresholding method was used to binarize the content of the bounding boxes from ScanStation and YOLO. The bounding boxes and OTSU semantic results are shown in Fig. 5.

3) These binarized bounding boxes combinations are then used as semantic labels to train a U-Net model, a convolutional neural network used for image segmentation.

The primary purpose of employing U-Net lies in two aspects. Firstly, it aims to reduce the possibility of undetected CFU by generate segments that may not contain bounding boxes.

Secondly, it seeks to establish semantic labels for all CFUs. This form of labeling can be utilized to train an instance segmentation model like Mask-RCNN [7], facilitating more accurate and comprehensive CFU detection. The output segments of the U-Net model can be seen in Fig. 6.

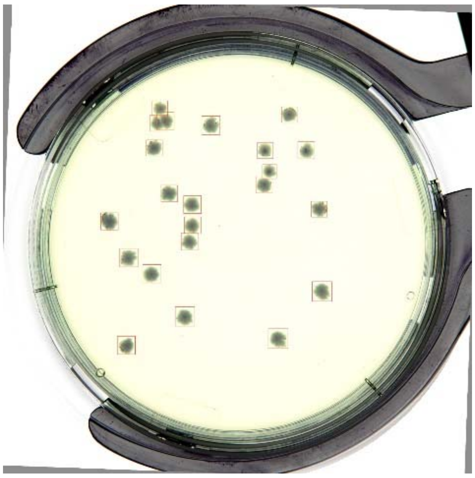


Fig. 4. CFUs detection by YOLO.

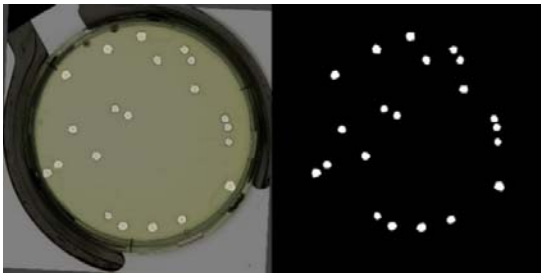


Fig. 5. CFUs segmentation with OTSU

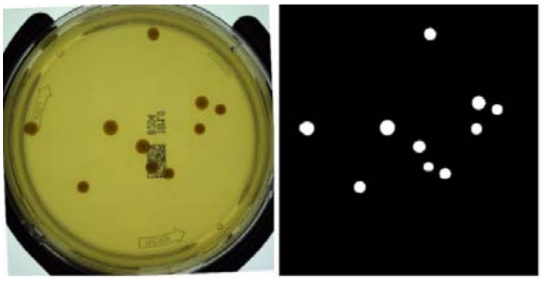


Fig. 6. Output of U-Net.

4) Several outputs of the U-Net are then classified as 'UFC' or 'outlier'. For this purpose, a model based on Xception, commonly used for image classification was trained with two datasets (one per class). For the 'UFC' class, we used the segments given by U-Net which were identified inside a bounding box given by ScanStation and YOLO. For the 'Outlier' class, we took the segments given by U-Net from boxes without CFU (before colony growth but with all artifacts like bubbles, annotations, impurities, etc.).

If a segment contains multiple bounding boxes, meaning it’s a cluster, we separated the segment according to the distance from the center to the bounding box limit. The results of this segment sectioning can be seen in Fig. 7, and an example of segments removed by Xception (bubble) is shown in Fig. 8.

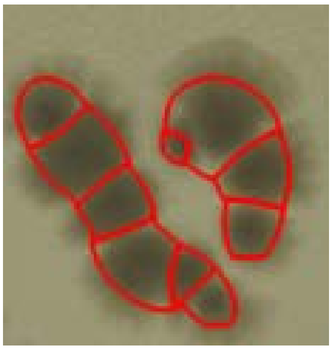


Fig. 7. Section of two agglomerate segments.

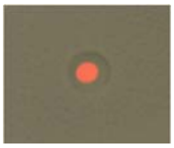


Fig. 8. Classification of an outlier (bubble).

2.2. Inference

Once learning is complete, inference is performed by injecting the new data into the U-Net model for segmentation followed by Xception for classification. As a consequence, the imperfect labels of bounding boxes from ScanStation are transformed into instance segmentation, incorporating certain missed CFUs while eliminating identified outliers that could have been detected. Fig. 9 shows the results of our improved Autolabel labeling algorithm. Individual colonies are identified by different shades of purple.

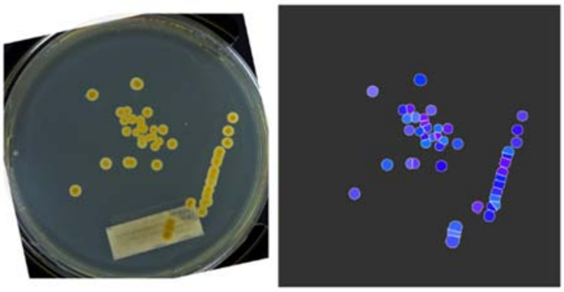


Fig. 9. Result of our method.

To compare the results with the original method, we transformed the individual segments into bounding boxes.

2.3. Training and Testing Datasets

In the training phase, we used a dataset consisting of 11900 annotated Petri dish image sequences acquired from ScanStation. Each sequence includes several images illustrating the incubation and growth stages of CFUs. We selected a total of 989 Petri dishes based on the best visually apparent ScanStation results on the last image of the sequence. For each selected

Petri dish, we extracted images containing CFUs at 2-hour intervals, resulting in a dataset of 11760 images for training the YOLO model. The U-Net dataset contained 7584 images, each from the last image of the sequence and selected based on the best OTSU binarization result from the bounding boxes generated by YOLO and Scanstation.

For training the Xception classifier, we provided an equal number of segments from both classes (13027). We performed the test on a separate set of 166 images captured by a system from the same manufacturer as the training data. The manually corrected results from ScanStation were used as ground truth.

It should be noted that, unlike the training dataset which contained only the best visual results from ScanStation, the test dataset used to evaluate our method contained mostly images of Petri dishes with difficult counting tasks, clumps, and/or high CFU numbers.

3. Results

The classical measure for evaluating object detection algorithms is the Intersection over Union (IoU) measure. However, given the non-reproducibility of our data, the delimitation of bounding boxes is very subjective, which can lead to low IoUs, even for close detections. We therefore propose to adopt a metric based on a distance criterion (between the predicted center and the ground truth center of the colony) and on a shape criterion (average radius of the predicted colonies and the ground truth colonies). This new metric called 'Distance Normalized by Mean Radius' (DNMR) is defined as follows (Eq. (1)):

$$\mathrm{DNMR} = 1 - \frac{dist(C_{GT}, C_{Pred})}{avg(Rad_{GT}, Rad_{Pred})}, \quad (1)$$

where *GT* is the ground truth, *Pred* is the prediction either by ScanStation or our method, *C* is a center in a 2D spatial position, and *Rad* is the radius.

With this metric, we decided to apply a threshold of 0.5 to define if a colony is detected or not compared to the ground truth. As CFUs have a low degree of overlapping the threshold was set to 0.5, we ensured that any score beyond this value would provide a reliable indication of successful detection. This allows us to evaluate our method using the classical criteria: Precision, Recall and F1 score.

Table 1 shows the precision, recall and F1 score obtained by ScanStation and our labeling method Autolabel on the Petri dish images of the inference base.

The results clearly show that the Autolabel method performs better than a conventional ScanStation-type method in terms of precision, recall and F1 score. However, we found that some CFUs are still not detected, especially very small ones, due to the U-Net resolution limit imposed by the memory size of the GPU we used (RTX 2080Ti, 11 Gb VRAM).

Table 1. Precision, Recall and F1 Score comparison.

Label	Precision	Recall	F1 Score
ScanStation	0.622	0.590	0.606
Autolabel	0.808	0.683	0.740

4. Conclusion

Our research proposes a novel approach to train machine learning models without using manually labeled images in the training phase. Instead, we relied on imperfect labels generated by an automated process (ScanStation). Despite this unconventional approach, our results demonstrated significant improvements in precision (+18 %), recall (+9 %), and F1 score (+14 %). It should be noted that the testing dataset utilized a different labeling approach, a manual one, which posed some challenges in replicating the same type of bounding boxes. To address this issue, we employed the ‘Distance Normalized by Mean Radius’ metric.

Overall, this study highlights the potential of using AI algorithms to improve detection accuracy before a manual correction phase, which is typically a time-consuming and labor-intensive task. Our approach leverages the power of AI generalization to improve the labeling accuracy of Petri dish images, using multiple cascaded algorithms, resulting in improved detection of CFUs. Although the results of our study demonstrate a significant improvement in the accuracy of Petri dish labeling, it is important to note that there is still tremendous potential for improvement.

Furthermore, our approach has the potential to be applied beyond CFUs detection and used to test large datasets of interpretative labels, which can help highlight errors made by both human and automatic labeling processes. This can be particularly useful in scenarios where manual labeling is not feasible due to resource constraints or time limitations. By utilizing machine learning to identify errors in interpretative labels, we can improve the accuracy and reliability of data-driven applications in various domains.

5. Perspectives

In future research, we plan to apply the proposed approach to improve the label accuracy of a larger dataset of Petri dish images. We aim to use the generated labels to train a detection model such as YOLO or instance segmentation model like Mask R-CNN. Our ultimate goal is to develop a robust and accurate detection model that can be used for a variety of applications, including bacterial colony detection and classification.

References

[1]. WHO, Food Safety, 2022, https://www.who.int/news-room/fact-sheets/detail/food-safety

[2]. A. Ferrari, et al., Bacterial colony counting with convolutional neural networks in digital microbiology imaging, *Pattern Recognition*, Vol. 61, 2017, pp. 629-640.

[3]. P. Zawadizki, P. Adamczuk, K. Jamka, et al., SDM for microbiological control of cosmetic products, *AAEM*, 2021, Vol. 28, Issue 4, 705.

[4]. J. Redmon, et al., You only look once: Unified, real-time object detection, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*, 2016, pp. 779-788.

[5]. O. Ronneberger, et al.. U-net: Convolutional networks for biomedical image segmentation, in *Proceedings of the 18th International Conference Medical Image Computing and Computer-Assisted Intervention (MICCAI'15)*, 2015, pp. 234-241.

[6]. F. Chollet, Xception: Deep learning with depthwise separable convolutions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17)*, 2017. pp. 1251-1258.

[7]. K. He, et al., Mask R-CNN, in *Proceedings of the International Conference on Computer Vision (ICCV'17)*, 2017, pp. 2961-2969.

(049)

High Accuracy Brain Tumor Classification with EfficientNet and Magnetic Resonance Images

J. M. Medina and J. Sánchez
University of Las Palmas de Gran Canaria, Computer Science Department,
Centro de Tecnologías de la Imagen (CTIM), 35017 Las Palmas de Gran Canaria, Spain
E-mail: jsanchez@ulpgc.es

Summary: The automatic detection of brain tumors is important for efficiently processing large amounts of data. This is a complex task due to the great variety that exists, and the inherent challenges associated with processing medical images. The aim of this work is to assess the performance of recent neural networks for the classification of brain tumors. We study the EfficientNet model, which has provided good results in many classification problems. We use two standard datasets with more than 3000 magnetic resonance images each. The classification includes four different classes with three tumor types (glioma, meningioma, pituitary tumors), and an additional class for brains without tumors. The experiments analyze three models of the EfficientNet architecture, using several techniques, such as transfer learning, early stopping and fine-tuning. The results show that the models attain an accuracy of 98.4 % and 97.5 % with the two datasets, which is on par with state-of-the-art methods.

Keywords: Brain tumor classification, Magnetic resonance imaging (MRI), Deep learning, Convolutional neural network (CNN).

1. Introduction

During the last years, many works have explored the application of deep neural networks for the classification of brain tumors. This is an intricate task due to the difficulty in the diagnosis of several types of tumors and the implications for the health of patients. Consequently, the quest for a neural network capable of delivering exceptional accuracy in tumor detection holds paramount importance for effective patient treatment.

Many works have obtained promising results with standard architectures, such as ResNet [13], Xception [11] or Vision Transformers [24], and a few works have compared the performance of multiple neural networks, like in [17-19].

The study presented in [18] showed that EfficientNet [23] provided good performance among many different models, with one of the best trade-offs between accuracy and model complexity. However, the authors did not search for the best hyper-parameters, using a general setting for all the models.

In this work, we are interested in deepening in the capabilities of this neural network. The aim is to analyze several techniques, such as transfer learning, fine tuning, or early stopping, and find the best configuration. We use two datasets of more than 3000 magnetic resonance images each. The first one, from Figshare [7], contains 3064 images with three types of tumors, whereas the second one, from Kaggle [4], has 3264 images with an additional label for images without tumors.

The experiments analyze the behavior of different models of the EfficientNet family, obtaining an accuracy of 98.4 % with the Figshare dataset and 97.5 % with the Kaggle dataset. In particular, we study three models with a high disparity in the number of parameters, so we may understand the behavior with respect to the size of the network. The use of transfer learning is important for attaining high levels of accuracy and fine-tuning is key for further improving the results. Comparing with state-of-the-art methods, our results rank in the top of the classification for both datasets.

Section 2 discusses the main works in the literature based on neural networks. Section 3 details the datasets that we use in our study, then it summarizes the main characteristics of the EfficientNet model and details the experimental setup. Section 3 analyzes the experimental results and compares with state-of-the-art methods. Some concluding remarks are given in Section 4.

2. Related Work

Over the past few years, many methods have addressed the problem of brain tumor classification with neural networks and a few studies have compared the performance of several convolutional neural networks (CNNs) for this task. For example, in [19], the authors trained five models (Xception, ResNet50, InceptionV3, VGG16, and MobileNet), achieving accuracies of 98.75 %, 98.50 %, 98.00 %, 97.50 %, and 97.25 %, respectively. However, this preliminary work lacks sufficient details to reproduce the results and it is not clear which dataset was used in the experiments.

The authors of the work presented in [17] investigated the performance of the VGG16, VGG19, ResNet50, and DenseNet121 models using the Figshare dataset, relying on transfer learning and adding three fully connected layers on top of the neural networks. They compared results using several optimization methods (Adam, Adadelta, RMSprop, and SGD), and obtained the best results with ResNet50 and DenseNet121, with an accuracy of 99.0 % and 98.9 %, respectively.

Nevertheless, these works are not directly comparable due to the use of different datasets. The work presented in [18] analyzes most of these CNNs and includes EfficientNet and the recent ConvNext, using the two datasets. They explored different versions of each family and provided sufficient details for reproducibility. They also obtained good performance for these models and draw some interesting conclusions. The EfficientNet model presented the best trade-off between accuracy and complexity, being one of the best candidates for this problem.

Many other articles have studied the behavior of other models, like in [25], where the authors showed the superiority of GoogLeNet [21] over AlexNet [12] using transfer learning.

The EfficientNet architecture has also been previously used in [10] and [14]. The first work was based on the EfficientNetB0 model and obtained an accuracy of 96.9 % using the Kaggle dataset. In this study, we obtain better results with both EfficientNetB0 and EfficientNetB3. For the work presented in [14], it is not clear which dataset was used.

A study on ResNet50 [13] provided results with an accuracy of 97.5 % using the Figshare dataset. They obtained better results without data augmentation, which is in line with the results obtained in [18].

More recently, in [11], the authors analyze several networks and propose an ensemble of three models based on VGG16, InceptionV3, and Xception, obtaining an accuracy of 96.9 % with the Kaggle dataset. The authors also studied three different Vision Transformers, but they did not obtain good results. In our work, we obtain higher accuracy with a single model. Although the accuracy of these methods is typically above 95 %, it is not easy to compare the results because they use different datasets and configurations.

3. Material and Methods

3.1. Datasets

In this work, we use two MRI datasets for brain tumor classification. The first one is available on Figshare [7] and was initially proposed in [6]. It is one of the first datasets for this purpose and many works have used it for evaluating their methods. It contains 3064 images with the three tumor types: 1426 images of gliomas, 708 images of meningiomas, and 930 images of pituitary tumors. In this case, there are no images of healthy brains, thus there is not any label for the no tumor class. The information is stored in MATLAB format in a data structure that contains the image, the label, a unique identifier, and the segmentation of the tumor region, including a mask and a polygon of the tumor contour. We converted the images into PNG format and split the images into the training and testing directories, with three subdirectories corresponding to the three tumor types.

The Kaggle dataset [4] contains images with the same types of tumors and an additional classification for images without tumors. It contains 3264 images, with 926 gliomas, 937 meningiomas, 901 pituitary tumors, and 500 without tumors. In this case, the dataset is organized in two directories (training and testing) and four subdirectories corresponding to each label. In a preprocessing step, we unified the size of the images to 512×512 by scaling and centering the original images. This dataset contains more variety than the previous one, because it was created by collecting images from other sources. For instance, it includes more than 2000 images from the Figshare dataset (as reported in [18]), as well as images from other datasets. Fig. 1 depicts some slices of this dataset from different views.

3.2. The EfficientNet Neural Network

EfficientNet [22] is a recent CNN that maintains a trade-off between accuracy and FLOPS by performing a compound scaling of the depth and width of the network. It also adapts the resolution of the input images for this purpose. This family of networks defines eight different models, from B0 to B7, with increasing sizes in width, depth, and image resolution. This network relies on the Mnasnet architecture, which uses inverted bottlenecks, and squeeze-and-excitation optimization.

In our study, we analyze the second version of this family [23], which introduces several improvements over the previous version. The main differences are the introduction of progressive learning, so that the size of the images and strength of regularization is gradually increased during training; the replacement of depthwise convolutions in the first layers to increase the speed; and the scaling of the network starting from later stages.

We report the results for the B0, the B3, and the Small configurations. The number of parameters is significantly smaller than in other neural networks with similar performance. The accuracy that we obtain with the models of this version is consistently higher than with the models of the first one.

For transfer learning, we remove the top layer and include a dense layer with the number of classes (three for the Figshare dataset and four for the Kaggle dataset) and SoftMax activation. Before the dense layer, we include dropout. In our experiments, we obtained slightly better results with a dropout rate of 70 % in comparison with 50 % and 30 %, therefore, we selected 70 % by default.

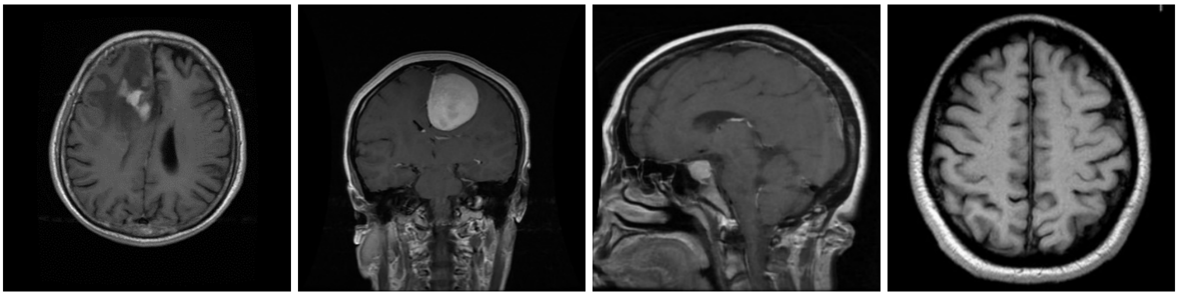


Fig. 1. Images of brain tumors from the Kaggle dataset: the first image on the left is an axial view of the brain with a *glioma*; the second image on the left is a coronal view with a *meningioma*; the third one is a sagittal view with a *pituitary* tumor; and the image on the right is a brain without tumor.

3.3. Experiments Setting

We use the `TensorFlow` and `Keras` libraries for the training of the neural networks, and the `scikit-learn` library to evaluate the models. The images are scaled down to 256×256 pixels with three channels. The range of the original pixel values is between 0 and 255, and we normalized the images between -1 and 1, dividing by 127.5 and subtracting 1. The experiments were conducted on an Intel Core i9-10940X CPU @3.30 GHz processor with 32 GB RAM, an NVIDIA GeForce RTX 2060 GPU with 8 GB RAM and an NVIDIA Geforce RTX 3060 GPU with 12 GB RAM, under Windows 10.

The models were trained using transfer learning with the weights corresponding to the ImageNet [8] dataset. We used the Adam optimizer with default parameters and early stopping with a patience parameter of 10. The maximum number of epochs was stablished to 50, although most executions finished before 25 epochs. For fine-tuning, we used the Adam optimizer with a learning rate of 10^{-4}, a decay rate of 10^{-5}, and 20 epochs. In this case, we unfroze the last 15 layers, comprising five convolutional layers.

During the experiments, we used several batch sizes, such as 32, 64, and 128, although we did not appreciate significant differences in accuracy but more stability during the optimization process.

We use one-hot encoding for coding the labels and categorical cross-entropy for the loss function. In the numerical results, we use the *accuracy*, *precision*, and *recall* metrics for comparing the methods, which are given by:

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN},$$

$$precision = \frac{TP}{TP+FP},$$

$$recall = \frac{TP}{TP+FN},$$

with *TP* the true positive values, *TN* the true negatives, *FP* the false positives, and *FN* the false negatives.

4. Results

The results for the Figshare dataset are given in Table 1. We observe that the best model is the EfficientNetB0, with an accuracy of 98.4 %, and a high precision and recall. The second-best method is EfficientNetB3, with a slightly inferior performance, and the third one is EfficientNetSmall.

Table 1. Results for the Figshare dataset.

Method	Accuracy	Precision	Recall
EfficientNetB0	98.4 %	98.1 %	98.0 %
EfficientNetB3	98.0 %	97.9 %	97.6 %
EfficientNetSmall	97.7 %	97.8 %	97.0 %

We observe a similar behavior when we use the Kaggle dataset; see Table 2. The accuracy is lower in this case for the three models. The best metrics are again for the EfficientNetB0 model. The training with this dataset is somehow more difficult as it has one more classification and image variety.

Table 2. Results for the Kaggle dataset.

Method	Accuracy	Precision	Recall
EfficientNetB0	97.5 %	97.3 %	97.6 %
EfficientNetB3	96.9 %	97.0 %	96.7 %
EfficientNetSmall	96.0 %	96.1 %	95.9 %

The number of parameters of the models is 6,2M for EfficientNetB0, 13,2M for EfficientNetB3, and 20,6M for EfficientNetSmall. The accuracy, precision, and recall are in general high for the three models. We note that the model with the lowest number of parameters provided the best accuracy in general. Fine-tuning, on the other hand, usually allowed to improve the accuracy by more than 2 %.

4.1. Precision per Tumor Type

In the following, we analyze the performance of the models with respect to each tumor type. In Figs. 2 and 3, we show the confusion matrices for the best model, *i.e.*, EfficientNetB0, with the Figshare and Kaggle datasets, respectively.

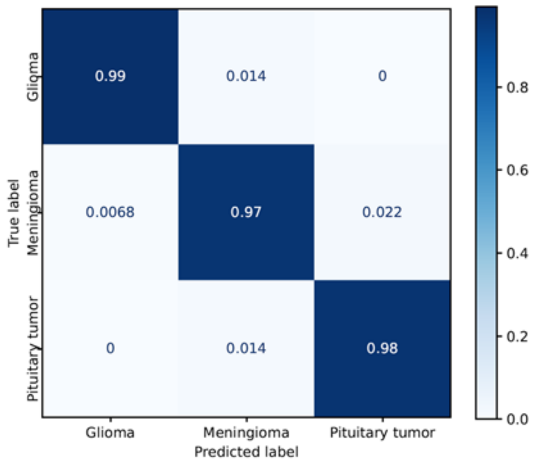


Fig. 2. Confusion matrix of the EfficientNetB0 model using the Figshare dataset.

In the case of the Figshare dataset, *gliomas* and *pituitary* tumors are correctly classified with high precision, with only one misclassification out of 146 for *gliomas* and two out of 87 for *pituitary* tumors. *Meningiomas* obtain the worse results with two errors out of 68 samples.

Looking at the confusion matrix corresponding to the Kaggle dataset, we observe that EfficientNetB0 classifies *pituitary* tumors with an accuracy of 100 %, whereas the *no tumor* class presents the worst results with an accuracy of 95 %. The performance for *gliomas* and *meningiomas* is similar, although we also observe that, for other experiments, *meningiomas* are slightly more difficult to classify than *gliomas*.

Tables 3 and 4 show the precision of the three methods for the three tumor types, the first one corresponding to the Figshare dataset and the second one to the Kaggle dataset. We observe that the precision for the *pituitary* tumor is the highest in general, followed by the *glioma* class. The *no-tumor* class also has a high classification rate for the Kaggle dataset. *Meningiomas* obtained the worst results in general.

Table 3. Precision of the methods with respect to each tumor type (Figshare dataset).

Method	Glioma	Meningioma	Pituitar
EfficientNetB0	99.3 %	97.1 %	97.8 %
EfficientNetB3	97.7 %	95.9 %	100 %
EfficientNetSm	97.3 %	97.1 %	98.9 %

Table 4. Precision of the methods with respect to each tumor type (Kaggle dataset).

Method	Glioma	Mening.	NoTum.	Pituit.
EfficientNetB0	97.0 %	97.1 %	95.1 %	100 %
EfficientNetB3	97.9 %	94.9 %	98.1 %	96.9 %
EfficientNetSm	98.9 %	88.5 %	98.1 %	98.9 %

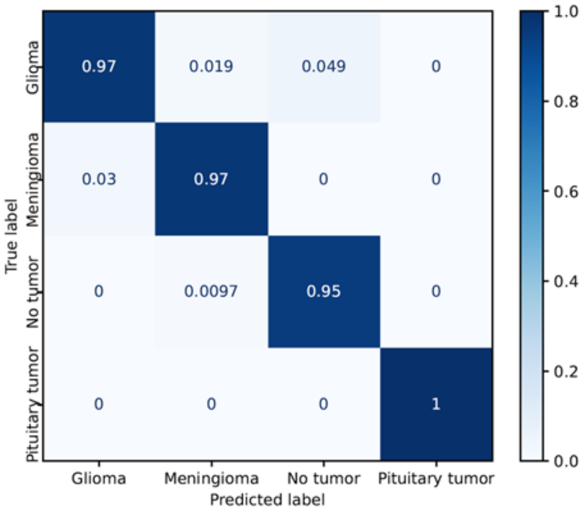


Fig. 3. Confusion matrix of the EfficientNetB0 model using the Kaggle dataset.

4.3. Comparison with State-of-the-art Methods

If we compare the results of the EfficientNet family with state-of-the-art methods, the models rank in the top of the classification.

Table 5 shows the best methods in the literature that have used the Figshare dataset. EfficientNetB0 ranks in the second position, whereas EfficientNetB3 ranks in the fourth position. The first work in the ranking [19] obtains a high precision with the ResNet model, only using transfer learning. We note that ResNet50 has 23,2M parameters, thus it is much larger than the EfficientNetB0 model. They also obtained good accuracy with other models, such as DenseNet and VGG, which are also much larger than EfficientNetB0. They add three fully connected layers on top of the network and analyze the performance using several optimizers. These models have also been analyzed in [18], also with transfer learning, although the results are not so accurate.

The third work [24] utilizes Vision Transformers for the classification. They obtain a slightly better result with an ensemble of Transformers. Nevertheless, they used an image resolution of 384×384 and the size of these networks is much bigger than EfficientNet.

In the case of the Kaggle dataset in Table 6, we observe that the EfficientNetB0 model provides the second-best result and EfficientNetB3 ranks in the third position. In this case, there are just a few works because this dataset is more recent.

The results presented in the first method [19], with an accuracy of 98.8 %, are, however, contradictory because it seems that they are using this dataset, but the number of images does not coincide, with more than 4000 images. On the other hand, it is not clear the configuration of the top layers they use after the CNN backbone and there are not enough details for reproducibility.

Table 5. Comparison with state-of-the-art methods using the Figshare dataset.

Method	Accuracy
Pashaei et al. [15]	93.7 %
Ayadi et al. [2]	94.7 %
Phaye et al. [16]	95.0 %
Ghassemi et al. [9]	95.6 %
Shaik et al. [20]	96.5 %
Badža et al. [3]	96.6 %
Kumar et al. [13]	97.1 %
EfficientNetSmall	**97.7 %**
Amin et al. [1]	98.0 %
Bodapati et al. [5]	98.0 %
EfficientNetB3	**98.0 %**
Tummala et al. [24]	98.2 %
EfficientNetB0	**98.4 %**
Polat et al. [17]	99.0 %

Table 6. Comparison with state-of-the-art methods using the Kaggle dataset.

Method	Accuracy
EfficientNetSmall	**96.0 %**
Hossain et al. [11]	96.5 %
Goutham et al. [10]	96.9 %
EfficientNetB3	**96.9 %**
EfficientNetB0	**97.5 %**
Saleh et al. [19]	98.8 %

5. Conclusion

This work has shown that EfficientNet can provide high accuracy for the classification of brain tumors. We selected two of the most important datasets and tried to search the best hyperparameters.

The best results for the two datasets were obtained with EfficientNetB0, which has the lowest number of parameters (6,2M). The second-best model was EfficientNetB3 with 13,2M parameters. This means that it is not necessary a neural network with a large capacity to deal with these datasets.

Transfer learning and fine-tuning are important for increasing the accuracy of the models. It is interesting to note that transfer learning was based on the ImageNet dataset, which is composed of natural images, very different from MR images. This denotes the generalization capacity of CNNs. On the other hand, fine-tuning allowed to further increase the accuracy by more than 2 %. The results that we have obtained in the experiments are on par with state-of-the-art methods, with the benefit that the EfficientNet models have less parameters.

Acknowledgements

This work was partly supported by the Servicio Canario de Salud, Gobierno de Canarias, under Contract F2022/03.

References

[1]. J. Amin, M. Sharif, N. Gul, M. Raza, M. A. Anjum, M. W. Nisar, S. A. C. Bukhari, Brain tumor detection by using stacked autoencoders in deep learning, *Journal of Medical Systems*, Vol. 44, Issue 2, 2020, pp. 1-12.

[2]. W. Ayadi, W. Elhamzi, I. Charfi, M. Atri, Deep CNN for brain tumor classification, *Neural Processing Letters*, Vol. 53, Issue 1, 2021, pp. 671-700.

[3]. M. Badža, M. Č. Barjaktarović, Classification of brain tumors from MRI images using a convolutional neural network, *Applied Sciences*, Vol. 10, Issue 6, 2020, 1999.

[4]. S. Bhuvaji, A. Kadam, P. Bhumkar, S. Dedge, S. Kanchan, Brain Tumor Classification (MRI), *Kaggle*, 2020.

[5]. J. D. Bodapati, N. S. Shaik, V. Naralasetti, N. B. Mundukur, Joint training of two-channel deep neural network for brain tumor classification, *Signal, Image and Video Processing*, Vol. 15, Issue 4, pp. 753-760, 2021.

[6]. J. Cheng, W. Huang, S. Cao, R. Yang, W. Yang, Z. Yun, Z. Wang, Q. Feng, Enhanced performance of brain tumor classification via tumor region augmentation and partition, *PLOS ONE*, Vol. 10, Issue 12, 2015, e0144479.

[7]. J. Cheng, Brain Tumor Dataset, *Figshare*, 2017.

[8]. J. Deng, W. Dong, R. Socher, L. Li, K. Li, L. Fei-Fei, ImageNet: A large-scale hierarchical image database, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'09)*, 2009, pp. 248-255.

[9]. N. Ghassemi, A. Shoeibi, M. Rouhani, Deep neural network with generative adversarial networks pre-training for brain tumor classification based on MR images, *Biomedical Signal Processing and Control*, Vol. 57, 2020, 101678.

[10]. V. Goutham, A. Sameerunnisa, S. Babu, T. B. Prakash, Brain tumor classification using EfficientNet-B0 model, in *Proceedings of the 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE'22)*, 2022, pp. 2503-2509.

[11]. S. Hossain, A. Chakrabarty, T. R. Gadekallu, M. Alazab, M. J. Piran, Vision transformers, ensemble model, and transfer learning leveraging explainable AI for brain tumor detection and classification, *IEEE Journal of Biomedical and Health Informatics*, 2023, pp. 1-14.

[12]. A. Krizhevsky, I. Sutskever, G. E Hinton, ImageNet classification with deep convolutional neural networks, in Advances in Neural Information Processing Systems (F. Pereira, C. J. Burges, L. Bottou, K. Q. Weinberger, Eds.), Vol. 25, *Curran Associates, Inc.*, 2012, pp. 1-9.

[13]. R. L. Kumar, J. Kakarla, B. V. Isunuri, M. Singh, Multi-class brain tumor classification using residual network and global average pooling, *Multimedia Tools and Applications*, Vol. 80, Issue 9, 2021, pp. 13429-13438.

[14]. S. D. Kumar, G. Kumar, S. Kumar, Multi-classification of brain MRI using EfficientNet, in *Proceedings of the International Conference for Advancement in Technology (ICONAT'22)*, 2022, pp. 1-6.

[15]. A. Pashaei, H. Sajedi, N. Jazayeri, Brain tumor classification via convolutional neural network and extreme learning machines, in *Proceedings of the 8th*

International Conference on Computer and Knowledge Engineering (ICCKE'18), 2018, pp. 314-319.

[16]. S. R. Phaye, A. Sikka, A. Dhall, D. R. Bathula, Dense and diverse capsule networks: Making the capsules learn better, *CoRR*, 2018, abs/1805.04001.

[17]. Ö. Polat, C. Güngen, Classification of brain tumors from MR images using deep transfer learning, *The Journal of Supercomputing*, Vol. 77, Issue 7, 2021, pp. 7236-7252.

[18]. D. Reyes, J. Sánchez, Performance of convolutional neural networks for the classification of brain tumors using magnetic resonance images, *Preprint*, 2023.

[19]. A. Saleh, R. Sukaik, S. S. Abu-Naser, Brain tumor classification using deep learning, in *Proceedings of the International Conference on Assistive and Rehabilitation Technologies (iCareTech'20)*, 2020, pp. 131-136.

[20]. N. S. Shaik, T. K. Cherukuri, Multi-level attention network: application to brain tumor classification, *Signal, Image and Video Processing*, Vol. 16, Issue 3, 2022, pp. 817-824.

[21]. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15)*, 2015, pp. 1-9.

[22]. M. Tan, Q. Le, EfficientNet: Rethinking model scaling for convolutional neural networks, in *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 6105-6114.

[23]. M. Tan, Q. Le, Efficientnetv2: Smaller models and faster training, in *Proceedings of the 38th International Conference on Machine Learning*, 2021, pp. 10096-10106.

[24] S. Tummala, S. Kadry, S. A. C. Bukhari, H. T. Rauf, Classification of brain tumor from magnetic resonance imaging using vision transformers ensembling, *Current Oncology*, Vol. 29, Issue 10, 2022, pp. 7498-7511.

[25]. Y. Yang, L. Yan, X. Zhang, Y. Han, H. Nan, Y. Hu, B. Hu, S. Yan, J. Zhang, D. Cheng, et al., Glioma grading on conventional MR images: A deep learning study with transfer learning. *Frontiers in Neuroscience*, Vol. 12, 2018, 804

.

(050)

A Robot-based Measurement Setup for Tactile Surface Classification

A. Jafari-Tabrizi, T. Ules, M. Grießer, D. Tranchida and D. P. Gruber
[1] Polymer Competence Center Leoben GmbH, Roseggerstrasse 12, 8700 Leoben, Austria
[2] Borealis Polyolefine GmbH, St.-Peter-Strasse 25, 4021 Linz, Austria
Tel.: +43 3842 429 62-715/707
E-mails: atae.jafari@pccl.at, thomas.ules@pccl.at

Summary: This work presents the preliminary results of an ongoing study on the classification of surfaces based on friction measurements. The investigated surfaces mainly differed in surface hardness, giving rise to differences in their tribological properties. The data used for the classification were force signals collected from an artificial fingertip mounted on a collaborative robot's end-effector, while sliding along the respective surface in contact mode. A deep learning-based time series classification method, namely InceptionTime, has been used for classification. This methodology allowed to differentiate the investigated surfaces with accuracies higher than 90 % which exceeded the classification accuracy achieved by a human test panel. The presented measurement setup is also equipped with a vibration sensor to capture the mechanical vibrations elicited in a finger pad upon the dynamic exploration of textured surfaces that can further improve the classification accuracy and lead to a holistic description of surface haptics in general. This paves the way to the implementation of an automatic in-line robot-based surface classification system into a production line.

Keywords: Time series classification, Tactile sensing, Artificial fingertip, Deep learning, Collaborative robots.

---

1. Introduction

With developments in the field of manufacturing high-end components, the need for sophisticated quality inspection techniques is increasing. In such cases traditional manual inspection strategies cannot meet the requirements. Advances in machine learning, robotics, and constant increases in the computational power of computers provide a solution for this challenge by introducing new ways of inspection and classification techniques. Currently, vision-based inspection is the standard methodology for automatic surface quality inspection in the manufacturing environment [1]. However, haptic properties of an end-product can be as important as its visual appearance. Therefore, a robot-based haptic inspection system, which can automatically inspect the surface characteristics of end products can make an important contribution to the production of high-quality components.

The setup introduced in this contribution consists of an artificial fingertip that is mounted on a collaborative robot's end effector. Five different samples were investigated to assess the feasibility of this method. For this, the artificial fingertip was slid along the respective surface. During sliding motion normal and friction forces were recorded and consequently used to train a deep learning algorithm for time series classification. This method reached classification accuracies higher than 90 %. However, optimizations and improvements are necessary both in software and mechanical aspects of the setup.

2. Experimental Setup

The setup consists of a collaborative robot (UR5e), a 3D-printed bracket mounted on its end-effector, holding a 3-axis force sensor (ME K3D40) to which an artificial fingertip replicating the human finger is attached. The fingertip is made of silicone with an additive (silicone deadener) to adjust the elasticity of the material. To reproduce the relatively hard uppermost skin layer, the stratum corneum, a water-based acrylic varnish was also applied to the fully cured finger.

For validation, 5 injection molded polypropylene (PP) plates were used. The dimensions of the surfaces were identical; however, their friction properties were different, due to the variations in their materials and resulting differences in surface hardness [2]. The artificial fingertip was slid along the surfaces at a constant speed of 2 cm/s holding a constant normal force of 0.5 N (F_z). The inspection setup along with the coordinate system can be seen in Fig. 1.

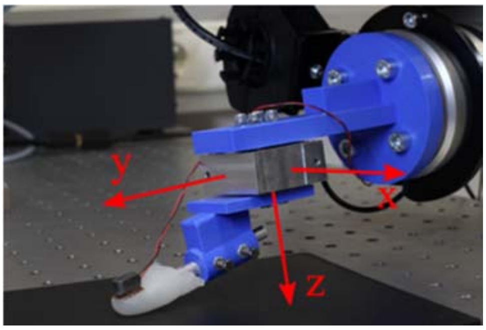


Fig. 1. Inspection setup.

3. Data Collection and Analysis

To classify the collected temporal data, a deep learning-based time series classification (TSC) method called InceptionTime was used [3]. Only recently deep learning algorithms for TSC have gained popularity [4]. InceptionTime is an ensemble of five deep learning networks, namely Inception networks. Inception network, a deep convolutional neural network architecture, was first introduced in [5] for image classification. The data collection process was divided into three cycles since the artificial fingertip showed some signs of damage during longer measurement periods that were reflected in the collected data. At each cycle the fingertip was slid 6 times on each sample. Thus, at the end there were 18 measurements from each surface. To train and test the networks, the time series of the coefficient of friction were calculated by dividing the friction force F_y by the normal force F_z (see the coordinate system shown in Fig. 1).

Ten ensemble networks were trained to classify each sample against all others. Out of the 18 collected measurements from each sample, 9 measurements were used for network training, and the rest for validation. For the selection of the training and validation data, the effect of wearing of the fingertip was considered.

4. Results

The performance of the networks was quantified by the degree of their accuracy with the test data. The validation accuracies are presented in Table 1. The plates used during the experiments are designated as S1 – S5. As it can be seen, the networks were able to successfully classify the surfaces with accuracies higher than 90 % based on the temporal data of the coefficient of friction.

Table 1. Accuracy of the trained networks in classifying the respective samples after training.

Network	Accuracy
S1 vs. S2	1.0
S1 vs. S3	0.92
S1 vs. S4	1.0
S1 vs. S5	0.92
S2 vs. S3	1.0
S2 vs. S4	0.92
S2 vs. S5	1.0
S3 vs. S4	1.0
S3 vs. S5	0.92
S4 vs. S5	0.94

5. Discussion and Conclusion

This methodology allowed to differentiate the investigated surfaces with accuracies higher than 90 %. For some of the classifications the accuracy was 100 %. This can be attributed to the relatively strong differences between the surface characteristics of the samples. Overall, the classification accuracy reached by this method clearly excelled the one obtained by an expert sensory study conducted with 7 test subjects.

This study presented a method of surface classification based on friction data collected by an artificial fingertip mounted on a collaborative robot. The results showed that friction data can be used for surface classification. Future studies will focus on the optimization of parameters like sliding speed, normal force and tilt angle of the fingertip with respect to the surface to further enhance the quality of the collected data regarding classification accuracy. In addition to the force data, it is also planned to include other sensory data channels like vibration and temperature for characterization. The results presented in this manuscript will also be compared with those obtained using other state-of-the-art time series classification methods.

Acknowledgements

The research work was performed within the COMET-project „Deep on-line learning for highly adaptable polymer surface inspection systems" (project-no.: 879785) at the Polymer Competence Center Leoben GmbH (PCCL, Austria) within the framework of the COMET-program of the Federal Ministry for Climate Action, Environment, Energy, Mobility, Innovation and Technology and the Federal Ministry for Digital and Economic Affairs. The PCCL is funded by the Austrian Government and the State Governments of Styria, Lower Austria and Upper Austria.

References

[1]. M. Babic, et al., Image based quality inspection in smart manufacturing systems: A literature review, *Procedia CIRP*, Vol. 103, 2021, pp. 262-267.

[2]. S. Derler, L-C. Gerhardt, Tribology of skin, review and analysis of experimental results for the friction coefficient of human skin, *Tribology Letters*, Vol. 45, 2012, pp. 1-27.

[3]. I. Fawaz, et al., InceptionTime: Finding AlexNet for time series classification, *Data Mining and Knowledge Discovery*, Vol. 34, 2020. pp. 1936-1962.

[4]. J. Faouzi, Time Series Classification: A review of Algorithms and Implementations. *Ketan Kotecha. Machine Learning (Emerging Trends and Applications)*, 2022. pp. 1-34, in press.

[5]. C. Szegedy, et al., Going deeper with convolutions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15)*, Boston, MA, USA, 2015, pp. 1-9.

(051)

Iterative Inversion of 2D Electrical Resistivity Tomography Data to Remove 3D Effects

L. Zanzi [1] **and A. Hojat** [1,2]
[1] Dipartimento di Ingegneria Civile e Ambientale, Politecnico di Milano, Milan, Italy
[2] Department of Mining Engineering, Shahid Bahonar University of Kerman, Kerman, Iran
Tel.: + 390223994271
E-mails: luigi.zanzi@polimi.it, azadeh.hojat@polimi.it, ahojat@uk.ac.ir

---

Summary: Electrical Resistivity Tomography (ERT) is a geophysical technique that is rapidly gaining popularity as a long-term monitoring method of hydrogeological risks such as landslides or floods. When applied to monitor embankments or levees, the tomographic image represents a 2D section of the earthen structure along its main axis but the inversion results are highly affected by the 3D effects generated by the particular 3D geometry of the structure. We propose an iterative 3D correction plus 2D inversion procedure as a more reliable alternative to direct 3D inversion that would be poorly constrained. Results on synthetic tests demonstrate that the procedure is effective. The method has been successfully applied to laboratory controlled experiments and to real data from a couple of ERT monitoring stations in operation along river levees. We are currently investigating the sensitivity of the convergence of the procedure to the complexity of the earthen structure.

Keywords: Electrical resistivity tomography, 2D/3D inversion, 3D effects, Dam, Embankment, River levees, Tailings dam.

---

1. Introduction

Electrical Resistivity Tomography (ERT) method is widely used in various disciplines to solve a variety of problems in groundwater studies, mineral explorations, subsurface characterization, mapping contaminations, and, in recent decade, long-term monitoring of hydrogeological risks [e.g., 1-4]. An ERT survey aims at determining the distribution of the electrical resistivity in the subsurface. In order to measure each data point for an ERT survey (either in a 2D survey along a line or a 3D survey in a grid of usually parallel lines), a current (*I*) is passed into the ground through two metal electrodes and the potential difference is measured at two potential electrodes (ΔV). The apparent resistivity (ρ_a) is obtained by multiplying the geometric factor of the electrode configuration (*K*) by the ratio of the potential difference to the injected current:

$$\rho_a = K\frac{\Delta V}{I} \tag{1}$$

By increasing the geometric factor, i.e., by increasing the electrode spacing, the measurement depth also increases and the resulting raw dataset represents the apparent resistivity distribution within a 2D vertical section or within a 3D volume of the subsoil. Raw data must be inverted to transform the apparent resistivity, which is a sort of average value within a large subvolume, into the real resistivity of each cell of the tomographic grid. The inversion is intrinsically non-linear and normally ill-conditioned [5, 6].

In recent decade, we have developed important research on applying different geophysical methods to monitor hydrogeological risks for earthen embankments, landslides, and rock slopes [e.g., 2, 4, 7-9]. Among different techniques and thanks to the development of multi-electrode systems a couple of decades ago [3] and rapid technological improvements in the last decade, ERT surveys became an efficient part of long-term monitoring strategies to mitigate hydrogeological risks. In this paper, we specifically address ERT monitoring systems installed along embankment structures like dams, tailings dams and river levees [e.g., 4, 7, 9-12]. There has been a continuous global demand to assess the stability of such earthen embankments that are considered as the main and final barrier against flooding, especially in extreme rainfalls triggered by climate changes. In response to this increasing demand, customized and low-cost ERT monitoring systems can be permanently installed along the critical sections of the earthen embankment [7]. Time-lapse analysis of the data measured by these ERT monitoring systems can detect any anomalous inhomogeneity such as water infiltration or cavities well in advance [2] making it possible to perform rapid mitigation efforts to prevent a probable failure of the embankment.

To achieve the desired goals in integrating an ERT monitoring system with early warning alarm strategies, reliable data processing algorithms must be employed. In this paper, we discuss the 3D effects as one of the main challenging data processing steps when 2D ERT profiles come to monitor an earthen embankment.

2. 3D Effects Distorting 2D ERT Data

An ERT system applied to monitor embankments is normally deployed as a long cable of electrodes parallel to the main axis of the structure (Fig. 1). The resulting tomographic image is expected to represent the resistivity distribution within a 2D section of the embankment showing resistivity distribution along the profile as well as in the vertical direction. However, the measured apparent resistivity values and accordingly the inversion results are highly affected by the 3D effects generated by the particular 3D geometry of the embankment structure (Fig. 1).

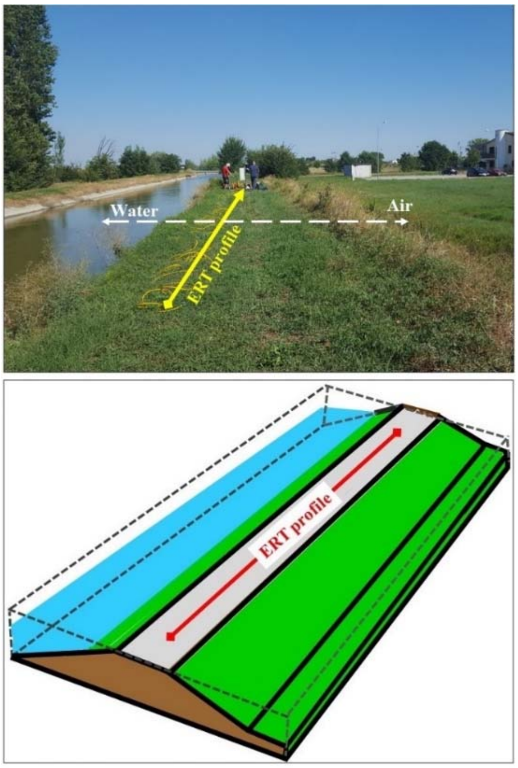


Fig. 1. Illustration of non-uniform 3D geometry of earthen embankments that affects 2D ERT data measured along a linear profile.

Air and water volumes, which are respectively highly resistive and conductive bodies, present on the two sides of the embankment affect the intrinsic 3D current distribution and the resulting potential fields and will thus influence the final result. Therefore, in order to obtain reliable results from 2D inversion programs that assume that resistivity does not change in the direction orthogonal to the ERT profile, it is necessary to quantify the 3D effects and to remove these effects from the measured ERT data before 2D inversion. It should be mentioned that 3D corrections vary whenever the boundary conditions of the embankment are changed, i.e., level of water/air on two sides of the embankment. Moreover, a priori information about resistivity distribution in the embankment body during different seasons can improve 3D corrections [12].

3. Correction of 3D Effects

In principle, 3D inversion would be necessary to handle the problem but a 3D inversion would be very poorly constrained due to the 2D geometry of the acquisition system. To prevent inversion instabilities, we propose an iterative 3D correction + 2D inversion procedure. The data corrected for the 3D effects ($\rho_a^{c_j}$) are obtained dividing the measured data (ρ_a^M) by correction factors (α_j) calculated as the ratio of the apparent resistivities simulated on a 3D model ($\rho_a^{3D_j}$), which includes the a priori information about the water level in the river, to the apparent resistivities simulated on the corresponding 2D model ($\rho_a^{2D_j}$):

$$\rho_a^{c_j} = \frac{\rho_a^M}{\alpha_j}, \tag{2}$$

with

$$\alpha_j = \frac{\rho_a^{3D_j}}{\rho_a^{2D_j}}, \tag{3}$$

where j is the iteration number.

To start the iterative procedure an initial model of the embankment is required. Thus, the iterative procedure to update the models and the corrections consists of the following steps:

1. Initial corrections based on 3D and 2D forward modelling on homogeneous models of the embankment (initial models), characterized by the resistivity measured with the smallest electrode aperture (to minimize the lateral effects);
2. 2D inversion of the corrected dataset;
3. Design of new 3D and 2D models with resistivity distribution within the embankment body based on the inversion result (previous step);
4. 3D and 2D forward modelling on the updated models (previous step);
5. Corrections based on the 3D and 2D forward modelling (previous step);
6. 2D inversion of the corrected dataset;
7. Convergence test;
8. Iterations of steps 3 to 7 until satisfying the convergence test.

We have developed the proposed correction procedure to process the ERT data obtained in our different projects where we have permanently installed an ERT monitoring system along an earthen embankment. The proposed correction strategy was tested on several synthetic models as well as laboratory data measured in a few controlled experiments. In this paper, we present the results of one of these tests to explain the main aspects of the proposed method.

We consider an ERT system that uses the Wenner electrode array, which is a symmetric configuration with equal distances between successive electrodes. The vertical section of the embankment body cutted along the longitudinal axis is defined as a simplified model with three layers (Fig. 2a): 0.5 m of superficial gravel with the resistivity of 400 Am, a homogeneous embankment with the height of 4 m and the resistivity of 50 Am, and the subsoil with the resistivity of

20 Am. The top gravel layer simulates the typical unpaved road that is often running along the levees (assumed 5 m wide in the 3D model of this example). The main body of the embankment is assumed to consist of clayey limes, much more conductive than gravel. The natural soil below the embankment may consists of similar clayey limes with lower resistivities assuming that they are fully saturated. The cross-section of the 3D embankment model is a typical trapezoidal section (as shown in Fig. 1) assuming slope angles of about 33°. A dry season is considered for the boundary conditions of the embankment and thus, highly resistive air is assumed to be present on both lateral sides. ERT measurements are simulated for a profile using 30 electrodes with unit electrode spacing of 2 m.

In order to have a reference for 3D corrections, we constructed the defined model in Res3dmod and Res2dmod forward modelling software [12] using which, we calculated the 3D effects on the synthetic data for an ERT line along the main axis of the embankment. The synthetic data calculated by Res3dmod software are actually representing the real measured data that are affected by the 3D geometry of the embankment while the synthetic data calculated by Res2dmod software are representing the measured data after the application of the true corrections of the 3D effects. We may call these corrections the ideal corrections of this example. The objective of the iterative procedure is to produce corrections that rapidly converge to the ideal corrections.

Using the proposed iterative procedure to correct 3D effects and assuming that we do not have any a priori information about the internal structure of the embankment body, we defined a homogeneous embankment with the resistivity value of 70 Am (Fig. 2b) selected as an average value measured with the smallest electrode aperture. 3D corrections were calculated for this homogeneous embankment and corrections were applied to the measured data, i.e., the synthetic data calculated by Res3dmod software for an ERT line along the main axis of the reference three-layered model (Fig. 2a). The corrected data were then inverted in Res2dinvx64 inversion software [12]. The inverted model obtained at this step (Fig. 3, middle) was used to improve the embankment model required for calculating 3D effects. Based on the inverse section, a three-layered model (Fig. 2c) was defined: a 1.5 m-thick superficial layer with the resistivity of 80 Am, a layer down to 5m with the resistivity of 40 Am, and the subsoil with the resistivity of 25 Am. 3D corrections were then calculated for this new model and corrections were applied again to the measured data, i.e., the synthetic data calculated by Res3dmod software for an ERT line along the main axis of the reference three-layered model (Fig. 2a).

Fig. 3 demonstrates the results of inverting the real data corrected for 3D effects after the ideal correction (top image), after initial corrections calculated for a homogeneous model (middle image), and after corrections from the first iteration (bottom image). By comparing the middle and bottom images with the top image we can appreciate the improved similarity produced by the refined corrections.

To better quantify the improvement, Fig. 4 shows the correction factors as a function of electrode aperture after ideal corrections, after initial corrections and after the first iteration. It is really interesting to observe the rapid convergence after only one iteration starting from a homogeneous model.

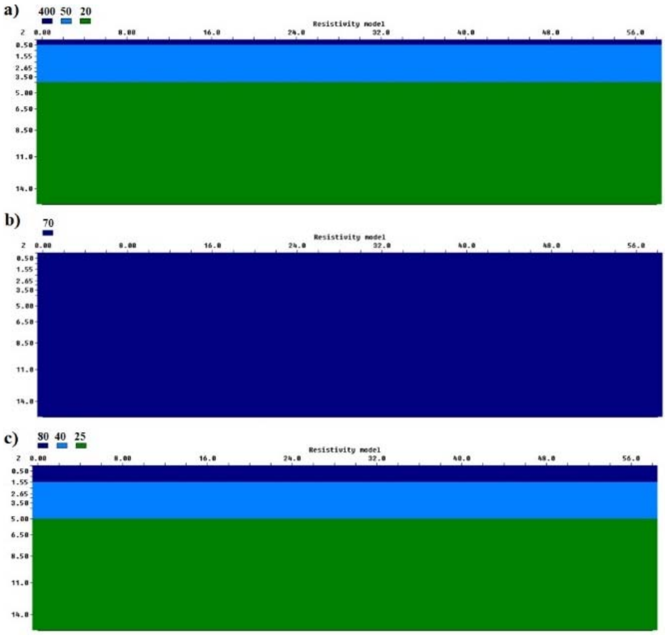


Fig. 2. 2D models used to calculate the corrections: a) the real model for calculating the ideal corrections, b) a homogeneous model for calculating the initial corrections, c) the simplified layered model derived from the inversion at the first iteration (Fig. 3 middle). The corresponding 3D models are obtained by extrapolating laterally these models and assuming the presence of air on both sides of the levee.

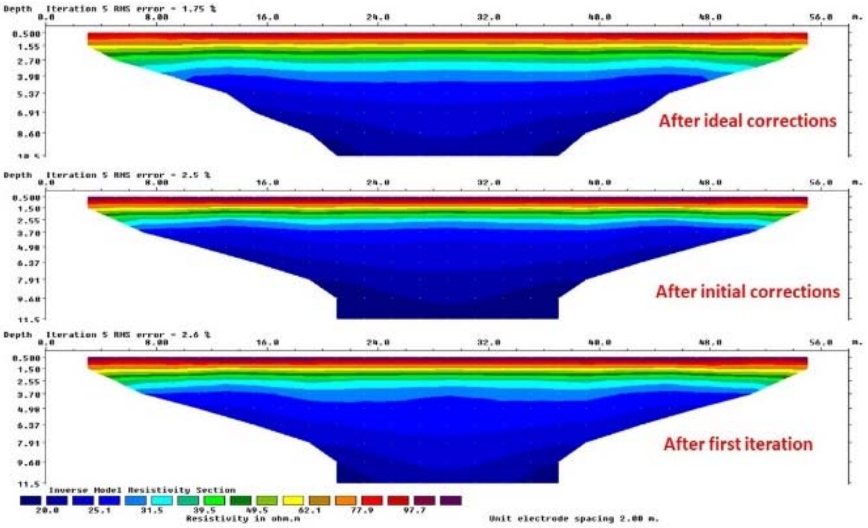


Fig. 3. Inversion results after ideal corrections (top), after initial corrections (middle) and after the first-iteration corrections of the real data (i.e., the synthetic data calculated by Res3dmod over the 3D version of the model shown in Fig. 2a).

Having obtained promising results, the procedure is being applied to datasets measured by the ERT monitoring systems that are permanently installed along earthen embankments, mainly river levees in Italy. Research is in progress and we are currently exploring how rapid the convergence is as a function of the complexity of the embankment structure using synthetic models.

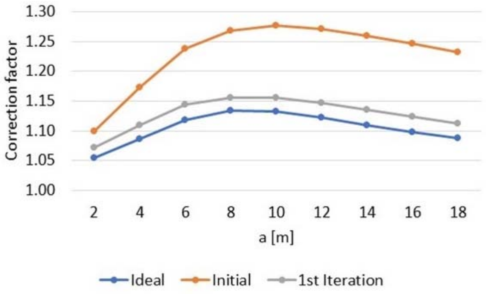


Fig. 4. Correction factor as a function of electrode aperture after ideal corrections, after initial corrections and after the first-iteration corrections.

4. Conclusions

ERT monitoring systems installed along the crest of dams or river levees generate 2D tomographic sections severely affected by 3D effects due to lateral water and air bodies. We demonstrated that the inverse problem can be successfully handled preserving the reliability and stability of the solution by running a procedure where 3D-effects corrections and 2D inversion are iteratively applied until convergence is reached. Synthetic tests and laboratory controlled experiments demonstrate the rapid convergence to the correct solution when the earth structure is relatively simple. We are currently testing the procedure on more complex models to explore how much the convergence is sensitive to the inhomogeneities of the embankment.

Acknowledgements

We would like to thank Dr. Meng Heng Loke for his valuable comments on quantifying 3D effects as well as guides on using different geotomo software. ERT monitoring systems in Italy are developed by technical support of LSI Lastem s.r.l. with the scientific support of Politecnico di Milano. Inversion of ERT data in Res2dinvx64 was generated using Seequent Software (Copyright © Seequent Systems, Incorporated).

References

[1]. M. Moradipour, H. Ranjbar, A. Hojat, S. Karimi-Nasab, S. Daneshpajouh, Laboratory and field measurements of electrical resistivity to study heap leaching pad no. 3 at Sarcheshmeh copper mine, in *Proceedings of the 22nd European Meeting of Environmental and Engineering Geophysics*, Barcelona, Spain, 4-8 September 2016, pp. 1-5.

[2]. A. Hojat, M. Ferrario, D. Arosio, M. Brunero, V. I. Ivanov, L. Longoni, A. Madaschi, M. Papini, G. Tresoldi, L. Zanzi, Laboratory studies using electrical resistivity tomography and fiber optic techniques to detect seepage zones in river embankments, *Geosciences*, Vol. 11, Issue 69, 2021, pp. 1-15.

[3]. M. H. Loke, J. E. Chambers, D. F. Rucker, O. Kuras, P. B. Wilkinson, Recent developments in the direct-current geoelectrical imaging method, *Journal of Applied Geophysics*, Vol. 95, 2013, pp. 135-156.

[4]. G. Tresoldi, A. Hojat, L. Zanzi, G.RE.T.A. installations for real-time monitoring of irrigation dams and canals, *Procedia Environmental Science, Engineering and Management*, Vol. 7, Issue 2, 2020, pp. 271-276.

[5]. M. Aleardi, A. Vinciguerra, A. Hojat, A convolutional neural network approach to electrical resistivity tomography, *Journal of Applied Geophysics*, Vol. 193, 2021, 104434.

[6]. M. Aleardi, A. Vinciguerra, A. Hojat, A geostatistical Markov Chain Monte Carlo inversion algorithm for

electric resistivity tomography, *Near Surface Geophysics*, Vol. 19, Issue 1, 2021, pp. 7-26.

[7]. A. Hojat, D. Arosio, L. Longoni, M. Papini, G. Tresoldi, L. Zanzi, Installation and validation of a customized resistivity system for permanent monitoring of a river embankment, in *Proceedings of the EAGE-GSM 2nd Asia Pacific Meeting on Near Surface Geoscience & Engineering*, Kuala Lumpur, Malaysia, 24-25 April 2019, pp. 1-5.

[8]. Z. Zhang, D. Arosio, A. Hojat, L. Zanzi, Tomographic experiments for defining the 3D velocity model of an unstable rock slope to support microseismic event interpretation, *Geosciences*, Vol. 10, Issue 327, 2020, pp. 1-23.

[8]. Z. Zhang, D. Arosio, A. Hojat, L. Zanzi, Reclassification of microseismic events through hypocentre location: Case study on an unstable rock face in Northern Italy, *Geosciences*, Vol. 11, Issue 37, 2021, pp. 1-20.

[9]. J. E. Chambers, D. A. Gunn, P. B. Wilkinson, P. I. Meldrum, E. Haslam, S. Holyoake, M. Kirkham, O. Kuras, A. Merritt, J. Wragg, 4D electrical resistivity tomography monitoring of soil moisture dynamics in an operational railway embankment, *Near Surface Geophysics*, Vol. 12, 2014, pp. 61-72.

[10]. P. Sjodahl, T. Dahlin, B. Zhou, 2.5D resistivity modeling of embankment dams to assess influence from geometry and material properties, *Geophysics*, Vol. 71, Issue 3, 2006, pp. G107-G114.

[11]. G. Tresoldi, D. Arosio, A. Hojat, L. Longoni, M. Papini, L. Zanzi, Tech-Levee-Watch: experimenting an integrated geophysical system for stability assessment of levees, *Rendiconti Online Della Società Geologica Italiana*, Vol. 46, 2018, pp. 38-43.

[12]. A. Hojat, D. Arosio, V. I. Ivanov, M. H. Loke, L. Longoni, M. Papini, G. Tresoldi, L. Zanzi, Quantifying seasonal 3D effects for a permanent electrical resistivity tomography monitoring system along the embankment of an irrigation canal, *Near Surface Geophysics*, Vol. 18, 2020, pp. 427-443.

[13]. Geotomosoft Web Portal, http://geotomosoft.com, www.aarhusgeosoftware.dk

(052)

The Effect of Stopwords Removal and Feature Engineering on Analysing the Sentiment of Air-traveller

M. S. Homaid [1,2], **I. Moulitsas** [1,2] **and K. W. Jenkins** [1]
[1] Cranfield University, Department of Computational Engineering Sciences,
MK43 0AL, Cranfield, United Kingdom
[2] Machine Learning and Data Analytics Laboratory, Digital Aviation Research
and Technology Centre (DARTeC)
E-mail: m.homaid@cranfield.ac.uk

Summary: Using sentiment analysis and machine learning to assess the level of airport service quality has received a lot of interest recently. Web-based opinion platforms such as Google reviews and SKYTRAX receive a large number of reviews and can effectively complement traditional service quality surveys. These data may however contain noise that does not add any value for the analysis. In this paper, we have employed six well-established supervised sentiment classification algorithms. In our work, we have studied how stopwords removal can reduce the noise existing in the data and improve the performance of the sentiment analysis in the domain of airport service quality. To this end, we have used the pre-compiled stopwords list of NLTK to study how the stopwords removal affects the supervised sentiment classification algorithms. We have applied pre-processing and feature engineering techniques and we have shown how effective stopwords removal significantly enhances the performance of the sentiment classifiers.

Keywords: Sentiment analysis, Stopwords, Machine learning, Natural language processing, Airport service quality.

1. Introduction

Air travel is the most common way of travelling around the world these days. Billions of people travel across airports annually [1]. Therefore, airport managements strive to deliver best services for travelers. Developing an effective tool to evaluate the quality of airport services has always been a critical issue for all involved stakeholders. There are a number of online opinion platforms, such as Google reviews [2] and SKYTRAX [3], which allow travelers to rate their experience, thus creating valuable review platforms.

In this paper, the data of SKYTRAX is used to assess the travelers' sentiments for the domain of Airport Service Quality (ASQ). Data was prepared employing several techniques which include stopwords removal and feature engineering. The impact of stopwords removal may vary depending on the context. Six well-established classification algorithms are utilised to analyse the sentiment of air-travelers. The primary objective of our work is to investigate the impact of stopwords removal on these classification techniques and propose such appropriate pre-processing techniques within the context of ASQ.

2. Related Work

2.1. Sentiment Analysis of ASQ

Several studies have assessed the quality of airport services using a variety of sentiment analysis techniques. A study conducted by [4] evaluated the level of airport service quality by employing AFINN sentiment lexicon. They utilised the data from Google reviews and compared the results with the well-measured service attributes of Airports Council International (ACI). The results of the study exhibited a high correlation between them. In [5], the authors applied conventional machine learning algorithms to analyse the sentiment of travelers. The dataset was collected from SKYTRAX and Twitter for analysis. The AdaBoost approach was developed which integrates the boosting and bagging ensemble. The model yields an average test accuracy of 78 %. Another study conducted by [6] used five different machine learning algorithms to analyse the sentiment of air travelers. The dataset was also obtained from the SKYTRAX website. The results indicated that XGBoost produced the best results across all the performance measures with an accuracy of 88 %. In [7] data from Twitter was used, with a focus on the analysis of six major US airlines. The authors carried out a multi-class sentiment analysis to gauge passenger satisfaction. They implemented seven different machine learning classifiers, among which random forest emerged as the most effective algorithm, yielding an F1 score of 86 %. Twitter data was also employed in study [8] where the authors developed two different deep learning models for sentiment analysis, namely CNN and LSTM. Results showed that LSTM produced better results, attaining accuracy and F1 score of 77 %. In [9] the authors implemented an analysis of more than 12000 tweets using an ensemble sentiment classification strategy based on the Majority Vote principle. This involved the use of various classification methods, including Naive Bayes, SVM, Bayesian Network, C4.5 Decision Tree, and Random Forest algorithms. Additionally, an ensemble approach was proposed and trained on the same dataset. The results demonstrated that the proposed ensemble approach surpassed the performance of individual classifiers, achieving an F1 score of 84 %. In [10], the authors compared two types of sentiment analysis techniques, lexicon-based and conventional machine learning sentiment analysis on the complete data of SKYTRAX. The results indicated that conventional machine learning algorithms outperformed the VADER sentiment lexicon. Logistic regression achieved an accuracy of 87 % while VADER provided poor accuracy of around 59 %.

2.2. Stopwords Removal

One of the main issues with generated data is that noise is naturally inherent in the textual data. For example, irregular expressions and frequent words can not only increase the computational time, but also hinder the performance of the sentiment classifiers. However, there is no certainty as to how stopwords removal can affect the performance of the analysis. While some works support the stopwords removal [11, 12], others show that stopwords still carry useful information for the analysis and help the classifiers to provide better results [13, 14]. This can vary depending on the context of the analysis. In [15], the authors leveraged the classical Van stoplist [16] to analyse the sentiment of Twitter. The results showed that using the classical stopwords removal has a negative effect on the performance of Twitter sentiment classification approaches.

3. Materials and Methods

3.1. Dataset Description

To study the effect of stopwords removal for the context of ASQ, we utilised the data of SKYTRAX. It comprised a total number of 38104 reviews. These reviews are labelled as recommended (yes) and not recommended (no). The dataset also consists of multiple features for the airport services and reviewers have the ability to rate these services. The dataset includes 25.19 % positive reviews and 74.18 % negative, spanning the period between July 2004 to November 2020.

3.2. Data Preparation

Several pre-processing steps were implemented in the context of this work. They involved lower-case, handling missing values, and replacing emojis with meaningful text. Furthermore, an essential step was required to use the selected machine learning classifiers. We applied the Term-Frequency Inverse-Document-Frequency (TF-IDF) to convert the review text into a numerical format. Along with the aforementioned steps, we also employed feature engineering on the dataset. Having studied the dataset in detail, we discovered that there are specific characteristics for each class. New features were added to achieve better results. These include the number of words in each review – usually frustrated people write longer reviews, the number of question marks – unhappy reviewers tend to use many question marks to pose a question, and the number of words with capital letters. We also applied the so-called "*max_features*" technique on the dataset, where the maximum number of features is specified, i.e., unique words or terms, to consider when creating the TF-IDF representation of a document. This can be useful when it comes to preventing overfitting and reducing computational complexity [17]. The value of *max_features* is typically set to a number that is lower than the total number of unique words in the corpus. We have tuned the value of *max_features* to 500 out of possible values, 100, 500, 1000, 2000, 3000 etc.

3.3. Stopwords

Stopwords are common words in a corpus that usually carry little semantic meaning and do not provide useful information for the analysis. These words usually fulfil grammatical functions such as connecting phrases or expressing possession. In natural language processing (NLP), stopwords are often removed during the pre-processing step to lower the data size, increase computational performance and concentrate on the most important words. By removing stopwords, machine learning algorithms can better identify important topics or keywords. However, as discussed earlier, there is no guarantee that removing stopwords will be advantageous. There are situations were keeping stopwords is necessary to comprehend a text's context or meaning, and this depends on the domain of the analysis.

In this paper, we have exploited the pre-compiled stopwords list of the Natural Language ToolKit (NLKT) for the English language [18]. To measure the efficiency of removing the stopwords, the non-removal of stopwords is identified as a baseline method.

3.4. Classification Methods

In this subsection, we outline the six different machine learning classifiers we used to assess the effect of the stopwords removal for analysing the sentiment of air-traveler. Namely, these are Logistic Regression (LR), Gradient Boosting (GB), Passive Aggressive (PA), Random Forest (RF), Perceptron and finally Multi-Layer Perceptron (MLP). These classifiers are trained and employed on the pre-processed dataset. The dataset ratio for training purposes is 70 % with the remaining 30 % used for testing purposes. The testing data was used to predict passengers' sentiments after having trained the classifiers' models. Then, these predictions were used to assess the performance of the models using the measures of accuracy, recall, precision and F1-score.

3.4.1. Logistic Regression

Logistic regression (LR) serves as a classification approach in machine learning, intending to determine the probability of a categorical dependent variable. The dependent variable in LR is binary, consisting of data encoded as 1 (YES) or 0 (NO). Within sentiment classification, logistic regression forecasts the sentiment group by utilizing an input of a word frequency integer vector [19].

3.4.2. Gradient Boosting

The gradient-boosted (GB) decision tree implementation is designed to enhance both performance and speed. It has also demonstrated effectiveness in handling imbalanced datasets. GB primarily relies on an ensemble of decision tree algorithms, employing a gradient-boosting framework [20].

3.4.3. Passive Aggressive

Passive aggressive (PA) algorithms are online learning techniques designed for binary classification tasks. They respond passively to correct classifications and aggressively to errors. The training process in PA algorithms occurs incrementally, with instances being introduced individually, in sequence, or in small clusters called mini batches. One of the primary benefits of online learning algorithms, such as PA, is continuous learning. As a result, PA classifiers are highly suitable for systems that process data in a continuous stream [21].

3.4.4. Random Forest

Random forest (RF) is a classification algorithm that employs an ensemble learning approach. It primarily operates by generating decision trees, each of which is built using random samples from the dataset. The output from the ensemble of trees is then combined to produce the final classification prediction. RF has been utilised across various applications and domains, including the medical field [22].

3.4.5. Perceptron

Perceptron is a supervised machine learning algorithm designed for binary classification and is regarded as one of the most basic forms of artificial neural networks. The algorithm iteratively processes data until the loss reaches an acceptably low level. To yield accurate predictions, the model's accuracy must surpass a specified threshold [23].

3.4.6. Multi-Layer Perceptron (MLP)

MLP is a neural network that consists of multiple perceptrons arranged in layers. Each layer's perceptrons are connected with those in other layers. The goal is to identify the most suitable connection weights and biases to minimise the prediction error [24].

Algorithm 1 demonstrates the proposed approach to extract passengers' sentiments.

Algorithm 1: Pseudocode of proposed experiments

Input: SKYTRAX reviews

Output: Classification results

```
Begin
Data preparation & cleaning
    Feature engineering
    Stopwords removal
    TF-IDF
Data splitting
Model building
Classification results
Evaluation of results
End
```

3.5. Performance Metrics

To evaluate the performance of our models, the task was formulated into a classification problem to utilise the evaluation metrics of accuracy, recall, precision and F1-score. These evaluation metrics are defined as follows:

Accuracy is defined as how close measurements to a certain value [25].

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{True Positives} + \text{True Negatives} + \text{False Positives} + \text{False Negatives}} \quad (1)$$

Accuracy on its own is not sufficient and should be supported with the concepts of Recall and Precision. The aim of Recall is to identify how many elements of actual positive were correctly detected [25].

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (2)$$

While Recall offers an evaluation of the model's accuracy for positive classes, Precision is employed to identify the ratio of correct positive predictions [26].

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (3)$$

The F1-score is utilised to assess the accuracy and aims to find a balance between Recall and Precision [25].

$$\text{F1 score} = 2\,\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

4. Results & Discussion

To analyse the effect of the stopwords in the domain of ASQ, we removed the NLTK stopwords list and evaluated how they affect the performance of the classifiers' classification. Table 1 illustrates the results for each one of the six models.

It is notable that the stopwords removal affects the performance positively in the PA, Perceptron and MLP. The accuracy improved by 11 %, 1 % and 56 %, respectively when compared to the baseline. To better comprehend the results, we moved towards the remaining performance measures, precision, recall and F1-score. When removing stopwords, PA, Perceptron and MLP classifiers show more improvements in identifying positive reviews which resulted in higher precision values when compared to the baseline. While in F1-score, the results indicate that the harmonic balance between precision and recall remains stable even after removing stopwords. This means that the model's performance is maintained, despite the reduction in the feature set.

Table 1. Performance of the models.

Model	Accuracy		Recall		Precision		F1-Score	
	Baseline	NLTK	Baseline	NLTK	Baseline	NLTK	Baseline	NLTK
PA	71 %	82 %	78 %	66 %	71 %	83 %	69 %	69 %
Perceptron	25 %	81 %	50 %	65 %	63 %	79 %	20 %	67 %
MLP	83 %	84 %	80 %	79 %	78 %	79 %	79 %	79 %
GB	88 %	88 %	83 %	83 %	84 %	84 %	84 %	83 %
RF	89 %	89 %	83 %	83 %	86 %	86 %	84 %	84 %
LR	86 %	86 %	79 %	80 %	82 %	82 %	80 %	81 %

It appears that stopwords hinder and misguide the performance of these models. The number of noisy words in our dataset is very frequent. Removing them helps these algorithms to only focus on the more important words and phrases that are more likely related to the domain. This can help these sentiment classification algorithms to process and analyse the data as they have fewer features to consider. Furthermore, these stopwords can cause misclassification as the algorithms may focus on these words rather than the more informative ones. Therefore, stopwords removal improves the performance of these algorithms by reducing the chances of misclassification. Also, stopwords removal can decrease the training time which leads to computational efficiency. On the other hand, the highest accuracy achieved was with the RF and GB classifiers. However, the removal of stopwords has almost no effect on these classifiers. We attribute this to the nature of tree-based algorithms such as RF and GB. We believe stopwords removal may have no effect on the performance of these models because they are designed to learn the most relevant features from the data, whether stopwords have been removed or not. It is also considered beneficial even if the performance metrics remain the same. By removing such frequent words in the dataset, faster training times can be achieved making the model more practical for real-world applications.

In Table 2, we provide a comparison between our results and previous research efforts, focusing on the F1 score achieved and the data used. The main purpose of this research was to analyse the effect of stopwords removal in the context of ASQ and our results show better or comparable performance with some of the most effective models in the field. This is attributed to the effective pre-processing and feature engineering steps proposed in our approach and data analysis.

Table 2. Comparative Analysis with previous studies.

Author/ Reference	Used dataset	Best Approach	F1-score
Prabhakar *et al.* [5]	SKYTRAX & Twitter	AdaBoost	68 %
Rane [7]	Twitter	RF	86 %
Barakat *et al.* [8]	Twitter	LSTM	77 %
Wan [9]	Twitter	Ensemble Classification	84 %
Our proposed approach	SKYTRAX	RF	84 %

5. Conclusions

In this paper, the main contribution is to investigate the implication of stopwords removal and the employed pre-processing techniques on the sentiment classification within the domain of ASQ. Six different machine learning classifiers were used on data from SKYTRAX spanning a timeline of sixteen years. We evaluated the results using the metrics of accuracy, recall, precision, and the F1-score. Our findings indicate that our pre-processing tasks can boost the overall performance of the selected classifiers. In particular, stopwords removal can positively affect the performance of the PA, Perceptron and MLP classifiers. Furthermore, the highest accuracy achieved was observed when using the RF and GB classifiers.

References

[1]. The World Bank. Air Transport, Passengers Carried, https://data.worldbank.org/indicator/IS.AIR.PSGR?end = 2018&start = 1970

[2]. Google Maps, https://www.google.com/maps/@21.5848387,38.9160206,10z

[3]. Skytrax. A-Z Airport Reviews, https://www.airlinequality.com/review-pages/a-z-airport-reviews/

[4]. K. Lee, C. Yu, Assessment of airport service quality: A complementary approach to measure perceived service quality based on Google reviews, *Journal of Air Transport Management*, Vol. 71, 2018, pp. 28-44.

[5]. E. Prabhakar, M. Santhosh, A. H. Krishnan, T. Kumar, R. Sudhakar, Sentiment analysis of US airline twitter data using new Adaboost approach, *International Journal of Engineering Research & Technology (IJERT)*, Vol. 7, Issue 1, 2019, pp. 1-6.

[6]. M. S. Homaid, I. Moulitsas, Measuring airport service quality using machine learning algorithms, in *Proceedings of the 6th International Conference on Advances in Artificial Intelligence (ICAAI'22)*, Birmingham, United Kingdom, 21-23 October 2022, pp. 8-14.

[7]. A. Rane, Sentiment classification system of Twitter data for US airline service analysis, in *Proceedings of the IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC'18)*, Tokyo, Japan, 23-27 July 2018, Vol. 1. pp. 769-773.

[8]. H. Barakat, R. Yeniterzi, L. Martín-Domingo, Applying deep learning models to twitter data to detect airport service quality, *Journal of Air Transport Management*, Vol. 91, 2021, 102003.

[9]. Y. Wan, Q. Gao, An ensemble sentiment classification system of Twitter data for airline services analysis, in *Proceedings of the 15th IEEE International Conference on Data Mining Workshop (ICDMW'15)*, Atlantic City, NJ, USA, 14-17 November 2015, pp. 1318-1325.

[10]. M. S. Homaid, D. B. Bisandu, I. Moulitsas, K. Jenkins, Analysing the sentiment of air-traveller: A comparative analysis, *International Journal of Computer Theory and Engineering*, Vol. 14, Issue 2, 2022, pp. 48-53.

[11]. A. Bakliwal, P. Arora, S. Madhappan, N. Kapre, M. Singh, V. Varma, Mining sentiments from tweets, in *Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis*, Jeju, Republic of Korea, 12 July 2012, pp. 11-18.

[12]. A. Pak, P. Paroubek, Twitter as a corpus for sentiment analysis and opinion mining, in *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May 2010, pp. 1320-1326.

[13]. H. Saif, Y. He, H. Alani, Semantic sentiment analysis of twitter, in *Proceedings of the 11th International Semantic Web Conference (Web–ISWC'12)*, Boston, MA, USA, 11-15 November 2012, pp. 508-524.

[14]. E. Martínez-Cámara, A. Montejo-Ráez, M. T. Martín-Valdivia, L. A. Ureña-López, SINAI: Machine learning and emotion of the crowd for sentiment analysis in microblogs, in *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval'13)*, Atlanta, Georgia, 14-15 June 2013, pp. 402-407.

[15]. H. Saif, M. Fernandez, Y. He, H. Alani, On stopwords, filtering and data sparsity for sentiment analysis of twitter, in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, 26-31 May 2014, pp. 810-817.

[16]. C. Van Rijsbergen, Information retrieval: Theory and practice, in *Proceedings of the Joint IBM/University of Newcastle upon Tyne Seminar on Data Base Systems*, Callaghan, Australia, 4-7 September 1979, pp. 1-4.

[17]. A. Heemann, D. Velinova, M. Gastegger, Textklassification – Decision Trees & Random Forests, 2017. https://www.researchgate.net/publication/320384121_Textklassification_-_Decision_Trees_Random_Forests

[18]. S. Bird, NLTK: The natural language toolkit, in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia, 17-21 July 2006, pp. 69-72.

[19]. M. A. Jenike, M. S. Albert, The elements of statistical learning data mining, inference, and prediction, *Journal of the American Geriatrics Society*, Vol. 32, Issue 6, 1984, pp. 441-444.

[20]. D. A. Al-Qudah, A. M. Al-Zoubi, P. A. Castillo-Valdivieso, H. Faris, Sentiment analysis for e-payment service providers using evolutionary extreme gradient boosting, *IEEE Access*, Vol. 8, 2020, pp. 189930-189944.

[21]. M. Popek, E. Swiatek-Najwer, M. Majak, J. Jaworowski, D. Szram, Online passive-aggressive algorithms, *Journal of Machine Learning Research*, Vol. 7, 2006, pp. 331-336.

[22]. L. Breiman, Random forests, *Machine Learning*, Vol. 45, 2001, pp. 5-32.

[23]. N. Aston, J. Liddle, W. Hu, Twitter sentiment in data streams with perceptron, *Journal of Computer and Communications*, Vol. 2, Issue 3, 2014, pp. 11-16.

[24]. S. K. Pal, S. Mitra, Multilayer perceptron, fuzzy sets, and classification, *IEEE Transactions on Neural Networks*, Vol. 3, Issue 5, 1992, pp. 683-697.

[25]. J. Brownlee, Machine learning mastery with Python: Understand your data, create accurate models, and work projects end-to-end, in Machine Learning *Mastery*, *Independently published*, 2016.

[26]. M. Buckland, F. Gey, The relationship between recall and precision, *Journal of the American Society for Information Science*, Vol. 45, Issue 1, 1994, pp. 12-19.

5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2023),
7-9 June 2023, Tenerife (Canary Islands), Spain

(053)

Feature Extraction of VLF Radio Signal for Solar Flares Detection

N. Christoff, I. Nachev, I. Iliev and P. Petkov
Technical University of Sofia, Faculty of Telecommunications,
8 Kl. Ohrisdski blvd, 1756 Sofia, Bulgaria
E-mail: nicole.christoff@tu-sofia.bg

Summary: A solar flare can be identified along a radio wave characteristic alteration of the route of the VLF radio trace, which is defined by the coordinates between VLFTX and SDRRX, using the data presented here. Thus, it has become possible to record a solar flare of class C or a higher level. Our data is obtained from open network software-defined radios (SDR) receivers, and although the devices are located in different geographical territories, they all capture information from the same transmitters. Analyzing the data in this manner, it is possible to ascertain whether or not the solar flare has had an effect on the region of the Earth that is between the transmitter and the receiver. The current research not only enables precise localization of the solar flare, but it also can allow to carry out an in-depth investigation into the way the nature of super long waves shifts when they come into contact with the ionosphere, as well as the impact that the ionization that takes place in the ionosphere during a solar flare has on those waves.

Keywords: Feature extraction, D-region, VLF, Solar flare, SDR, SID, Radio astronomy.

1. Introduction

Radio astronomy encompasses a broad spectrum of investigations into celestial objects, with a focus on studying solar phenomena. Solar flares, in particular, are of great interest, and researchers are actively studying their classification, techniques for capturing them, and methods for processing the data gathered. These flares are characterized by the sudden release of magnetic energy stored in the sun's atmosphere, resulting in intense conditions and emissions that span the electromagnetic spectrum. From super-long wavelengths to high-frequency X-rays and gamma rays, solar flares emit radiation across the entire electromagnetic spectrum [1].

Solar flares are forceful eruptions that emanate from the surface of the sun and release enormous amounts of electromagnetic radiation, charged particles, and magnetic fields into space. The classification of these flares is determined by their X-ray flux, which is a measure of the energy they emit. Solar flares are classified into five categories – A, B, C, M, and X – based on their peak flux, which is generally measured in watts per square meter (W/m2). The X-ray classes are divided into nine subdivisions on a logarithmic scale, from B1 to B9, C1 to C9, and so on. An X2 flare is twice as powerful as an X1 flare and four times more potent than an M5 flare. The X-class is a distinct category that extends beyond X9. Flares with strengths of X10 or greater are occasionally referred to as Super X-class solar flares [1, 2].

Solar flares can cause disruptions in the Earth's ionosphere, which can lead to a phenomenon called Sudden Ionospheric Disturbances (SID). SIDs occur when the intense X-ray and ultraviolet radiation from a solar flare ionize the Earth's upper atmosphere, causing a temporary increase in the density of electrons in the ionosphere. This can affect radio communication signals, particularly those in the lower frequency range. The severity of SIDs is influenced by the strength of the solar flare that initiated them, as well as the time of day and the Earth's location. SIDs usually endure for short intervals, ranging from a few minutes to a few hours [3, 4].

Solar flares and SIDs have a big impact on very low frequency (VLF) radio waves. SIDs, caused by solar flares, can generate ionospheric disturbances that lead to the absorption and scattering of VLF radio waves, affecting communication systems in these frequency bands. Additionally, solar flares can create radio emissions in the VLF range that can interfere with VLF radio communication. Therefore, monitoring solar activity is crucial to predict and mitigate any adverse effects on VLF radio communication.

VLF SID monitoring systems play a critical role in detecting and measuring the impact of SIDs on communication systems. These systems utilize VLF radio waves to measure changes in the ionosphere caused by SIDs, while also taking into account the Earth's ionosphere and its waveguide effect on VLF signal [3].

During SIDs, the ionosphere undergoes significant disturbances that can cause VLF waves to experience attenuation, phase shifting, and dispersion. By analyzing changes in VLF wave propagation caused by SIDs, VLF monitoring systems can provide valuable insights into the state of the ionosphere and its response to solar activity. This information can be used to improve ionospheric models and prediction methods, leading to more accurate forecasts of space weather impacts on communication systems. VLF SID monitoring systems can also provide crucial data for research on space weather and ionospheric dynamics, making them an essential tool in understanding and mitigating the impacts of solar disturbances on communication systems. Therefore, the development and implementation of VLF SID monitoring systems that take into account the ionosphere-earth waveguide effect are crucial for ensuring reliable and robust communication systems in the face of solar disturbances [3-5].

In the literature, the effects of solar flares on the D-layer are often referred to as SID. These disturbances occur when the ionization/plasma density in the D region of the ionosphere becomes abnormally high due to a solar flare. The resulting increase in radio wave absorption is most severe in the VHF, UHF, HF, and VFL frequency bands and can disrupt or interfere with telecommunications systems. Attenuations are sudden and can last for minutes or hours [3].

One consequence of this ionospheric disturbance is a change in the level of the VLF wave signal. Scientists can utilize this enhancement to detect solar flares by monitoring the signal strength of a remote VLF transmitter. SIDs are recorded and indicate the occurrence of a solar flare. The slight geomagnetic effect in the lower ionosphere appears as a small hook on magnetic records and is commonly known as the "geomagnetic crochet effect" or "sudden field effect" [5, 6].

It is possible to create a public SDR network operating in such frequency ranges, including the frequency bands in which various VLF transmitters operate. Such a monitoring system could be optimized for real-time data recording and processing.

The article proceeds as follows: Section 2 presents a comprehensive overview of the current state-of-the-art in solar flare detection and feature extraction techniques aimed at enhancing VLF signal analysis and solar flare detection. Section 3 presents the algorithm, which comprises three main steps, and the attained outcomes. Section 4 refers to the conclusion and future work.

2. State-of-the-arts

Many different kinds of technology may detect solar flares, including the following:

– Optical telescopes using specialized filters to observe specific wavelengths of light emitted during a solar flare. They can detect visible light and the ultraviolet wavelengths that the Sun emits during an eruption. Telescopes collect light and direct it to a detector that captures an image of the eruption. Captured images can be analyzed to study flare behavior and the mechanisms behind it [7].

– X-ray detectors are carried on satellites and space probes to observe X-rays emitted by the Sun during a flare. X-rays are emitted at high energies during flares and can penetrate Earth's atmosphere, making them ideal for space-based detection. X-ray detectors collect and measure the energy of the X-rays emitted by the Sun during a flare [8].

– Magnetometers detect changes in the Earth's magnetic field caused by a solar flare. Solar flares can cause disturbances in the Earth's magnetic field, which can be detected by magnetometers placed around the world. These changes can be used to study the flare's magnetic properties and its impact on the Earth [9].

– Ultraviolet detectors are also carried on satellites and space probes to observe the ultraviolet light emitted by the Sun during a flare. Ultraviolet detectors are designed to capture specific wavelengths of light and can be used to study the ionized gases in the Sun's atmosphere during a flare [10].

– Radio telescopes detect radio waves emitted by the Sun during an eruption. These waves can be observed at longer wavelengths than visible light and ultraviolet waves detected by optical telescopes. Radio telescopes can be used to study the Sun's magnetic fields, which are responsible for producing solar flares. This type of telescope included the technique for detecting solar flares by detecting spontaneous ionospheric distributions by monitoring radio signals. This type of technology allows us to deepen the knowledge and techniques for analyzing the impact of radio waves on the ionosphere, in addition to what is described for finding solar flares [11].

Radio telescopes have several advantages over other technologies for solar flare detection. One advantage is that they can detect solar flares in all weather conditions, as radio waves can penetrate through clouds and Earth's atmosphere. In contrast, optical telescopes, ultraviolet detectors, and X-ray detectors are all affected by weather conditions or blocked by Earth's atmosphere.

Another advantage of radio telescopes is that they can detect flares in real-time, providing researchers with immediate data on solar activity. Optical telescopes, ultraviolet detectors, and X-ray detectors often require data processing before the flare can be detected, which can delay the analysis of solar activity.

Radio telescopes can also study the magnetic fields of the Sun during a flare. Radio waves produced by the magnetic fields of the Sun during a flare can be detected by radio telescopes. This allows researchers to better understand the magnetic fields and how they influence solar flares.

Radio telescopes can detect solar flares with lower energy levels compared to X-ray detectors. This allows for a wider range of solar flares to be studied and a better understanding of solar activity. Additionally, radio telescopes can be used for long-term monitoring of solar activity, allowing for changes in solar activity to be observed and better understood. This can be useful for predicting solar activity and its potential impacts on Earth's systems.

The VLD SID monitoring system in the general case consists of a type of sensor-loop antenna, a preamplifier (this item is highly recommended but not required), a receiver that can receive the frequency of the VLF band, a computer used to control the receiver and to record and analyze the data and software in order to record and analyze the data. One popular software package for this purpose is the "Spectrum Lab" software or Cambridge SUPER SID software, but it is also possible to use specially made software that has specific data storage. The software will typically display the recorded data in a graphical format, allowing you to analyze the effects of solar flares on the Earth's ionosphere.

Typically, for a receiving device, many are developed using an audio card that provides lower sensitivity than a dedicated software-defined radio. This means that they can detect weaker signals, and provide a clearer and more accurate picture of the signals they are receiving. VLF signals can be extremely weak, so this is a key advantage when it comes to monitoring them. SDRs also typically have a higher signal-to-noise ratio and higher dynamic range than audio cards, which means that they can detect signals that would be lost in the noise when using an audio card which simplified signal processing. In summary, the advantages of using an SDR with a high-impedance input for VLF SID monitoring include better sensitivity, wider frequency range, higher dynamic range, flexibility, and better noise performance. These advantages can result in clearer, more accurate signals, and more detailed analysis of the signals being monitored.

Recent advances in feature extraction methods have improved the accuracy of VLF radio signal analysis for solar flare detection. Verma et al. [12] proposed a wavelet feature extraction method for characterizing VLF waves, which has been shown to improve efficiency. George et al. [13] used linear regression analysis to calculate the peak flux from VLF amplitude, which has demonstrated promising results for nowcasting X-class solar flares. Lotz and Clilverd [14] used nonlinear smoothers to detect phenomena in the narrow band VLF transmission phase produced by perturbation events in the D region of the ionosphere. Nardi et al. [15] proposed an automatic detection system using LSTM neural networks to classify relevant VLF patterns for seismically active Earth crust deformation monitoring. These studies collectively demonstrate the potential of feature extraction techniques for improving VLF signal analysis and solar flare detection.

3. Proposed Method

Both the sunrise and the sunset may be defined and extracted as features. We take the event with the maximum intensity, which allows us to define a solar flare as a possible occurrence.

3.1. Data Collection

We employ information that was recorded using an open system of software-defined radios (SDRs) (see Fig. 1a). They function within the frequency range of 0 MHz to 30 MHz, including the very low frequency (VLF) band. The receivers are positioned around the landscape at various geographical coordinates. This opens up the opportunity to receive signals and make recordings from a single or more VLF transmitters located in separate territories. The system operates in real-time, and the data was collected throughout the period covering the last several months.

For the purpose of this article, we gathered data from the receiving station Germany Dhoffo Frankfurt on 15.11.2022 and saved it as a CSV file. You can see this data in Fig. 1a. For the aim of carrying out visual data validation, the use of space weather is chosen (see Fig. 1b).

3.2. Data Preprocessing

Due to the independence of the system, the data can come in a variety of levels, and there is no alternative to overcome hardware issues. There is nothing that can be done about a malfunctioning receiver until it is powered back on and its functionality is restored. As a direct consequence of this, it is likely to obtain an empty report. It is also possible the appearance of noise. To guarantee the correctness of the data, we look for gaps in the period that the data covers.

Finding date/time gaps

First, we read the CSV file, gathered the field names from the first row, and then extracted each data row individually. This allowed us to locate the temporal gaps. We determined the difference between each consecutive time. Then, the difference in times was expressed in minutes. Since the software is supposed to do a data registration once every five minutes, the gap value was set to 6 minutes. To locate such gaps, we first determined the beginning and ending times. For all differences, bigger than six minutes, we produced several date/time values in the missing range. We made a new time frame that includes the absent rows. We included the rows that were previously absent from the Time Frame. We let the measurement be presented without any values in the areas where it should be presented.

Noise filtration

A signal's noise, and more specifically its impulse noise or spikes, can be removed with the help of a median filter. We took the median value of the samples in a sliding window and use it over the data measurements. The structural representation of a median filter with a window size of N can be expressed by the following equation:

$$y[n] = median(x[n-(N-1)/2], x[n-(N-3)/2],\dots, x[n+(N-3)/2], x[n+(N-1)/2]) \tag{1}$$

The output sample at time n is denoted as $y[n]$, while the input sample at time n is represented by $x[n]$. The median function is utilized to determine the median value of the samples within the sliding window. In the present article, the value of N has been assigned as 3. It is noteworthy that empty positions are excluded from the filtration process.

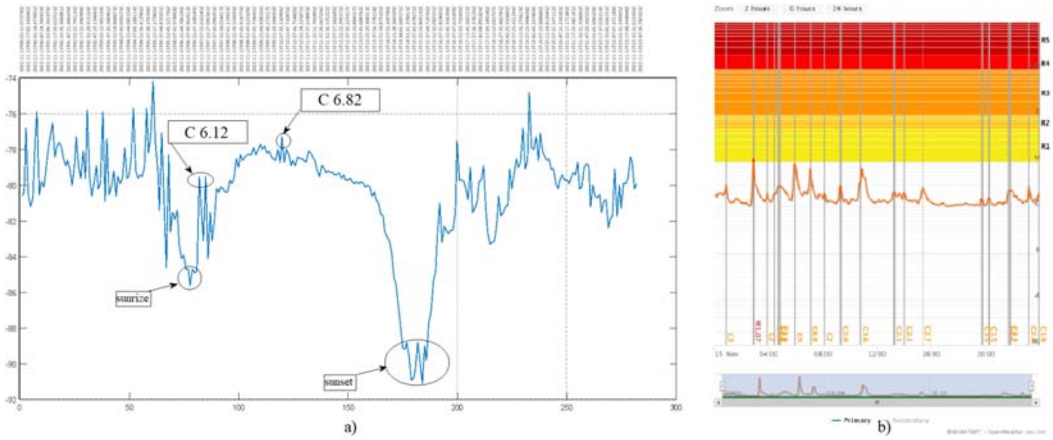


Fig. 1. Solar activity: a) Example from Germany_Dhoffo_Frankfurt_26_7_dBm on 15.11.2022, b) from GOES X-ray satellite on 15.11.2022.

Data interpolation

Data interpolation refers to the technique of approximating absent values within a given dataset by predicting the values that exist between established data points. The utilization of a prevalent method in signal processing and data analysis is imperative when faced with the challenge of missing or irregularly sampled data, which may obstruct further analysis. For the current article, we used polynomial interpolation (Eq. (3)) of second order and we also apply linear interpolation (Eq. (2)) for comparison.

The linear interpolation between two points (x_0, y_0) and (x_1, y_1), where x is the independent variable and y is the dependent variable, can be expressed as:

$$y = y_0 + (y_1 - y_0) * ((x - x_0) / (x_1 - x_0)) \qquad (2)$$

The polynomial function of order n, given a set of $n+1$ data points (x_0, y_0), (x_1, y_1),..., (x_n, y_n), and can be expressed as:

$$y = a_0 + a_1 * x + a_2 * x^2 + ... + a_n * x^n, \qquad (3)$$

where the coefficients $a_0, a_1,..., a_n$ are determined by solving a system of linear equations that ensure the function passes through all of the data points.

Linear interpolation and polynomial interpolation differ in their underlying assumptions regarding the relationship between data points. Specifically, linear interpolation assumes a linear relationship between data points, whereas polynomial interpolation is capable of capturing more complex relationships. Polynomial interpolation, even a widely used technique, is susceptible to overfitting and may yield unpredictable outcomes when extrapolating beyond the established range of the available data points. Linear interpolation, on the other hand, maybe less accurate but is more stable and produces smoother results. The outcome after the data interpolation is displayed on the left side of Fig. 2. The signal that has undergone applied linear interpolation can be seen in blue, while the polynomial interpolation is represented in orange.

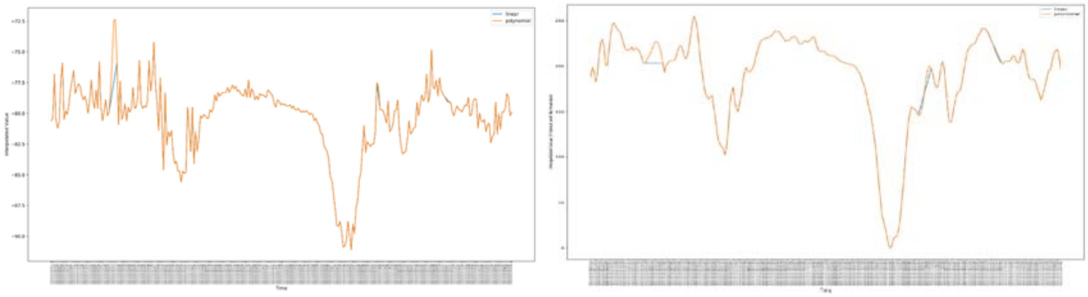


Fig. 2. Data preparation. On the left: found and filled gaps on the data Germany_Dhoffo_Frankfurt_26_7_dBm on 15.11.2022, On the right: Noise filtered, smoothed, and normalized interpolated data.

Data smoothing

The Savitzky-Golay filter is a common technique employed for data smoothing. It operates by fitting a polynomial to a sliding window of data and subsequently utilizing the polynomial to approximate the smoothed value for each data point. This approach proves to be particularly advantageous when dealing with data that exhibits a high degree of noise, yet possesses noteworthy characteristics, such as peaks or troughs, that necessitate preservation.

The formula for the Savitzky-Golay filter can be written as follows:

$$y_k = \sum_{i=-m}^{m} c_i x_{k+i} \tag{4}$$

The smoothed value at the k point is denoted by y_k, while x_{k+i} represents the data points in the neighborhood of k. The filter coefficients c_i are dependent on the window length 2m + 1, and the degree of the polynomial fit. We used the Savitzky-Golay filter, where *m* is equal to *3* and the polynomial is from 3 order.

3.2. Data Normalization

The process of data normalization, which is also referred to as data scaling, involves the standardization of the value range within a given dataset. The reasoning behind this practice is to ensure that all features or variables are given equal significance or weight during the process of data analysis or modeling. Enhanced precision can be achieved by standardizing the range of values in a dataset, thereby facilitating more accurate data analysis or modeling. It occurs since features with higher magnitudes do not exert a disproportionate influence on the analysis or modeling procedures. Enhanced interpretability can be achieved by the normalization of data can potentially facilitate comprehension, as it ensures uniformity in the range of values across various features or variables.

We normalized the smoothed interpolated signal in the range [0, 255] using the minimum-maximum scaling method:

$$X_{STD} = \frac{X - X.min}{X.max - X.min}, \tag{5}$$

$$X_{norm} = (X_{STD}.(max - \min) + \min).255, \tag{6}$$

X represents a group of data samples that should be normalized, *X.min* represents the minimum value in this group of data, and *X.max* represents the maximum value in this group of data. Where *max* and *min* represent the range to which we scale the data. This technique is used to scale and alter each feature individually so that the final value corresponds to a value range of 0 to 255. Fig. 2 (right) displays the outcome after the application of noise filtration, smoothing, and normalization techniques to the interpolated data.

3.3. Feature Extraction

The process of identifying, modifying, and consolidating the initial data to generate significant and informative characteristics that could be further employed in a machine learning algorithm is referred to as feature extraction. The primary objective of feature extraction is to convert unprocessed data into significant attributes of the data while eliminating insignificant or superfluous information. The selection of the feature extraction methodology is contingent upon the characteristics of the data and the particular issue under consideration.

Then, it uses a for loop to slide a window over the signal, with a fixed window size and stride length. The range for the loop is determined by the length of the signal minus the window size, plus one, with a step size of the stride length. The minimum values are stored in a list, and the maximum values are stored in an array.

We use a discrete signal of a given solar activity, which consists of 12 samples each hour and has a length of 288 samples to perform measurements over 24 hours. Using a sliding window of size 24, we will first find the two minimum values, which stand for sunrise and sunset. We found 122 minimal values. We chose only the two minimum values with the highest amplitude. The next step is to locate the local maximum to establish a solar flare as a potential event. We specified a condition to identify local maxima. In this case, the local maxima are defined as values greater than their neighboring points. The indices of the maximum values are used to plot red dots on the original data to indicate the maximum positions.

The result of our method is presented in Fig. 3. We successfully extract the two points, which refer to the start and the end of the day. We obtained 30 maximal values, which can be referred to as picks of real solar activities in the given moments, which can be validated after visual inspection.

4. Conclusions

The field of radio astronomy encompasses a wide range of research activities aimed at exploring celestial entities, with a particular emphasis on analyzing solar events. Solar flares have garnered significant attention from the scientific community, with ongoing efforts to investigate their categorization, capture, methodologies, and data processing techniques. In the current article, we propose a method of solar flares extraction using the VLF system that was stated is a lot more cost-effective than traditional registration methods.

In future work, we intend to enhance the method, to classify the solar flares and sustain the present algorithm, which will function as the input of a deep neural network architecture. This will help in dealing with a limitation of the present implementation, namely the predetermined dimensions of the sliding windows. Another limitation is that the use of polynomial interpolation renders our model susceptible to noise in the data, thereby reducing its ability to generalize effectively to new data.

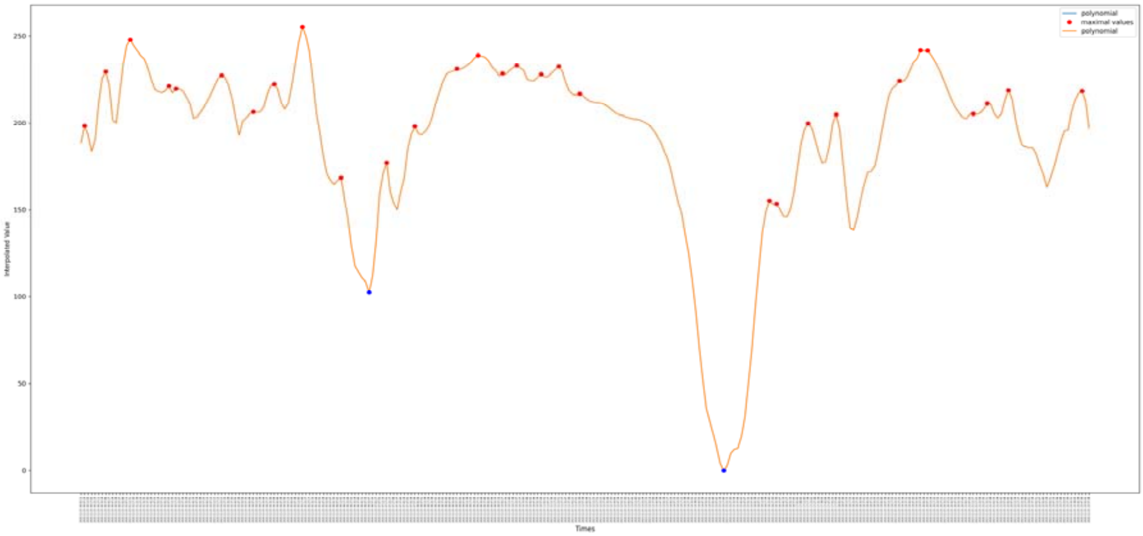


Fig. 3. Feature extraction. The start/end of the day is marked with blue dots and potential solar flares with red dots.

Acknowledgements

This article is supported by Horizon 2020 program, Scientific and Technological Excellence by Leveraging LOFAR Advancements in Radio Astronomy (STELLAR) project, Grant ID: 952439.

References

[1]. S. M. Hamdi, D. Kempton, R. Ma, S. F. Boubrahimi, R. A. Angryk, A time series classification-based approach for solar flare prediction, in *Proceedings of the IEEE International Conference on Big Data (Big Data'17)*, Boston, MA, USA, 2017, pp. 2543-2551.

[2]. D. P. Grubor, D. M. Šulić, V. Žigman, Classification of X-ray solar flares regarding their effects on the lower ionosphere electron density profile, *Ann. Geophys.*, Vol. 26, Issue 7, 2008, pp. 1731-1740.

[3]. W. P. Wah, M. Abdullah, A. M. Hasbi, S. A. Bahari, Development of a VLF receiver system for Sudden Ionospheric Disturbances (SID) detection, in *Proceedings of the IEEE Asia-Pacific Conference on Applied Electromagnetics (APACE'12)*, Melaka, Malaysia, 2012, pp. 98-103.

[4]. A. C. Moral, E. C. Kalafatoglu Eyiguler, Z. Kaymaz, Sudden Ionospheric Disturbances and their detection over Istanbul, in *Proceedings of the 6th International Conference on Recent Advances in Space Technologies (RAST'13)*, Istanbul, Turkey, 2013, pp. 765-768.

[5]. R. A. Marshall, T. Wallace, M. Turbe, Finite-difference modeling of very-low-frequency propagation in the earth-ionosphere waveguide, *IEEE Transactions on Antennas and Propagation*, Vol. 65, Issue 12, Dec. 2017, pp. 7185-7197.

[6]. H. Lu, J. Yang, Q. Li, S. Hao, F. Guo, J. Wu, J. Chen, G. Ma, T. Xu, ELF/VLF communication experiment by modulated heating of ionospheric auroral electrojet at EISCAT, *IEEE Transactions on Antennas and Propagation*, Vol. 69, Issue 4, April 2021, pp. 2267-2273.

[7]. Z. Švestka, Optical observations of solar flares, *Space Science Reviews*, Vol. 5, Issue 3, 1966, pp. 388-418.

[8]. D. F. Ryan, R. O. Milligan, P. T. Gallagher, B. R. Dennis, A. K. Tolbert, R. A. Schwartz, C. A. Young, The thermal properties of solar flares over three solar cycles using GOES X-ray observations, *The Astrophysical Journal Supplement Series*, Vol. 202, Issue 2, 2012, 11.

[9]. K. G. McCracken, J. Beer, F. Steinhilber, Evidence for planetary forcing of the cosmic ray intensity and solar activity throughout the past 9400 years, in Coronal Magnetometry, *Springer*, 2014, pp. 585-607.

[10]. M. Dominique, A. N. Zhukov, P. Heinzel, I. E. Dammasch, L. Wauters, L. Dolla, S. Shestov, M. Kretzschmar, J. Machol, G. Lapenta, First detection of solar flare emission in mid-ultraviolet Balmer continuum. *The Astrophysical Journal Letters*, Vol. 867, Issue 2, 2018, L24.

[11]. W. P. Wah, M. Abdullah, A. M. Hasbi, S. A. Bahari, Development of a VLF receiver system for sudden ionospheric disturbances (SID) detection, in *Proceedings of the IEEE Asia-Pacific Conference on Applied Electromagnetics (APACE'12)*, 2012, pp. 98-103.

[12]. S. Verma, H. Jalori, A. K. Gwal, VLF waves characterization: Wavelet feature extraction method, *International Journal of Science and Research (IJSR)*, Vol. 5, Issue 2, 2016, pp. 1892-1897.

[13]. H. E. George, C. J. Rodger, M. A. Clilverd, K. Cresswell‐Moorcock, B. J. Brundell, N. R. Thomson, Developing a nowcasting capability for X‐Class solar flares using VLF radiowave propagation changes, *Space Weather*, Vol. 17, Issue 12, 2019, pp. 1783-1799.

[14]. S. I. Lotz, M. Clilverd, Demonstrating the use of a class of min-max smoothers for D region event detection in narrow band VLF phase, *Radio Science*, Vol. 54, Issue 3, 2019, pp. 233-244.

[15]. A. Nardi, A. Pignatelli, E. Spagnuolo, A neural network based approach to classify VLF signals as rock rupture precursors, *Scientific Reports*, Vol. 12, Issue 1, 2022, 13744.

(054)

Deep Learning for Coronary Artery Disease Severity Classification

Ariadna Jiménez-Partinen [1], **Karl Thurnhofer-Hemsi** [1,3,4], **Esteban J. Palomo** [1,3]
and Ana I. Molina-Ramos [2,3,4]
[1] Department of Computer Languages and Computer Science, University of Málaga,
Bulevar Louis Pasteur, 35, 29071, Malaga, Spain
[2] Cardiology Department, Hospital Universitario Virgen de la Victoria, 29010, Malaga, Spain
[3] Instituto de Investigación Biomédica de Málaga y Plataforma en Nanomedicina-IBIMA Plataforma
BIONAND. C/ Severo Ochoa, 35, Málaga TechPark, Campanillas, 29590, Malaga, Spain
[4] Centro de Investigación Biomédica en Red de Enfermedades Cardiovasculares (CIBERCV),
Instituto de Salud Carlos III (ISCIII). Avenida Monforte de Lemos,
3-5. Pabellón 11. Planta 0, 28029, Madrid, Spain
Tel.: +34 952133392
E-mail: ariadnapartinen@uma.es

Summary: Medical imaging evaluations are one of the fields where computed-aid diagnosis could improve the efficiency of diagnosis supporting physician decisions. Cardiovascular Artery Disease (CAD) is diagnosed using the gold standard, Invasive Coronary Angiography (ICA). In this work, performance analysis for binary classification of ICA images considering the severity ranges separately is reported, evaluating how performance is affected depending on the degree of lesions considered. For this purpose, an annotated dataset of ICA images was employed, which contains the ground truth, the location and the category of lesions into seven possible ranges: <20 %, [20 %, 49 %], [50 %, 69 %], [70 %, 89 %], [90 %, 98 %], 99 %, and 100 %. The ICA images were pre-processed, divided into patches and balanced by downsampling and data augmentation. In this study, four known pre-trained CNN architectures were trained using different categories of lesion degree as input, whose F-measures are computed. Results report that the F-measures showed a behavior dependent on the narrow presents of the image, being lesions with more than 50 % severity were better classified, achieving an F-measure of 75 %.

Keywords: Invasive coronary angiography, Deep learning, Classification, Healthcare.

---

1. Introduction

The use of Machine Learning, and more concrete Deep Learning, techniques are widely extended in different image areas. Medical imaging evaluations are one of the fields where computed-aid diagnosis could improve the efficiency of diagnosis supporting physician decisions. Cardiovascular Artery Disease (CAD) is usually diagnosed using Invasive Coronary Angiography (ICA), which remains the gold standard [1]. ICA imaging is an X-ray-based evaluation image, where a radiocontrast is injected into the arteries of the myocardium through a catheter which is inserted by a percutaneous incision in the radial or femoral artery [2].

The assessment of the degree of the narrowing is usually done visually, involving a substantial impact of subjectivity because it depends on the experience of the clinician, generating interobserver variability [3]. It has been proven the successful use of Convolutional Neural Networks (CNN) classifying objects in the image classification system [4-6].

The aim of this work is to evaluate how the performance of the binary classification problem of ICA images is affected depending on which category is considered as the positive class, being affected by the morphology and grade of narrowing of the lesion.

2. Methodology

The ICA images used in the present study are from a dataset composed of 42 anonymized videos acquired at Hospital Universitario Virgen de la Victoria (Málaga, Spain) by the cardiac angiography equipment Artis Zee (Siemens AG, München, Germany) under the corresponding regulations and permissions of the local ethical committee of the hospital. The original Digital Imaging and Communication in Medicine (DICOM) files were converted to PNG files for ease of use, obtaining images with a size of 512×512 pixels. The database is composed of different projections of the right coronary artery (RCA) and left coronary artery (LCA), such as right and left anterior obliques (RAO, LAO), with cranial and caudal angulations.

The frames with enough radiocontrast to visualize the arteries correctly were annotated (bounding box and category) by both the informatic and cardiology team. The possible severity categories of a lesion are <20 %, [20 %, 49 %], [50 %, 69 %], [70 %, 89 %], [90 %, 98 %], 99 %, and 100 %. In Fig. 1, a sample of each category is reported. In total, there are 3900 images with at least one lesion and 1943 images without visible lesions.

We followed a patch classification methodology, that is, the raw images were divided into a 4×4 grid. These patches were resized to 32×32 pixels and assigned to the category corresponding if the centroid of the bounding box of a lesion falls into it, and the rest were considered "non-lesion" patches. This procedure implied a huge increase in the "non-lesion" class, unbalancing the distribution between both classes. To lessen it, the "non-lesion" class was filtered, removing background patches using a basic mask of the ICA images, which were obtaining applying morphological operations segmenting the vessels. Those patches whose mask had less than 2 % of vessel pixels were discarded. Additionally, the negative class was randomly reduced to equalize the positive class. Then, data augmentation was applied using the following basic operations:

- Translations in the X and Y axis in a random range of [−4, 4] pixels;
- Scaling of the images randomly with a scale factor in a range of [0.9, 1.2];
- Flip horizontally and vertically.

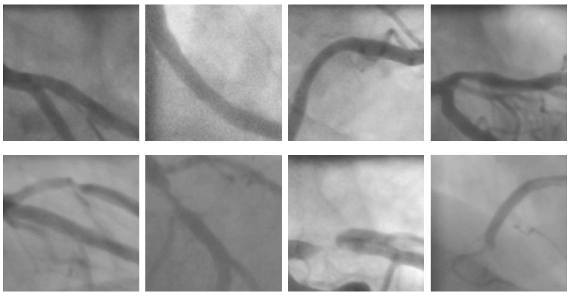


Fig. 1. Samples of the eight lesion categories. From left to right and top to bottom: non-lesion, <20 %, [20 %, 49 %], [50 %, 69 %], [70 %, 89 %], [90 %, 98 %], 99 %, and 100 %.

Finally, for the training process, the patches obtained for each input established were: 5904, 3536, 2992, 2620, 2268, 84, and 668, for <20 %, [20 %, 49 %], [50 %, 69 %], [70 %, 89 %], [90 %, 98 %], 99 %, and 100 %, respectively.

Four pre-trained deep network architectures widely used were chosen: MobileNet-V2, ResNet-18, ResNet-50, and DenseNet-201.

The performance of the different models was evaluated and compared using the F-measure, which provides an overall performance, where the precision (P) and recall (R) measures under the concept of the harmonic mean are included. Considering the main four representative parameters: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) [7].

$$F-measure = 2 \cdot \frac{P \cdot R}{P+R} = \frac{2 \cdot TP}{2 \cdot TP+FP+FN}$$

3. Experimental Results

3.1. Experiments Description

The present work studies how the performance is affected depending on the lesion category considered the "lesion" class. Both classes were divided into training (80 %) and test (20 %) sets by videos, i.e., frames of the same videos in the test set are unavailable for the train set because frames of a sequence could be very similar, so a fairer performance evaluation is carrying out. To compare in a reliable way the results attained in test sets, K-fold cross-validation was implemented, with K = 5.

For the training, we set: validation frequency = 50, validation patience = 5, maximum epochs = 70, initial learning rate = 0.0001, and SGDM solver (Stochastic Gradient Descent with Momentum). The batch size was set according to the number of training patches to keep the rate of iterations in all training processes.

The methods were implemented in MATLAB R2023a on a computer system with an Intel Core i9-10900X processor, 128 GB of RAM, and an NVIDIA GeForce RTX 3080 Ti GPU card. Moreover, no layer of chosen pre-trained models was frozen, updating all weights during training.

3.2. Results

In Table 1, the average and standard deviation of F-measure values obtained in the test sets are reported, showing in bold the highest values for each positive class. The first fact to stand out is that the four architectures employed have the same tendency in each positive class, without large differences between them.

Another remarkable aspect is the poor performance obtained in the classification task of 100 % lesions. However, in the rest of the ranges, performance increased slightly, although it decreased again with low lesion degrees ([20 %, 49 %] and <20 %). It could be explained because of the morphology of the lesion, although there are more patches to train, these lesions are more difficult to discern between non-lesion vessels being a more complex classification task. Whereas lesions higher than 50 %, i.e., 99 %, [90 %, 98 %], [70 %, 90 %], and [50 %, 69 %], are clearly distinguishable from non-lesion vessels getting higher outcomes, 0.698, 0.676, 0.685, and 0.752, respectively.

4. Conclusions

This study reports the evaluation of how the performance of the binary classification is affected depending on the severity degree of ICA images. Four CNN architectures were employed for the binary classification task, using lesion and non-lesion patches. The lesion class, positive class, was established considering the seven possible ranges of lesion degree separately. Results reported that severe lesions, higher than 50 % (75 % F-measure), are better classified than 100 % (36 % F-measure) and lower than 50 % (60 % F-measure) lesions.

Table 1. F-measure obtained on the test set using 5-fold cross-validation for each category.

Severity	MobileNet	ResNet-18	ResNet-50	DenseNet-201
100 %	0.328 ± 0.211	0.254 ± 0.161	**0.360 ± 0.234**	0.211 ± 0.089
99 %	**0.698 ± 0.047**	0.577 ± 0.128	0.585 ± 0.242	0.632 ± 0.163
[90 %, 98 %]	0.619 ± 0.087	0.569 ± 0.053	0.555 ± 0.061	**0.676 ± 0.069**
[70 %, 89 %]	0.594 ± 0.058	**0.685 ± 0.015**	0.581 ± 0.046	0.623 ± 0.033
[50 %, 69 %]	**0.752 ± 0.015**	0.745 ± 0.035	0.695 ± 0.032	0.742 ± 0.027
[20 %, 49 %]	0.562 ± 0.031	**0.571 ± 0.028**	0.553 ± 0.034	0.511 ± 0.038
<20 %	**0.602 ± 0.022**	0.559 ± 0.029	0.585 ± 0.020	0.600 ± 0.017

Acknowledgments

This work is partially supported by the Autonomous Government of Andalusia (Spain) under grant UMA20-FEDERJA-108, and by the University of Málaga (Spain) under grant B1-2021_20 and grant UMA of Pla, and by the Instituto de Investigación Biomédica de Málaga – IBIMA Plataforma BIONAND (Spain), under grant PLAN_PROPIO (Área Ibima Tech) 2021. The authors thankfully the funding from the University of Malaga (Spain).

References

[1]. J. Knuuti, W. Wijns, A. Saraste, D. Capodanno, E. Barbato, C. Funck-Brentano, et al., 2019 ESC guidelines for the diagnosis and management of chronic coronary syndromes: The task force for the diagnosis and management of chronic coronary syndromes of the European Society of Cardiology (ESC), *European Heart Journal,* Vol. 41, Issue 3, 2020, pp. 407-477.

[2]. G. Rigatelli, F. Gianese, M. Zuin, Modern atlas of invasive coronary angiography views: a practical approach for fellows and young interventionalists, *The International Journal of Cardiovascular Imaging,* Vol. 38, Issue 5, 2022, pp. 919-926.

[3]. L. M. Zir, S. W. Miller, R. E. Dinsmore, J. P. Gilbert, J. W. Harthorne, Interobserver variability in coronary angiography, *Circulation,* Vol. 53, Issue 4, 1976, pp. 627-632.

[4]. G. Litjens, F. Ciompi, J. M. Wolterink, B. D. de Vos, T. Leiner, J. Teuwen, I. Išgum, State-of-the-art deep learning in cardiovascular image analysis, *JACC: Cardiovascular Imaging*, Vol. 12, Issue 8 Part 1, 2019, pp. 1549-1565.

[5]. J. H. Moon, W. C. Cha, M. J. Chung, K. S. Lee, B. H. Cho, J. H. Choi, Automatic stenosis recognition from coronary angiography using convolutional neural networks, *Computer Methods and Programs in Biomedicine*, Vol. 198, 2021, 105819.

[6]. K. Pang, D. Ai, H. Fang, J. Fan, H. Song, J. Yang, Stenosis-DetNet: Sequence consistency-based stenosis detection for X-ray coronary angiography, *Computerized Medical Imaging and Graphics*, Vol. 89, 2021, 101900.

[7]. E. Ovalle-Magallanes, J. G. Avina-Cervantes, I. Cruz-Aceves, J. Ruiz-Pinales, Hybrid classical-quantum convolutional neural network for stenosis detection in X-ray coronary angiography, *Expert Systems with Applications*, Vol. 189, 2022, 116112.

(055)

Revising and Reexamining Angluin's Algorithm: Implications for Unified Regular Language Learning Algorithms

A. Fellah
Northwest Missouri State University, School of Computer Science and Information Systems
Maryville, MO, 64468 USA
E-mail: afellah@nwmissouri.edu

Summary: While residuality adds a linguistic meaning to the automaton's states in the context of regular languages and discerns significant facts from the semantics of each state, alternation provides a very succinct representation of regular languages and generalizes nondeterminism. In this paper, we encapsulate the two-mode properties, residuality and alternation, into a more expressive automaton model that we refer to as *reversal state Residual Alternating Finite Automata* (rs-RAFA). Then, along the analogical lines of Angluin's algorithm, we present a new regular language learning algorithm, namely rs-AL^*, which is built on the *Learner-Teacher-Oracle* framework.

Keywords: Machine learning, Deep learning, Residual languages, Alternating finite automata, Angluin's algorithm $L^\star$.

---

1. Introduction

The property of *residuality* has been introduced by Denis *et al* [9, 13, 14] in the context of finite state automata. It is considered as a natural distillation of the essence of the automaton's states language recognition. Residuality adds a foundational linguistic meaning to the automaton's states in the context of regular languages and discerns significant facts from the semantics of each state of the automaton. In the context of regular languages, *residual finite automata* (RFA) also called *residual state finite automata* (RSFA) [9, 13, 14] are a subclass of nondeterministic finite automata (NFA) where each state represents a language called *residual language* of the language recognized by the NFA. RFA are introduced as a solution to the well-known problem of NFA not having unique minimal (in terms of the number of states) representatives. It has shown that every regular language admits a unique canonical RSFA [14]. An automaton $\mathcal{A}$ accepting a language $\mathcal{L}$ is residual if every state q of $\mathcal{A}$ represents a residual language. Such a refined class of residual automata allows one to eventually observe each state independently and describe its formal semantic subsequently. We adopt RFA notation.

The notion of residual languages plays an important role in the context of learning algorithms. The class of RFA lies between deterministic finite automata (DFA) and nondeterministic finite automata (NFA). Such a class of automata has their own specific properties which are usually exploited in learning regular language algorithms. In addition, RFA share several significant properties in the context of determinism and nondeterminism settings. For instance, RFA share with NFA the existence of automata that are exponentially smaller, in the number of states, than the corresponding minimal DFA for the same language. But more importantly, every DFA exhibits the property of residuality which underlines several active learning algorithm techniques for finite state automata, such as the seminal algorithm $L^\star$ of Angluin's [15] for leaning DFA and the generalized algorithm NL of Boling's *et al* [10] for learning NFA. Both algorithms have provoked a tremendous amount of research [5, 6, 8, 10, 11, 18] in several areas of computer science such as machine learning, artificial intelligence, and software verification.

The notion of alternation is a natural generalization of nondeterminism. It received its historical and formal treatment in the seminal paper of Chandra, Kozen, and Stockmeyer [4]. Alternation provides a very succinct representation of regular languages while residuality adds a natural meaning to the automaton's states. For AFA, it is proved that they are precisely as powerful as both DFA and NFA as far as language recognition, but they differ in efficiency. However, beyond this seemingly negative result the presence of alternation can lead to simplified construction in formal languages [1, 2, 3, 12, 15, 17, 18]. In terms of the number of states, a minimal DFA might be exponentially larger than an NFA and double-exponentially larger than an AFA, respectively. Furthermore, AFA have particularly emerged as a practical tool in several applications such as model checking, learning algorithms, formal methods, and software programs' verification. Furthermore, learning regular languages has gained widespread in a wide variety of applications such as pattern recognition, data mining, and learning algorithms. The uniqueness of the minimality of DFA plays an important role in the language learning algorithm $L^\star$ where it has been proved that the class of regular languages could be learned efficiently in polynomial time, in terms of the number of states, of the DFA.

In Angluin's algorithm, the model is composed of a *learner and teacher*. The learner, who initially knows nothing about the regular language $\mathcal{L}$, tries to learn $\mathcal{L}$

by interacting with the teacher. Information sources is available from the teacher who has knowledge about the language in question and guide the learner by answering *membership* and *equivalence queries*. A membership query consists in asking the teacher "*Is the word in the language* $\mathcal{L}$". Additionally, an equivalence query consists in asking the teacher whether a hypothesized DFA $\mathcal{H}$ recognizes the language $\mathcal{L}$. The teacher returns either *yes* if the equivalence query $\mathcal{H}$ and the inferred model are equivalent, otherwise a counter example is returned.

In this paper, we investigate a pair of semantic properties termed residuality and alternation that complement one another and are amenable to automata learning. Then, we encapsulate the two-mode properties into a more expressive automaton model that we refer to as *reversal state residual alternating finite automata (*rs-RAFA). In the classical active learning algorithms and roughly speaking, typically algorithms are mainly built on the minimal *Learner-Teacher* framework. In this paper, we extend such a framework with an *Oracle and* a prefix rs *that* we refer to as rs-*Learner-Teacher-Oracle* paradigm that finds the unique minimal and consistent rs-RAFA in polynomial time for each regular language.

The remaining of the paper is organized as follows. In Section 2, we introduce basic preliminary concepts, and notations. Language interpretation and regular expressions are described in Subsection 2.1. Section 3 covers alternating finite automata (AFA) with straightforward details by showing that AFA are a suitable framework for active automata learning algorithms. Section 4 looks at the related work of derivatives of regular expressions and languages. Section 5 discusses residuality and describes such a property as an important framework in algorithmic learning languages. In Section 6, we show the importance of alternation and revisualization properties which lead us to two main theorems between rs-RAFA and DFA. Section 7 extends yet the results of the native AFA in two directions, languages over Boolean operations and residual languages. We also present a new regular language learning algorithm, namely rs-AL^*, along the lines of L^*. Finally, in Section 8 we conclude the paper with a summary and discuss looming future work.

2. Preliminaries

In this section we briefly recall some relevant definitions. An *alphabet is* a finite, nonempty set. The elements of an alphabet are called *symbol*s or *letters.* A string over an alphabet Σ is a finite sequence consisting of zero or more symbols of Σ. Without loss of generality, we assume in the sequel that alphabets do not contain any of the "special" symbols, ($\emptyset$, ε, $\cup$, $\cap$, $*$,$\vee$, $\wedge$, $+$, $\cdot$, (,)). The string consisting of zero letters is called the *empty string,* denoted by ε. The length of a string w, denoted by $|w|$, is the number of symbols in w. By definition, $|\varepsilon| = 0$. The set of all strings (respectively all nonempty strings) over an alphabet Σ is denoted by Σ^* (respectively Σ^+). A language $\mathcal{L}$ over Σ is a (possibly infinite) set of finite strings $\mathcal{L} \subseteq \Sigma^*$. We denote the language of an automaton $\mathcal{A}$ by $\mathcal{L}(\mathcal{A})$, and the language accepted by a state $q \in Q$ by $\mathcal{L}(\mathcal{A}_q)$. Given a language $\mathcal{L}$ over an alphabet Σ, the Kleene star closure ("$*$") of $\mathcal{L}$ is the set $\mathcal{L}^* = \cup_{i=0}^{\infty} \mathcal{L}^{i}$ and the positive Kleene plus "+") of $\mathcal{L}^+$ is $= \cup_{i=1}^{\infty} \mathcal{L}^{i}$. The language $\overline{\mathcal{L}} = \Sigma^* \backslash \mathcal{L}$ is the complement of $\mathcal{L}$. The concatenation of two strings u and v is the string consisting of the symbols of u followed by the symbols of v, denoted $u \cdot v$ (also often written as uv). We denote the reversal of a string w by w^r while the reversal of a language $\mathcal{L}$, denoted $\mathcal{L}^r$ is defined as $\mathcal{L}^r = \{w^r \mid w \in \mathcal{L}\}$. A *prefix-closed set* is a set where every prefix of every member is also a member of the set. A *suffix-closed set* is a set where every suffix of every member is also a member of the set. An NFA is a quintuple $\mathcal{A} = (\Sigma, Q, Q_0, \delta, F)$ where Σ is the alphabet, Q is a finite set of states, Q_0 is a set of initial states $Q_0 \subseteq Q$, $\delta: Q \times \Sigma \rightarrow 2^Q$ is the transition function, and $F \subseteq Q$ is a set of final states. If $|Q_0| = 1$ and $\delta: Q \times \Sigma \rightarrow Q$, then the NFA is called a DFA. The language accepted by $\mathcal{A}$ is $\mathcal{L}(\mathcal{A}) = \{w \in \Sigma^* \mid (Q_0, w) \cap F \neq \phi\}$.

2.1. Languages and Regular Expressions

Regular expressions are formal notations for describing regular languages. Let e denote a regular expression, we define the regular language of e to be $\mathcal{L}(e)$. Furthermore, we define the set of regular expressions by E over Σ as the subset of $(\Sigma \cup \{\epsilon, \phi +,\cdot,*, (,)\})^*$ that recursively satisfies the following conditions:

(i) $\Sigma \cup \{\epsilon, \phi\} \subseteq E$
(ii) if $(e_1 + e_2) \in E$ then
  (a) $(e_1 + e_2), (e_1 \cdot e_2), (e^*) \in E$
  (b) $(e_1 \cap e_2), \overline{e} \in E$

In addition to the above operations of union, concatenation, and star as defined in (a), regular expressions which include the intersection ($\cap$) and complement ($-$) operations as shown in (b) are called *extended regular expressions.*

3. Alternating Finite Automata (AFA)

Alternating finite automata (AFA) has the property of alternation in the following sense: If in a given state q the automaton reads an input symbol a, it will activate all states of the automaton to work on the remaining part of the input in parallel. Once the states have completed their tasks, q will evaluate their results using a Boolean function and pass on the resulting value by which it was activated. A word w is accepted if the starting state computes the values of 1. Otherwise, it is rejected. In a nondeterministic computation all configurations are *existential* in the sense that there exists at least one successful path that leads to acceptance. An AFA may have also *universal* configurations from which the computation branches into several parallel computations that must all lead to acceptance. We represent existential and universal choices by a Boolean formula. Formally, let Q be a set, we use $\mathcal{B}^Q$ to be the set of all Boolean formulas over Q. That is, $\mathcal{B}^Q$ is built from the elements $q \in Q$, 1, and 0 using the binary operations or ($\vee$), and ($\wedge$), and not ($-$). We now formalize this idea.

Definition 3.1. An alternating finite automaton (AFA) is a quintuple $\mathcal{A} = (\Sigma, Q, s, F, g)$ where (a) Σ is an alphabet, the input alphabet; (b) Q is a finite set, the set of states; (c) $s \in Q$ is the starting state; (d) $F \subseteq Q$ is the set of final states; (e) g is a mapping of Q into the set of all mappings of $\Sigma \times \mathcal{B}^Q$ into $\mathcal{B}$.

We turn to defining the sequential behavior of an AFA. For $q \in Q$ and $a \in \Sigma$, let $g_q(a)$ be the Boolean function defined as:

$$g_q(a, \hat{u}) \colon \Sigma \times \mathcal{B}^Q \to \mathcal{B},$$

where a $\in \Sigma$ and $\hat{u} \in \mathcal{B}^Q$. Also, for $a \in \Sigma$, $q \in Q$, and $\hat{u} \in \mathcal{B}^Q$, $g_q(a, \hat{u}) = g_q(a)(\hat{u})$, is equal to either 0 or 1.

Now define $f \in \mathcal{B}^Q$ by the condition: $f_q = 1 \Leftrightarrow q \in F$. f is called the characteristic vector of F.

We extend g to a mapping of Q into the set of all mappings of $\Sigma^* \times \mathcal{B}^Q$ into $\mathcal{B}$ as follows:

$$g_q(w, \hat{u}) = \begin{cases} u_q \text{ if } w = \epsilon \\ g_q(a, g(v, \hat{u})) \text{ if } w = av \; with \; a \, \Sigma, v \in \Sigma^{*}{}' \end{cases}$$

where $w \in \Sigma^*$ and $\hat{u} \in \mathcal{B}^Q$.

Definition 3.2. Let $\mathcal{A} = (Q, \Sigma, s, F, g)$ be an AFA. A word $w \in \Sigma^*$ is accepted by $\mathcal{A}$ if and only if $g_s(w, f) = 1$. The language accepted by $\mathcal{A}$ is the set $\mathcal{L}(\mathcal{A}) = \{w \mid w \in \Sigma^* \wedge g_s(w, f) = 1\}$.

We denote the language of $\mathcal{A}$ by $\mathcal{L}(\mathcal{A})$ and the language accepted by a state $q \in Q$ by $\mathcal{L}(\mathcal{A}_q)$. Note that in the same spirit as the characteristic vector of F, we extend g to languages.

Definition 3.3. Let $\mathcal{A} = (Q, \Sigma, s, g, F)$ be an AFA and $\mathcal{L}(\mathcal{A})$ the language accepted by $\mathcal{A}$. Then, the characteristic output of $\mathcal{A}$ is defined as

$$g_{\mathcal{A}}(w, \hat{u}) = \begin{cases} 1 \text{ if } g_q(w, \hat{u}) = 1 \text{ for all } w \in \mathcal{L}(\mathcal{A}) \\ 0 \; otherwise \end{cases}$$

Example 3.1. Consider the following AFA $\mathcal{A} = (Q, \Sigma, s, g, F)$ where $Q = \{q_0, q_1, q_2\}$, $\Sigma = \{\text{a, b}\}$, $s = \{q_0\}$, $F = \{q_2\}$, and g is given by the following Table 1.

Example 3.2. Let $w = bab$ be a string. We will check whether the input w is accepted by the above AFA $\mathcal{A}$.

$$g_{q0}(bab, f) = g_{q1}(ab, f) \vee \overline{g_{q2}(ab, f)} =$$

$$= g_{q0}(b, f) \vee \overline{(g_{q0}(b, f) \vee \overline{g_{q1}(b, f)})} =$$

$$= \left(g_{q1}(\epsilon, f) \vee \overline{g_{q2}(\epsilon, f)}\right) \vee$$

$$\overline{((g_{q1}(\epsilon, f) \vee \overline{g_{q2}(\epsilon, f)}) \vee \overline{\overline{(g_{q0}(\epsilon, f)} \wedge g_{q2}(\epsilon, \text{f}))})} =$$

$$= (0 \vee \bar{1}) \vee \overline{((0 \vee \bar{1}) \vee \overline{(\bar{0} \wedge 1)})} =$$

$$= (0 \vee 0) \vee \overline{((0 \vee 0) \vee \overline{(1 \wedge 1)})} =$$

$$= (0) \vee \overline{((0) \vee \overline{(1)})} = 0 \vee \overline{(0 \vee 0)} = 0 \vee \overline{(0)} = 0 \vee 1 = 1$$

Table 1. AFA's state table.

states	**a**	**b**
q_0	$q_0 \wedge q_1$	$q_1 \vee \overline{q_2}$
q_1	q_0	$\overline{q_0} \wedge q_2$
q_2	$q_0 \vee \overline{q_1}$	1

Therefore, the string *bab* is accepted $\mathcal{A}$.

We now introduce the notion of *r*-AFA [2, 3] which can be seen as an AFA except that it reads its input in reverse order. Moreover, such automata are usually used for implementing regular languages and their operations efficiently as summarized in the next theorem. We will adapt and extend *r*-AFA far beyond their original scope. That is, we will exploit such results in the context of learning algorithms for regular languages and automata.

Theorem 3.1. [2, 3]. If $\mathcal{L}$ is accepted by an n-state complete DFA; then $\mathcal{L}$ is accepted by an *r*-AFA with at most $\lceil log_n \rceil$ states.

4. Derivatives of Regular Expressions

The notion of derivative regular expressions has been introduced by Brzozowski [6] to find quotient on regular expressions and give corresponding derivatives and their auxiliary functions. Let e be an extended regular expression and u is a word over Σ^*. We denote by $\partial_u(e)$ the derivative of e with respect to u, which is formally defined as follows:

Definition 4.1. The derivative of an extended regular expression e with respect to a word $u \in \Sigma^*$ is defined to as: $\delta_u(e) = v \in \Sigma^*$ such that $uv \in e\}$ and $\delta_u(\mathcal{L}) = \{v \in \Sigma^*$ such that $wv \in \mathcal{L}\}$.

Intuitively, $\partial_u(e)$ is the set of all remaining strings obtainable from e by taking off the prefix u, if possible. The derivatives of an extended regular expression e with respect to a symbol $a \in \Sigma$ are defined as follows:

Properties 4.1.

$$\partial_a(\phi) = \phi; \partial_a(\epsilon) = \phi; \partial_a(a) = \epsilon; \partial_\epsilon(a) = a,$$
$$\partial_a(b) = \phi \text{ if } b \neq a; \partial_a(e_1 + e_2) = \partial_a(e_1) + \partial_a(e_2),$$
$$\partial_a(e_1^*) = \partial_a(e_1) \cdot e_2^*; \partial_a(e_1 \cap e_2) = \partial_a(e_1) \cap \partial_a(e_2),$$
$$\partial_a(e_1) = \overline{\partial_\text{a}(e_1)},$$
$$\partial_\text{a}(e_1 \cdot e_2) = \begin{cases} \partial_\text{a}(e_1) \cdot e_2 + \partial_\text{a}(e_2) \text{ if } e_1 \text{ is } nullable \\ \partial_\text{a}(e_1) \cdot e_2 \; otherwise \end{cases}$$

In addition, the concept of derivatives applies to languages. For a language $\mathcal{L}$, the derivative of $\mathcal{L}$ with respect to a string w is the set of remainder words after having read w from any word in $\mathcal{L}$. The following two properties are a consequence from the a definition of the derivative where $u \in \Sigma^*$ and $a \in \Sigma$.

i. $\delta_\varepsilon(\mathcal{L}) = \mathcal{L}$,
ii. $\delta_{ua}(\mathcal{L}) = \delta_a(\delta_u(\mathcal{L}))$

Example 4.1. Given the NFA $\mathcal{A} = (Q, \Sigma, Q_0, F, \delta)$, where $Q = \{x_1, x_2, x_3\}$, $\Sigma = \{a, b\}$, $Q_0 = x_1$, $F = \{x_3\}$ and δ is given by the following system of language equations. The regular language $\mathcal{L}(\mathcal{A})$ generated by $\mathcal{A}$ is obtained by solving the following system of language equations. Let $\mathcal{A}$ be a NFA as depicted below in Fig. 1.

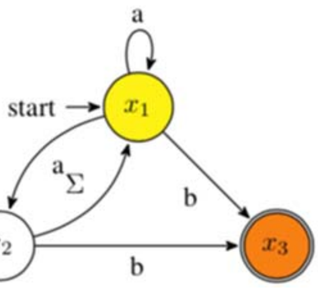


Fig. 1. $\mathcal{L}(\mathcal{A}) = \mathcal{L}x_1 = (a + a\Sigma)^*(ab + b)$.

$$\mathcal{L}x_1 = a\mathcal{L}x_1 + a\mathcal{L}x_2 + b\mathcal{L}x_{3;}\ \mathcal{L}x_2 = \mathcal{L}x_1 + a\mathcal{L}x_3,$$
$$\mathcal{L}x_3 = \epsilon$$

Using backward propagation yields the solution, $\mathcal{L}x_1$ which is obtained by using a series of substitutions of Arden's rule [8] which yields $\mathcal{L}x_1 = (a + a\,\Sigma)^*(ab + b)$.

5. Residuality

Residual finite state automata (RFA) [9, 13, 14] are a subclass of NFA where each state represents a residual language of the language that is accepted by the automaton. Importantly, it adds a foundational meaning for a better understanding of regular language learning algorithms and computational learning theory. The class of RFA lies between DFA and NFA; and they share several significant properties. For instance, RFA share with NFA the existence of automata that are exponentially smaller, in the number of states, than the corresponding minimal DFA for the same language.

Let $\mathcal{A} = (Q, \Sigma, q, F, \delta)$ be a finite state automaton. The language $\mathcal{L}(\mathcal{A})$ is the set of all accepted strings by $\mathcal{A}$. For a state $q \in Q$, we write $\mathcal{A}_q$ for the automaton that starts in the configuration (*i.e.,* state) q.

Definition 5.1. A language $\mathcal{L}(\mathcal{A}_q) \in \Sigma^*$ is a residual language of $\mathcal{L}$ if there is $w \in \Sigma^*$ such that $\mathcal{L}(\mathcal{A}_q) = \partial_w(\mathcal{L})$, where $\partial_w(\mathcal{L}) = \{v \in \Sigma^* \mid wv \in \mathcal{L}\}$.

Example 5.1. Let $\mathcal{A} = \mathcal{A} = (Q, \Sigma, Q_0, F, \delta)$ be a DFA. $Q = \{q_0, q_1, q_2, q_3, q_4\}$, $\Sigma = \{a, b\}$, $Q_0 = \{q_0\}$, $F = \{q_2, q_4\}$ and δ as shown in the following graph in Fig. 2.

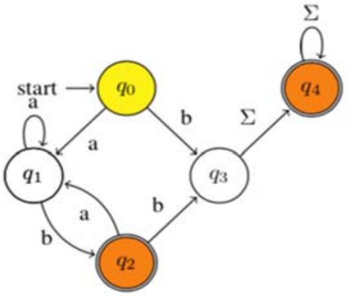


Fig. 2. □$(\mathcal{A}) = \mathcal{L}_{q0} = a^+\text{b} + (a^+b)^+(\epsilon + b\Sigma^+) + b\Sigma^+$.

For notational purpose, we use $\mathcal{L}_q$ to indicate the residual language of state q**.** We derive a simple set of residual languages of $\mathcal{L}$ from Fig. 2., denoted by *Res(*$\mathcal{L}$*)*, $\delta_\varepsilon(\mathcal{L}) = \mathcal{L}_{q0}$, $\delta_\alpha(\mathcal{L}) = \mathcal{L}_{q1}$, $\delta_\alpha(\mathcal{L}) = \mathcal{L}_{q2}$, $\delta_\beta(\mathcal{L}) = \mathcal{L}_{q4}$, where α and $\beta \in \Sigma^*$, $\alpha = a^+b$ and $\beta = b\Sigma^+$.

An automaton $\mathcal{A}$ accepting a language $\mathcal{L}$ is residual if every state q of $\mathcal{A}$ corresponds to a residual language. However, the reverse is not always true. That is, not every residual language should be accepted by a unique state. To this end, several states may accept the same language, consequently, we categorize the set of residual languages as *prime* and *composed.* residual languages [1, 5, 6, 11, 13].

Definition 2.3. A residual finite automaton (RFA) $\mathcal{R} = (\Sigma, Q, s, F, \delta)$ is a nondeterministic finite automaton (NFA) where for every state $q \in Q$, $\mathcal{L}(\mathcal{R}_q)$ is a residual language $\in \mathcal{L}(\mathcal{R})$.

6. rs-RAFA: A State Machine Framework for Residual-Alternating Learning Algorithm

Learning regular sets from queries and counter examples by Angluin [15] forms the basis of many modern state machine inference algorithms. Given a regular language $\mathcal{L}$ a canonical *residual alternating finite automata* (RAFA) is an alternating finite automaton (AFA) such that every composed state of the equivalent minimal DFA can be represented as unions and intersections of prime residual languages of $\mathcal{L}$. With respect to minimization of RAFA, we consider a special kind of RAFA that we call s-RAFA. A s-RAFA is a RAFA $\hat{\mathcal{A}} = (Q, \Sigma, q, F, \delta)$ such as every $a \in \Sigma$ and every $\hat{u} \in \mathcal{B}^Q$, $g_q(a, \hat{u})$) does not depend on $\hat{u}$. Intuitively, this means that the starting state s cannot be reached in any computation. Obviously, for every RAFA one construct an equivalent s-RAFA which has just one more state.

Theorem 6.1. Let $\mathcal{L}$ be a language and $\mathcal{L}^r$ be the reverse of $\mathcal{L}$. $\mathcal{L}$ is accepted by an s-RAFA with $k+1$ states if and only if $\mathcal{L}^r$ is accepted by a DFA with 2^k states.

Proof: We show the proof by construction. The proof is not included due to page limits. Refer to the author's contact for details.

Now, we introduce a variation of s-RAFA which we call rs-RAFA (r short for reversal). The new rs-RAFA is the same as a s-RAFA except that the input string is read in reverse.

Theorem 6.2. For each language $\mathcal{L}$ that is accepted by a n-states DFA, there exists an equivalent rs-RAFA with at most $1 + \lceil(\log_2 n)\rceil$ states.

Proof: The proof is not included due to page limits. Refer to the author's contact for details.

7. Revisiting and Reexamining Angluin's Learning Algorithm

Regular languages can be described by different types of finite state automata which all have the same expressive power – they accept regular languages. Finite state automata are usually classified into several classes: deterministic, nondeterministic, universal, and alternating automata. Despite being equivalent from the point of view of the class of languages they can recognize they differ in succinctness and efficiency when solving formal languages and automata related problems. It would be interesting to consider finite automata that exploit the potential benefits in succinctness. In this paper for instance, we only exhibit few cases of succinctness – in terms of the number of states. That is, a minimal DFA for a regular language $\mathcal{L}^r$ might be exponentially larger than an equivalent s-RAFA accepting $\mathcal{L}^r$ (Theorem 6.1). Furthermore, every s-RAFA has a useful property of residuality which makes them exponentially more succinct than DFA and the preferable automaton model to work with in practical learning applications. Technically, regular languages are not easy to learn using rs-RAFA and rs-RAFA. In contrast, special attention is devoted to RAFA (*i.e.*, rs-RAFA) that can be learned efficiently along the lines of $L^\star$. We introduce a new query learning paradigm based on the prefix *rs* (*i.e.*, rs-RAFA) called rs-*Learner-Teacher-Oracle* paradigm that finds the unique minimal and consistent rs-RAFA in polynomial time for each regular language. Technically, RAFA (*i.e.*, s-RAFA, rs-RAFA) have the property that the states of the automaton correspond to residual languages. Along the analogical lines of $L^\star$ for learning DFA, this paper presents a new inference algorithm for RAFA. A reexamination and revision of $L^\star$ has substantially foreground a new active algorithm for RAFA, called rs-AL^*.

7.1. Active Learning of RAFAs via rs-Learner-Teacher-Oracle Inference

The learner wants to learn a regular target language $\mathcal{L}$ over an alphabet Σ^* by actively querying a teacher using two different types of queries: *membership query* MQ($\mathcal{L}$) and *equivalence queries* EQ($\mathcal{L}$). The teacher has access to the language through the oracle by reversing the learner's language in question using another variant of queries called *reversal membership query* and *equivalence query* MQ($\mathcal{L}^r$) and EQ($\mathcal{L}^r$), respectively, as shown in Fig. 3.

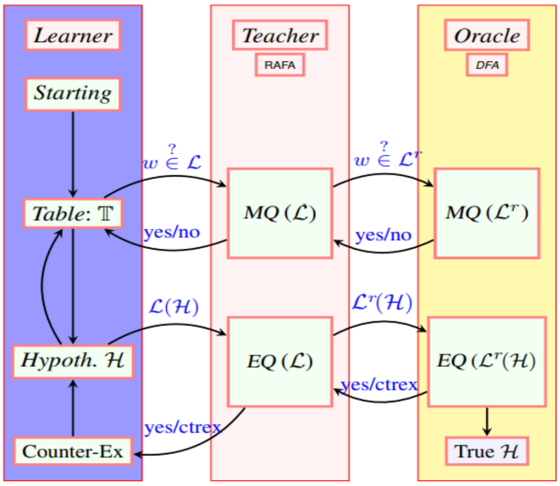


Fig. 3. RAFA: Active learning algorithm using two variants of queries.

The primordial active learning task is to find a RAFA by converting it to DFA using the oracle who satisfies the membership and equivalence queries of the learner, formally defined as follows:

Algorithm: rs-AL*

Input: MQ *($\mathcal{L}$) and* EQ *($\mathcal{L}$), target language: $\mathcal{L} \subseteq \Sigma^*$*
Output:

The oracle replies "*yes*" or "*no*" depending on whether $w \in \mathcal{L}^r$ or not. Moreover, the oracle answers "*yes*" if $\mathcal{H}$ is correct or supplies *counterexample*.

a) Learner-Teacher: MQ ($\mathcal{L}$): consists in asking the teacher if a word $w \in \Sigma^* \in \mathcal{L}$. The teacher replies "*yes*" or "no" depending on whether $w \in \mathcal{L}$ or not.

b) *Learner-Teacher:* EQ ($\mathcal{L}$) consists in asking the teacher whether a hypothesis RAFA (of minimal size) $\mathcal{H}$ is correct, *i.e.*, whether $\mathcal{L}(\mathcal{H}) = \mathcal{L}$. The teacher answers *yes* if $\mathcal{H}$ is correct or returns a *counterexample*.

c) *Teacher-Oracle:* MQ ($\mathcal{L}^r$) consists in asking the oracle if a word $w \in \Sigma^* \in \mathcal{L}^r$ (Theorem 6.1 and Theorem 6.2). The oracles replies "*yes*" or "*no*", depending on whether $w \in \mathcal{L}^r$ or not.

d) *Teacher-Oracle:* EQ($\mathcal{L}^r$) consists in asking the oracle to use the relationship between RAFA and DFA (Theorem 6.1 and Theorem 6.2). That is, whether $\mathcal{L}^r(\mathcal{H}) = \mathcal{L}^r$. The oracle answers "*yes*" if $\mathcal{H}$ is correct or returns a *counterexample*.

During the learning process, the learner constructs and maintains two sets, a prefix-closed set $\mathcal{U} \subseteq \Sigma^*$ of words that are candidates for identifying states, and a suffix-closed set $\mathcal{V} \subseteq \Sigma^*$ of words that are used to distinguish such states. Both $\mathcal{U}$ and $\mathcal{V}$ contain the empty string ε. Using membership queries, the learner checks whether all strings uv are in $\mathcal{V}$ or $uav \in \mathcal{U}\Sigma\mathcal{V}$ belong to $\mathcal{L}$, and organizes the results into an

Observation table $\mathbb{T} = (\mathcal{U}, \mathcal{V}, \mathcal{T})$ for $\mathcal{L}$ with entries where $\mathcal{T}$ is a mapping function defined as

$$\mathcal{T}(w) = \begin{cases} (\mathcal{U} \cup \mathcal{U}\Sigma) \times \mathcal{V} \mapsto \{yes\}\, if\ w \in \mathcal{L} \\ (\mathcal{U} \cup \mathcal{U}\Sigma) \times \mathcal{V} \mapsto \{no\}\, if\ w \notin \mathcal{L} \end{cases},$$

where $w \in (\mathcal{U} \cup \mathcal{U}\Sigma)\mathcal{V}$. The table $\mathbb{T}$ is of dimension $|(\mathcal{U} \cup \mathcal{U}\Sigma)\mathcal{V} \times \mathcal{V}|$, where the elements entries of both rows and columns are $\{yes, no\}$. We call such a function *row* of u and the set of all rows is denoted by *Rows* ($\mathbb{T}$). Rows labeled by the elements of $\mathcal{U}$ are the candidates for states of the automaton being constructed, and columns labeled by the elements of $\mathcal{V}$ correspond to distinguishing experiments for these states. Rows labeled by elements of $\mathcal{U}\Sigma$ are used to construct the transition function.

7.2. Closedness and Consistency

a. The table $\mathbb{T}$ is closed if for all $u \in \mathcal{U}$ and $a \in \Sigma$, there is $u' \in \mathcal{U}$ such that $row(ua) = row(u')$;

b. The table $\mathbb{T}$ is consistent if for all $u, u' \in \mathcal{U}$ and $a \in \Sigma$, there is $u' \in \mathcal{U}$ such that $row(u) = row(u')$ which implies that $row(ua) = row(u'a)$.

If the table $\mathbb{T}$ is closed and consistent the *Learner* constructs hypothesized RAFA $\mathcal{H} = (Q, \Sigma, q_0, F, \delta)$.

8. Conclusion

In conclusion, we revised, reexamined, and extended Angluin's learning approach for a variety of residual alternating finite automata, RAFA, s-RAFA, and sr-RAFA, by exploited the relationship between these variants of RAFA and DFA using Theorems 6.1 and 6.2. Furthermore, we developed a regular language learning algorithm called rs-AL^* along the line of $L^\star$. A possible direction for future work is to enhance is to improve the table-based learning algorithm of rs-AL^*.

Reference

[1]. D. Angluin, S. Eisenstat, D. Fisman, Learning regular languages via alternating automata, in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'15),* 2015, pp. 3308-3314.

[2]. A. Fellah, H. Jurgensen, S. Yu, Constructions for alternating finite automata, *Inter. Comp. Math*., Vol. 35, 1991, pp.117-132.

[3]. S. Yu, Regular languages, Chapter 2, in Handbook of Formal Languages, *Springer*, 1997, pp. 41-110.

[4]. A. K. Chandra, D. C. Kozen, L. J. Stockmeyer, Alternation, *J. ACM*, Vol. 28, Issue 1, 1981, pp. 114-433.

[5]. S. Berndt, et al., Learning residual finite automata, in *Proceedings of the 31st AAAI Conference Artificial Intelligence,* 2017, pp. 1749-1755.

[6]. F. W. Vaandrager, Model learning, *Commun. ACM*, Vol. 60, Issue 2, 2017, pp. 86-95.

[7]. A. Okhotin, Conjunctive grammars, *Journal of Auto., Lang. and Combin*., Vol. 6, Issue 4, 2011 pp. 519-535.

[8]. D. N. Arden, Delayed-logic and finite state, in *Proceedings of the 2nd Annual Symposium on Switching Circuit Theory and Logical Design (SWCT'61)*, 1961, pp. 133-151.

[9]. F. Denis, A. Lemay, A. Terlutte, Residual finite state automata, in *Proceedings of the Symposium on Theoretical Aspects of Computer Science (STACS'10)*, 2010, pp. 144-157.

[10]. B. Bollig, P. Habermehl, C. Kern, M. Leucker, Angluin-style Learning of NFA, in *Proceedings of the of the Int. Joint Conference on Artificial Intelligence (IJCAI'19)*, 2019, pp. 1004-1009.

[11]. J. Moerman, M. Sammartino, Residual nominal automata, in *Proceedings of the 31st Int. Conference on Concurrency Theory (CONCUR'20)*, 2020, pp. 44:1-44:21.

[12]. A. Fellah, Equations and regular-like expressions for AFA, *Int. J. Comput. Math*., Vol. 51, Issues 3-4, 1994, pp. 157-172.

[13]. F. Denis, A. Lemya, A. Terlutte, Residual finite state automata, *Fund. Inform*., Vol. 51, Issue 01, 2002, pp. 339-368.

[14]. F. Denis, A. Lemay, A. Terlutte, Learning regular languages using RFSAs, *Theoret. Comput. Sci*., Vol. 313, Issue 2, 2004, pp. 267-294.

[15]. D. Angluin, Learning regular sets from queries and counter examples, *Information and Computation*, Vol. 75, Issue 2, 1987, pp. 87-106.

[16]. J. A. Brzozowski, Derivatives of regular expressions, *J. ACM*, Vol. 11, Issue 4, 1964, pp. 481-494.

[17]. A. Fellah, Real-time languages, timed alternating automata, and timed temporal logics: relationships and specifications, *J. Procedia Computer*, Vol. 62, 2015, pp. 47-54.

[18]. A. Fellah, On alternating and residuality language learning algorithms, in *Proceedings of the 3rd Int. Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI'21),* 2021, pp. 87-93.

(056)

An Improved Multiview Stereo for Semantic Reconstruction

Sk. Mohammadul Haque
Indian Institute of Technology Kharagpur, Kharagpur, India – 721302
E-mail: mohammadul.ee@gmail.com

Summary: We present an improved multiview stereo scheme for semantic reconstruction of 3D scenes that provides both geometry with colour and semantic models of scenes from their photographic images. Specifically, our proposed scheme is based on a popular and efficient multiview stereo method called the patched-based multiview stereo (PMVS). Our work applies to scenarios when per-view semantic estimates are available on the image planes. We present our improved scheme on the PMVS that can be used in the 3D geometry estimation as well as the simultaneous fusion of the per-view semantics all in a joint multiview fashion to form a final reconstructed semantic point cloud with colour information. We finally test our scheme on a number of datasets and show its efficacy and compared performance with the existing base method. Specifically, we show that with little trade-off in completeness error, our method achieves huge improvement in accuracy.

Keywords: Multiview stereo, 3D-reconstruction, Semantic reconstruction, Point-cloud, Image.

---

1. Introduction

Multiview stereo (MVS) has been a classical problem in geometric computer vision. Given a collection of photographic images of a particular scene along with camera calibration is used as input in MVS. The output of the multiview stereo scheme is given usually in the form of a dense point cloud. Among such methods, which have been very popular is the [1]. In a different direction, solution schemes for semantic segmentation have evolved tremendously in the last decade. It may be interesting to the research community to look into how the new learning-based methods, especially in our present case, semantic segmentation schemes can improve the standard and popular methods of geometry estimation through appropriate adaptation to use the raw semantic information available from them. In this setting, we specifically consider the semantic segmentation on photographs that output semantic labels at the pixel level on the image planes. Then, we use these pixel-level semantic estimates as additional inputs to multiview stereo pipelines to study their influence.

2. Related Work

While there are several popular methods for multiview stereo proposed in the geometric vision community, the scheme [1] is one of the most widely used methods and implementation versions of the scheme are also available with open-source. Hence, we choose it to be the baseline geometry estimation scheme in our present study. Our current study is to investigate how we can adapt baseline geometry estimation to incorporate the per-view semantic information readily available from deep-neural-network schemes that readily provide semantic maps at outputs, to improve the geometry and also semantics. We assume that the possibly noisy semantic images are already available as input. While there are deep neural network-based methods that are proposed recently, they are not as scalable as [1].

3. Patch-based Multi-view Stereo

The PMVS [1] scheme builds up of three stages. In the first stage, the input images along with their camera pose and intrinsic calibration matrices are taken and initial stable seed patches are generated through sparse set of matched keypoints using Difference of Gaussian and Harris detector. In the second stage, the seed patches are expanded by adding new candidate neighbouring patches and then using visibility constraints, erroneous matches are filtered out. These expansion and filtering steps are repeated a number of times until no more expansion is expected.

4. Proposed Method

In our proposed method, we explicitly use the additional semantic information available at the input. Fig. 1 shows the pipeline where we can see that we have made use of the additional semantic information in all the blocks. This additional semantic information is propagated to both the seed generation stage and the expansion and filtering stage. Our methods use this semantics to aid in the matching and filtering procedure. The semantics in 3D is also itself refined in a joint fashion with the geometry estimation. As the output, we obtained a point cloud where we have the final estimates of the coordinates, normal, rendered colour, and the semantic for each point.

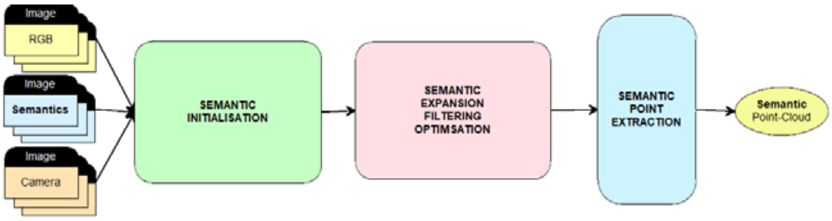


Fig. 1. Our proposed scheme that is based on the PMVS [1] method.

5. Experiments

We present performance evaluation experiments on a number of synthetic datasets obtained from. Specifically, we use synthetic data from 3DRMS Challenge 2018 dataset [2] and Slagboom en Peeters Aerial Survey dataset [3]. Fig. 2 shows the visual comparison on dataset from [2] where we have used the true semantics as input. A large number of outliers are present in the base method while output from our proposed method do not contain such outliers and simultaneously provides the semantics in 3D.

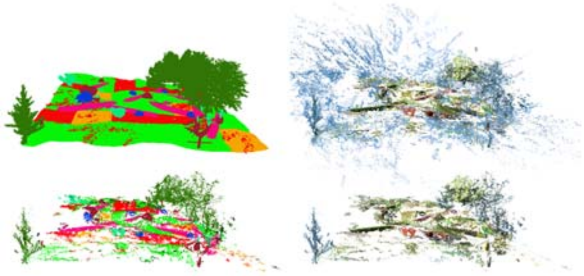


Fig. 2. Result on a dataset clear_0128 from [2]. Top left: Ground truth semantic point cloud. Top right: Output point cloud with colour map from [1]. Bottom left: Output semantic point cloud from our proposed scheme. Bottom right: Output point cloud with colour map from our proposed scheme.

In Table 1, we compare the quantitative geometry errors. Completeness errors measure the mean distances of points of ground truth to nearest points in outputs. Accuracy errors measure the mean distances of points in outputs to nearest points in ground truth. Our proposed method with little trade-off in completeness error achieves huge improvement in accuracy.

Table 1. Performance Comparison.

Data		Geometry Completeness Error (abs. scale)	Geometry Accuracy Error (abs. scale)
twilight 0001	Base [1]	**0.227**	0.716
	Our proposed	0.275	**0.045**
clear 0128	Base [1]	**0.202**	0.398
	Our proposed	0.206	**0.039**

Fig. 3 shows the visual comparison on real (noisy images and semantics) dataset from [3]. Our proposed method could able to generate more points and also a semantic point cloud.

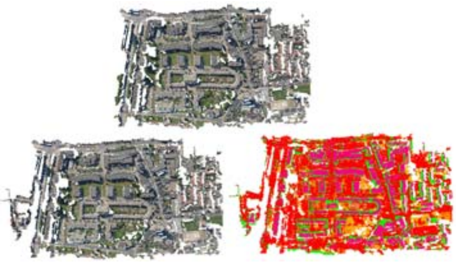


Fig. 3. Visual comparisons on real dataset from [3]. Top: Output point cloud with colour map from base method [1]. Bottom: Output point cloud with colour map (left) and semantic point cloud (right) from our proposed scheme.

6. Conclusions

We have presented our proposed method for joint geometry-semantics reconstruction under a multi-view framework. As our modification maintains the low complexity of the existing framework of PMVS and only requires additional input semantic segmentation maps per image, our method is easy to adapt as a modified plugin to already existing reconstruction software packages that use PMVS. We provided experimental results that show the usefulness and efficacy of our method.

Acknowledgements

The author is supported by the Department of Science and Technology, Government of India through the DST INSPIRE Faculty Award.

References

[1]. Y. Furukawa, J. Ponce, Accurate, dense, and robust multiview stereopsis, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 32, Issue 8, 2010, pp. 1362-1376.

[2]. R. Tylecek, et al., 3D Reconstruction Meets semantics: Challenge results discussion, ECCV Workshops, http://trimbot2020.webhosting.rug.nl/events/3drms/challenge/

[3]. Slagboom en Peeters Aerial Survey, http://www.slagboomenpeeters.com/3d.htm

(057)

Application of the Adaptive Wavelet Filtering to the Identification of Non-stationary Systems Parameters

A. Klepka
AGH University of Science and Technology, Department of Mechatronics and Robotics,
Al. Mickiewicza 30, 30-059 Krakow, Poland
E-mail: klepka@agh.edu.pl

Summary: Operational modal analysis (OMA) is a tool for identifying modal parameters of mechanical systems during operation. Modal analysis is used for linear systems with small or proportional damping. Applying this technique to other types of systems, including non-stationary systems, requires different, new procedures. The article focuses on the use of time-frequency filtering in combination with recursive filtering, and algorithm for identifying modal parameters based on operational measurements. The presented technique uses an adaptive signal filtering method to separate the frequency components, which reduces the model's order for parametric identification. Thanks to this, the estimation of mechanical parameters can be carried out based on relatively simple mathematical expressions. At the same time, the computational complexity and time needed to determine the parameters are significantly reduced. This is extremely important for real-time methods. The algorithm can be implemented on any hardware platform. The verification of the algorithm was carried out on simulation data and on data from a real non-stationary object.

Keywords: Identification of non-stationary systems, Wavelet transform, Time-frequency analysis, Wavelet filtering, Recursive identification.

---

1. Introduction

The operational modal analysis is the method to identify the modal parameters based only on response measurements of the structure. It is widely used in civil, mechanical, and aerospace engineering communities and applied to identify the modal parameters of such structures as buildings, towers, bridges, offshore platforms, airplanes, etc. [1]. Unfortunately, OMA has some limitations. One of them is the assumption that the system is time-independent. It means that the system's parameters (e.g., stiffness, damping) are not a function of time. This imposes the possibility of using this method only for stationary systems. In practice, many mechanical systems can operate in a non-stationary state. These include planes, rockets, bridges, cranes, or rotating machines operating at variable speeds. Classical modal parameter identification methods assume the eigenvector's and the eigenvalue's matrix invariance. It means that they cannot be directly used to identify the parameters of non-stationary systems. The solution to the problem is the assumption of quasi-stationarity in a given interval. It requires a compromise between the quality of the results (small number of samples) and resolution in the time domain. Another approach is the use of recursive methods of identification. In this case, the eigenvalue matrix can be estimated for every sample. The problem, in this case, is the selection of the appropriate order of the model. It directly translates into the quality of the estimates and computational complexity.

Despite the above limitations, OMA techniques enable the identification of parameters of non-stationary systems. They are widely used, e.g., as fault detection methods based on state subspace identification [2], modal filter [3], or 3D machine vision [4]. In addition, OMA is commonly used to identify: rotating machinery (e.g., turbines, helicopters, engines, etc.) in which the assumption of broadband excitation is not met [5], structures with variable mass [6], and geometry [7], systems with closely spaced modes [8], large structures [9] and systems with varying boundary conditions [10].

Although many applications use OMA techniques to identify modal parameters, it is still a highly complex issue in some cases. In addition to the limitations mentioned above, there are also practical problems in the implementation of OMA methods. These are:

- Length of data required for analysis – to get a good enough estimate of the results, it is necessary to ensure the right quality and quantity of data;
- Duration of the estimation procedure – very often, it is necessary to provide information about the dynamic state of the structure online. Long data series and complex (time-consuming) modal parameter estimation methods are not conducive to obtaining results quickly;
- Requires the intervention of an experienced operator to select the correct results from a set of solutions. This drawback restricts the use of these methods to offline modal parameter identification only;
- Influence of variability of working conditions on the identification of results;
- Selection of an algorithm for modal parameter estimation;
- Effect of the estimation procedure order on data obtained from a series of sub-experiments.

The presented method uses the Continuous Wavelet Transform (CWT) to separate the signal's frequency components. The modal parameters (frequency and modal damping) are then estimated for each component using the Recursive Least Square (RLS) algorithm. Thanks to signal decomposition, the order of the estimated model is known and equals two. This makes it possible to use simple procedures to determine the modal parameters.

2. Recursive Identification of Modal Parameters

The applied RLS algorithm contains the following steps:

Step 1: Acquiring the current system response signal from the measuring device.

Step 2: Estimating a priori prediction error based on evaluation from the previous iteration.

$$\hat{\varepsilon}(i) = y(i) - \phi^T(i)\hat{\theta}(i-1), \tag{1}$$

where *i* indicates successive number of sample. $\phi(i)$ is the regressor vector, $\hat{\varepsilon}(i)$ is the estimator of prediction error, $\theta(i)$ is the vector of model parameters.

Step 3: Gain vector $L(i)$ and covariance matrix $P(i)$ updating:

$$L(i) = \frac{\boldsymbol{P}(i-1)\phi(i)}{\lambda+\phi^T(i)\boldsymbol{P}(i-1)\phi(i)}, \tag{2}$$

$$\boldsymbol{P}(i) = \frac{1}{\lambda}\left[\boldsymbol{P}(i-1) - \frac{\boldsymbol{P}(i-1)\phi(i)\phi^T(i)\boldsymbol{P}(i-1)}{\lambda+\phi^T(i)\boldsymbol{P}(i-1)\phi(i)}\right], \tag{3}$$

where λ is the forgetting factor. For $i = 0$, $P(0) = \delta I$, where δ is a large natural number, for example. 10^6, I is the unit matrix.

Step 4: Updating vectors of estimated model parameters and observation vector

$$\begin{gathered}\hat{\theta}(i) = \hat{\theta}(i-1) + \\ +L(i)\left[y(i) - \phi^T(i)\hat{\theta}(i-1)\right],\end{gathered} \tag{4}$$

$$\phi(i) = \begin{bmatrix} y(i), y(i-1), \dots, y(i-n_A), \\ \hat{\varepsilon}(i), \hat{\varepsilon}(i-1), \dots, \hat{\varepsilon}(i-n_C) \end{bmatrix} \tag{5}$$

Step 5: Wait for the following sample and return to Step 1.

As a result of RLS algorithm, the vector of model parameters is obtained

$$\theta^T(k) = [-a_1, \dots, -a_{n_A}, 1, c_1, \dots, c_{n_C}], \tag{6}$$

where n_A and n_C are model order, $a_1, \dots, -a_{n_A}, c_1, \dots, c_{n_C}$ are model coefficients.

Fig. 1. presents a diagram of the proposed algorithm.

An essential element of the identification process based on the RLS algorithm is selecting the appropriate value of the forgetting factor λ. Large values of the forgetting coefficient (slightly less than 1) cause the algorithm to react slowly to changing modal parameters. When the value of the forgetting coefficient decreases, the algorithm reacts faster to changes in modal parameters. Problems related to the selection of the forgetting factor are not the subject of this work. In the cases described in the following chapters, the forgetting factor was selected based on a series of experiments conducted during the development of the method.

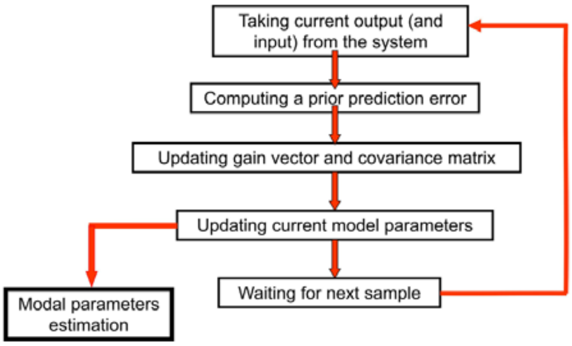


Fig. 1. Organization of the Recursive Least Square algorithm.

3. Wavelet Filtration

Mathematically, CWT can be described by the formula

$$\begin{gathered}\left(W_{g_i} x_i\right)(a_i, b_i) = \\ = \frac{1}{\sqrt{a_i}}\int_{-\infty}^{+\infty} x_i(t) g_i^{*}\left(\frac{t_i-b_i}{a_i}\right) dt,\end{gathered} \tag{7}$$

where *i* is a given time moment, b_i is a translation (displacement) representing the region, a_i is dilatation (expansion) or scale parameter, $g_i(t)$ is the basic wavelet function, and $x(t)$ is the analyzed signal. In the classical approach, wavelet filtering decomposes particular signal frequency components with resolution depending on the wavelet function parameters. Such a transformation enables the decomposition of frequency components depending on the scale parameter [3]. The schematic process is shown in Fig. 2.

There are two main advantages of using wavelet filtering. First, it is possible to filter very short signals. Secondly, there is no significant signal delay caused by the filtering process. Classical methods of narrowband filtering introduce a delay in the filtered signal depending on the order of the filter. In separating signal components with very close frequency values, it is necessary to use a very high-order filter. This makes it impossible to analyze short-duration signals. A similar problem also occurs in the wavelet transform, and it is called the “end result” [11]. Although the wavelet is focused on a given frequency and time, the window function describing a wavelet exceeds data length and wavelet coefficients in terminal regions based on incomplete information. This results in an inaccurate representation of the filtered signal in these regions. The impact of this effect is incomparably smaller than with narrowband filters.

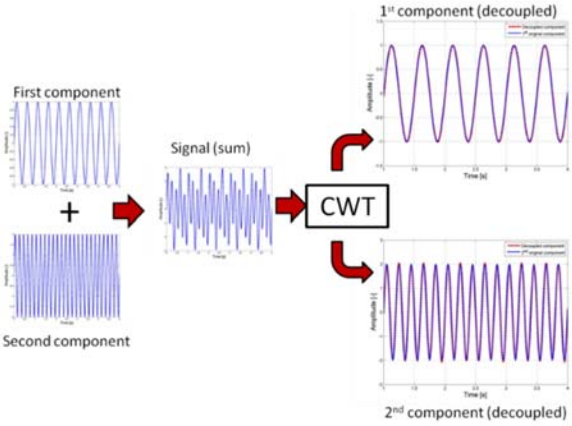


Fig. 2. Wavelet-based signal filtering.

4. The Algorithm

In the classical approach, wavelet transform decomposes the individual frequency components of the signal with time-frequency resolution, depending on the parameters of the wavelet function [12]. The choice of wavelet function parameters requires some compromise between filtering quality in the frequency and time domains. After decoupling, each signal component can be analyzed separately. This approach significantly reduces the computational effort due to the low order of the signal model. Also, tracking all eigenfrequencies and system damping coefficients is not always required.

The adaptive identification algorithm consists of two main parts. The algorithm's core is the RLS algorithm – which is responsible for the estimation of model parameters. The input data are filtered using wavelet transform, which reduces the model order and estimates the modal parameters based on analytical formulas. An integral part of the identification process is an adaptation of a wavelet filter that allows to tune the filter characteristics to the current value of the frequency. The organization of the proposed adaptive wavelet filtering procedure presents in Fig. 3.

The adaptation process compares the scale parameter *a* (wavelet frequency) with the frequency estimated from the RLS algorithm. Suppose the difference between the estimated frequency and the frequency corresponding to the current scale parameter is within a range (±δ). In that case, the identification process continues unchanged, but if the difference between the two frequencies is more significant than the assumed frequency, the frequency of the scale parameter is changed to a value corresponding to the frequency estimated from the RLS algorithm. The adaptation process and the scheme of the method with adaptive wavelet filtering are presented schematically shown in Fig. 3, where f_e, f_w and δ are respectively: the current estimated frequency, the frequency corresponding to the scale parameter (wavelet frequency), and the adaptive step.

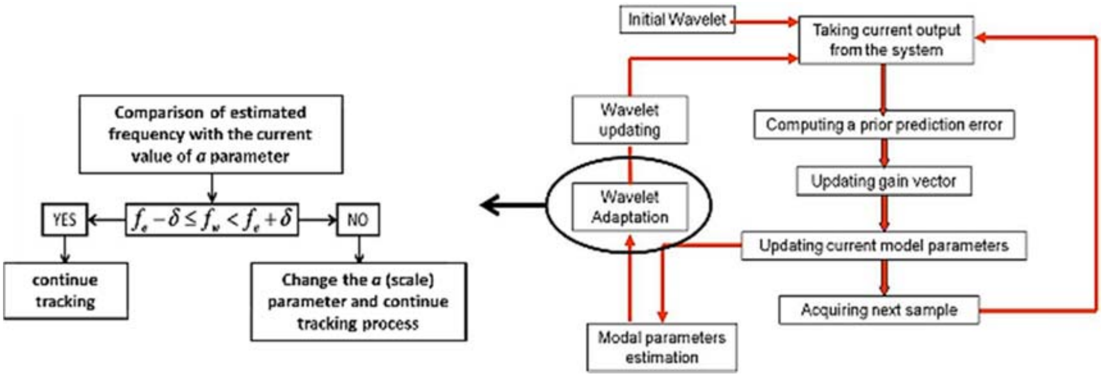


Fig. 3. Organization of proposed adaptive wavelet filtering procedure.

5. Results

Firstly, A two-degree of freedom system with varying stiffness, excited by the white noise, was used in the simulation. The stiffness parameters of the system were described as follows:

$$k_1 = \begin{cases} 26000 & for \quad 0 < t \le t_1 \\ 26000 * e^{0,001} t & for \quad t_1 < t \le t_2 \\ 59722 & for \quad t_2 < t \le t_3 \end{cases},$$

$$k_2 = \begin{cases} 8000 & for \quad 0 < t \le t_1 \\ 8000 * e^{0,001} t & for \quad t_1 < t \le t_2 \\ 3591 & for \quad t_2 < t \le t_3 \end{cases}$$

An example of the time history of the system response and the stiffness parameters are presented in Fig. 4.

Examples of parameter estimation results using the proposed method are shown in Fig. 5 and Fig. 6, where comparisons of the identification results using non-adaptive and adaptive filtering are presented.

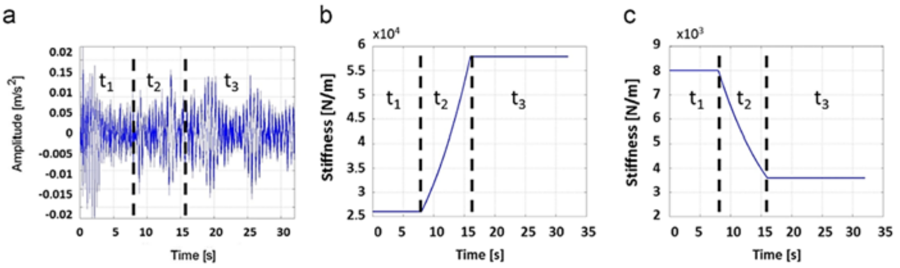


Fig. 4. (a) System response for white noise excitation, (b) stiffness changes for mode 1 and (c) stiffness changes for mode 2.

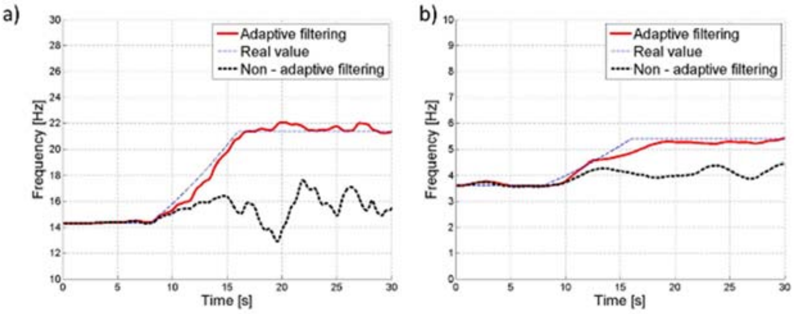


Fig. 5. Comparison of identification results: a) natural frequency – mode1, b) natural frequency – mode 2.

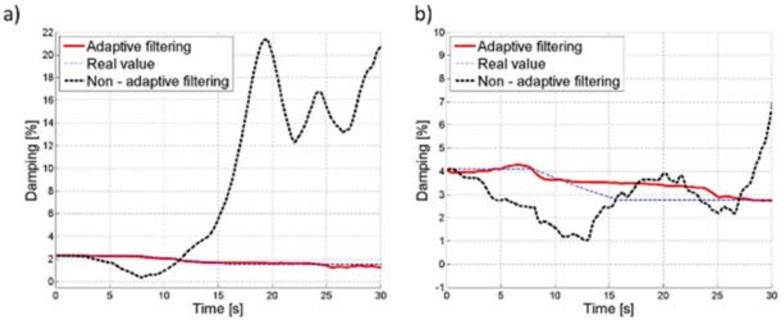


Fig. 6. Comparison of identification results: a) damping ratio – mode1, b) damping ratio – mode 2.

As can be noticed, the method with an adaptive wavelet filter gives better results. The best results were achieved for the natural frequency of the system. For the damping ratio, the results are satisfactory but it should be remembered that this parameter is very hard to identify, especially when the value of the damping ratio is large (mode 2).

The second experiment was performed on the laboratory test rig. The test bed was built out of three main parts: a frame, a cart, and two metal bellows (Fig. 7). Two metal bellows are mounted between the cart and the frame. The friction between the cart and the frame has been eliminated thanks to air bearings.

An electromagnetic shaker and a signal generator were used to excite the structure. The white noise signal was used as an excitation signal. During the experiment, the pressure in the metal bellows was changed. As a result of system stiffness changes, the natural frequency of the system was shifted. The time history of the system response, the result of the natural frequency identification, and the adaptation process of the wavelet function are presented in Fig. 8. Also, in this case, the results obtained with the use of adaptive methods are considerably better than the ones obtained by non-adaptive wavelet filtering. The algorithm reacts much faster to the changes in the system's natural frequency. Both algorithms were run with the same initial parameters.

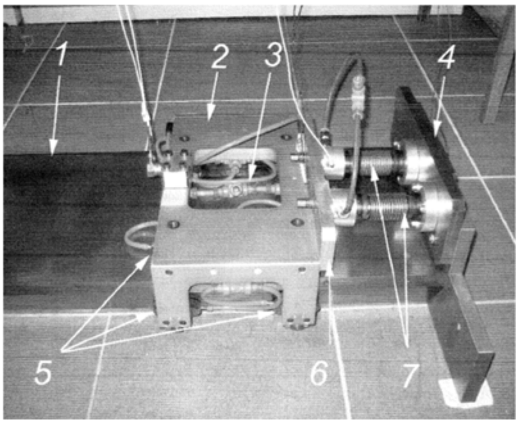


Fig. 7. Equipment for experiment prepared to free vibration run test 1. Frame, 2. Cart on the air bearings, 3. Air bearing hoses, 4. Metal bellows bracket mounted to the frame, 5. Air bearings, 6. Metal bellows bracket mounted to the sliding cart, and 7. Two metal bellows.

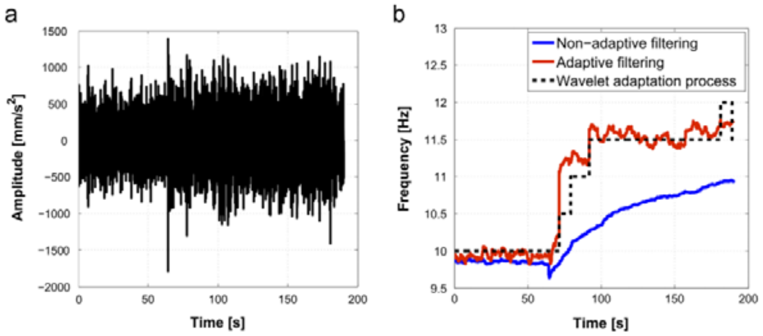


Fig. 8. System response (a), comparison of identified natural frequency of the system (b).

6. Conclusions

Application of the adaptive wavelet filtration to recursive identification of modal parameters has been investigated. The performed test confirmed that the wavelet transform is a useful tool to support the identification process of non-stationary systems. The adaptive wavelet filtration allows the separation of the signal frequency components and reduces the model order of the analyzed signal. This approach significantly reduces the computation time of modal parameters thanks to analytical formulas for damping ratio and natural frequencies. It facilitates the hardware implementation of the algorithm. The application of signal decomposition enabled the determination of the initial parameters of the RLS algorithm. This approach increased the speed of convergence of the algorithm. In the author's opinion, the algorithm has two main advantages. First, it is insensitivity to the initial parameters of wavelet function and the possibility of their adaptation to the system's current state; thus, the algorithm correctly identifies the modal parameters of non-stationary systems, although their changes occur rapidly, and the damping ratio has a large value. The second is the possibility of filtration of very short signals with closely spaced frequency components. Filters with narrow bandwidths introduce a delay to the signal.

This delay is related to the order of the applied filter. For signals with closely spaced frequency components, the order of the filter can be very high. It is difficult (or even impossible) to use that filters to identify processes where parameters change rapidly, and the acquisition system allows to collect only a small number of samples.

References

[1]. T. Uhl, W. Lisowski, P. Kurowski, In-Operation Modal Analysis and Its Application, *AGH University of Science and Technology*, Krakow, 2001.

[2]. M. Basseville, L. Mevel, A. Vecchio, B. Peeters, H. Van der Auweraer, Output-only subspace-based damage detection – Application to a reticular structure, Structural Health Monitoring, in *Proceedings of the 1st European Workshop on Structural Health Monitoring (EWSHM'03)*, 2003, pp. 161-168.

[3]. P. Kurowski, K. Mendrok, T. Uhl, An application of operational modal analysis in modal filtering, *Journal of Physics. Conference Series*, Vol. 305, 2011, 012094.

[4]. P. Kuhut, P. Kurowski, Application of modal analysis supported by 3D vision-based measurements, *Journal of Theoretical and Applied Mechanics*, Vol. 47, Issue 4, 2009, pp. 868-870.

[5]. R. Pintelon, B. Peeters, P. Guillaume, Continuous-time operational modal analysis next term in the presence of harmonic disturbances – The multivariate case, *Mechanical Systems and Signal Processing*, Vol. 24, Issue 1, 2010, pp. 90-105.

[6]. M. Goursat, M. Dohler, L. Mevel, P. Andersen, Crystal clear SSI for operational modal analysis of aerospace vehicles, in *Proceedings of the 28th International Modal Analysis Conference (IMAC'10),* 2010, pp. 1421-1430.

[7]. M. D. Spiridonakos, S. D. Fassois, Parametric identification of a time-varying structure based on vector vibration response measurements, *Mechanical Systems and Signal Processing*, Vol. 23, Issue 6, 2009, pp. 2029-2048.

[8]. A. Agnenia, L. B. Cremaa, G. Coppotell, Output-only analysis of structures with closely spaced poles, *Mechanical Systems and Signal Processing*, Vol. 24, Issue 5, 2010, pp. 1240-1249.

[9]. P. Andersen, R. Brincker, M. Goursat, L. Mevel, Automated modal parameter estimation for operational modal analysis of large systems, in *Proceedings of the 2nd International Operational Modal Analysis Conference (IOMAC'07),* 2007, pp. 299-308.

[10]. V. H. Vu, M. Thomas, A. A. Lakis, L. Marcouiller, Operational modal analysis of non-stationary mechanical systems by short-time autoregressive (star) modelling, in *Proceedings of the 3rd International Conference on Integrity, Reliability and Failure (IRF'09)*, 22-24 July 2009, P0212.

[11]. T. Kijewski, A. Karem, On the presence of end effects and their melioration in wavelet-based analysis*, Journal of Sound and Vibration*, Vol. 256, Issue 5, 2002, pp. 980-988.

[12]. T. Uhl, A. Klepka, Application of wavelet transform for identification of modal parameters of nonstationary systems, *Journal of Theoretical and Applied Mechanics*, Vol. 43, Issue 2, 2005, pp. 277-296.

(058)

The Tone of Superficial Electromyography Provides Characteristics that Allow the Discrimination of Hand Gestures

C. L. Sandoval-Rodriguez [1], A. F. Jimenez-Quezada[1], D. M. Reyes-Bravo [2],
N. A. Castillo-Zambrano [1] and O. Lengerke [1]
[1] Unidades Tecnológicas de Santander, Calle de los estudiantes 9-82, 680005, Bucaramanga, Colombia
[2] Universidad Autonoma de Bucaramanga, Av. 42 #48 – 11, 680002, Bucaramanga, Colombia
Tel.: (+57) 607 6917700
E-mail: csandoval@correo.uts.edu.co

Summary: Many studies related to electromyography focused functional prostheses. Also, to the diagnosis of pathologies and recognition of movements, thus allowing the development of human-machine interfaces. The objective of this work was to estimate the ability to discriminate between palmar and cylindrical moves, using three types of characteristics in the time domain (maximum peak of the signal, average signal, root mean square (RMS) value, and total energy Teager Kaiser-Skteo) at through variance analysis with original signals and with enveloping-tone. The work showed results where it is evident that there is a separation between the movements analyzed for all the characteristics obtained from the tone of the sEMG as opposed to the same obtained from the original sEMG. Therefore, for the features analyzed in the time domain, it is much easier to differentiate the palmar gesture from the cylindrical when it takes from the sEMG tone than when from the original sEMGs.

Keywords: sEMG, Sliding windowing, Pattern recognition.

1. Introduction

In daily life, the movements of the hand represent the freedom of each person. Some time ago, the works that used analysis and processing of electromyography signals (sEMG) were associated with facilitating and helping people who do not have upper extremities or cerebrovascular diseases [1-4] and thanks to that, have awakened a considerable interest in the development interfaces man-machine, being one of the most researched subjects over the last few years [5]. Electromyography studies use different alternatives for the processing and the characteristics extraction associated with each movement [2, 6-10].

RMS is one of the most widely used statistical measures for sEMG signal features extraction as it has shown good results when used in classifier such as: Support Vector Machine (SVM) or Linear Discriminant Analysis (LDA), with an accuracy of around 90 % [11].

In signal processing, windowing techniques are used to obtain relevant information in the time domain. This information provides unique features that allow movements and gestures to be classified [9, 12-15]. In the present work, we used the technique of sliding windowing to obtain the tone of surface electromyography signals (sEMG). And from these, extract characteristics for classification. Finally, an analysis of variance has been carried out to estimate the discrimination capacity of the feature extracted to both original signals and tone signals.

The rest of paper will be structured in the following sections: In Section 2, the movements studied are shown, the obtained of envelope, the windowing and the characterizes extractions, the results is presented in Section 3, the discussion is presented in Section 4 and ultimately, the conclusions are presented in Section 5.

2. Materials and methods

2.1. Database

The following movements were studied: palmar and cylindrical (Fig. 1). There are 60 records for each move of healthy subjects. We used the ML880 Power-lab data acquisition system and the ML135 Dual Bio Amp signal conditioner and observed them on a computer with the ADINSTRUMENTS Labchart interface [13]. We applied a sequence of treatments to reach the proposed characteristics.

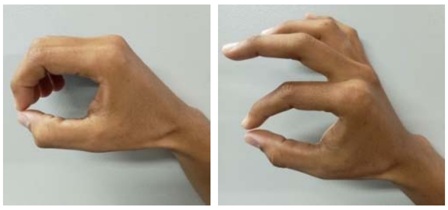


Fig. 1. Cylindrical type and palmar type movement images.

2.2. Signal Processing

We obtained the envelope (Fig. 2B) using the Hilbert transform since this transform does not alter the representation of the signal from one domain to another. In this sense, the envelope is just function of original signal [12]. For this case, we used the upper envelope, see Equation (1).

$$H(t) = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{f(u)}{u-t}du = -\frac{1}{\pi t} * f(t), \quad (1)$$

where H (t) is the upper envelop of original sEMG signal. Subsequently, the envelope is windowed to improve the extraction of characteristics [9]. We obtained the sEMG tone with a moving average of the samples in windows of 200 ms [9, 13] (Fig. 2C).

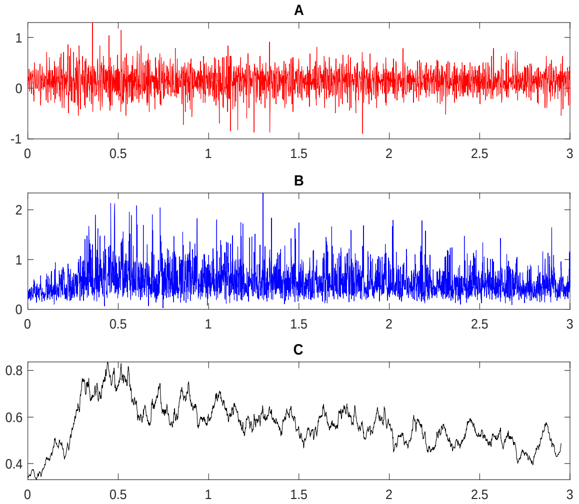


Fig. 2. sEMG signal processing. (A) Original signal. (B) Upper envelope (via Hilbert transform). (C) sEMG tone (using sliding windows).

2.3. Features

The expression for the characteristics used in this article are shown below. Here x_i is the EMG signal.

RMS Root mean square: allows to classify the signal assessing the contents of energy, relating your range in an interval given [1], is expressed as:

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} {x_i}^2} \quad (2)$$

Dc: Also is known as Mean Value and is expressed as:

$$Dc = \frac{1}{N}\sum_{i=1}^{N} x_i \quad (3)$$

Pico: It recognizes as the maximum value that achieve by the signal, as follows:

$$\text{Pico} = \max(x_i) \quad (4)$$

SKTEO total energy teager Kaiser-skteo: it permits as to increase the relation between the noises vs. signal to have a better detection of the muscular activity. For this (1), the equation is defined as:

$$STKEO = \sum_{i=2}^{N-1} {x_i}^2 - x_{i+1}x_{i-1} \quad (5)$$

2.3. Statistic Analysis

It's used the method Analysis of Variance (ANOVA) in order to observe the capacity of discrimination and separation between individuals and the movements, being useful the boxplot for this procedure. A comparison has been made between the characteristics of the two be study gestures, being the non-processing signal (Fig. 3A) and the processing (tone) signal (Fig. 3B), the result of comparison can be seen in the Fig. 3. In this step, the boxes were placed in parallel (boxplot) to see if they overlap and afterwards, to obtain the different values of the medians and Interquartile range (IQR) recorded in the Tables 1 and 2.

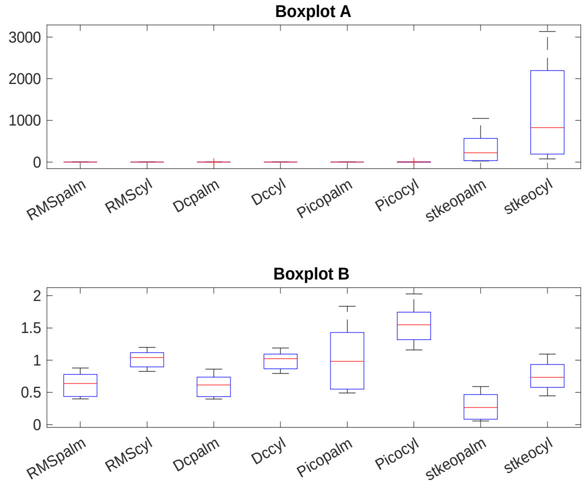


Fig. 3. Boxplot features (A) original signal. (B) Processed signal.

Table 1. Median and IQR, Features with raw signals.

Features	Movement type	Median	IQR
RMS	Palm	0.2784	(0.1726,0.4232)
	Cylindrical	0.5018	(0.2783,0.8273)
Dc	Palm	0.1433	(0.1426,0.1444)
	Cylindrical	0.1437	(0.1430,1.1444)
Peak	Palm	1.1970	(0.5083,1.9240)
	Cylindrical	2.8933	(1.1460,3.8625)
Skteo	Palm	221.4128	(33.3998,569.9844)
	Cylindrical	824.3306	(189.4301,2195.361)

Table 2. Median and IQR, Features with processed signals.

Features	Movement type	Median	IQR
RMS	Palm	0.63683	(0.4353,0.7780)
	Cylindrical	1.0395	(0.89586,1.116)
Dc	Palm	0.61597	(0.4343,0.7381)
	Cylindrical	1.0217	(0.8657,1.0942)
Peak	Palm	0.98404	(0.5513,1.4281)
	Cylindrical	1.5479	(1.3173,1.7438)
Skteo	Palm	0.26586	(0.0835,0.4663)
	Cylindrical	0.73579	(0.5770,0.9319)

3. Results

The comparison between the populations of each movement is shown in Fig. 3. In the upper panel the results of the characteristics extracted from the original signal. In the lower panel the results after obtaining the envelopes. The evaluated characteristics correspond from left to right: RMS, DC, Peak and Skteo. Here it is observed that the medians and interquartile of the characteristics in each movement overlap in each evaluated feature. However, when we obtain the features from the envelope, the populations at each move are widely separated.

The medians values and IQR of each movement and characteristics of the non-processing signal were annotated in Table 1, where it has been tested that there is overlap and that, in the RMS, Pico and DC characteristics, the values are lower compared to the Stkeo values.

Likewise, it's found a separation in the RMS, Peak and Dc characteristics values of the processing signal (tone), whereas, in the Pico characteristic continue with overlap (Table 2).

4. Discussion

In the previous work, it was extracting features from sEMG in time domain [11]. The features extracted was: RMS, mean value zero crossing times, slope sign change. The objective was to classify hand gestures. These features are the input for a neural network, where the output goal is above 87.5 %. Although the result is good, feature extraction it was made on the stochastic signal (original sEMG). In our work, we demonstrate that the discrimination increases for tone (envelope) sEMG. Similarly, other works has employed the envelope of sEMG to evaluates dynamic behavior (velocity and force prediction) with good results [13, 16].

Also, we observed that all evaluated characteristics have low overlap between evaluated hand gestures.

In order to emphasize the aforementioned, we have calculated the percentage of interquartile difference (PIQD) according to equation (5)

$$PIQD = \left(\frac{75th\ percentile\ palm - 25th\ percentile\ cylindrical}{75th\ percentile\ palm}\right) * 100 \qquad (6)$$

Therefore, if PIQD < 0, there is overlap, otherwise for positive PIQD there is no overlap.

In Figs. 4 and 5 it can see that the processing sEMG processing favors separability of populations. Therefore, the use of sEMG tone can markedly improve the accuracy in hand gesture scoring. On the other hand, after obtaining the envelope, the characteristic associated with the maximum value (peak) may not be appropriate for the classification, see Fig. 5. Here it is seen that the PIQD is negative. However, the other three characteristics contribute to the separation of the two hand gestures that have been analyzed. With this simple processing to obtain the sEMG pitch, the chance of success in hand gesture classification is increased, even using simple techniques such as case-based reasoning (CBR) [17-19] and linear discriminant analysis (LDA) [6, 20, 21]. Finally, we have classified only two hand gestures, and we provide relevant knowledge contributing to the state of the art in the context of the classification of hand movements and encourages the use of this process with databases with more classes or movements such as those proposed in the background [5].

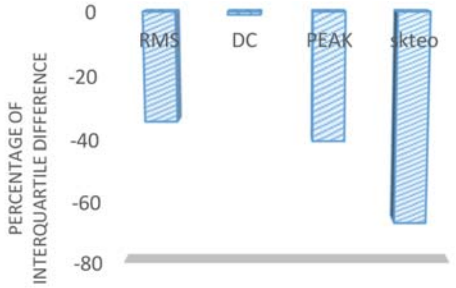


Fig. 4. PIQD for each feature obtained from original sEMG.

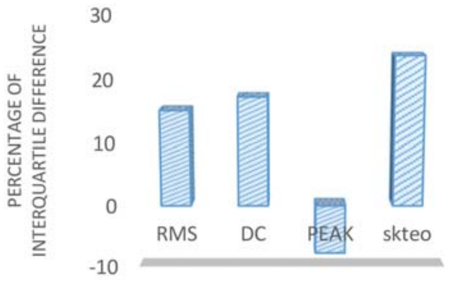


Fig. 5. PIQD for each feature obtained from envelope sEMG.

5. Conclusion

We proposed a method for feature extraction using the sEMG tone. We observed that separation capacity between both analyzed gestures (palm and cylindrical) increases remarkably using temporal characteristics for tone sEMG compared to the original signals. the peak feature is not recommended for use as a descriptor. The total energy teager Kaiser shows a high differentiation and it can be more appropriate to gestures classification.

References

[1]. C. Rocha León, Sistema de reconocimiento de patrones de agarre en base a un solo canal de electromiografía superficial como medio de entrenamiento y guía para los pacientes con amputación transradial en la fundación FABRILAB, 2020, http://hdl.handle.net/20.500.12495/3944.

[2]. J. Too, A. R. Abdullah, T. N. S. T. Zawawi, N. M. Saad, H. Musa, Classification of EMG signal based on time domain and frequency domain features, *Int. J.*

Hum. Technol. Interact., Vol. 1, Issue 1, 2017, pp. 25-30.

[3]. W.-J. Li, C.-Y. Hsieh, L.-F. Lin, W.-C. Chu, Hand gesture recognition for post-stroke rehabilitation using leap motion, in *Proceedings of the International Conference on Applied System Innovation (ICASI'17)*, 2017, pp. 386-388.

[4]. R. Ma, L. Zhang, G. Li, D. Jiang, S. Xu, D. Chen, Grasping force prediction based on sEMG signals, *Alexandria Eng. J.*, Vol. 59, Issue 3, 2020, pp. 1135-1147.

[5]. P. Kaczmarek, T. Mańkowski, J. Tomczyński, putEMG – A surface electromyography hand gesture recognition dataset, *Sensors*, Vol. 19, Issue 16, 2019, 3548.

[6]. A. C. B. Rezende, G. Lopes Filho, F. H. T. Vieira, Aplicação da Análise Discriminante Linear (LDA) para Classificação de Sinais Eletromiográficos (EMG) de Movimentos da Mão, in *Anais da VII Escola Regional de Informática de Goiás*, 2019, pp. 351-360.

[7]. C. M. D. Acevedo, A. L. J. Mogollón, Optimización y clasificación de señales EMG a través de métodos de reconocimiento de patrones, *Iteckne*, Vol. 10, Issue 1, 2013, pp. 67-76.

[8]. L. S. Vega Escobar, Metodología basada en entrenamiento automático para el reconocimiento del movimiento individual de los dedos de la mano usando análisis de señales electromiográficos de superficie, 2018, http://hdl.handle.net/20.500.12622/293.

[9]. A. Ullah, S. Ali, I. Khan, M. A. Khan, S. Faizullah, Effect of analysis window and feature selection on classification of hand movements using EMG signal, in *Proceedings of the Intelligent Systems Conference (IntelliSys'20)*, Vol. 3, 2020, pp. 400-415.

[10]. N. Jose, R. Raj, P. K. Adithya, K. S. Sivanadan, Classification of forearm movements from sEMG time domain features using machine learning algorithms, in *Proceedings of the IEEE Reg. 10 Annu. Int. Conference (TENCON'17)*, 2017, pp. 1624-1628.

[11]. Á. L. Valdivieso Caraguay, J. P. Vásconez, L. I. Barona López, M. E. Benalcázar, Recognition of Hand Gestures Based on EMG Signals with Deep and Double-Deep Q-Networks, *Sensors*, Vol. 23, Issue 8, 2023, 3905.

[12]. C. O. Jiménez Tintaya, Procesamiento Digital de Señales Sísmicas con MATLAB, *Revista De Investigación De Física,* 10, 02, 2007, pp. 45–50.

[13]. C. L. Sandoval-Rodriguez, A. C. Pita-Mejia, Villamizar-Mejia R, B. E. Tarazona-Romero, O. Lengerke-Perez, Model to Relationship the Speed of Hand Movements with the SEMG Signals from the Forearm, *Journal of Physics: Conference Series*, Vol. 2224, 2022, 012094.

[14]. W. Seok, Y. Kim, C. Park, Pattern recognition of human arm movement using deep reinforcement learning, in *Proceedings of the Int. Conference Inf. Netw.*, 2018, pp. 917-919.

[15]. M. Kim, K. Kim, W. K. Chung, Simple and Fast Compensation of sEMG Interface Rotation for Robust Hand Motion Recognition, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 26, Issue 12, 2018, pp. 2397-2406.

[16]. C. L. Sandoval-Rodriguez, R. V. Pita-Mejia, B. E. Tarazona-Romero, A. D. Rincon-Quintero, A. J. Rodriguez-Nieves, Relationship between force signal and superficial electromyographic signals associated to hand movements, *Periodicals of Engineering and Natural Sciences*, Vol. 11, Issue 1, 2023, pp. 64-73.

[17]. C. Cárdenas, C. Sandoval, A. Rincón, D. Galván, H. Téllez, Data Collector Design for Vibration Analysis by Raspberry pi 3B Embedded System Means for Industrial Applications, *Journal of Physics: Conference Series*, Vol. 2224, Issue 1, 2022, 12032.

[18]. C. L. Sandoval-Rodriguez, J. G. A. Villabona, C. G. Cárdenas-Arias, A. D. Rincon-Quintero, B. E. Tarazona-Romero, Characterization of the mechanical vibration signals associated with unbalance and misalignment in rotating machines, using the cepstrum transformation and the principal component analysis, *IOP Conference Series: Materials Science and Engineering*, Vol. 844, Issue 1, 2020, 12057.

[19]. C. L. S. Rodriguez, E. A. Correa-Quintana, B. E. Tarazona-Romero, A. D. Rincón-Quintero, J. G. Maradey-Lazaro, Characterization of mechanical vibrations in a metal structure using the transform Cepstrum, *Period. Eng. Nat. Sci.*, Vol. 9, Issue 4, 2021, pp. 767-777.

[20]. M. Ortiz-Catalan, B. Håkansson, R. Brånemark, Real-time and simultaneous control of artificial limbs based on pattern recognition algorithms, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 22, Issue 4, 2014, pp. 756-764.

[21]. X. Ren, Y. G. Soo, M. Odagaki, F. Duan, SEMG-based hand motion recognition system using RMSR and AR model, in *Proceedings of the Chinese Control Conference (CCC'17)*, 2017, pp. 5410-5415.

(059)

Building Kalypso: The Construction

N. Manos, M. Vasileiou and E. Kavallieratou
University of the Aegean, Department of Information and Communication Systems Engineering,
Palama 2, 83200 Karlovasi, Samos, Greece
Tel.: +30 2273082280
E-mail: kavallieratou@aegean.gr

Summary: Underwater vehicles can play an essential role in fish farming and their construction can be a challenging aspect of their development owing to the necessity of waterproofing electronic components. This paper describes the construction and basic framework of a robotic vehicle named "Kalypso," which was developed to inspect nets in fish farms. Kalypso can distinguish between clean areas on the net and areas that are either torn or covered in algae, based on an algorithm. Also, it is equipped with sensors to estimate position, depth, temperature and leakage, which can detect water in the watertight housing. The watertight part of the robot includes cable plugs, some of which are detachable to facilitate the connection of various devices, such as sensors and lights. Kalypso has two cameras, one facing the front and another facing the bottom. Furthermore, the robot has underwater ultrasonic sensors, which measure the distance from the environment, and LED lights on the front for increased visibility in low-light environments. Several tests were carried out, both inside fish farm nets and in the open sea.

Keywords: Fish farms, Leak sensor, Ultrasonic sensor, Artificial intelligence, Autonomous Unmanned Vehicle (AUV), Plugs.

---

1. Introduction

Fish farming is an essential aspect of the world's food production, and the growth of the industry has led to the development of innovative technologies to improve the efficiency and sustainability of the process.

In this context, the development of Kalypso is a significant step towards the use of underwater robotic vehicles in the fish farming industry. Kalypso is a cost-effective and efficient underwater vehicle used to inspect net cages in fish farms featuring stable design, flexible movement capabilities, and advanced sensors. Its ability to distinguish between clean and damaged areas on the net is a critical feature, as it can help identify potential issues and prevent the spread of diseases in the fish population. Overall, Kalypso is a significant technological innovation that can have a positive impact on the sustainability and efficiency of the fish farming industry.

The research presented in [1] deals with the second phase of AUV rendezvous with a docking station and the process of docking itself. Further research [2] analyses an optimization framework for the design of AUVs using two state-of the-art population-based optimization algorithms, namely non-dominated sorting genetic algorithm (NSGA-II) and infeasibility driven evolutionary algorithm (IDEA). Another research [3] presents the design of a low cost AUV and the implementation for autonomy, acoustic and environmental research in stand-alone configuration. In addition, the research described in [4] progresses in the definition of the requirements needed to be applied to these kinds of vehicles, considering the low cost of construction and operation as one of the main goals. Furthermore, LoCO (Low Cost Open) AUV, a general-purpose, vision-guided AUV with rated depth of 100 meters that is open source is presented [5]. Another study analyses an AUV which uses main board Arduino Mega with 8 bit Atmel microcontroller [6]. Similarly, the development and design of an open source and low cost AUV for optical mapping of the deep sea is presented in [7]. Moreover, the research [8] deals with localization and navigation system for low cost AUVs that are limited to a micro-electro-mechanical system (MEMS).

Besides, a study [9] examines the development of a new robotic vehicle for monitoring underwater and presents experimental results. Research introduced in [10] proposes a new low cost AUV-based method of seafloor observation with three important attributes to achieve surveys. Initially, it can be applied in highly rugged terrains, it can also produce a 3D image of seafloor based on a depth sensor and finally it can estimate the flow velocity field. More specifically, another study [11] presents hydrographic research with the objective of the implementation of AUV. In addition, research [12] deals with the development of a modular, low-cost mobile networking infrastructure for underwater domains. A study [13] proposes how a low cost commercial ROV can be as a foundation for developing versatile and affordable AUVs, while [14] presents a platform designed to encourage broader use of AUVs to reduce the timeline for new technology development. Research [15] describes an AUV named Blue Duck II, which is semi-modular with 3D printable parts, low cost and easy to modify based on needs. Similarly, a low-cost underwater vehicle is presented in [16] intended for shallow-depth inspections.

What is more, a study [17] analyses an effective cruise speed estimation strategy for low cost AUVs, which is based on the motor feedback information. In addition, a low-cost AUV Alba-13 which is the second prototype with the purpose of validation as a sampling platform, is presented [18]. Research [19] describes an AUV named Folaga which is characterised by an actuation system that integrates the actuator set for propulsion with buoyancy.

Study [20] deals with a new low-cost open-source vehicle called "Hipp oCampus X". It has a modular software and hardware architecture and is adaptable to any mission. Finally, research [21] presents Pirajuba, a low-cost AUV with innovative components that emphasizes on the hardware and software control architectures.

The main objective of this paper is the analysis of the construction and overall preparation of Kalypso ROV which can be seen in Fig. 1. Section 2 introduces the robotic vehicle and presents in detail its electronic parts (e.g., sensors, motors, board, etc.). Section 3 describes the experimental results performed in Kefalonia fishery. Finally, Section 4 concludes the paper.

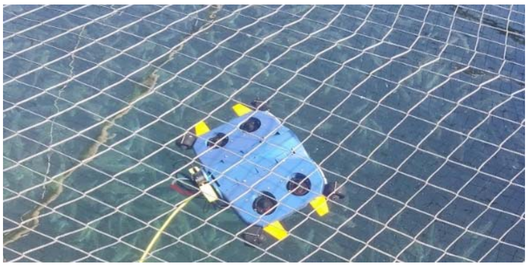


Fig. 1. Kalypso in Kefalonia Fishery.

2. The Robotic Vehicle

The robotic vehicle, named "Kalypso", was designed and constructed to inspect nets in fish farms. In more detail, its components are quoted as follows:

- 3D skeleton – printed from blue, black and orange PLA plastic;
- 1 Acrylic tube 6" with 4 O-rings and 2 flanges for absolute tightness and protection of electronic circuits (BlueRobotics [22]);
- Specially improved 11 penetrator plugs with O-rings in the form of a gland to insulate the cables entering and exiting the central tube (BlueRobotics [22]);
- 3 Additional plugs (BlueTrailEngineering [23]) that can be inserted and removed from the robot for various functions, such as charging, adding sensors and other accessories (Fig. 2);
- 8 Three-phase motors (4 clockwise and 4 counterclockwise);
- 8 Motor drivers ESCs (BlueRobotics [22]);
- 1 Raspberry Pi 3 mainboard;
- 1 Pixhawk board which is responsible for the robot's navigation;
- 1 Mini PC Intel Core i7, 16 GB RAM, 512 GB SSD, Nvidia GeForce GTX 1660 Ti for computer vision;
- Sensors:
  - 1 Digital water pressure sensor, which can measure up to 300 m depth (BlueRobotics [22]);
  - 1 Leak sensor with 4 detectors (BlueRobotics [22]);
  - 1 Digital battery voltage and current meter and correct operation (BlueRobotics [22]);
  - 1 Temperature sensor (BlueRobotics [22]);
  - 3 Ultrasonic distance sensor (2 front and 1 left) which can measure distances up to 100 m and 300 m depth rating (BlueRobotics [22]);
  - Front camera with 1920×1080 resolution and bottom camera with 720 MP.

Finally, it includes:

- Power supply: 2×14.8 V, 10000 mAh and 30 C Lithium batteries;
- 12 v led lights.

Fig. 2. Plug Connectors.

The robotic vehicle is equipped with advanced sensors, including leak sensor, underwater ultrasonic sensor, dual camera front and down and led lights from led stripes, which are insulted with silicone. The sensors are essential for maintaining the vehicle's stability and providing critical data for the inspection process. Kalypso can distinguish between clean areas on the net and areas that are either torn or covered in algae, based on an algorithm partitioning. An entropy filter is first applied to the image, and then the different regions are separated using multi-Otsu thresholding. This allows the vehicle to identify potential issues and prevent the spread of diseases in the fish population.

Below is the design and placement of the three ultrasonic distance sensors:

- The two sensors on the right and left measure the distance in the straight line of the robot;
- The third sensor in the center turns left and detects distance to the left of the robot.

The design and construction of Kalypso were carefully considered to ensure that the vehicle could operate efficiently in the harsh and unpredictable underwater environment.

The wiring and grouping of the plugs (15 in total) are shown in Fig. 3 and explained as follows:

- 4-pin plug, 2 of the pins are used for charging the batteries and 2 are for powering the LEDs. So, when the robot needs charging, the LED lights plug is disconnected and the charger is connected;
- 8-pin connector, of which 3 are needed for the balanced charging of the lithium battery cells, and 5 are for the connection of the arm;

- 8-pin connector, 4 of the pins are used for the wired networking of the mini pc and the remaining 4 for the networking of the Raspberry pi. A special adapter is connected to establish communication between them;
- 8 gland plugs for the internal connection of the motors;
- Plug – switch for central activation / deactivation of the robot;
- Pressure and temperature sensor plug;
- Distance sensor plug;
- Bottom camera connector.

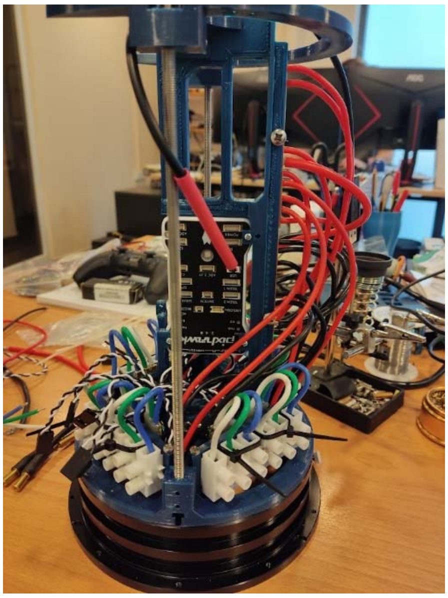


Fig. 3. Hardware Connections.

3. Experimental Results

One key aspect of the testing involved evaluating Kalypso's ability to identify damaged areas on the net using the algorithm partitioning. The vehicle was able to distinguish between clean areas on the net and areas that were either torn or covered in algae with high accuracy. This capability is essential for identifying potential issues and preventing the spread of diseases in the fish population.

The leak sensor was also evaluated during the initial testing, and it proved to be a critical component in detecting water leakage in the watertight housing and alerting the operator. This capability is essential for ensuring the safety of the vehicle and preventing potential damage or malfunction.

Overall, the experimental results from the initial testing phase of Kalypso demonstrated the effectiveness and potential of the robotic vehicle in the fish farming industry. The vehicle's stable design, advanced sensors, and precise movement capabilities make it a cost-effective and efficient solution for inspecting nets in fish farms. The ability to identify potential issues and prevent the spread of diseases in the fish population is critical to the success of the industry, and Kalypso's success in this area provides a promising model for future innovations in the field.

4. Conclusions

Overall, the development of Kalypso demonstrates the immense potential of robotics technology to address some of the most pressing challenges facing the fish farming industry. As this technology continues, Kalypso AUV is low cost and adaptive with simple materials and 3D printed parts which anyone can assemble. The tested depth of robot is about 17 meters in sea water. A strong point of the robot is the detachable plugs for batteries charging and other applications, compare to other robots in this category. Another point is the bottom camera that allows taking pictures and videos at the bottom of the net, without the robot having to make additional movements. Finally, there are future extensions such as long-distance communication and navigation of the robot.

Acknowledgements

This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship, and Innovation, under the call RESEARCH – CREATE – INNOVATE. (Project code: T2EDK- 02504).

References

[1]. A. V. Inzartsev, A. M. Pavin, N. I. Rylov, Development of the AUV automatic docking methods based on echosounder and video data, in *Proceedings of the 24th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS'17)*, 2017 May, pp. 1-6.

[2]. K. Alam, T. Ray, S. G. Anavatti, Design and construction of an autonomous underwater vehicle, *Neurocomputing*, Vol. 142, 2014. pp. 16-29.

[3]. O. A. Viquez, E. M. Fischell, N. R. Rypkema, H. Schmidt, Design of a general autonomy payload for low-cost AUV R&D, in *Proceedings of* the *IEEE/OES Autonomous Underwater Vehicles Conference (AUV'16)*, 2016 November, pp. 151-155.

[4]. J. Busquets, J. V. Busquets, F. Zilic, F. Pérez, F. Gonzalez, C. Aron, J. Gilabert, Low cost AUV in multiple deployment configurations for oceanographic research, in *Proceedings of the MTS/IEEE OCEANS-Bergen Conference*, 2013 June, pp. 1-9.

[5]. C. Edge, S. S. Enan, M. Fulton, J. Hong, J. Mo, K. Barthelemy, J. Sattar, Design and experiments with LoCO AUV: A low cost open-source autonomous underwater vehicle, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'20)*, 2020 October, pp. 1761-1768.

[6]. J. Busquets, J. V. Busquets, D. Tudela, F. Pérez, J. Busquets-Carbonell, A. Barberá, J. Gilabert, Low-cost AUV based on Arduino open source microcontroller board for oceanographic research

applications in a collaborative long term deployment missions and suitable for combining with an USV as autonomous automatic recharging platform, in *Proceedings of the IEEE/OES Autonomous Underwater Vehicles Conference (AUV'12)*, 2012, pp. 1-10.

[7]. E. Iscar, C. Barbalata, N. Goumas, M. Johnson-Roberson, Towards low cost, deep water AUV optical mapping, in *Proceedings of the MTS/IEEE OCEANS Conference*, Charleston, 2018 October, pp. 1-6.

[8]. N. R. Rypkema, E. M. Fischell, A. L. Forrest, M. R. Benjamin, H. Schmidt, Implementation of a hydrodynamic model-based navigation system for a low-cost AUV fleet, in *Proceedings of* the *IEEE/OES Autonomous Underwater Vehicle Workshop (AUV'18)*, 2018 November, pp. 1-6.

[9]. M. Dunbabin, J. Roberts, K. Usher, G. Winstanley, P. Corke, A hybrid AUV design for shallow water reef navigation, in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2005 April, pp. 2105-2110.

[10]. Y. Noguchi, T. Sakamaki, S. Ito, M. Humblet, Y. Furushima, T. Maki, Wide area seafloor imaging by a low-cost AUV, in *Proceedings of the Global Oceans 2020: Singapore–US Gulf Coast Conference*, 2020 October, pp. 1-7.

[11]. A. Martínez, Y. Rodríguez, L. Hernández, C. Guerra, H. Sahli, Hardware and software architecture for AUV based on low-cost sensors, in *Proceedings of the 11th International Conference on Control Automation Robotics Vision*, 2010 December, pp. 1428-1433.

[12]. B. Moridian, L. Wei, J. Hoffman, W. Sun, B. Page, M. Sietsem, N. Mahmoudian, A low-cost mobile infrastructure for multi-AUV networking, in *Proceedings of the IEEE/OES Autonomous Underwater Vehicle Workshop (AUV'18)*, 2018 November, pp. 1-6.

[13]. J. S. Willners, I. Carlucho, S. Katagiri, C. Lemoine, J. Roe, D. Stephens, S. Wang, From market-ready ROVs to low-cost AUVs, in *Proceedings of the OCEANS 2021: San Diego-Porto Conference*, 2021 September, pp. 1-7.

[14]. A. Underwood, C. Murphy, Design of a micro-AUV for autonomy development and multi-vehicle systems, in *Proceedings of the OCEANS 2017-Aberdeen Conference*, 2017 June, pp. 1-6.

[15]. D. A. Corbeanu, S. Krupiński, F. Maurelli, Blue duck II: A low-cost AUV design based on a moving mass for shallow-water operations, in *Proceedings of the IEEE/OES Autonomous Underwater Vehicles Symposium (AUV'20)*, 2020 September, pp. 1-5.

[16]. M. Vasileiou, N. Manos, E. Kavallieratou, A low-cost 3D printed mini underwater vehicle: Design and fabrication, in *Proceedings of the 20th International Conference on Advanced Robotics (ICAR'21)*, 2021, pp. 390-395.

[17]. S. Gao, B. He, X. Zhang, J. Wan, X. Mu, T. Yan, Cruise speed estimation strategy based on multiple fuzzy logic and extended state observer for low-cost AUV, *IEEE Transactions on Instrumentation and Measurement*, Vol. 70, 2020. 2502513.

[18]. J. Busquets, D. Proserpio, F. J. Martin, J. V. Busquets, Low-cost AUV Alba13 as multi-sensor platform with water sampler capabilities, for application in multi-agent ocean research applications, in *Proceedings of the OCEANS-St. John's Conference*, 2014 September, pp. 1-8.

[19]. A. Caffaz, A. Caiti, G. Casalino, A. Turetta, The hybrid glider/AUV Folaga, *IEEE Robotics Automation Magazine*, Vol. 171, 2010, pp. 31-44.

[20]. D. A. Duecker, N. Bauschmann, T. Hansen, E. Kreuzer, R. Seifried, Hipp oCampusX-A hydrobatic open-source micro AUV for confined environments, in *Proceedings of the IEEE/OES Autonomous Underwater Vehicles Symposium (AUV'20)*, 2020 September, pp. 1-6.

[21]. E. A. De Barros, L. O. Freire, J. L. Dantas, Development of the Pirajuba AUV, *IFAC Proceedings*, Vol. 4320, 2010, pp. 102-107.

[22]. BlueRobotics, https://bluerobotics.com/

[23]. BlueTrailEngineering, https://www.bluetrailengineering.com/

(060)

YOLO-based Multi-Modal Analysis of Vineyards using RGB-D Detections

T. Clamens [1], J. Rodriguez [1], M. Delamare [2], L. Lew-Yan-Voon [1], E. Fauvet [1] and D. Fofi [1]
[1] VIBOT, ImViA EA 7535, Université de Bourgogne, 12 rue de la fonderie 71200 Le Creusot, France
[2] CESI Engineering School, 80 avenue Edmund Halley Rouen Madrillet Innovation,
76800 Saint-Etienne-du-Rouvray, France
E-mail: thibault.clamens@u-bourgogne.fr

Summary: Agricultural robotics is a rapidly growing research area due to the need for new practices that are more environmentally responsible. It involves a range of technologies including autonomous vehicles, drones and robotic arms. These systems can be equipped with sensors and cameras to gather data and perform tasks autonomously or with minimal human intervention. For robot navigation and manipulation, and plant monitoring and analysis, perception is of prime importance and is still a challenging task today. For instance, visual perception using color images only for disease detection in vineyards, such as Mildew in which the symptoms manifest as small spots on or beneath the leaves, is still a hard task that does not allow to achieve high detection accuracy. To extract more representative features to improve the detection accuracy, other modalities must be used in addition to the Red Green and Blue (RGB) information of color images. In this paper, we present first a multimodal acquisition system that we have developed. It is composed of a multi-spectral (MS) camera and an RGB-D camera that are mounted on a mobile robot for data acquisition in a vineyard. Next, we describe the multi-modal dataset that we have built based on the data acquired with our system in a commercial vineyard. Finally, we implemented an Early RGB and depth data fusion technique together with the YOLOv5m Deep Learning network to detect the main parts of the vine: leaves, branches, and grapes using our dataset. The results that we have obtained, compared to those obtained using RGB images only with the YOLOv5m architecture, demonstrate the benefits of adding multi data fusion techniques to the object detection pipeline. These results are encouraging and show that multi-sensor data fusion is a technique that is worth considering as it can be useful for improving grapevine disease recognition technologies.

Keywords: Viticultural robotics, Vineyard analysis, Multi-modal dataset, RGB-D camera, Multi-spectral camera, RGB-D fusion, Object detection.

---

1. Introduction

Nowadays, in modern agriculture, farmers must deal with several trade-offs every day. They must satisfy a rising public demand while maintaining the quality of their products and of their land for the good health of the consumers of their products. The new environmental considerations imply an evolution of agricultural and viticultural practices [1] to increase sustainability and farmers' health and safety. For these reasons, robotic and precision agricultures are developing at a large pace as they can contribute to a reduction in the use of phytosanitary products such as pesticides, herbicides, and fungicides in farming as well as a reduction of human labor in harsh working conditions.

Robotic agriculture and precision agriculture are two different concepts but they are related. Robotic agriculture involves the use of robots or automated machines to perform specific tasks in the agricultural process, such as planting, harvesting, or spraying crops. On the other hand, precision agriculture is a farming method that uses technology to optimize crop production by analyzing and managing various factors, such as soil characteristics, weather patterns, and crop health, with high precision and accuracy. Although both concepts are different, they nevertheless have one common point in that they both rely heavily on visual perception and more particularly on one important application of visual perception that is object detection [2]. Indeed, object detection allows robots to navigate autonomously while avoiding obstacles in robotic agriculture. In precision agriculture it allows to identify and locate objects of interest, such as weeds, crops, and pests for weed and pest management and crop health monitoring. It is typically achieved through machine learning and deep learning algorithms, which are trained on large datasets of annotated images [3]. Once trained, these algorithms can accurately detect and classify objects in real-time, enabling robots to perform tasks such as crop monitoring [4], weed control [5, 6], and fruit picking [7].

Object detection using visual cues is challenging due to factors that are outside human control such as illumination, geometric properties of agricultural fields, weather conditions, and plant structure uncertainty. Appropriate detection and localization of plants, fruits and weeds are the backbones for inspection, robotics, and autonomous systems for agriculture. It helps farmers in several ways: to monitor crop health more efficiently, to identify and to respond to potential pest or disease outbreaks before they become severe resulting in a reduction in the need for phytosanitary products, and to estimate production yield and increase product quality.

Our work is targeted towards vineyard inspection and analysis. In this field, standard RGB cameras and computer vision can provide affordable and versatile solutions for object detection such as leaves, branches and grapes. The Faster R-CNN architecture provides accurate results for grape detection. Recognition results can be integrated through data association approaches that use object tracking or mapping to perform fruit counting in the vineyard. An alternative to the two-stage object detection algorithm is the one-stage algorithms such as the YOLO (You-Only-Look-Once) architectures. YOLOv5 model has outstanding performance in terms of speed and accuracy [8, 9]. To further improve object detection, multi-modal data fusion is crucial. The use of depth cameras can provide 3D information and highlight the unique geometry of objects present in the scene [10].

In the next section, we describe the multimodal acquisition system that we have developed as well as the multimodal dataset that we have built with data acquired with it. In Section 3, we present the data fusion method that we have developed and used together with the YOLOv5m Deep Learning network to detect leaves, branches and grapes in a vineyard. In Section 4, we present the results obtained and a comparison of the results with and without multi-sensor data fusion. Finally, we conclude in Section 5.

2. Data Acquisition and Multimodal Dataset

In this study, a multimodal acquisition system was developed to acquire vineyard data in several modalities. It is composed of a SILIOS CMS-V multi-spectral camera sensitive to eight different wavelengths bands from 550 to 830 nm and of a Microsoft Kinect V2 camera for RGB and depth information. Both cameras have been mounted on a Summit XL mobile robot as shown in Fig. 1 for ease of acquisition in a vineyard field which can have an irregular or an uneven shape. The positions of the cameras on the robot have been carefully chosen for accurate capture of the fine details of the scene, i.e., leaves, branches, and berries. Also, the cameras are fixed on the mobile robot so that their relative positions are perfectly known and do not vary during acquisition.

Fig. 1. Top-left: Summit XL mobile robot instrumented with the multi-sensors system; Top-right: RGB image; Bottom left: MS image; Bottom right: Depth image.

To acquire data with our acquisition system, the cameras must be calibrated, and the acquired images registered. Camera calibrations are achieved using the calibration method described in [11] for the multi-spectral camera and the iai_kinect2 package by [12] for the Microsoft Kinect V2 camera. Regarding image registration, the feature-based image registration algorithm is used [13]. This algorithm uses the Harris corner detector to extract the corner features from the reference images, the RGB images in our case, and the target images, the multispectral images. Then, it matches the corner features using the Scale Invariant Feature Transform (SIFT) algorithm to find the best affine transformation that aligns the images. Only the RGB and the multispectral images need to be registered. The depth image is already registered with the RGB image by the Microsoft Kinect V2 camera that is used to acquire both images.

Image acquisition is done at a frame rate of 15 images per second and with the mobile robot moving at a speed of 0.6 meters per second to reduce motion blur [14]. An example of the three types of images produced by our acquisition system, namely RGB images, depth images, and multispectral images is shown in Fig. 1. All the acquired images depict vine plants and 300 of them have been manually labelled with four classes to build our multimodal dataset. The four classes are branches, leaves, grapes, and background which represents any area that is not assigned to anyone of the other three classes.

3. Object Detection with Multi-sensor Data

The method presented in this paper aims to improve object detection in vineyards by combining RGB and depth information using an Early Fusion architecture together with the YOLOv5 Deep Learning network [15]. The whole network architecture is shown in Fig. 2. YOLOv5 is a popular convolutional neural network (CNN) for real-time object detection with high accuracy [8, 9]. The architecture employs a single neural network to analyze the entire image, subsequently dividing it into regions and predicting bounding boxes and class probabilities for each. The network comprises a backbone, which consists of convolutional layers that extract and generate image features at multiple scales; a neck, which generates feature pyramids to facilitate scale-invariant object detection; and a head, which utilizes anchor boxes to produce final output vectors containing class probabilities, objectness scores, and bounding box coordinates. Compared to its predecessor YOLOv4, YOLOv5 is 88 % smaller in size and 180 % faster in performance while maintaining comparable accuracy on the same task.

Since the YOLOv5 model is designed to process images in the RGB format, it can only accept an image with the three color channels (Red, Green and Blue) as input. However, our RGB-D data contains four channels: three channels for the RGB image and one channel for the depth image. Thus, the four channel RGB-D image must be transformed to a three-channel image to be able to be processed by YOLOv5. This is achieved by first transforming the depth image into a three-channel image so that it is of the same dimension as the RGB image and can thus be fused with it channel by channel. We have considered two types of transformation: replicating the raw depth image two times to obtain a depth image with three identical channels and colorizing the depth image using the Jet color palette to obtain a colored depth image in the RGB format. The effect of both transformations on the detection accuracy will be studied in the experiment section.

After transforming the depth image to a three-channel image, we now have a six-channel RGB-D image. To transform this six-channel image to the three-channel input of the YOLOv5 network, we have defined an Early data fusion network. It consists in first stacking the RGB and the depth information into a single tensor. Next, a convolution operation with six input filters, and three output filters is performed. The filter size has been chosen to be 3×3, and by doing zero-padding, the input and the output image sizes are the same. Finally, a batch normalization function is added to prevent vanishing gradients during training. Thus, at the output of this fusion module, a three-channel image is obtained that is then input to the YOLOv5 pipeline for training, test, and validation.

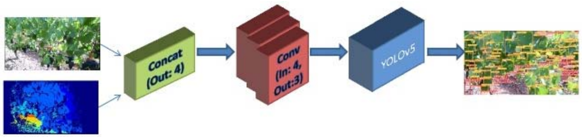


Fig. 2. Early fusion network architecture together with YOLOv5 network.

4. Experimental Results

In all our experiments, we have used the multimodal dataset that we have built with 300 manually labeled images of three classes of objects: leaves, grapes, and branches and a fourth background class which represents all areas that are none of the three object classes. The images of our dataset are images of vine plants where the "leaves" object is abundantly present. Moreover, the leaves overlap with each other such that it is very difficult to label all of them to generate the ground truth data. Thus, in the images many leaves are left unlabeled. In supervised Deep Learning networks, unlabeled data are automatically assigned to the no-class or the background class. In our case, they are assigned to the no-class. In the general case, if a valuable information (leaves, grapes, or branches) is not labeled, then during training the network might get confused because this information will be considered as pertaining to both the object class and the no-class. This will result in a degradation of the performance. For example, if a leaf is unlabeled, there will be the same type of object in two different classes, the "leaves" object class and the no-class. Then, during training similar features will be used to identify the leaf and the no-class object. Consequently, during testing, detected leaves will be assigned a lower probability than if all the leaves in the dataset were correctly labeled. To overcome this problem, a pre-processing step has been implemented to clear (i.e., set to zero) all the non-labeled regions. An example of the resulting images of this pre-processing step is shown in Fig. 3.

Fig. 3. Example of a labeled image processed to clear out no labeled regions. Vineyard RGB images (left) and Image with no labeled regions clear out (right).

Three training and test cases have been considered using two Deep Learning architectures: a Data Fusion network that we have defined and the YOLOv5m, a medium size YOLOv5 network, with the Ranger optimizer and pretrained with the MS COCO dataset. We have used and compared both the Adam and the Ranger optimizers [16]. Finally, we have chosen the Ranger optimizer, which is a combination of RAdam and Lookahead, because it offers improved training efficiency, faster convergence, and better performance compared to Adam. The three cases are:

- YOLOv5m network trained, validated and tested with the RGB images only;
- Data fusion and YOLOv5m networks trained, validated, and tested with RGB and raw depth images;
- Data fusion and YOLOv5m networks trained, validated, and tested with RGB and colored depth images.

For the first training and test case where only the RGB images are needed, the other modalities of our multimodal dataset are just ignored and not used. In the two other cases, the depth information is used in addition to the RGB ones. For the experiments, the dataset has been randomly split into 70 % of training, 15 % of validation and 15 % of testing data. Regarding the network parameters, the number of epochs is 100, the batch size is 4 and the image size is set to 640×640 pixels.

To evaluate our model, we have computed the precision and recall scores, and the Mean Average Precision (mAP) at 50 % and in the interval [50-95 %]. mAP@0.5 is a measure used to evaluate the overall performance of an object detection model by considering a prediction as correct if its Intersection over Union (IoU) with manual annotation is greater than or equal to 0.5, and by taking the average of the mean accuracies for each object class.

The quantitative results are summarized in Table 1. They show that the network with data fusion and multi-sensor data achieves better precision, and especially better robustness. The combination of data fusion with the YOLOv5m network allows for better adaptability to different vineyards, which is our ultimate goal. We can clearly see in Table 1 that for the mAP@0.5:0.95, we have a gap of almost double in the robustness of the system. We can therefore conclude that the RGB-D system allows for better robustness and therefore better adaptation to different configuration changes such as luminosity and seasonality.

Table 1. Quantitative results of the combined data fusion and YOLOv5m model trained with the multimodal dataset.

	P	R	mAP@0.5	mAP@0.5:0.95
RGB only	0.82862	0.68724	0.76331	0.35314
RGB + D	0.82562	0.69211	0.76367	0.3593
RGB + D colored	0.83125	0.68456	0.75909	0.34817

Fig. 4 represents an example of the output of the combined data fusion and YOLOv5m network trained with RGB images only, and with RGB-D images. The red, green and blue boxes are respectively the leaves, the grapes, and the branches classes.

Fig. 4. Example of detection using the combined data fusion and YOLOv5m network trained with RGB images (left) and with RGB-D images (right).

5. Conclusions

We have developed a multimodal acquisition system that is particularly suitable for data acquisition in a vineyard. Using our acquisition system, we have acquired data in a commercial vineyard and built a multimodal dataset with 300 manually labelled images of vine leaves, grapes, and branches. We then used this dataset to study the effects of using depth information in addition to RGB information and data fusion techniques to detect object in vineyards. To do so, we have trained a combined data fusion and medium size YOLOv5 network, denoted by YOLOv5m, to detect leaves, grapes, and branches in vineyard images. The results that we have obtained show that the use of multimodal data allows to increase the detection accuracy while reducing false negatives.

These results are encouraging, and we intend in future work to either use our dataset or create a new dataset with even more modalities, and data fusion techniques to address viticultural challenges such as the rapid and accurate detection of plant pathologies for vine plant health monitoring and disease management, berry detection for automatic harvesting or production yield estimation, and weed detection for autonomous weed removal in robotic agriculture. For example, in disease detection and recognition, the detection result mixed with the localization of the robot, can help to create a map of where the different diseases are. In this way, the vineyard owners can localize problems earlier, and solve them before they become a real problem.

Acknowledgements

We gratefully acknowledge the support of the French government's Plan France Relance initiative via the European Union, which provided funding for this project under contract: ANR-21-PRRD-0047-01.

References

[1]. H. O. Pörtner, D. C. Roberts, et al., Climate Change 2022: Impacts, Adaptation and Vulnerability, *IPCC*, 2022.

[2]. V. Sharma, R. N. Mir, A comprehensive and systematic look up into deep learning-based object detection techniques: A review, *Computer Science Review*, Vol. 38, November 2020, 100301.

[3]. L. Mohimont, F. Alin, M. Rondeau, N. Gaveau, L. A. Steffenel, Computer, vision and deep learning for precision viticulture, *Agronomy*, Vol. 12, Issue 10, 2022, 2463.

[4]. H. Kishan Das Menon, D. Mishra, D. Deepa, Automation and integration of growth monitoring in plants (with disease prediction) and crop prediction, *Proceedings of Materials Today*, Vol. 43, 2021, pp. 3922-3927.

[5]. A. S. M. Mahmudul Hasan, F. Sohel, D. Diepeveen, H. Laga, M. G. K. Jones, A survey of deep learning techniques for weed detection from images, *Computers and Electronics in Agriculture*, Vol. 184, May 2021, 106067.

[6]. N. Rai, Y. Zhang, B. G. Ram, L. Schumacher, R. K. Yellavajjala, S. Bajwa, X. Sun, Applications of deep learning in precision weed management: A review, *Computers and Electronics in Agriculture*, Vol. 206, March 2023, 107698.

[7]. Z. Guan, H. Li, Z. Zuo, L. Pan, Design a robot system for tomato picking based on YOLO v5, *IFAC-PapersOnLine*, Vol. 55, Issue 3, 2022, pp. 166-171.

[8]. G. Jocher, YOLOv5 by Ultralytics, https://github.com/ultralytics/yolov5, 2020

[9]. A. Bochkovskiy, C. Y. Wang, H. Y. Liao, YOLOv5: Improved real-time object detection, *arXiv Preprint*, 2021, arXiv:2103.14030.

[10]. Z. Wu, G. Allibert, C. Stolz, C. Demonceaux, Depth-adapted CNN for RGB-D cameras, in *Proceedings of the Asian Conference on Computer Vision (ACCV'20)*, Nov. 2020.

[11]. ROS, CAMERA CALIBRATION, http://wiki.ros.org/camera_calibration

[12]. T. Wiedemeyer, IAI Kinect2, https://github.com/code-iai/iai_kinect2

[13]. B. Islam Md, Mir J. Kabir Md, A new feature-based image registration algorithm, *Computer Technology and Application*, Vol. 4, 2013, pp. 79-84.

[14]. T. Clamens, G. Alexakis, R. Duverne, R. Seulin, E. Fauvet, D. Fofi, Real-time multispectral image processing and registration on 3D point cloud for vineyard analysis, in *Proceedings of the International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP'21)*, 2021, pp. 388-398.

[15]. A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, W. Burgard, Multimodal deep learning for robust RGB-D object recognition, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'15)*, 28 September – 02 October 2015.

[16]. Wright Less, Ranger – A Synergistic Optimizer, GitHub Repository, https://github.com/lessw2020/Ranger-Deep-Learning-Optimizer

(062)

4D Joint Harmonic Retrieval and Model Order Estimation with Convolutional Neural Networks

S. Schieler, S. Semper, R. Faramarzahangari, C. Schneider and R. S. Thomä
Technische Universität Ilmenau, FG EMS, Helmholtzplatz 2, 98693 Ilmenau, Germany
Tel.: +49 (0)3367-691153
E-mail: steffen.schieler@tu-ilmenau.de

Summary: Harmonic retrieval is essential in radio channel sounding, estimation, and modeling. In our previous work, we proposed a CNN-based approach combined with additional steps on the likelihood function. This paper extends the approach to perform joint 4D harmonic retrieval by utilizing the samples from a multi-antenna receiver in frequency, time, and the spatial domains of a radio channel transfer function.

The proposed architecture also reliably estimates the number of spectral components in the measurement. Hence, our approach can estimate four-dimensional parameters from a signal without prior knowledge of the unknown number of paths. Therefore, the architecture jointly solves the model order selection problem and the parameter estimation task in 4D.

Keywords: Harmonic retrieval, Convolutional neural network, Wireless channel, Parameter estimation.

1. Introduction

Spectral estimation is a ubiquitous problem in signal processing, arising in various applications such as channel estimation, radar localization, and direction finding. The available solutions can be categorized into four groups: subspace algorithms [1, 2], iterative maximum likelihood (ML) [3], Sparse Signal Recovery (SSR) [4] and most recently Deep Neural Network (DNN)-based algorithms [5-10]. DNN-based spectral estimation algorithms have demonstrated improved performance and robustness in low-Signal-to-Noise Ratio (SNR) conditions compared to traditional methods.

1.1. State-of-the-art

Various studies have used different neural networks for frequency and direction-of-arrival estimation. These methods have shown improvements in performance in the low-SNR domain compared to traditional methods like MUSIC [1]. In addition, a DNN has also been applied successfully to separate and reconstruct an unknown number of sources [11]. However, many works model the task using grid-based classification, e.g., [6, 7], and therefore suffer from inherent estimation bias due to grid-mismatch, and the non-linearly increase in grid complexity at higher dimensions. Modeling parameter estimation as a regression task, as shown in [10, 12], yields estimate comparable in accuracy to subspace methods. When combined with gradient steps on the likelihood function, the grid-free estimates serve as sufficient initial guesses for a second-order Newton method to converge [10, 12]. This approach combines the best of two worlds: fast, robust, but approximate initial estimates combined with a high-resolution iterative method allowing quadratic convergence, but only after successful initialization by the deep neural network.

Further promising applications for DNN-based estimators are in the area of Integrated Communication and Sensing (ICAS). Those systems demand estimators that are capable of real-time updates, can handle measurement system imperfections, varying resource allocations in the illuminating signal, and are capable of including known reference points in the environment.

1.2. Contributions

Compared to previous works, our approach can estimate up to 60 deterministic sources from a single snapshot in terms of their delay, Doppler-shift, and 2D Direction of Arrival (DoA). For this we extend the approach from [12] from 2D to 4D. Our work presents a joint regression and classification tasks: to estimate the number of sources and their parameters.

We compare the performance of the mean squared error (MSE) of the Convolutional Neural Network (CNN) estimates with existing methods. Additionally, we analyze the model order selection performance and compare our approach to an iterative ML. Furthermore, we also show that the DNN estimates are sufficient initial guesses to warm-start a second-order gradient iteration and enable convergence with high accuracy.

Our proposed methodology performs reliable standalone model order and parameter prediction with comparably short computation time, yielding a robust 4D parameter estimator with only moderate computational complexity.

2. Signal Model

Our task is retrieving the spectral components of deterministic sources from frequency and time samples of a radio channel, i.e., their propagation delays τ, Doppler-shifts α, azimuth φ, and elevation θ.

For the spatial dimensions, we assume a Uniform Rectangular Array (URA) with $N_v, N_h \in \mathbb{N}$ vertical and horizontal antenna elements, respectively. We model the wireless channel transfer-function measurement of bandwidth B with $N_f \in \mathbb{N}$ frequency samples, $N_t \in \mathbb{N}$ snapshots, and employ the narrowband assumption since $B \ll f_c$. We denote the sampled observation in complex baseband ($f_c = 0$) by $\boldsymbol{S}$. The sampling process in time- and frequency is characterized by the sampling intervals in frequency $\Delta f > 0$ and time $\Delta t > 0$ with $\boldsymbol{S}_\tau, \boldsymbol{S}_\alpha$ sampled $N_f, N_t \in \mathbb{N}$ times at

$$\boldsymbol{f_k} = \boldsymbol{f_0} + \boldsymbol{k} \cdot \boldsymbol{\Delta f}, \boldsymbol{t_l} = \boldsymbol{t_0} + \boldsymbol{l} \cdot \boldsymbol{\Delta t},$$

where $k = 0, \dots, N_f - 1, l = 0, \dots, N_t - 1$, $f_0 = -\frac{B}{2}$ and $t_0 = 0$.

Similarly, the sampling process in the spatial domain is carried out over the URA antenna elements with

$$n_v = n \cdot d_v, n_h = m \cdot d_h,$$

where $n = 0, \dots, N_v - 1, m = 0, \dots, N_h - 1$ and d_v, d_h define the vertical and horizontal antenna element spacings of the array.

$$S_{n,m}(\theta_p, \varphi_p) = \\ \exp\left(-j2\pi\lambda^{-1} d_v n \sin(\theta_p)\cos(\varphi_p)\right) \otimes \\ \otimes \exp\left(-j2\pi\lambda^{-1} d_h m \sin(\theta_p)\sin(\varphi_p)\right)$$

With that, the discrete signal model $\boldsymbol{S} \in \mathbb{C}^{N_f \times N_t \times N_v \times N_h}$ becomes

$$\boldsymbol{S}(\boldsymbol{\gamma}, \boldsymbol{\tau}, \boldsymbol{\alpha}, \boldsymbol{\theta}, \boldsymbol{\varphi}) = \sum_{p=1}^{P} \gamma_p \cdot \boldsymbol{S}(\tau_p) \otimes \\ \otimes \boldsymbol{S}(\alpha_p) \otimes \boldsymbol{S}(\theta_p, \varphi_p),$$

where the index $p = 1, \dots, P$ denotes the source index and $\boldsymbol{\gamma} \in \mathbb{C}^P$, $\boldsymbol{\tau} \in \mathbb{R}^P$, $\boldsymbol{\alpha} \in \mathbb{R}^P$, $\boldsymbol{\varphi} \in \mathbb{R}^P$, and $\boldsymbol{\theta} \in \mathbb{R}^P$ contain the corresponding complex weights, delays, and Doppler-shifts, Azimuth of Arrival, and Elevation of Arrival, respectively.

The noisy observation $\boldsymbol{Y} \in \mathbb{C}^{N_f \times N_t \times N_v \times N_h}$ is then formulated as

$$\boldsymbol{Y} = \boldsymbol{S}(\boldsymbol{\gamma}, \boldsymbol{\tau}, \boldsymbol{\alpha}, \boldsymbol{\theta}, \boldsymbol{\varphi}) + \boldsymbol{N}, \quad (1)$$

where $\boldsymbol{N} \in \mathbb{C}^{N_f \times N_t \times N_v \times N_h}$ is a complex, zero-mean Gaussian noise process with variance σ^2.

Estimating P, $\boldsymbol{\tau}$, $\boldsymbol{\alpha}$, $\boldsymbol{\theta}$, and $\boldsymbol{\varphi}$ from $\mathbf{Y}$ constitutes a joint model order selection and harmonic retrieval problem.

3. Neural Network

The goal of the presented approach is to use a deep convolutional neural network to estimate the model order P (number of paths in the signal) and their parameters from $\boldsymbol{Y}$.

3.1. Preprocessing

We use a similar preprocessing approach and parameter encoding as presented in [12], but extend it to 4 dimensions. This means, we use the same windows and DFT processing to create $N_w = 8$ views of the original data, denoted by $\boldsymbol{Y_1}$. As our network implementation in PyTorch [13] requires real numbers, we use 4 different mapping functions to obtain real numbers and present them to the CNN as different channels for the convolution operation by stacking them. Despite the redundancy in the mappings, our experiments showed, that the additional features helped training the network.

$$f_1(\boldsymbol{Y}_1) = \mathrm{R}(\boldsymbol{Y}_1), \\ f_2(\boldsymbol{Y}_1) = \mathrm{I}(\boldsymbol{Y}_1), \\ f_3(\boldsymbol{Y_1}) = \log_{10}(|\boldsymbol{Y_1}|), \\ f_4(\boldsymbol{Y}_1) = \angle(\boldsymbol{Y}_1)$$

3.2. Network Architecture

The architecture of our neural network, as depicted in Fig. 1, is structured into four distinct stages described below. We implemented the 4D convolutional layers using the same approach as used in [14].

The first stage feeds the input into a series of three consecutive blocks, each containing two submodules of 4D convolutional layer, Batch Normalization, and a Rectified Linear Unit (ReLU) activation function. The parametrization of the first block doubles the number of channels at the output, while the second one uses the same number of input and output channels. We denote these block as Double-Convolution blocks.

In the second stage, the network undertakes the task of downsampling. This is performed using convolutional layers with a stride of 2, effectively reducing the data dimension by half with each block. Each downsampling block is also followed by a Double-Convolutional block parametrized to preserve the number of channels.

The third stage is tasked with generating parameter predictions. The process begins with the reduction of channels through two blocks of convolutional layers. Following this, the data is propagated through two fully-connected (FC) layers with a ReLU activation function interspersed. The output of the final FC layer encodes the relative parameter estimates, denoted as η.

The fourth is tasked with the prediction of the number of paths P based on the output from the second stage. It employs a single convolutional block and two

FC layers with a ReLU activation function. The output of this stage, denoted as $\hat{\rho}$, is a one-hot encoded vector representing the estimated model order $\hat{P}$.

3.3. Loss Functions and Hyperparameters

We train our approach with the loss function and hyperparameters used in [12]. This means, we compute the MSE of the parameter estimates and weight it with the estimated probability that the corresponding path exists. The main differences compared to [12] are the increased number of dimensions (4 instead of 2) and number of possible paths (60 instead of 20). For further information, Table 1 outlines the hyperparameters of the synthetic datasets and the training hyperparameters.

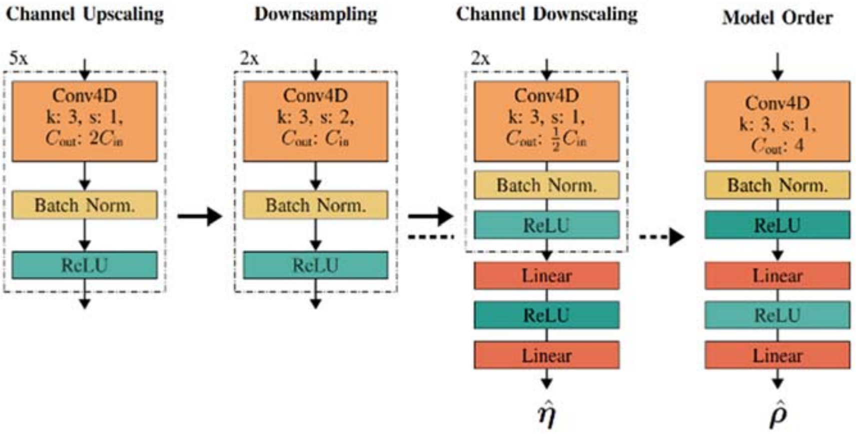


Fig. 1. The architecture uses convolutional layers to perform upscaling, downsampling, and downscaling. The encoded parameters in η and model order ρ are estimated from the downscaled or down sampled result, respectively.

4. Analysis

As outlined in Section 1, we analyze and compare our approach to other SOTA algorithms used to perform 4D parameter estimation. To assess the accuracy of the parameter estimations, we provide a comparison of the MSE for three different methods: a periodogram-based peak search with (low computational complexity), a iterative ML approach similar to the RIMAX estimator [3] (high computational complexity), and our approach. In addition, we use these estimates to warm-start a gradient iteration

$$\begin{aligned}(\boldsymbol{\gamma}^{k+1}, \boldsymbol{\tau}^{k+1}, \boldsymbol{\alpha}^{k+1}, \boldsymbol{\theta}^{k+1}, \boldsymbol{\varphi}^{k+1}) &= \\ = (\boldsymbol{\gamma}^{k}, \boldsymbol{\tau}^{k}, \boldsymbol{\alpha}^{k}, \boldsymbol{\theta}^{k}, \boldsymbol{\varphi}^{k}) &- \epsilon^k z^k,\end{aligned} \qquad (2)$$

where the descent direction is given by

$$z^k = [(\boldsymbol{F}^k)^{-1} \cdot \boldsymbol{J}^k](\boldsymbol{\gamma}^k, \boldsymbol{\tau}^k, \boldsymbol{\alpha}^k), \qquad (3)$$

where F is the Fisher-Information Matrix and J is the Jacobian of the negative log-likelihood function. This defines a second-order Gauss Newton scheme as

$$\lambda(\gamma, \tau, \alpha) = \frac{1}{\sigma^2} ||Y - S(\gamma, \tau, \alpha, \theta, \varphi)||_F^2$$

The results for the estimation performance are shown in Fig. 2. In line with the 2D results in [12], we demonstrate that our approach is able to overcome the grid-limited Peak-Search without any further interpolation. The accuracy increases slightly with SNR but quickly saturates for SNR > 10 dB. Compared to the iterative ML approach, our approach falls short to achieve the same accuracy. However, when using the estimates from our model to initialize (1) the estimation accuracy improves significantly from only 10 iterations. This finding is in-line with the results in [10] and [12], where similar improvements were observed. It shows that the estimates from our approach are close enough to the global minimum of (2) for (1) to converge. For SNRs > 10 dB the initializations start to become suboptimal, preventing convergence of the Gauss-Newton algorithm. We suspect, that the impact of missing or wrong values is more severe at higher SNRs, whereas missing estimates in low-SNR that are below noise level contribute less to convergence. However, a more in-depth analysis is required for a more conclusive statement.

To compare the computational complexity of the algorithms we make an assessment of the average runtime required to obtain a single sample on an identical system. This measure is mainly used, as the use of iterative numerical methods in the iterative ML approach is not straightforward. While the iterative ML approach requires 15218 ms to obtain the estimates, our approach requires only 34 ms. With the additional overhead of the Gauss Newton approach this time increases to 289 ms (10 iterations). This reaffirms the results from [10, 12], and highlights, how that our approach is well suited for applications which require a trade-off between accuracy and runtime, as this can be regulated in our approach via the number of gradient iterations.

Table 1. Dataset and training hyperparameters.

Name	Value
Dataset	
Distribution **($\boldsymbol{\tau_p, \alpha_p, \theta_p, \varphi_p}$)**	$\mathfrak{U}_{[0,1]}$ for all
Magnitudes	$\mathfrak{U}_{[0.001,1]}$
Phases	$\mathfrak{U}_{[0,2\pi]}$
SNR	-10 – 50 dB
Number of paths	$\mathfrak{U}_{[1,60]}$
Training set	10000,000 snapshots
Validation set	1000 snapshots
Test set	4000 snapshots
Training	
Optimizer	Adam [15], $\gamma = 0.0003$, $\beta_1 = 0.9, \beta_2 = 0.999$
Mini-batch size	256
Epochs	20
Trainable parameters	34127141

To assess the performance of the model order selection of our approach, we compare our approach's model order selection performance with the number of sources estimated via from the iterative ML approach. We configured the latter with a stop-criterion based on the D'Agostino's K-squared test, which measures the goodness-of-fit of the residual. If the residual is estimated to stem from the normal distribution with a 95 % confidence interval, the iterative ML stops and we select the number of paths from the last result as the model order.

Our results in Fig. 3 illustrate, that our approach achieves a much better performance compared to the iterative ML approach. However, we note, that the chosen stop criterion for the iterative ML approach might not be an optimal choice. Therefore, the results mainly reaffirm the results from previous works, that show that DNN-based model order selection delivers superior performance compared to previous solutions [5, 8, 11, 12].

5. Conclusion

This work introduces a CNN-based approach to 4D model order selection and parameter estimation. It is an extension of the approach in [12] to four signal dimensions, that is able to estimate signal parameters via a supervised-learning regression task. Similar to prior works, the approach can be used to warm-start a well-performing second-order gradient iteration for non-convex likelihood function optimization. Compared to previous works, it presents an improvement in terms of parameter dimensions, maximum number of paths, and NN parameter quantity.

The approach has a low runtime, making it suitable for time-constraint harmonic retrieval tasks, e.g., for ICAS applications. However, further research is required to better understand the misaligned results at higher SNRs. Furthermore, testing it with real measurement data is necessary. The approach might also have potential for wideband channel data processing, where the narrowband assumption is invalid.

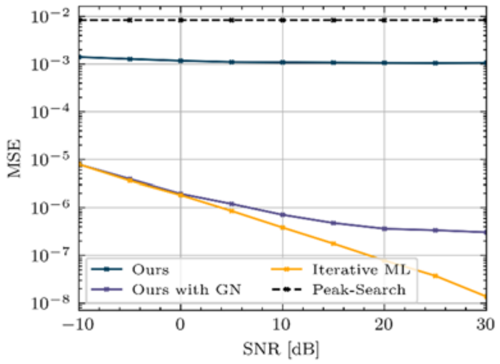


Fig. 2. Results for the MSE of the estimated parameters from a Peak-Search, an iterative ML estimator and our approach. While our approach outperforms the Peak-Search, it is outperformed by the iterative ML, reaffirming the results from [12].

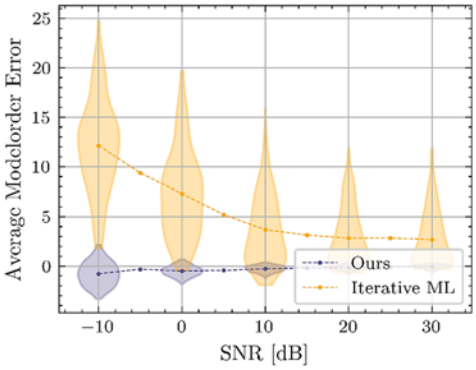


Fig. 3. Our approach shows promising results with regard to the model order selection. It can estimate the correct number of paths in the data with comparably low variance. Please note that the results were averaged across a single batch prior to avoid spikes.

Acknowledgements

The authors acknowledge the financial support by the Federal Ministry of Education and Research of Germany in the project "Open6GHub" (grant number: 16KISK015), "KOMSENS-6G" (grant number: 16KISK125), and DFG project HoPaDyn with Grant-No. TH 494/30-1.

References

[1]. R. Schmidt, Multiple emitter location and signal parameter estimation, *IEEE Trans. Antennas Propag.*, Vol. 34, Issue 3, 1986, pp. 276-280.

[2]. R. Roy, T. Kailath, ESPRIT-estimation of signal parameters via rotational invariance techniques, *IEEE Trans. Acoust. Speech Signal Process.*, Vol. 37, Issue 7, 1989, pp. 984-995.

[3]. R. Thomä, M. Landmann, A. Richter, and U. Trautwein, Multidimensional high-resolution channel sounding measurement, in Smart Antennas State of the Arts, T. Kaiser, A. Bourdoux, H. Boche, J. Rodriguez Fonollosa, J. Andersen Bach, and W. Utschick (Eds.), *Hindawi Publishing Corporation*, 2005.

[4]. D. Malioutov, M. Cetin, A. Willsky, A sparse signal reconstruction perspective for source localization with sensor arrays, *IEEE Trans. Signal Process*., Vol. 53, Issue 8, 2005, pp. 3010-3022.

[5]. G. Izacard, S. Mohan, C. Fernandez-Granda, Data-driven estimation of sinusoid frequencies, in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2019, pp. 5127–5137.

[6]. G. K. Papageorgiou, M. Sellathurai, Y. C. Eldar, Deep networks for direction-of-arrival estimation in low SNR, *IEEE Transactions on Signal Processing*, Vol. 69, 2021, pp. 3714-3729.

[7]. D. Chen, et al., Robust DoA estimation using denoising autoencoder and deep neural networks, *IEEE Access*, Vol. 10, 2022, pp. 52551-52564.

[8]. M. Naseri, et al., Machine learning- based angle of arrival estimation for ultra-wide band radios, *IEEE Communications Letters*, Vol. 26, Issue 6, 2022, pp. 1273-1277.

[9]. W. Liu, Super resolution DOA estimation based on deep neural network, *Scientific Reports*, Vol. 10, 2020, 19859.

[10]. A. Barthelme, W. Utschick, A machine learning approach to DoA estimation and model order selection for antenna arrays with subarray sampling, *IEEE Transactions on Signal Processing*, Vol. 69, 2021, pp. 3075-3087.

[11]. S. Schieler, et al., Estimating multi-modal dense multipath components using auto-encoders, in *Proceedings of the 30th European Signal Processing Conference (EUSIPCO'22)*, 2022, pp. 1716-1720.

[12]. S. Schieler, et al., Grid-free harmonic retrieval and model order selection using deep convolutional neural networks, *arXiv Preprint*, 2023, arXiv:2211.04846.

[13]. A. Paszke, et al., PyTorch: An imperative style, high-performance deep learning library, in Advances in Neural Information Processing Systems, Vol. 32, *Curran Associates, Inc.*, 2019.

[14]. J. P. Vizcaíno, Learning to reconstruct confocal microscopy stacks from single light field images, *IEEE Transactions on Computational Imaging*, Vol. 7, 2021, pp. 775-788.

(063)

2D Spectral Analysis of OFDM Radar Data using Deep Learning

R. Faramarzahangari, S. Schieler, R. Thomae and C. Schneider
Ilmenau University of Technology, Electronic Measurements and Signal Processing Group EMS,
29 Ehrenbergstr., 98693 Ilmenau, Germany

Summary: Spectral estimation is often encountered in a variety of signal processing tasks. One of the most common methods to estimate the spectrum of sampled data, is to use the Discrete Fourier Transform (DFT). However, using sampled data to estimate the spectrum with the DFT results in a limited accuracy and sidelobes. The limited accuracy is due to grid-oriented nature of periodogram and though this can be mitigated by interpolation, but it can lead to misinterpretation if the paths are too close to each other. These effects can cause issues in practical applications, e.g., sidelobes can mask weaker spectral components in the proximity of stronger components. To alleviate the problem of sidelobes, windowing techniques can be used on the cost of Signal-to-Noise ratio (SNR) loss. In this paper we present a novel approach which uses a Convolutional Neural Network (CNN) trained on synthetic data to estimate multidimensional spectra and predict the model-order of noisy input data. We compare our approach to conventional DFT methods using a harmonic retrieval data model.

Keywords: OFDM radar, Deep learning, JCRS, CNN, Periodogram.

1. Introduction

Joint Communication and Radar Sensing (JCRS) is one of the key technologies in the next generation of the mobile communication systems, 6G. The idea of JCRS is to make dual use of the communication resources to perform radar sensing, in which the objects that reflect electromagnetic waves are detected. One of the most effective waveforms for communication is Orthogonal Frequency-Division Multiplexing (OFDM) and is also a suitable waveform for radar sensing [1] both in active and passive manner. To detect the propagation paths using OFDM waveform, a spectrum called delay-Doppler spreading function is estimated and a criterion to determine the path bins in the spectrum is deployed to extract the path parameters, and then estimate the relative range and velocity of the reflecting objects.

Periodogram spectrum estimation is one of the most convenient methods that uses windowed Discrete Fourier Transform (DFT) but it is prune to distortions due to the side-lobes of the deployed window. This can be mitigated by tweaking the window at the cost of reducing the Signal-to-Noise Ratio (SNR). There are other grid-oriented (e.g., MUSIC [2]) and grid-free approaches (e.g., ESPRIT [3]) that have more accuracy at the cost of a higher computation time. Another grid-free approach for parameter estimation is to use model-based maximum likelihood estimators (such as RiMAX [4]) have a superior accuracy, yet they are prune to model imperfections and unavailability of calibration references. Moreover, maximum likelihood methods have a higher runtime, compared to periodogram.

The propagation paths can be detected from the estimated spectrum such that a Constant False Alarm Rate (CFAR) is ensured [5]. CFAR methods define the means (such as thresholds and statistical properties), by which the paths are distinguished from the distortions (or false alarms). Another approach is to estimate the number of existing paths k (e.g., using EDC [6]) and presume that the highest amplitude k peaks refer to the propagation paths.

The recent advances in deep learning suggest that deep Convolutional Neural Networks (CNN) [7] can be used for 1D spectral estimation of noisy multi-sinusoidal data, such that a lower false alarm rate is achieved, compared to state-of-the-art methods such as MUSIC, especially in lower SNRs ([8] and [9]).

Our approach is to use a CNN for estimating the 2D path parameters (delay and Doppler frequency shift) by training it to generate a high-resolution delay-Doppler spectrum, without the distortions caused by windowing. We also introduce another CNN architecture to estimate the number of paths using the estimated spectrum.

2. Data Model

In an OFDM transmission, the discrete time-variant channel frequency response can be acquired by dividing the received OFDM frame by the transmitted one. For P paths, each with a path delay τ_p, a Doppler shift α_p, and a complex path loss γ_p, the discrete time-variant channel frequency response is [1]:

$$\boldsymbol{H}_{\boldsymbol{m},\boldsymbol{n}} = \sum_{p=1}^{P} \gamma_p \, e^{i2\pi n T_0 \alpha_p} e^{-i2\pi m \Delta f \tau_p} + \\ + \boldsymbol{Z}_{\boldsymbol{m},\boldsymbol{n}} \in \mathbb{C}^{M \times N}, \tag{1}$$

where T_0 is the OFDM symbol duration, Δf is the sub-carrier spacing, M is the number of sub-carriers, N is the number of symbols, and $\boldsymbol{Z}_{\boldsymbol{m},\boldsymbol{n}}$ is additive white Gaussian noise. To obtain the path parameters from $\boldsymbol{H}_{\boldsymbol{m},\boldsymbol{n}}$ a consecutive forward and inverse DFT can be applied on $\boldsymbol{H}_{\boldsymbol{m},\boldsymbol{n}}$ to get the delay-Doppler spectrum $\boldsymbol{S}_{\boldsymbol{m}^*,\boldsymbol{n}^*}$:

$$\boldsymbol{S}_{\boldsymbol{m}^*,\boldsymbol{n}^*} = \sum_{k=0}^{N^*-1} \left(\sum_{l=0}^{M^*-1} \boldsymbol{H}_{\boldsymbol{k},\boldsymbol{l}}\, e^{-j2\pi \frac{l\cdot m}{M^*}} \right) e^{j2\pi \frac{k\cdot n}{N^*}}, \quad (2)$$

where $\boldsymbol{H}_{\boldsymbol{k},\boldsymbol{l}}$ is the zero-padded $\boldsymbol{H}_{\boldsymbol{m},\boldsymbol{n}}$ to the expanded dimensions $\mathbb{C}^{M^* \times N^*}(M^* \gg M$ and $N^* \gg N)$ and $\mathcal{F}_i$ denotes a DFT along the i-th axis. Note that $\boldsymbol{S}_{\boldsymbol{m}^*,\boldsymbol{n}^*}$ includes side-lobe distortions due to the limited symbols and subcarriers. The infinite bandwidth (orgroundtruth) spectrum consists of only Dirac delta functions as in

$$\boldsymbol{S}^{\boldsymbol{gt}}_{\boldsymbol{m}^*,\boldsymbol{n}^*} = \sum_{p=1}^{P} \gamma_p \delta(\tau - \tau_p) \cdot \delta(\alpha - \alpha_p), \quad (3)$$

where $\delta(.)$ is the Dirac delta function.

3. The Deep Learning Approach

The functions in (1) and (3) can be used as the input and output of a supervised learning task, in which a CNN learns to generate the distortion free delay-Doppler spectrum from the discrete time-variant channel frequency response $\boldsymbol{H}_{\boldsymbol{m},\boldsymbol{n}}$.The loss function for this task is the Mean Squared Error (MSE) of the generated spectrum $\widehat{\boldsymbol{S}}_{\boldsymbol{m}^*,\boldsymbol{n}^*}$ and the groundtruth spectrum $\boldsymbol{S}^{gt}_{\boldsymbol{m}^*,\boldsymbol{n}^*}$:

$$\mathcal{L}_{spec} = \frac{1}{M^* \cdot N^*} \sum_{m=1}^{M^*} \sum_{n=1}^{N^*} \left(\boldsymbol{S}^{gt}_{\boldsymbol{m}^*,\boldsymbol{n}^*} - \widehat{\boldsymbol{S}}_{\boldsymbol{m}^*,\boldsymbol{n}^*} \right)^2, \quad (4)$$

where M^* and N^* refer to the dimensions of the spectrum. Note that in $\boldsymbol{S}^{gt}_{m,n}$, we replace the Dirac delta function in (3) with narrow Gaussian functions. The choice is made for three reasons:

1) The shape of the Gaussian function fixed and hence known in advance. So, the parameters of the resulting pseudo-spectrum can be estimated by using the model;
2) To have a high resolution (interpolated), grid-based representation. Unlimited bandwidth would require a different representation of parameters;
3) To enable training via regression instead of classification.

We developed a network that estimates the $\widehat{\boldsymbol{S}}_{m,n}$, extended by another one that uses the features, from which $\widehat{\boldsymbol{S}}_{m,n}$ is generated, for estimating the number of existing paths. As shown in Fig. 1 the first part deploys multiple windowing functions (namely rectangular, Hann, Tukey, Taylor, Blackman, Chebyshev, flat top, and cosine) and performs a 2-dimensional Fast Fourier Transform (FFT) to get the complex DFT result of the windowed input $\boldsymbol{Y}_{m,n}$, which is then normalized and pre-processed by a feature extraction step. The feature extraction step provides it's next layer with 4 properties of $\boldsymbol{Y}_{m,n}$, namely it's real and imaginary parts, absolute value, and phase. These properties are used by 9 consecutive CNN blocks, each consisting of a 2D convolutional layer followed by a batch normalization and Rectified Linear Unit (ReLU) [10] layer. Finally, to generate the spectral estimation, we use a 2D transpose convolution layer to reconstruct the spectrum into expanded dimensions.

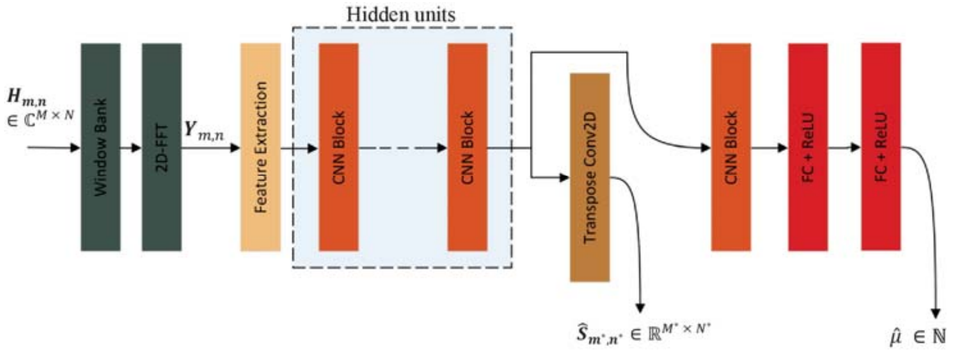


Fig. 1. The architecture of the spectral estimation network. The spectrum output is estimated by applying a transpose convolution to the CNN output, while the model order is estimated by using the CNN input by a smaller CNN.

The model order part is an extension to the spectral estimation network by passing the data before the transpose convolutional layer to a CNN block, identical to the previously mentioned CNN blocks. The output of the CNN block is flattened to be processed by 2 consecutive fully connected (FC) layers with ReLU nonlinearities to generate the output model order $\hat{\mu}$. Note that a non-negative output is ensured by applying a ReLU function after the final FC layer. The path parameters can then be estimated by finding the peaks of the estimated spectrum using a 2D peak search algorithm. The loss function used for training the model order part is also the MSE of the network's model order estimation and the groundtruth model order.

We implemented and trained our model using PyTorch [11] framework. The training process was done on an NVIDIA A100-PCIE-40 GB graphics processing unit (GPU). The training dataset consists of 400000 sample of $\langle \boldsymbol{H}_{\boldsymbol{m},\boldsymbol{n}}, \boldsymbol{S}^{gt}_{\boldsymbol{m}^*,\boldsymbol{n}^*} \rangle$ pairs to train the network in a supervised manner. The maximum number of propagation paths in the dataset is 20 paths. The channel frequency responses used as the input of the network have dimensions of $\mathbb{C}^{50 \times 10}$ and the network was trained to generate outputs with dimensions $\mathbb{R}^{1000\times 80}$. To ensure the performance of the CNN for different SNRs, the network was trained on different SNRs ranging from 0 to ∞. Each of the convolutional layers of the CNN has 64 filters and the output of each filter is an inner representation of the input data and has the dimensions of $\mathbb{R}^{125\times 10}$. The properties of the training dataset and other training hyper-parameters are denoted in Table 1.

Table 1. Training hyper-parameters.

Parameter	**Value**
Dataset Distribution	
Path parameters (τ_p and α_p)	$\mathrm{uniform}(0,1)$
Path weights ($|\gamma_{p}|$)	$\mathrm{uniform}(10^{-4},10^{-1})$
Path phases ($\measuredangle\gamma_p$)	$\mathrm{uniform}(0,2\pi)$
Received SNRs [dB]	$\mathrm{uniform}(0,\infty)$
Number of paths	$\mathrm{uniform}(0,20)$
Dataset Samples	
Number of samples for training	400000
Number of samples for validation	4000
Training	
Optimizer	Adam [12], γ=0.0003, β1=0.9, and β2=0.999.
Mini-batch size	256
Epochs	20
Trainable parameters	6989875

4. Results

The performance of the deep learning approach can be evaluated in terms of accuracy of the estimation (MSE error), false-alarm rates, and missed detections. A comparison between our approach and the periodogram method using simulation data (based on (1) and (2)) is presented in Fig. 2. In this example the periodogram deploys a rectangular window to distinguish between close paths. As it can be seen, the result from the CNN approach does not include any side-lobe distortions, which simplifies the parameter estimation using a peak search algorithm. The only shortcoming of the CNN approach is that it can drop some of the paths (missed detections) but the rate of this type of errors (also the false alarms and MSE) is much lower than similar approaches.

For further evaluation, the output of the developed CNN network is used with a 2-dimensional peak search algorithm to extract the path parameters. Conveniently, CFAR methods are used to extract the valid object detections out of a distorted periodogram. As the estimated spectrum by the network is distortion free, CFAR algorithm is not applicable in this case. This is then compared against the periodogram method. The MSE histogram of both methods is presented in Fig. 3. The evaluation is done over a dataset of 4000 sample with the similar distribution of the training dataset for different SNRs. It can be observed that the work in this paper not only outperforms the periodogram in terms of the mean estimation error, but it also has a lower maximum error and error variance. Note that both MSEs in Fig. 3 has a discrete distribution since the estimated parameters are bound to the estimated spectral grid, which limits the accuracy of parameter estimation.

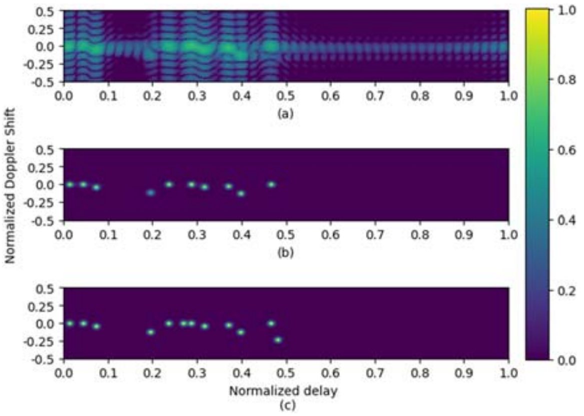


Fig. 2. Comparing the performance of the CNN approach (b) with Periodogram (a) given the groundtruth (c). The CNN model detects 12 out of 14 paths and by using a 2D-peak search algorithm, the MSE of the parameter estimation is 2.36×10^{-5}.

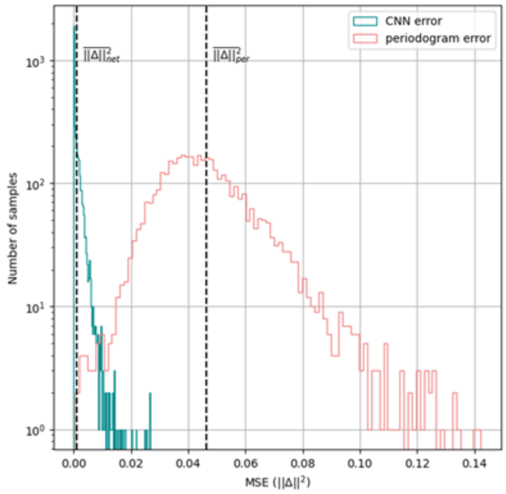


Fig. 3. MSE error histogram of periodogram parameter estimation compared to our CNN approach (SNR=10 dB). The MSE of the CNN approach is 16 dB lower than the periodogram.

5. Conclusion

In this paper we introduced a new approach based on CNNs for spectral analysis of OFDM signals for estimation of multi-path propagation parameters. This can outperform the periodogram method in parameter estimation using a peak search algorithm, because the CNN learns to estimate the spectrum without the distortion caused by the side-lobes of the sampling window. This can reduce the effect of side-lobe interference and mis-detecting the side-lobes for propagation paths. The developed spectral estimation method comes with an extension to estimate the number of paths based on the channels extracted by spectral estimation network. Furthermore, the performance of the developed CNN. can be evaluated against measurement data. This enables deployment of this method in real-time applications which is the requirement of many JCRS use-cases.

The performance of the introduced model can be enhanced by doing an ablation study to find the best architecture and hyper-parameters for this task.

Acknowledgements

The authors acknowledge the financial support by the Federal Ministry of Education and Research of Germany in the project “Open6GHub” (grant number: 16KISK015) and “KOMSENS-6G” (grant number: 16KISK125).

References

[1]. M. Braun, OFDM radar algorithms in mobile communication networks, PhD Thesis, *Karlsruher Institut für Technologie (KIT)*, 2014.

[2]. R. Schmidt, Multiple emitter location and signal parameter estimation, *IEEE Transactions on Antennas and Propagation*, Vol. 34, 1986, pp. 276-280.

[3]. R. Roy, T. Kailath, Esprit-estimation of signal parameters via rotational invariance techniques, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 37, 1989, pp. 984-995.

[4]. M. Landmann, M. Käske, R. S. Thomä, Impact of incomplete and inaccurate data models on high resolution parameter estimation in multidimensional channel sounding, IEEE Trans. on Antennas and Propagation, Vol. 60, Issue 2, February 2012, pp. 557-573.

[5]. M. A. Richards, Fundamentals of Radar Signal Processing, *McGraw-Hill Education*, 2014.

[6]. L. C. Zhao, P. R. Krishnaiah, Z. D. Bai, On detection of the number of signals in presence of white noise, *Journal of Multivariate Analy*sis, Vol. 20, 1986, pp. 1-25.

[7]. Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proceedings of the IEEE*, Vol. 86, Nov. 1998, pp. 2278-2324.

[8]. I. Gautier, S. Mohan, C. Fernandez-Granda, Data-driven estimation of sinusoid frequencies, *Advances in Neural Information Processing Systems*, Vol. 32, 2019, pp. 5127-5137.

[9]. G. K. Papageorgiou, M. Sellathurai, Y. C. Eldar, Deep networks for direction-of-arrival estimation in low SNR, *IEEE Transactions on Signal Processing*, Vol. 69, 2021, pp. 3714-3729.

[10]. A. F. Agarap, Deep learning using Rectified Linear Units (ReLU), https://arxiv.org/abs/1803.08375

[11]. A. Paszke, et al., PyTorch: An imperative style, high-performance deep learning library, in Advances in Neural Information Processing Systems, Vol. 32, *Curran Associates, Inc.*, 2019.

[12]. D. P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, in *Proceedings of the Conference on Learning Representations (ICLR’15)*, 2015.

(064)

Classification of Heart Sounds Using Quantum Machine Learning Models

F. Plou-Llorente [1], **E. F. Combarro** [2], **A. J. Muñoz-Montoro** [3] **and J. Ranilla** [2]
[1] University of Zaragoza, Pedro Cerbuna, 12, 50009-Zaragoza, Spain
[2] University of Oviedo, Campus de Gijón, 33204-Gijón, Spain
[3] University of Málaga, Campus de Teatinos, 29071-Málaga, Spain
E-mails: 735805@unizar.es, ajmmontoro@uma.es, {efernandezca, ranilla}@uniovi.es

---

Summary: Heart sound analysis is a crucial task for detecting and diagnosing cardiovascular diseases. With the advancement of digital signal processing and artificial intelligent techniques, automatic classification of heart sounds has become an active research area. In this work, we discuss the challenges and opportunities in applying Quantum Machine Learning (QML) models to real-world heart sound analysis. Traditional machine learning algorithms, such as support vector machines and neural networks, have shown promising results in classifying heart sounds based on Mel-frequency cepstral coefficients (MFCCs). However, the emerging field of QML offers new possibilities for enhanced performance and improved accuracy. Quantum Neural Networks (QNNs) and Quantum Support Vector Machines (QSVMs) are two prominent QML models that exploit the principles of quantum mechanics to process information more efficiently. In our study, we employ QNNs and QSVMs to explore their applicability in heart sound classification. Through our experimentation, we have evaluated different architectures and embeddings to identify the most suitable models. Our findings demonstrate the potential of QML models in certain situations for heart sound classification, opening up new avenues for more accurate and effective detection of cardiovascular diseases.

Keywords: Quantum machine learning, Heart sound classification, Quantum support vector machines, Quantum neural networks.

---

1. Introduction

Cardiovascular diseases have a profound impact on global health, resulting in millions of deaths annually and imposing significant economic burdens on healthcare systems. Early detection and accurate diagnosis of these diseases are crucial for effective treatment and prevention of severe complications. Heart sound analysis plays a vital role in providing valuable insights into heart function and aiding physicians in identifying anomalies that may require further testing or treatment.

Traditionally, heart sound analysis has relied on techniques such as auscultation, where physicians listen to the sounds produced by the heart using a stethoscope. However, manual auscultation is subjective and heavily dependent on the experience and expertise of the practitioner, leading to potential variability in interpretation. Moreover, it can be challenging to detect subtle abnormalities or classify heart sounds accurately solely based on auditory cues.

To overcome these limitations and improve the accuracy and efficiency of heart sound analysis, researchers have turned to digital signal processing and machine learning techniques. One widely used method in this domain is the application of Mel-frequency cepstral coefficients (MFCCs), which are commonly employed in speech recognition, music analysis, and other audio-related applications. By utilizing MFCCs, important features related to the frequency and energy distribution of heart sounds can be extracted. These features then serve as inputs to machine learning algorithms for heart sound classification [1], allowing the differentiation between normal heart sounds, abnormal heart sounds, and specific cardiovascular conditions.

Various machine learning algorithms, including support vector machines (SVMs) [1], neural networks (NNs) [1], and random forests (RF) [2], have demonstrated promising results when combined with MFCC-based features for heart sound classification. These approaches offer non-invasive, cost-effective, and automated solutions for early detection of cardiovascular diseases, enabling timely interventions that can improve patient outcomes and reduce healthcare costs.

2. Quantum Machine Learning

Quantum Machine Learning (QML) represents an exciting frontier at the intersection of quantum computing and machine learning. It harnesses the principles of quantum mechanics to develop more powerful and efficient methods for processing and analyzing data [3]. By leveraging quantum phenomena, QML offers the potential to enhance the capabilities of traditional machine learning algorithms.

One prominent approach within QML is the utilization of Quantum Neural Networks (QNNs) [4]. QNNs employ parameterized quantum circuits, serving two purposes: embedding classical data into the Hilbert space of quantum states and functioning as variational forms with free parameters that can optimize a loss function [5, 6]. This hybridization of quantum and classical elements enables QNNs to perform complex computations and extract meaningful features from the input data. By exploring different architectures and embeddings, researchers can unlock the full potential of QNNs in various applications, including heart sound analysis.

Another noteworthy QML model is the Quantum Support Vector Machine (QSVM) [7]. QSVM exploits the exponential dimensionality of the Hilbert space to efficiently compute quantum kernels, which can enhance the data separation. By leveraging quantum algorithms, QSVMs have the potential to achieve better classification performance compared to their classical counterparts. The choice of feature maps plays a crucial role in determining the effectiveness of QSVMs. By selecting appropriate feature maps and optimizing hyperparameters, researchers can harness the power of QSVMs for heart sound analysis.

In this work, we adopted both QNNs and QSVMs to explore their applicability in the field of heart sound analysis. By employing QNNs, we sought to leverage the unique properties of quantum computation to extract and classify features from heart sound data. Likewise, QSVMs provided an avenue to exploit the advantages of quantum kernels for accurate heart sound classification. To implement and evaluate these models, we utilized PennyLane, a versatile quantum machine learning library that enabled us to simulate quantum computations on classical hardware [8].

Through our exploration of QML models in heart sound analysis, we aim to leverage the inherent strengths of quantum computing to improve the detection and diagnosis of cardiovascular diseases. By incorporating quantum algorithms into the heart sound analysis pipeline, we anticipate enhanced accuracy, improved classification performance, and the potential to uncover novel insights from the data. Our study provides a foundation for future research in this emerging field, paving the way for further advancements in applying QML to tackle complex challenges in cardiovascular health.

3. Experimental Results

We conducted our experiments using a variant of the "Yaseen Heart Sound Database" (YHSD) [9], which comprises high-quality recordings of heart sounds and murmurs obtained from patients with various cardiac conditions. The database consists of two distinct sets: a normal set and an abnormal set. The abnormal set encompasses four categories: aortic stenosis, mitral stenosis, mitral regurgitation, and mitral valve prolapse. Each category contains 200 audio files, resulting in a total of 1.000 audio files in the database. The recordings were meticulously curated and captured using electronic stethoscopes to ensure accuracy and reliability. The YHSD is part of a growing collection of publicly available databases, created to facilitate research and advancements in the diagnosis of cardiovascular diseases through signal processing and machine learning techniques. The dataset served as the foundation for evaluating the performance of our proposed model, enabling us to assess its effectiveness in heart sound classification and its potential impact on improving cardiovascular health outcomes.

The audio signals were subjected to a preprocessing stage, where we applied MFCCs to extract relevant features. The resulting dataset consisted of audio segments with a duration of 0.45 seconds each, representing distinct cycles of heart sounds. To optimize the feature extraction process, we carefully selected a combination of parameters, including 10 MFCCs, 60 % overlap, and 30 filters. These parameter choices were determined through rigorous experimentation and analysis. Additionally, we maintained consistency in other preprocessing parameters across all dataset, including a window length of 0.03 seconds and a sampling frequency of 8000 Hz.

Although YHSD encompasses four different heart diseases, our objective was to establish a binary classification framework by grouping all diseases into two classes: healthy hearts and unhealthy hearts. This binary classification scheme allowed us to focus on distinguishing between normal heart sounds and those indicative of various cardiovascular abnormalities. By consolidating the diseases into these two classes, we aimed to develop a robust classification model capable of accurately identifying the presence of cardiovascular conditions based on heart sound characteristics.

For the training, the classical ML models used were SVMs and NNs. For the comparison with quantum methods, we reduced the variables to 8. This was achieved via a PCA transformation. In terms of the size of training and test sets, we used an 80/20 division in training/test for the classical models. Meanwhile, for QSVM, we used 20 % of the total examples for test, but barely 200 examples for training, and for QNNs we used 991 examples in the 0.45 s cycles case and 600 examples in the complete audios case for training. This was, again, because of limitations in the amount of quantum computation we could simulate classically. In the case of QNNs, the computational cost is lower than QSVM, so we can use more examples for training.

Then, we trained SVMs and QSVMs performing an hyperparameter tuning and using a cross-validation ($k = 5$) approach. QSVM hyperparameter tuning consisted in choosing the best quantum kernel based on different feature maps: the ZZ Feature Map, Angle Embedding and Amplitude Embedding (see [6]).

For classical and quantum NNs, we were not performing cross-validation, instead we used a validation set and early stopping to reduce overfitting. As with the previous methods, we made a hyperparameter tuning to choose the best models. Tuning in a quantum network consists in choosing between different configurations of feature maps and variational forms. Also, for QNNs, we did not use a PCA-transformation, instead we used a classical layer that reduces the variables to 8.

Table 1 shows the results obtained for 0.45 s cycle dataset (380 variables, 4767 examples). The results show that the QSVM outperforms classical SVMs when using a reduced number of variables

(SVM-PCA), with performance very close to using all variables. However, QSVMs did not outperform classical neural networks. On the other hand, hybrid quantum neural network (QNN) competes against classical methods, obtaining promising results. It is important to note that this is an initial study, and we plan to improve the performance of quantum neural networks in future work by increasing the number of parameters and using data re-uploading techniques [10]. Despite these limitations, our results demonstrate the potential of quantum machine learning methods for diagnosing cardiovascular diseases.

Table 1. Results obtained for 0.45 s cycle dataset.

Method	Train score	Test score
SVM-PCA	90.69	90.04
SVM-All	96.83	94.86
NN-PCA	97.90	98.32
NN-All	99.84	98.53
QSVM	97.08	93.61
QNN	98.79	94.13

4. Conclusions

Heart sound analysis plays a crucial role in the detection and diagnosis of cardiovascular diseases. In this study, we explored the applicability of QML models for heart sound analysis, specifically QNNs and QSVMs. Our experimental results demonstrated promising potential for the classification of cardiovascular diseases using these QML models.

The QSVM outperformed classical SVMs when using a reduced number of variables, while QNNs exhibited competitive performance compared to classical methods. Although limitations in computational resources and dataset size were present, these findings underscore the value of QML models in improving the accuracy and efficiency of heart sound analysis.

Further research involving larger datasets and expanded parameter complexity is necessary to comprehensively evaluate the capabilities of QML models in this field. Nevertheless, this study highlights the opportunities presented by quantum machine learning, showcasing its potential in enhancing the detection and classification of cardiovascular diseases.

Acknowledgements

This work was supported in part under grant PID2020-119082RB funded by MCIN/AEI/10.13039/501100011033, grant AYUD/2021/50994 funded by Gobierno del Principado de Asturias (Spain), and QUANTUM SPAIN project funded by the Ministry of Economic Affairs and Digital Transformation of the Spanish Government and the European Union through the Recovery, Transformation and Resilience Plan – NextGenerationEU. We would also like to express our gratitude to the CTIC of Asturias for providing us with the infrastructure necessary to carry out this work.

References

[1]. G. Y. Son, S. Kwon, Classification of heart sound signal using multiple features, *Applied Sciences*, Vol. 8, Issue 12, 2018, 2344.

[2]. M. Nassralla, Z. El Zein, H. Hajj, Classification of normal and abnormal heart sounds, in *Proceeding of the Fourth International Conference on Advances in Biomedical Engineering (ICABME'17)*, Beirut, Lebanon, 19-21 October 2017, pp. 1-4.

[3]. J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, S. Lloyd, Quantum machine learning, *Nature*, Vol. 549, Issue 7671, 2017, pp. 195-202.

[4]. A. Abbas, D. Sutter, C. Zoufal, et al., The power of quantum neural networks, *Nature Computational Science*, Vol. 1, Issue 6, 2021, pp. 403-409.

[5]. M. Schuld, F. Petruccione, Machine Learning with Quantum Computers, *Springer*, 2021.

[6]. E. F. Combarro, S. González-Castillo, A Practical Guide to Quantum Machine Learning and Quantum Optimization, *Packt*, 2023.

[7]. V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces. *Nature*, Vol. 567, Issue 7747, 2019, pp. 209-212.

[8]. V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, S. Ahmed, V. Ajith, N. Killoran, Pennylane: Automatic differentiation of hybrid quantum-classical computations, *arXiv Preprint*, 2018, arXiv:1811.04968.

[9]. Yaseen Heart Sound Database Portal, https://github.com/yaseen21khan/Classification-of-Heart-Sound-Signal-Using-Multiple-Features-

[10]. A. Pérez-Salinas, A. Cervera-Lierta, E. Gil-Fuster, J. I. Latorre, Data re-uploading for a universal quantum classifier, *Quantum*, Vol. 4, 2020, 226.

(066)

An Approach Using Deep Learning for Forest Fire Detection

Huong Nguyen Thu [1,2,3] **and Long Nguyen The** [1,2]
[1] Department of Informatics, Institute for Cybersecurity and Digital Technologies,
MIREA – Russian Technological University
[2] Laboratory of Artificial Intelligence and Machine Learning, Institute of Information Technology
and Data Science, Irkutsk National Research Technical University, 664074, Irkutsk, Russia
[3] University of Information and Communication Technology, Thai Nguyen University,
70000, Thai Nguyen, Viet Nam
Tel.: +79246039341
E-mail: {thuhuongyb, thelongit88}gmail.com

Summary: Due to the influence of climate change, natural disasters are becoming more complicated and have serious consequences. Forest fire is an environmental problem that poses threats to the safety of human life, infrastructure and environment, it is considered an important agent of forest pattern formation. Due to climate change, i.e. less rainfall and/or increase in day temperatures, longer dry seasons and interference from human activities, the trend and frequency of wildfires has increased and reached alarming levels in many parts of the world. In particular, the phenomenon of forest fires is affected by factors such as humidity, temperature and vegetation characteristics, etc., so the detection and prediction of forest fires face many challenges. In this study, we focus on using image processing techniques and visual algorithms to preprocess data and filter noise, in order to create a complete database for training and testing model. Natural elements are also used in combination with object features (forest fire images) to build feature vectors. The fast-RCNN deep learning model is proposed to optimize the forest fire detection system. The experimental results show that this method has positive results and high sensitivity to the data.

Keywords: Deep learning, Feature extraction, Fire forest, Object detection, Fast – RCNN, Artificial intelligence.

---

1. Introduction

Types of forest fire forecasting methods are mainly according to the composite index (P) of Nesterov calculated from the value of meteorological factors including temperature, air humidity and precipitation, forest type, wind force and fog level. It can be seen that these forest fire forecasting methods do not use other important factors such as vegetation indices (NDVI, NDWI, NDMI), wildfire weather index (FWI) or Topographic humidity index (TWI), distance to road, distance to residential area, slope, slope direction, are the factors that have proven to have high predictive ability in forest fires [1]. Statistical methods have been used for forest fire research due to the inherent randomness of wildfire phenomena. However, for problems with large data volumes and multiple inputs, the predictive accuracy of statistical models is still limited. The available literature dictates that flame detection using visible light camera is the generally used fire detection method, which has three categories including pixel-level, blob-level, and patch-level methods. The pixel-level methods, are fast due to usage of pixel-wise features such as colors and flickers, however, their performance is not attractive as such methods can be easily biased. Compared to pixel-level methods, blob-level flame detection methods show better performance as such methods consider blob-level candidates for features extraction to detect flame. The major problem with such methods is the difficulty in training their classifiers due to numerous shapes of fire blobs. Patch-level algorithms are developed to improve the performance of previous two categories of flame detection algorithms, however, such methods result in many outliers, affecting their accuracy [2, 3].

Recently, machine learning models have been proposed for wildfires because they work better with big data with many inputs. In general, the accuracy of machine learning models is better than statistical models. Deep learning, a branch of machine learning based on artificial neural networks, is becoming one of the core tools of the Industrial Revolution 4.0. were developed. The algorithms can automatically extract complicated fire features and successfully apply them to various fire scenarios [4]. In conclusion, deep-learning-based forest fire detection has a higher real-time performance and better robustness than the traditional approaches. Since forest fires are difficult to extract features from as dynamic targets, it is still a great challenge to balance the real-time performance and maintenance of a good recognition accuracy in forest fire detection at the same time [5, 6]. The model is inspired from GoogleNet architecture, considering its reasonable computational complexity and suitability for the intended problem compared to other computationally expensive networks such as AlexNet. To balance the efficiency and accuracy, the model is fine-tuned considering the nature of the target problem and fire data.

The main purpose of this study is to propose building an optimal model for forest fire detection using Fast – RCNN combined with advanced image processing techniques.

2. Methodology

The forest fire detection system is divided into 3 main stages: (1) image preprocessing, (2) feature extraction and generation of feature vectors, (3) model training and testing (Fig. 1).

The basic idea is that the classical Fast – RCNN accepts images of any size as input. Each convolutional layer in a Fast -RCNN generates feature maps from the input image that represent the intensity of each feature.

Transfer learning enables us to repurpose these models for any relevant task, from object identification for self-driving vehicles to caption generation for video clips. We customize a pre-trained model as a feature extractor and fine-tuner in this work, and a brief note on customization follows: This method uses previously learned representations to extract meaningful features from new samples. On top of the pre-trained model, we constructed a new classifier to reuse the feature mappings extracted by the previous dataset. It is not necessary to retrain the entire model in this method. The fundamental convolutional network already has features that can be used to identify images in general.

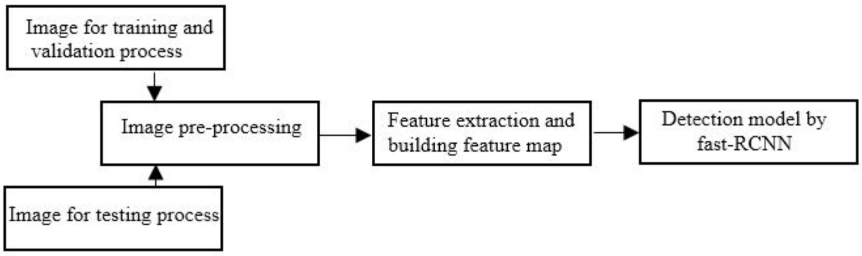


Fig. 1. The main steps of forest fire detection system.

We classify the input image by comparing its feature maps with the feature maps of the previous input images. For this purpose, feature vectors are extracted from the feature maps generated by each convolutional layer. The steps for creating feature maps and classifying an object using these maps are as follows (Fig. 2):

Step 1: pre-trained Fast – RCNN model for object classification. Including image input, keep-pro (to control the dropout rate), layer-3, layer-4, and layer-7.

Step 2: The upsampling of these low-resolution feature maps is done using transposed convolutions. Using transposed convolution with parameters:

- kernel = (4,4);
- stride = (2,2);
- padding = 'same'.

The next step is to optimize our neural network, aka building loss functions and optimizer operations. Here we use cross-entropy as our loss function and Adam as our optimization algorithm.

Step 3: layer-10: layer-9 is unsampled 2 times to match the dimensions with layer-3 of Fast – RCNN, using transposed convolution with parameters:

- kernel = (4,4);
- stride = (2,2);
- padding = 'same'.

After that, a skip connection was added between layer-3 and layer-10. The upsampling process is further refined by adding features from coarser but higher resolution feature maps from the lower layer. We take in important parameters including a number of epochs, batch size, loss function, optimizer operation, and place holder for the input image, label images, learning rate. For the training process, we also set keep-probability to 0.5 and learning-rate to 0.0001 to keep track of the progress

Step 4: Skip connection is introduced after each convolution block to enable the subsequent block to extract more abstract, class-salient features from the previously pooled features. Layer-10 is up-sampled 4 times to match the dimensions with input image size so we get the actual image back and depth is equal to a number of classes, using transposed convolution with parameters:

- kernel = (16,16);
- stride = (8,8);
- padding = 'same'.

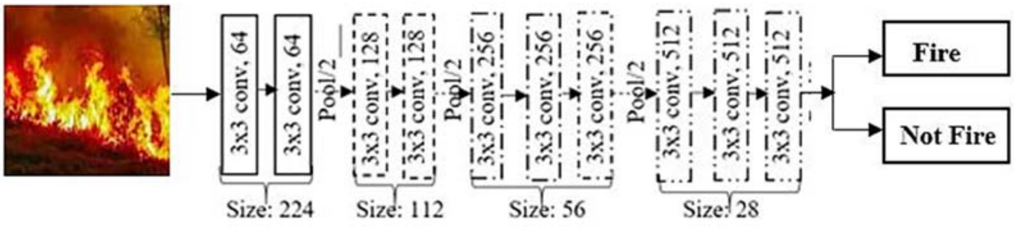


Fig. 2. An architecture of the Fast -RCNN to detect forest fire image.

3. Analysis and Evaluate Experimental Results

To evaluate the performance of a classifier machine learning model, the accuracy (ratio of correct predictions to the total number of points) and confusion matrix are often used. The tests show that the model trained by the Adam weight update algorithm, the number of epoch loops = 4000 and batch_size = 100 gives the best results.

It can be seen that the accuracy on the test dataset of the Fast-RCNN model is superior to that of other tested models, that is, it has the best generalization ability (Fig. 3). On the training dataset, the accuracy of Fast-RCNN model is better than that of ANN and VGG16-CNN models (Table 1).

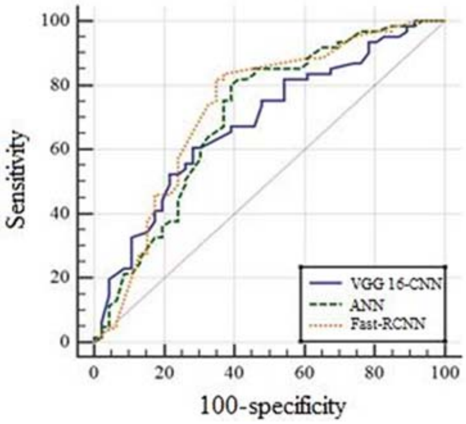


Fig. 3. Result of analysis ROC of 3 methods.

In this technique, we unfreeze a few top layers of the models and train both the newly added classification and unfrozen layers of the models. The higher-order feature representations of the underlying model can be "fine-tuned" in this way to make them more relevant to the dataset under consideration. Applying fine-tuning enables us to use pre-trained networks to distinguish classes in untrained datasets. Since the weights of the uppermost layers are retrained on a new dataset, fine-tuning will result in more accuracy than feature extraction-based transfer learning.

Table 1. The results of detection based on VGG16-CNN, ANN, proposed method (Fast-RCNN).

Criteria	VGG16-CNN	ANN	Proposed method
Accuracy (%)	96.79	93.11	99.50
MSE	0.330	0.511	0.016

4. Conclusions

A model based on the combination of the Fast-RCNN method and the computer vision algorithm is developed to detection forest fire. Extraction of features on the forest fire to create a feature map using the fast recurrent convolutional neural network method has improved the performance of the model under objective conditions such as contrast and light of weather. The main advantage is the fast and efficient processing time with high data sensitivity. The factors were all analyzed to design and arrange the parameters involved in the training and testing process to achieve the best model performance (> 99 %). The results were compared with other methods and show that the proposed method is the most suitable for system development

References

[1]. N. V. Baranovskiy, V. A. Kirienko, Mathematical simulation of forest fuel pyrolysis and crown forest fire impact for forest fire danger and risk assessment, *Processes*, Vol. 10, Issue 3, 2022, 483.

[2]. K. Muhammad, J. Ahmad, S. W. Baik, Early redetection using convolutional neural networks during surveillance for effective disaster management, *Neurocomputing*, Vol. 288, 2012, pp. 30-42.

[3]. C. B. Liu, N. Ahuja, Vision based Fire detection, in *Proceedings of the 17th Int. Conference Pattern Recognit. (ICPR'04)*, Aug. 2004, pp. 134-137.

[4]. T. H. Chen, P. H. Wu, Y. C. Chiou, An early fire-detection method based on image processing, in *Proceedings of the Int. Conference Image Process. (ICIP'04)*, 2004, pp. 1707-1710,

[5]. J. Choi, J. Y. Choi, Patch-based fire detection with online outlier learning, in *Proceedings of the 12th IEEE Int. Conference Adv. Video Signal Based Surveill. (AVSS'15)*, Aug. 2015, pp. 1-6.

[6]. G. Marbach, M. Loepfe, T. Brupbacher, An image processing technique for fire detection in video images, *Fire Safety J.*, Vol. 41, Issue 4, 2006, pp. 285-289.

5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2023),
7-9 June 2023, Tenerife (Canary Islands), Spain

(069)

Swin Transformer Tiny for Music Genre Classification using SpecAugment

Wanjun Liu, Yumeng Li and Haicheng Qu
School of software, Liaoning Technical University, Liaoning 125105, Huludao, China
Tel.: +86-18740159937, fax: +86-04295084330
E-mail: lym19970408@163.com

Summary: We present a Swin Transformer Tiny based learning network for addressing the weakness of deep learning networks in feature extraction from Mel spectrograms for music genre classification. Our proposed network employs a window mechanism that partitions the input Mel spectrogram into multiple small blocks, enabling feature extraction and classification on each small block. This mechanism allows for effective utilization of the temporal information contained in Mel spectrograms and improves the classification accuracy of the model. The lightweight structure of our proposed network reduces the number of parameters and computational cost. Additionally, we enhance the Mel spectrograms using the masking blocks of frequency channels method in SpecAugment and optimize the model using the Adam optimizer. Experimental results demonstrate that the proposed Swin Transformer Tiny-based model achieves a classification accuracy of 98.5 % on the GTZAN dataset.

Keywords: Swin transformer, Music genre classification, SpecAugment, Adam optimizer, Data augmentation.

---

1. Introduction

Music is an important tool for human entertainment. Due to variations in the atmosphere of music venues, diverse music customs and habits in different regions, and continuous innovation by music producers, music has gradually evolved into a rich and colorful array of genres. With the development of digital music media platforms, online music has become the mainstay of popular music consumption. The massive amount of music data has led to personalized demands such as music retrieval, playlist categorization, and preference recommendation. These demands depend on the classification of music genres. However, the diversity expressed by music genres makes audio algorithm classification a challenging task [1]. Efficient and accurate intelligent classification of music genres is of great significance for the development of music platforms and is also one of the pressing issues in the field of music information retrieval.

In this paper, we propose a learning network based on Swin Transformer Tiny, which has fewer parameters and higher computational efficiency. Compared to Swin Transformer and Transformer, Swin Transformer Tiny reduces computational complexity and memory usage while maintaining a certain classification accuracy. Additionally, we employed the Adam optimizer acceleration method and the method of randomly masking certain frequency channels from SpecAugment to enhance the Mel-spectrogram.

2. Related Work

At present, the music genre classification process can be roughly divided into two parts: feature extraction and machine learning. In 2002, Tzanetakis et al. [2] collected music data to create the GTZAN dataset, which contains 10 music genres, 1000 music samples, and extracted three groups of feature samples including pitch, timbre, and rhythm, which were then input into KNN and GMM for classification. The classification accuracy exceeded 60 %, marking one of the pioneering studies in the field of music genre classification. With the rapid development of machine learning, many researchers have proposed innovative feature extraction methods and classification models in the field of music genre classification. Mu X. [3] studied and designed a new method to speed up the KNN algorithm process. The method judges the closest genre based on the distance from the test sample to the centroid, although it performed poorly in accuracy, it did speed up the algorithm process. Kostrzewa D. [4] pointed out that the creation of deep neural networks is more challenging and requires more time for learning, resulting in poor classification results. Therefore, they proposed to assemble the neural network into a width ensemble for music genre classification, achieving 65.8 % classification accuracy on the FMA-small dataset. Rahardwika D.S., et al. [5] analyzed the impact of feature selection on the accuracy of music genre classification using SVM as a classifier. Seventeen features including acousticness, instrumentalness, popularity, energy, tempo, loudness, artist name, and beat were assigned values and grouped in experiments based on the importance of these 17 features. The results showed that correct feature selection helps to improve the accuracy of music genre classification. Ma Z. [6] compared the performance and feature extraction ability of neural networks and traditional machine learning algorithms in music genre classification. Yang R., et al. [7] designed a parallel recursive convolutional neural network (PRCNN), where parallel CNN and dual RNN blocks focus on extracting spatial features and time frame order, respectively, and the outputs of the two blocks are fused into a powerful representation of time-series data. The fused vector is then fed into a SoftMax function for classification. Gusain R., et al. [8] extracted the MFCC features of the dataset and analyzed and compared the performance of neural networks and XGBoost algorithms. The accuracy achieved on the dataset collected from the Kaggle website was 90.28 % and 89.52 %, respectively. Chang P. C., et al. [9] designed an end-to-end neural network called MS-SincResNet. JIE GAN [10] used a recurrent neural network and channel attention to obtain feature maps of music applied to music genre classification tasks. A 91 % accuracy was achieved on the GTZAN dataset.

From related work, it can be seen that there are difficulties in feature extraction, model complexity, and low accuracy in music genre classification tasks. To address these problems, we propose a Swin Transformer Tiny model to handle music genre classification problems and adopt the SpecAugment method to enhance the Mel spectrogram.

3. Methodology

3.1. Swin Transformer Tiny

The Swin Transformer [11] is a type of transformer-based neural network architecture that has achieved state-of-the-art results on a variety of computer vision tasks. The "Swin Transformer Tiny" is a smaller and more lightweight version of the original Swin Transformer model, designed to be faster and more efficient while still maintaining high levels of accuracy. The Swin Transformer Tiny uses a hierarchical architecture with a combination of patch-based and window-based self-attention mechanisms, allowing it to model both local and global dependencies in the input data. It also incorporates a series of bottleneck layers that reduce the computational complexity of the model while preserving its representational power.

3.2. SpecAugment

SpecAugment [12] is a data augmentation technique that was developed to improve the performance of automatic speech recognition (ASR) systems. The augmentation policy consists of warping the features, masking blocks of frequency channels, and masking blocks of time steps. Due to the need to further segment the Mel spectrograms in this study, it is difficult to control the hyperparameters of time warping and time masking blocks. Therefore, this paper abandons the methods of time warping and time masking blocks and only applies the method of masking frequency channels in SpecAugment to enhance the Mel spectrograms. It operates on the spectrogram representation of an audio signal, which is a visual representation of the sound frequency over time. By randomly masking portions of the spectrogram, SpecAugment encourages the ASR system to learn more robust features that are invariant to small variations in the input signal. This can improve the accuracy of the ASR system, especially in situations where there is a lot of background noise or other sources of variation in the input signal.

3.3. Model Structure

The model architecture that combines the masking of blocks of frequency channels in Swin Transformer Tiny and SpecAugment is depicted in Fig. 1.

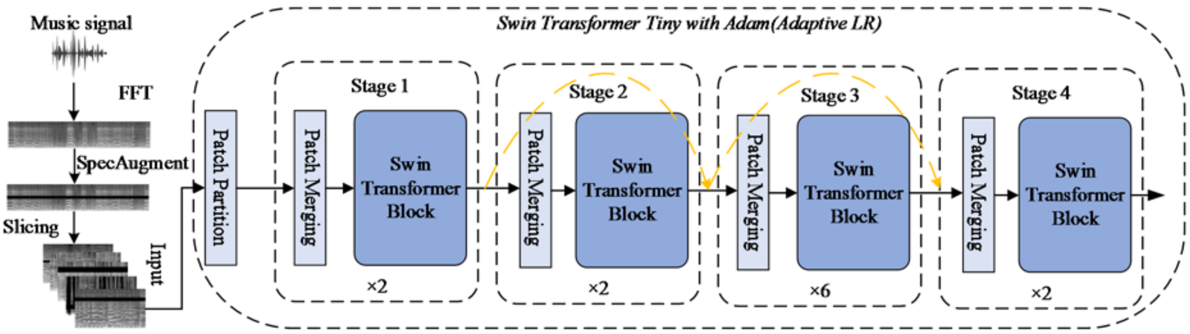


Fig. 1. Model structure diagram.

The Swin Transformer proposes a hierarchical construction method of Transformer that includes sliding window operations [13]. The sliding window operation restricts the attention calculation to a window, which can introduce local perception of the CNN convolution operation while saving computation. As shown in Fig. 2, the image is divided into multiple small blocks and these blocks are shared between consecutive self-attention layers. This strategy is helpful for processing the Mel spectrogram and can effectively reduce memory consumption and computational complexity. By preprocessing long audio signals into frequency spectrum slices, the computational complexity can be reduced to an acceptable range. Moreover, the use of the moving window partitioning strategy can also improve the model's response to short-term changes in audio signals, enabling the model to better capture the time-series features of audio signals. The Swin Transformer Tiny employs a strategy of limiting the computation to a window-based region, which greatly reduces the computational burden of the network and linearly scales down the complexity with respect to the image size.

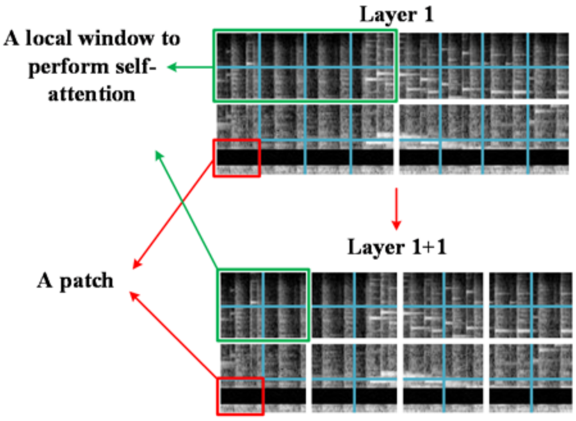


Fig. 2. Windows and patches of model.

The Swin Transformer Tiny designed in this paper is a variant of the Swin Transformer, which is a lightweight model suitable for computation in resource-constrained environments. Compared to the Swin Transformer, Swin Transformer Tiny has several differences:

(1) Model size: Swin Transformer Tiny is much smaller than the Swin Transformer, with fewer layers and fewer channels, requiring less computation resources to run, and with shorter training times.

(2) Residual connection mode: Residual connections are introduced in the four feature layers (yellow dashed line in Fig. 1) to better preserve the information of input features and improve the model's expressiveness and robustness.

(3) Adam Optimizer is used in the training process [14]: Its adaptive learning rate can automatically adjust the learning rate size based on the gradient of each parameter, making the training process more efficient. The model can converge quickly during training and obtain excellent results faster. The adaptive learning rate changes of the Swin Transformer Tiny model trained for 20 epochs using Adam Optimizer are shown in Fig. 3.

Music signals are often influenced by environmental noise, recording equipment, and other factors, resulting in varying degrees of signal variation. Randomly masking some frequency channels in SpecAugment can simulate noise and interference in audio signals, enabling the model to better handle such situations, reduce the model's reliance on local features, and focus more on global features, thereby improving the robustness of the network and mitigating overfitting.

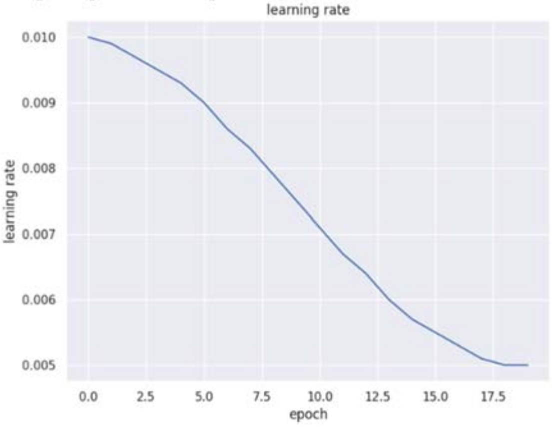


Fig. 3. Adam Optimizer's learning rate.

4. Experimental Evaluation

4.1. Dataset

The GTZAN [2] dataset is a widely used dataset for music genre classification research. It consists of 1000 audio files of 30 seconds each, with each file belonging to one of ten music genres. The genres included in the dataset are: blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae, and rock. We divided the GTZAN dataset into training and testing sets in an 8:2 ratio.

4.2. Dataset Preprocessing

In this study, we adopted a spectrogram segmentation approach to preprocess audio signals. Specifically, we transformed the audio signal into partial Mel spectrograms of size 1876×128, as shown in Fig. 4, and applied SpecAugment data augmentation (Fig. 5). Next, we divided Fig. 3 into 11 spectrograms of size 128×313, as illustrated in Fig. 6. This approach not only reduced the scale of training samples, but also expanded the size of the training data and enhanced the model's ability to capture local details. The final genre classification result could be obtained by aggregating the classification results of all spectrogram segments. Spectrogram segmentation is beneficial for improving genre classification performance. As we found that using time warping, time masking, and their combination did not perform as well as using frequency masking alone, we only applied frequency masking from SpecAugment to augment the Mel spectrograms.

4.3. Results Analyse

Our evaluation metric is the accuracy of the testing set, and we employ the Adam optimizer with adaptive learning rate. The Adam optimizer addresses the issue of sparse gradients on noisy data by combining the advantageous properties of the AdaGrad and RMSProp algorithms. Additionally, we apply decoupled weight decay to the Adam optimizer, which enhances the model's generalization performance. The experimental results indicate that our proposed algorithm achieves an accuracy of 98.5 % on the GTZAN dataset after only 20 epochs of training.

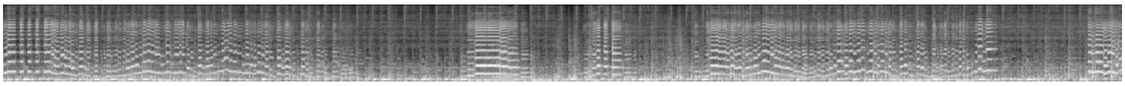


Fig. 4. Mel spectrum of Jazz027.

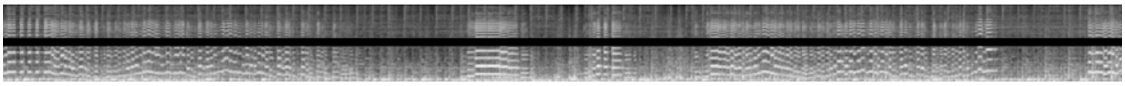


Fig. 5. Mel spectrum of Jazz027 for masking blocks of frequency channels.

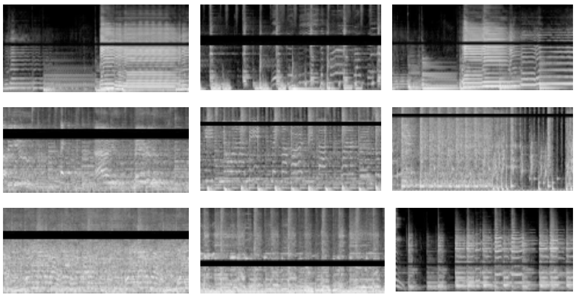


Fig. 6. Enhanced Mel spectrum after cutting.

We compared the test accuracy of AlexNet, GoogLeNet, Transformer, Swin Transformer, and our proposed model, Swin Transformer Tiny, under the same parameter conditions after training for 20 epochs. The results are shown in Table 1. It can be seen that Swin Transformer Tiny has certain advantages in handling music genre classification tasks compared to other models.

Table 1. Comparison of test accuracy among different models.

Model	Accuracy
Alexnet	83.96 %
GoogLeNet	89.76 %
Transformer	92.33 %
Swin Transformer	95.72 %
Swin Transformer Tiny	98.5 %

The training set accuracy curve for Swin Transformer Tiny trained for 20 epochs is shown in Fig. 7, and the test set accuracy is shown in Fig. 8. Fig. 9 shows the confusion matrix generated by GTZAN test set after training 20 epochs.

The 1.5 % inaccuracy of Swin Transformer Tiny is due to insufficient feature extraction of the spectral characteristics of the music genres Country, Jazz, and Rock. Specifically, the Mel-spectrograms of Jazz and Country are more similar, while the Mel-spectrograms of Rock and Metal are more similar. This confusion is also reflected in the confusion matrix shown in Fig. 9, which demonstrates significant misclassification among these four genres. Therefore, the classification performance of this model is compromised for these genres, resulting in the observed 1.5 % error.

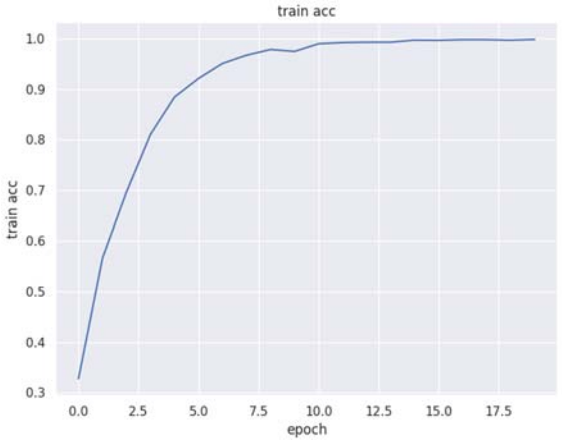


Fig. 7. Accuracy after 20 epochs of Train set.

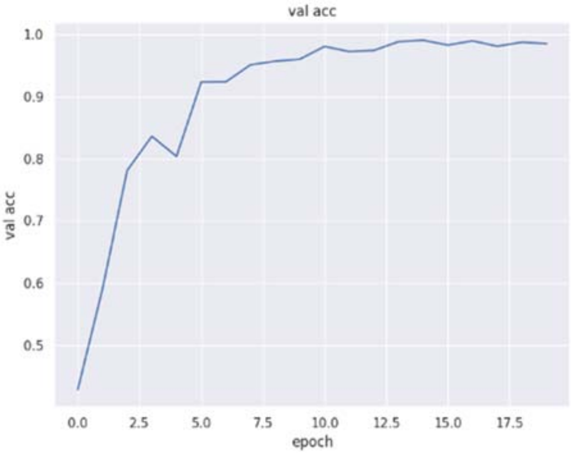


Fig. 8. Accuracy after 20 epochs of Testing set.

We reviewed excellent papers in the field of MGC over the past five years and selected the following innovative methods for comparison. The comparison of the accuracy of each model is shown in Table 2. It can be seen that Swin Transformer Tiny is suitable and efficient for handling music genre classification problems. The brief introduction of various models can be found in Section 2, "Related Work".

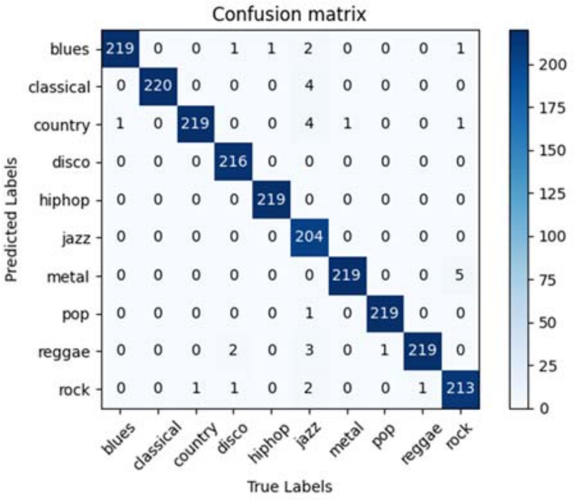


Fig. 9. Confusion matrix of Testing set.

Table 2. Comparison of accuracy among innovative models in the past five years.

Model	Accuracy
MK-NN[3]	55.3 %
Wide ensemble En10[4]	65.8 %
Improved-SVM[5]	81.2 %
SVM-PCA[6]	83 %
CNN+Bi-GRU[7]	89.3 %
XGBoost[8]	89.52 %
MS-SincResNet[9]	91.49 %
BLSTM[10]	93.1 %
S3T[13]	97.2 %
Swin Transformer Tiny	98.5 %

5. Conclusions

Experimental results indicate that Swin Transformer Tiny is effective in addressing the problem of music genre classification. The application of Adam optimizer and the masking of blocks of frequency channels in SpecAugment can also effectively improve the accuracy of the model. However, the drawback lies in the excessive preprocessing steps of the audio and the weak discriminative ability for the music genres of Country, Jazz, Rock, and Metal. Our next step will be to investigate how to process audio signals more efficiently to achieve higher feature extraction efficiency and model classification accuracy.

References

[1]. T. Łukaszewicz, D. Kania, A Music Classification approach based on the trajectory of fifths, *IEEE Access*, Vol. 10, 2022, pp. 73494-73502.

[2]. G. Tzanetakis, P. Cook, Musical genre classification of audio signals, *IEEE Transactions on Speech and Audio Processing*, Vol. 10, 2002, Issue 5, pp. 293-302.

[3]. X. Mu, Implementation of music genre classifier using KNN algorithm, Highlights in Science, *Engineering and Technology*, Vol. 34, 2023, pp. 149-154.

[4]. D. Kostrzewa, W. Mazur, R. Brzeski, Wide ensembles of neural networks in music genre classification, in *Proceedings of the 22nd International Conference Computational Science (ICCS'22)*, London, UK, June 21-23, 2022, 2022, pp. 64-71.

[5]. D. S. Rahardwika, E. H. Rachmawanto, C. A. Sari, et al., Effect of feature selection on the accuracy of music genre classification using SVM classifier, in *Proceedings of the International Seminar on Application for Technology of Information and Communication (iSemantic'20)*, 2020, pp. 7-11.

[6]. Z. Ma, Comparison between machine learning models and neural networks on music genre classification, in *Proceedings of the 3rd International Conference on Computer Vision, Image and Deep Learning & International Conference on Computer Engineering and Applications (CVIDL&ICCEA'22)*, 2022, pp. 189-194.

[7]. R. Yang, L. Feng, H. Wang, et al., Parallel recurrent convolutional neural networks-based music genre classification method for mobile devices, *IEEE Access*, Vol. 8, 2020, pp. 19629-19637.

[8]. R. Gusain, S. Sonker, S. K. Rai, et al., Comparison of neural networks and XGBoost algorithm for music genre classification, in *Proceedings of the 2nd International Conference on Intelligent Technologies (CONIT'22)*, 2022, pp. 1-6.

[9]. P. C. Chang, Y. S. Chen, C. H. Lee, MS-SincResnet: Joint learning of 1D and 2D kernels using multi-scale SincNet and ResNet for music genre classification, in *Proceedings of the International Conference on Multimedia Retrieval*, 2021, pp. 29-36.

[10]. J. Gan, Music feature classification based on recurrent neural networks with channel attention mechanism, *Mobile Information Systems*, Vol. 2021, 2021, pp. 33-42.

[11]. Z. Liu, Y. Lin, Y. Cao, et al., Swin transformer: Hierarchical vision transformer using shifted windows, in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV'21)*, 2021, pp. 10012-10022.

[12]. D. Spark, W. Chan, SpecAugment: A simple data augmentation method for automatic speech recognition, https://arxiv.org/pdf/1904.08779.pdf

[13]. H. Zhao, C. Zhang, B. Zhu, et al., S3T: Self-supervised pre-training with Swin transformer for music classification, in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'22)*, 2022, pp. 606-610.

[14]. Y. Wang, A convolutional neural network method based on Adam optimizer with power-exponential learning rate for bearing fault diagnosis, *Journal of Vibroengineering*, Vol. 24, 2022, Issue 4, 13.

(071)

Classifying Musical Instruments Using Temporal and Spectral Features with Deep Neural Nets

Cai Xiu Chiah, Lee Choo Tay and Weng Kin Lai
Tunku Abdul Rahman University of Management and Technology, 53300 Kuala Lumpur, Malaysia
Tel.: 6034145123, fax: 60341423166
E-mail: taylc@tarc.edu.my

Summary: Recognising musical instruments from an audio track has not been an easy task for machines, as compared to a trained human ear. However, machines trained with artificial intelligence can do this task as well as a human. The audio signal produced by an instrument has unique temporal and spectral features, which could be extracted for machine learning purpose. While most researchers have used spectral features, this research investigates the use of both temporal and spectral features of monophonic music instead. The performance of a recurrent neural network model called Bidirectional Long Short Term Memory (Bi-LSTM) network and two convolutional neural networks (CNNs) in musical instruments classification with log Mel-spectrogram were analysed and evaluated. When tested on a dataset of 14 musical instruments, the Bi-LSTM and 1-dimensional CNN models obtained a macro F1 score of 0.977 while 2-dimensional CNN model achieved 0.978.

Keywords: Convolution neural network, Long short term memory network, Deep learning, Musical instruments classification, Music information retrieval

---

1. Introduction

Musical instrument recognition is an active research area in the music information retrieval (MIR) field and it has contributed significantly in providing essential and useful information for various audio or music applications. Some examples are playlist generation, music recommendation system, music browsing, music transcription, and genre classification. Compared to text search when both input queries and data are text, music search is a more complex task as the music files are in digital format. Hence, it is imperative to convert music information into text format, i.e. music tags, to facilitate music search. Musical instrument recognition plays an important role as it provides instrumental information in music search with the proliferation of digital music files. Automatic music transcription and playlist generation in music streaming platforms rely heavily on MIR tasks to provide better users experience.

However, audio information extraction is a challenging task, especially from polyphonic music that includes human voice. The task of musical instruments identification in polyphonic music with multiple musical instruments requires a complicated system, as the sound generated by different instruments may varies in timbre, instrument quality and playing style, not to mention the overlapping of sound from multiple instruments. Convolution neural network (CNN) approach using spectral features has been widely used and has shown a better performance over the traditional machine learning approach. However, MIR is intrinsically a temporal task and past literature have indicated that temporal features provide important information that is useful in musical instrument classification [1]. It is necessary to develop an effective model on processing temporal information and learning from time series data, i.e., music. In this paper, a temporal model is developed and its performance compared to two other commonly employed spectral models.

2. Prior Work

Musical instrument recognition is a research topic that has been extensively studied in various forms with the term "*instrument classification*" or "*instrument identification*". Most of the research have utilized either machine learning or deep learning as their classifier models. This research can be categorized into two different research topics based on the type of input data, namely instrument identification in either monophonic or polyphonic music. The former is the focus of this study.

For musical instruments recognition in monophonic music, Toghiani-Rizi and Windmark [2] collected audio samples from the London Philharmonic Orchestra dataset and converted them into frequency domain signals. The authors trained an artificial neural network (ANN) model and carried out experiments analyzing the effect of different audio signal features on recognition accuracy – the whole audio sample, only the attack of audio signal, all characteristics of signal except attack, the first 100 Hz of the signal in frequency domain and the following 900 Hz of the same spectrum. The results showed that the ANN model with complete music sample reached the highest average accuracy of 93.5 %.

Chakraborty and Parekh [3] extracted 6 audio descriptors including Mel Frequency Cepstral Coefficients (MFCC), linear predictive coding (LPC), cepstral coefficients (CC) and spectral centroids as spectral descriptors, pitch salience and hierarchical clustering on principal components (HPCP) as tonal descriptors. All the features respectively acted as inputs to 4 different classifiers, namely, k-nearest neighbour (k-NN), support vector machine algorithm (SVM), ANN and random forest. As a result, an optimally tuned ANN classifier showed the best performance. Meanwhile, CC was considered the best feature among other features, even for the regular approaches with MFCC.

In [4], the authors introduced the implementation of an ANN-based model with specific transient length selection which contained most of the significant music information. Two experiments were carried out on string instruments recognition where the first experiment had applied the first 1000 Hz normalized frequency spectrum with selected onset position as input data and achieved an accuracy of 96.47 %. In the second experiment, the authors had extended the signal attack length which included signal peak amplitude that carried more fundamental information, resulting in the final accuracy of 98.22 %.

Kakarwal and Chaudhary [5], proposed a k-NN model on classifying different Indian string instruments by analyzing features of audio samples including zero crossing rate (ZCR), roughness, spectral centroid, brightness and roll-off. The research claimed that the average accuracy of the classifier with the combination of more than 3 audio features was much higher than that with only a single feature extraction, which are 90 % and 67 % respectively. It showed that a major advantage of audio feature extraction involving multiple temporal and spectral features is that it provides sufficient music information for retrieval.

RNN network has shown good success in music genre and emotion classification. However, due to the drawbacks of vanishing gradient and long-term dependencies problem in RNN, Long Short Term Memory (LSTM) was proposed as an improved version and has been widely used in the research of music genre classification. Rajesh and Nadini [6] achieved more than 85 % accuracy in classifying 4 instruments with 4 emotions using RNN structure embedded with LSTM. Deepak and Prasad [7] showed that LSTM could process time series data, music in this case, to discover patterns over time which are distinctive to each data class. In this research, the LSTM model achieved an accuracy between 95 % to 96 % in music genre classification on the GTZAN dataset. Combining LSTM with SVM improved the accuracy to 98 %.

With the motivation from previous LSTM-based tasks, Yun and Bi [1] implemented a single layer (LSTM) network on the task of musical instrument classification through temporal features extraction of audio signals. Like the CNN approach, the authors trained the LSTM model with log Mel-spectrogram but with temporal features. The LSTM model was then compared with a LeNet CNN model on the performance of instrument recognition and both models obtained the accuracies of 81.85 % and 83.04 % respectively, which are pretty close to each other.

In addition, Han et al [8], Yu et al [9], Solanki and Pandey [10] Racharia et al [11] and Gururani et al. [12, 13] had achieved significant progress in using various temporal and spectral features as inputs to various machine learning algorithms and neural network architectures in their work.

Motivated by these past researchers, this paper explored the performance of a LSTM network and two convolution networks in recognising musical instruments in monophonic music taken from a standard dataset.

3. System Architecture

The dataset used in this project were extracted from monophonic audio samples in Kaggle dataset developed by Dibakar Sil [14]. The audio samples of 14 instruments from the family of string, woodwind and brass were categorized into different subfolders annotated with the instrument name. The audio tracks were pre-processed before feeding into the models for training, validating and testing. The models built for evaluation are Bi-LSTN, one-dimensional CNN (1D CNN) and two-dimensional CNN (2D CNN). Kapre (Keras Audio Preprocessing) library developed by Choi et al [15] was used to implement and modify the models.

3.1. Audio Signal Preprocessing

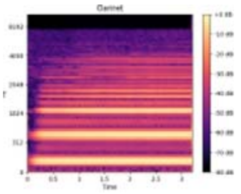
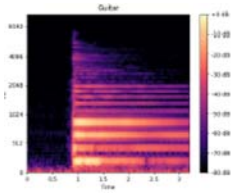
The audio samples in 16-bit WAV format were first enveloped to remove silence in order to avoid the problem of every audio clip having similar patterns. Then each clip was trimmed into pieces of 1 second to form the actual input data. Consequently, the total number of audio clips for training and validation was 2470 clips of audio signal of equal length, in which 80 % was used for training and 20 % for validation. A fresh set of data was prepared in a similar manner for testing.

The trimmed audio clips were down-sampled with a rate of 22050 Hz which was the Nyquist rate, and the stereo audio was converted into mono input by taking the average of the left and right audio channels. The sampling rate of 22050 Hz allows the use of up to 11025 Hz as Nyquist frequency which can include most of the instrument information. Noises above the frequency range were removed.

In this research, a spectrogram was used as the time-frequency represented input data to the system and hence the implementation of short-time Fourier transform (STFT) with 512 sample length was introduced. Each of the 1-s audio signals was further divided into short term sequences of fixed window size of 25 ms and a hop size of 10 ms, and then the fast Fourier transform (FFT) was applied on the signals. This approach is similar to [6].

To obtain a Mel-spectrogram, a Mel filter-bank with 128 bands was used to reduce the dimensionality of the spectrogram generated, as suggested by [8]. The Mel-spectrogram was then converted into logarithmic scale for better audio visualization. The waveforms and the log Mel-spectrograms (MFCC) of two instruments used are illustrated in Table 1.

Table 1. Examples of Waveform and MFCC for Clarinet and Guitar.

Instrument	Waveform	*MFCC*
Clarinet		
Guitar		

3.2. Network Architecture

The models of the three networks were implemented using the Kapre library.

A) Bidirectional Long Short Term Memory (Bi-LSTM) Architecture

Our LSTM model was implemented with two dense layers in a bidirectional manner, as illustrated in Table 2 The audio data was first normalised before going through the actual LSTM model, a time distributed dense layer was created to filter out the signal feature over time from the input data. Next, a bidirectional layer was constructed to compute the gradient in both forward and backward manner. The output from the bidirectional layer was then concatenated with the features filtered from the previous time-distributed dense layer

Table 2. Bi-LSTM Structure.

Layer Type	Description
Input layer	Mel-spectrogram
Time-distributed layer	64 units with tanh activation
Bidirectional layer	32 units with return sequence
1st dense layer	64 units with ReLU activation
Max-pooling layer	1D max-pooling
2nd dense layer	32 units with ReLU activation
Flatten and fully connected layer	–
Dropout layer	Dropout rate of 0.5
Classification layer	14 units of Softmax function

By doing so, the LSTM network could not only learn the signal features from the input data but also took reference in the concatenated features from these two layers. Subsequently, two dense layers with ReLU activation were created with a max-pooling layer in between to summarize the features from the first layer by calculating its maximum value. Since LSTM was prone to overfitting, according to [6], a dropout of 0.5 was added before the classification layer to minimize the problem. On the classification layer, a SoftMax function was used to indicate the probability for each instrument.

B) 2D CNN Architecture

A simple 2-dimensional convolutional neural network (2D CNN) model similar to that of computer vision-based approaches like VGG-Net structure was constructed. The Mel-spectrogram formed the input data to the 5 layers CNN model. First layer was the hyperbolic tangent (tanh) convolution layer with a kernel size of 3×3 to learn the small features of data. The kernel size was maintained in the following convolution layers and the number of filters in each layer was increased by a factor of 2 for every two convolution layers to obtain the more specific information of the data as illustrated in Table 3.

Zero padding was applied on the input data for each convolution layer to prevent the loss of some useful information with the increase of convolution layers. Also, a max-pooling method was used after each convolution layer to compute the maximum value of the feature for each patch. However, after the last convolution layer, a global max-pooling layer was constructed to obtain the maximum value of all features before the fully flatten connected layer. After that, a dropout layer was added with the dropout rate of 0.3 as a slight improvement over [10] and eventually a SoftMax function was used for the classification layer.

Table 3. 2D CNN Structure.

Layer Type	Description
Input layer	Mel-spectrogram
1st convolution layer	3×3 convolution with 8 filters
Max-pooling layer 1	2×2 max-pooling
2nd convolution layer	3×3 convolution with 16 filters
Max-pooling layer 2	2×2 max-pooling
3rd convolution layer	3×3 convolution with 16 filters
Max-pooling layer 3	2×2 max-pooling
4th convolution layer	3×3 convolution with 32 filters
Max-pooling layer 4	2×2 max-pooling
5th convolution layer	3×3 convolution with 32 filters
Global max-pooling layer	–
Flatten and fully connected layer	–
Dropout layer	Dropout rate of 0.3
Classification layer	Softmax function

C) 1D CNN Architecture

A 1-dimensional convolutional neural network (1D CNN) was constructed with reference from [16]. The model has 5 time-distributed convolution layers which is the same number of layers as the 2D CNN model. With the time-distributed layers, the model would learn only the temporal features from the Mel-spectrogram. Allamy and Koerich [16] had reported that the small convolutional filters with the size of 3 gave the best accuracy. Hence, the kernel size for each convolutional layer was fixed at 3. The number of filters in each layer was increased by a factor of 2 for every layer as presented in Table 4. A 1D max-pooling layer was added after each convolution layer for maximum value computation from the previous layer. Similar to the 2D CNN model, a dropout layer was added before the classification layer and SoftMax function was used.

Table 4. 1D CNN Structure.

Layer Type	Description
Input layer	Mel-spectrogram
1st convolution layer	8 filters with the size of 3
Max-pooling layer 1	1D max-pooling
2nd convolution layer	16 filters with the size of 3
Max-pooling layer 2	1D max-pooling
3rd convolution layer	32 filters with the size of 3
Max-pooling layer 3	1D max-pooling
4th convolution layer	64 filters with the size of 3
Max-pooling layer 4	1D max-pooling
5th convolution layer	128 filters with the size of 3
Global max-pooling layer	–
Flatten and fully connected layer	–
Dropout layer	Dropout rate of 0.3
Classification layer	Softmax function

3.3. Performance Evaluation

The performance of the models was evaluated through predictions on new audio files taken from the same dataset. To determine accuracy on instrument recognition, the predicted results were visualized in a confusion matrix. The values of True Positive (*TP*), False Positive (*FP*), True Negative (*TN*) and False Negative (*FN*) were used to compute the overall accuracy, precision (*P*) and recall (*R*) of the model using equations (1) to (3). Following the evaluation method recommended by Han, Kim and Lee [8], the *F1* measure in equation (4) was used to obtain the overall performance of the model, which is the harmonic mean of precision.

$$Accuracy = \frac{TP+TN}{Total\ Predictions}, \tag{1}$$

$$P = \frac{TP}{TP + FP}, \tag{2}$$

$$R = \frac{TP}{TP + FN}, \tag{3}$$

$$F1 = \frac{2 \times P \times R}{P + R} \tag{4}$$

In addition, as the dataset used has multi-class labels with non-uniform distribution, macro-average computation would be more impartial than micro-average. The *P* and *R* scores were obtained individually for each class before the calculation of macro P_{macro} and R_{macro}, as illustrated in equations (5) and (6), where k is the total number of classes. $F1_{macro}$ was then computed using P_{macro} and R_{macro}, as shown in (7).

$$P_{macro} = \frac{\sum_{i=1}^{k} P_i}{k}, \tag{5}$$

$$R_{macro} = \frac{\sum_{i=1}^{k} R_i}{k}, \tag{6}$$

$$F1_{macro} = \frac{2 \times P_{macro} \times R_{macro}}{P_{macro} + R_{macro}} \tag{7}$$

With the overall accuracy and F1 measure obtained, a comparison was made among the Bi-LSTM model and the CNN models in processing temporal and spectral features in monophonic audio signals.

4. Results and Discussions

All three trained models were able to achieve validation accuracy of 1 after 30 epochs of training. It is worth to note that 2D CNN achieved the maximum accuracy the earliest on the 8th epoch, the LSTM model on the 12th epoch but fluctuated before stabilising on the 24th epoch, while the 1D CNN model achieved stable maximum accuracy only on the 29th epoch.

4.1. Predictive Performance of Models

A new set of data was prepared for the musical instruments prediction to examine the practical performance of the three models. The test results are illustrated in Fig. 1. All the three models produce very similar accuracies. The predictive accuracy of B-LSTM is the same as the 1D CNN model of 0.997 while the overall accuracy of the 2D CNN model was 0.998.

Fig. 1. Performance comparison of the models.

The macro precision of Bi-LSTM and 1D CNN are similar at 0.977 while 2D CNN is slightly better at 0.987. However, the macro recall of Bi-LSTM is the worst at 0.975, 1D CNN is marginally better at 0.978 while 2D CNN is the winner at 0.984. The macro F1 score of in ascending order is therefore Bi-LSTM at 0.976, 1D CNN at 0.977 and 2D CNN at 0.985.

4.2. Performance on Instrument-wise Identification

In order to have a detailed evaluation on the model performance, instrument-wise analysis was done in this section as the predictive performance of the proposed model varied according to instruments. The results are presented in three clustered column charts in Fig. 2.

As shown in Fig. 2(a), Bi-LSTM performed the best on cello and trombone with an accuracy and F1 score of 1 and the worst in sax soprano with the lowest F1 score of 0.91. The low F1 score is mainly attributed by the low precision of 0.833 caused by high false positive (FP) where the model wrongly predicted 12 cases of violin and 1 case of flute, French horn, trumpet and tuba each as sax soprano. The model is found to perform poorly on violin in recall due to high false negative with 12 cases wrongly predicted as sax soprano mentioned above.

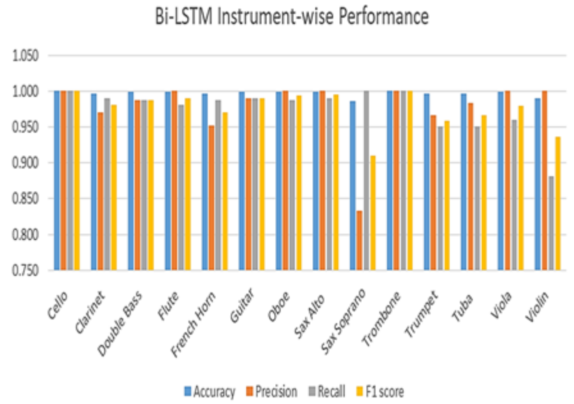
Even though the 1D CNN model only learned the temporal features, its performance is fairly consistent across all the instruments as shown in Fig. 2(b). It obtained perfect accuracy and F1 score in trumpet and the lowest F1 score of 0.941, due to a low precision of 0.923 caused by high false positive with 3 cases of cello and 1 case of violin wrongly predicted as viola. Note that all three are string instruments and it is possible that the temporal features of some musical notes are similar.

2D CNN that was trained on the same MFCC as Bi-LSTM was observed to perform poorly in sax soprano with a low precision of 0.909 and F1 score of 0.952, due to cases where cello, clarinet, flute, French horn and tuba were wrongly predicted as sax soprano. This is shown in Fig. 2(c). However, compared to Bi-LSTM, it performed well for violin with a perfect score of 1. Similar performance was achieved for both guitar and trombone.

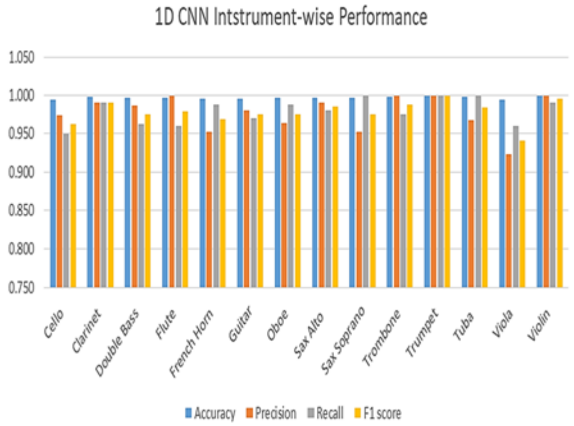
Nonetheless, with the use of Mel-spectrogram as the input training data, all three models had performed better than that of reported prior work which was done on monophonic music from London Philharmonic Orchestra dataset using spectral features. However, Chakraborty and Parekh [3] achieved an overall accuracy of 93 % with their ANN model using cepstral coefficients as features whereas Toghiani-Rizi and Windmark [2] obtained the highest accuracy of 93.5 % by using an ANN model with a feature vector based on the first 1000 Hz of the audio frequency spectrum.

5. Conclusions

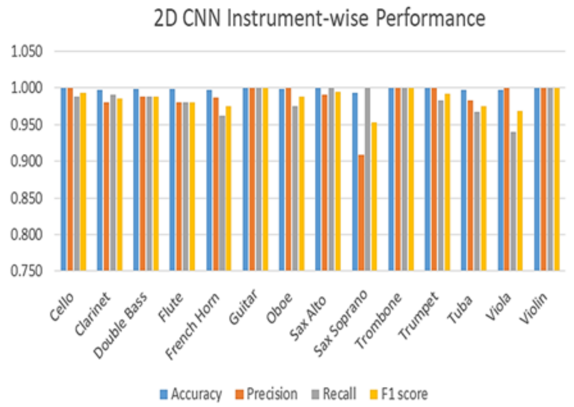
This paper evaluated the use of temporal and spectral features on three neural networks, namely Bi-LSTM recurrent neural network, 1-dimensional convolution network and 2-dimensional neural network in recognizing 14 musical instruments through the monophonic musical tones. The log Mel spectrogram of audio samples in Kaggle dataset were used for training and testing. All three models performed well with accuracy above 99.6 % with 2D CNN achieved an F1 score of 0.985, 1D CNN at 0.977 and Bi-LSTM at 0.976. Bi-LSTM performed the worst in the precision of sax soprano and the recall of violin as the model wrongly predicted large samples of the latter as the former, even though they belong to different families of instrument. This confusion did not occur in the CNN models.


Bi-LSTM Instrument-wise Performance

(a) Bi-LSTM


1D CNN Intstrument-wise Performance

(b) 1D CNN


2D CNN Instrument-wise Performance

(c) 2D CNN

Fig. 2. Instrument-wise performance by models.

The concept of image data processing was applied in this paper as Mel-spectrogram was selected as the input data. However, audio signal especially music data is much more complex than an image data due to its spectral characteristics overlapping with the temporal features. Moreover, music is a continuous signal and it can be in various lengths with the pattern varying according to the sound tuning and filtering. Hence, it is imperative to apply the spectral and temporal characteristics in designing a musical instruments recognition model.

For future work, the proposed model can be enhanced by training it with polyphonic audio samples to investigate its ability to identify musical instruments on polyphonic music accurately.

Acknowledgements

The authors are grateful to *Tunku Abdul Rahman University of Management and Technology* for various support in this research

References

[1]. M. Yun, J. Bi, Deep Learning for Musical Instrument Recognition, https://www.semanticscholar.org/paper/DEEP-LEARNING-FOR-MUSICAL-INSTRUMENT-RECOGNITION-Yun/ad4201d862fd0952d8028697d505ad7697337292

[2]. B. Toghiani-Rizi, M. Windmark, Musical instrument recognition using their distinctive characteristics in artificial neural networks, *arXiv Preprint*, 2017, arXiv:1705.04971.

[3]. S. S. Chakraborty, R. Parekh, Improved musical instrument classification using cepstral coefficients and neural networks, in Methodologies and Application Issues of Contemporary Computing Framework (J. K. Mandal, S. Mukhopadhyay, P. Dutta, K. Dasgupta, Eds), *Springer*, Singapore, 2018, pp. 123-138.

[4]. P. Roy, S. Roy, D. De, TMIR: transient length extraction strategy for ANN-inspired musical instrument recognition, in *Proceedings of the IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE'20)*, 2020, pp. 267-271.

[5]. S. N. Kakarwal, S. R. Chaudhary, Analysis of musical string instruments using k-NN, in *Proceedings of the International Conference on Smart Innovations in Design, Environment, Management, Planning and Computing, (ICSIDEMPC'20)*, 2020, pp. 283-286.

[6]. S. Rajesh, N. J. Nalini, Musical instrument emotion recognition using deep recurrent neural network, *Procedia Computer Science*, Vol. 167, 2020, pp. 16-25.

[7]. S. Deepak, B. G. Prasad, Music classification based on genre using LSTM, in *Proceedings of the 2nd International Conference on Inventive Research in Computing Applications (ICIRCA'20)*, 2020, pp. 985-991.

[8]. Y. Han, J. Kim, K. Lee, Deep convolutional neural networks for predominant instrument recognition in polyphonic music, *IEEE/ACM Transactions on Audio Speech and Language Processing*, Vol. 25, Issue 1, 2017, pp. 208-221.

[9]. D. Yu, H. Duan, J. Fang, B. Zeng, Predominant instrument recognition based on deep neural network with auxiliary classification, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 28, 2020, pp. 852-861.

[10]. A. Solanki, S. Pandey, Music instrument recognition using deep convolutional neural networks, *International Journal of Information Technology*, Vol. 14, Issue 3, 2019, pp 1659-1668.

[11]. K. Racharla, V. Kumar, C. B. Jayant, A. Khairkar, P. Harish, Predominant musical instrument classification based on spectral features, in *Proceedings of the 7th International Conference on Signal Processing and Integrated Networks (SPIN'20)*, 2020, pp. 617-622.

[12]. S. Gururani, C. Summers, A. Lerch, Instrument activity detection in polyphonic music using deep neural networks, in *Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR'18)*, April, 2018, pp. 569-576.

[13]. S. Gururani, M. Sharma, A. Lerch, An attention mechanism for musical instrument recognition, in *Proceedings of the 20th International Society for Music Information Retrieval Conference (ISMIR'19)*, Vol. 2, 2019, pp. 83-90.

[14]. D. Sil, Dibakar, Novice, Kaggle, https://www.kaggle.com/dibakarsil/music-instruments-and-2dfigures

[15]. K. Choi, D. Joo, J. Kim, Kapre: On-GPU audio preprocessing layers for a quick implementation of deep neural network models with Keras, in *Proceedings of the 34th International Conference on Machine Learning (ICML'17),* Sydney, Australia, 2017.

[16]. S. Allamy, A. L. Koerich, 1D CNN architectures for music genre classification, in *Proceedings of the IEEE Symposium Series on Computational Intelligence (SSCI'21)*, May 2021, pp. 01-07.

(072)

Enhancing Graph Representation Learning with Attention-driven Spiking Neural Networks

Huifeng Yin, Mingkun Xu, Jing Pei and Lei Deng
Center for Brain-Inspired Computing Research (CBICR), Department of Precision Instrument,
Tsinghua University, Beijing, China
E-mail: {peij,leideng}@mail.tsinghua.edu.cn

Summary: Graph representation learning has become a crucial task in machine learning and data mining due to its potential for modeling complex structures such as social networks, chemical compounds, and biological systems. Spiking neural networks (SNNs) have recently emerged as a promising alternative to traditional neural networks for graph learning tasks, benefiting from their ability to efficiently encode and process temporal and spatial information. In this paper, we propose a novel approach that integrates attention mechanisms with SNNs to improve graph representation learning. Specifically, we introduce an attention mechanism for SNN that can selectively focus on important nodes and corresponding features in a graph during the learning process. We evaluate our proposed method on several benchmark datasets and show that it achieves comparable performance compared to existing graph learning techniques.

Keywords: Graph representation learning, Attention mechanisms, Spiking neural networks, Graph neural networks.

---

1. Introduction

Graphs are ubiquitous data structures that can represent a wide range of complex systems, from social networks to biological networks. Graph representation learning seeks to create effective embeddings that encapsulate a graph's structural and semantic information, facilitating tasks like node classification, edge classification, and graph classification. Recently, spiking neural networks (SNNs) have been proposed as a powerful approach to graph representation learning due to their ability to encode and process spatiotemporal information in a more biologically plausible and energy-efficient way than traditional neural networks [1].

Recently, various approaches have been proposed for graph representation learning, including spectral methods, random walk-based methods, and neural network-based methods [2]. Among these, neural network-based methods, such as graph convolutional networks (GCNs) [3] and graph attention networks (GATs), have gained significant attention due to their capacity to capture complex nonlinear relationships in graph structures. However, traditional neural networks have limitations in processing spatiotemporal information, which is essential for numerous graph learning tasks due to the human brain's constant interaction with dynamic stimuli and evolving environments. SNNs, inspired by the way neurons communicate in the brain through spikes, are capable of encoding and processing spatiotemporal information more efficiently and in a biologically plausible manner, making them an attractive alternative for graph representation learning. But they still face challenges in processing large-scale graphs efficiently.

To address this issue, attention mechanisms have been introduced to selectively focus on relevant parts of the node features, allowing the network to selectively attend to important nodes and corresponding features in the graph [4]. In this paper, we propose an attention-driven SNN model that combines the benefits of attention mechanisms with the efficiency and interpretability of SNNs for graph representation learning. We demonstrate the effectiveness of our proposed model through several experiments on various benchmark datasets, showing comparable performance and better biological plausibility compared to existing graph representation learning methods.

2. Methods

Our proposed Spiking Graph Attention Network (SpikingGAT) model combines attention mechanisms with Graph Spiking Neural Network (Graph-SNN) to enable efficient graph representation learning [1]. Specifically, we introduce a graph attention mechanism that computes attention coefficients for each pair of nodes, allowing the model to weigh the contributions of neighboring nodes appropriately. This mechanism enables the SNN to focus selectively on relevant nodes and their corresponding features and effectively capture the underlying graph structure, during the learning process, as shown in Fig. 1. By integrating attention mechanisms into the Graph-SNN architecture, our SpikingGAT model achieves superior graph representation learning outcomes.

Specifically, in our SpikingGAT model, we first compute the attention coefficients for each node pair by applying a compatibility function that takes into account the nodes' features and their relative positions in the graph. The attention coefficient between node i and node j for the k^{th} head is computed as follows:

$$e_{ij}^{(k)} = \mathrm{LeakyReLU}\left(\mathbf{a}^{(k)T}\left[\mathbf{W}^{(k)}\mathbf{h}_i \parallel \mathbf{W}^{(k)}\mathbf{h}_j\right]\right), \qquad (1)$$

where $\mathbf{W}^{(k)}$ is a learnable weight matrix for the k^{th} head, $\mathbf{h}_i$and $\mathbf{h}_j$are the feature vectors of nodes i and j, respectively, $\mathbf{a}^{(k)}$ is a learnable attention vector, and LeakyReLU is the activation function. These coefficients are then normalized using a softmax function to ensure that they sum to one, promoting a smooth distribution of attention weights among neighboring nodes:

$$\alpha_{ij}^{(k)} = \frac{\exp\left(e_{ij}^{(k)}\right)}{\sum_{k\in\mathcal{N}_i}\exp\left(e_{ik}^{(k)}\right)}, \qquad (2)$$

where $\mathcal{N}_i$ is the set of neighboring nodes of node i, and $\alpha_{ij}^{(k)}$ is the normalized attention coefficient for the k^{th} head.

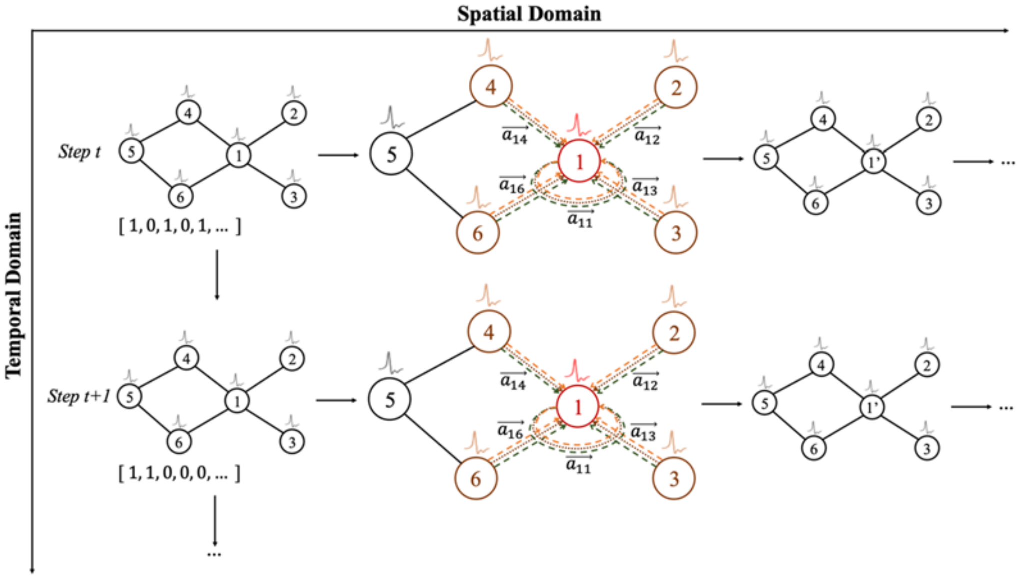


Fig. 1. The schematic diagram of SpikingGAT, showcasing the integration of multi-head attention and temporal domain within SNNs. Various arrow styles and colors signify distinct attention heads being performed independently with K = 3 heads.

Once we have computed the attention coefficients for each node pair, we apply the multi-head attention mechanism to the input features of each node by multiplying them with the corresponding coefficients. This operation is carried out at each layer of the SpikingGATs, updating the node representations with attention-weighted information from their neighbors. For each head k, the output feature matrix is given by:

$$\mathbf{H}^{(l+1,k)} = \sigma\left(\textstyle\sum_{j=1}^{N}\alpha_j^{(k)}\mathbf{W}^{(l,k)}\mathbf{H}^{(l)}\right), \qquad (3)$$

where σ is a nonlinear activation function, such as the ReLU function. Then, the output feature matrices from all k heads are concatenated to form the final output feature matrix:

$$\begin{aligned}&\mathbf{H}^{(l+1)} = \\ &= \mathrm{concat}\left(\mathbf{H}^{(l+1,1)}, \mathbf{H}^{(l+1,2)}, \dots, \mathbf{H}^{(l+1,K)}\right)\end{aligned} \qquad (4)$$

To account for the spiking behavior in SNNs, we employ a Leaky Integrate-and-Fire(LIF) [5] neuron model, which integrates the input features weighted by attention over time. The membrane potential is described by the following differential equation:

$$\tau\frac{du}{dt} = -[u(t) - u_{\mathrm{rest}}] + RI(t), \qquad (5)$$

where τ is the membrane time constant, u_{rest} is the resting membrane potential, R is the membrane resistance, and $I(t)$ represents the input current. When the membrane potential exceeds a certain threshold value u^{th}, the neuron generates a spike and the membrane potential is reset to the resting potential u_{rest} This process can be formulated as:

$$\begin{cases} o^t = 1, u^t = u_{rest} & \text{if } u^t \geq u_{th} \\ o^t = 0 & \text{if } u^t < u_{th} \end{cases}, \qquad (6)$$

where o^t represents the output spike at the time step t.

The SpikingGAT model architecture consists of multiple layers of LIF neurons, with each layer responsible for aggregating and transforming the attention-weighted features of the neighboring nodes. The output of the final layer is a set of effective embeddings that capture both the graph structure and node features.

We trained our SpikingGAT model using the iterative spiking message passing [1, 6] method of SNN that adjusts the connection weights between neurons based on the spike propagation and attention mechanism. This learning rule enables the network to adapt its connection strengths in response to both the graph structure and node features. Additionally, incorporating attention mechanisms enhances the model's ability to learn graph representations, focusing on the most informative connections in the graph.

3. Experiments

3.1. Basic Experiments

To evaluate the performance of our proposed SpikingGAT model, we first conducted experiments on three standard citation datasets, Cora, Citeseer, and Pubmed, where nodes represent paper documents and edges are undirected citation links. We provide a summary of the dataset statistics in Table 1, which is utilized to assess the model's ability to learn meaningful node representations within a single graph [3].

Table 1. Overview of citation datasets.

Dataset	Cora	Citeseer	Pubmed
Nodes	2708	3327	19717
Edges	5429	4732	44338
Node feat.	1433	3703	500
classes	7	6	3
Training/Validation/ Testing	140/500 /1000	120/500 /1000	60/500/ 1000

The Cora dataset contains 2708 nodes, 5429 edges, 7 classes, and 1433 features per node. The Pubmed dataset contains 19717 nodes, 44338 edges, 3 classes, and 500 features per node. The Citeseer dataset contains 3327 nodes, 4732 edges, 6 classes, and 3703 features per node. Each document node has a class label. We only use 20 labels per class during training with all feature vectors.

In these experiments, we maintain consistent settings for models across each dataset to ensure fairness. We employ the Adam [7] optimizer, setting an initial learning rate of 0.01 for GCN, and 0.005 for GAT and our SpikingGAT. All models are executed for 200 epochs, and we conduct 10 trials with varying random seeds. In each trial, the models are initialized using a uniform initialization method and trained by minimizing the cross-entropy loss on the training nodes. From the perspective of SNN, we set the time window (T) to 8 and establish a threshold (u^{th}) of 0.25 for basic performance evaluation. Specially, we set the leakage factor of 1, which does not exhibit decay in historical membrane potential as the time step progresses. For GATs and SpikingGATs, we adopt an MLP structure [Input-64-Output] with 8 attention heads. For GCNs, we adopt an MLP structure [Input-400-16-Output]. We implemented these models using the PyTorch deep learning framework and the Deep Graph Library (DGL). We trained the model on GPU-enabled hardware for efficient parallel processing of the spiking neurons and attention mechanisms.

Our results from 10 trials are presented in Table 2. The results indicate that, despite employing binary spiking communication, our SpikingGAT models achieve performance comparable to the state-of-the-art results with a slight gap. This demonstrates the feasibility and capability of the spiking mechanism and spatiotemporal dynamics in handling diverse features from different nodes, as well as their effectiveness in graph scenarios with few labels.

Table 2. Performance comparison on citation datasets.

Method	Cora (ACC.±s.d.)	Citeseer (ACC.±s.d.)	Pubmed (ACC.±s.d.)
GCN [4]	81.4±0.5	70.9±0.5	79.0±0.3
GAT [4]	83.0±0.7	72.5±0.7	79.0±0.3
SpikingGAT	**79.9±0.9**	**68.4±0.5**	**78.0±0.5**

Furthermore, we employed t-SNE [8] to visualize the feature representation capabilities of our SpikingGAT model compared to GCN and GAT in their last hidden layer features, as shown in Fig. 2. Specifically, on the Cora, Citeseer, and Pubmed datasets, our SNN model demonstrated superior performance in distinguishing between different classes and convergence within the same class, thereby demonstrate more effective representation learning capabilities compared to GCN and GAT.

3.2. Extended Experiments

In addition to the basic experiments on single graph datasets, we conducted extended experiments on multi-graph datasets to further validate the effectiveness and versatility of our SpikingGAT model in various graph learning tasks. By evaluating our model on tasks such as node classification, edge classification, and graph classification, we aimed to demonstrate the model's capability to generalize and adapt to different types of graph tasks. We summarize these dataset statistics used in extended experiments in Table 3.

Specifically, for node classification tasks, we used the SBM CLUSTER datasets, which are generated with the Stochastic Block Model (SBM). SBM is a traditional graph generation model in which each node belongs to a different community, and each community is connected with different probabilities [9].

For edge classification tasks, we employed the TSP dataset, which is about the traveler’s problem, as follows: “Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city and returns to the origin city?”. This dataset is a collection of complete graphs with weighted edges representing the distances between cities. The goal of this task is to classify edges as part of the optimal tour or not [10], testing our model's capacity to learn edge-level representation effectively.

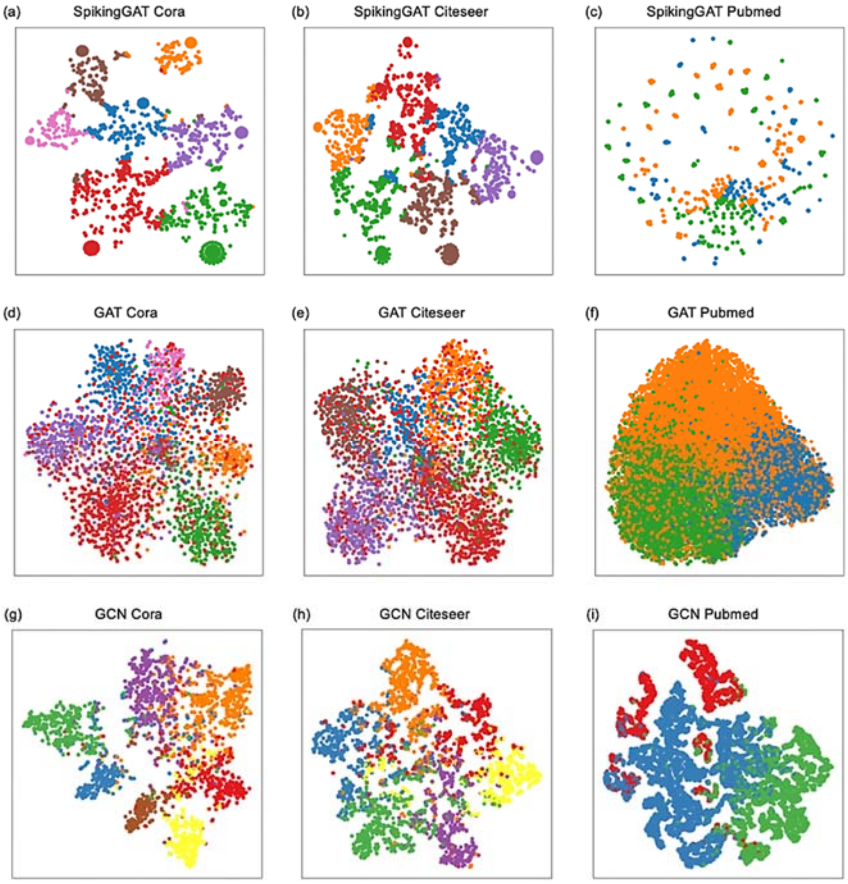


Fig. 2. (a), (b), (c) Visualization of the last hidden layer features of the SpikingGAT on the Cora, Citeseer, and Pubmed datasets produced by t-SNE; (d), (e), (f) Visualization of the last hidden layer features of the GAT on the Cora, Citeseer, and Pubmed datasets produced by t-SNE; (g), (h), (i) Visualization of the last hidden layer features of the GCN on the Cora, Citeseer, and Pubmed datasets produced by t-SNE.

Table 3. Overview of multi-graph datasets [9].

Dataset	SBM CLUSTER	TSP	MNIST
Graphs	12000	12000	70000
Avg. Nodes	117.20	275.76	70.57
Avg. Edges	4301.72	6894.04	564.53
Node feat.	Node Attr(7)	Coord(2)	Pixel+ Coord(3)
classes	6	2	10
Training/ Validation/ Testing	10000/ 1000/ 1000	10000/ 1000/ 1000	55000/ 5000/ 10000
Task Type	Node Classification	Edge Classification	Graph Classification

Finally, for graph classification tasks, we convert each image in the popular MNIST datasets into graphs using super-pixels and classify these graphs. The node features of the graph are generated by the intensity and position of the super-pixels, and the edges are k nearest neighbor super-pixels, which is set to 8 [9]. The resulting graphs are of sizes 40-75 nodes. In this experiment, we aimed to evaluate the SpikingGAT model's performance in learning meaningful representations for entire graphs, capturing the global structure and relationships between nodes.

In these extended experiments, we employ the Adam optimizer, setting the initial learning rate of 0.001 and the minimum learning rate of 10^{-5}. All models run up to 500 epochs, and we conduct 4 trials with varying random seeds. From the perspective of SNN, we maintained the same experimental settings as in the basic experiments with the time window $T = 8$, the firing threshold $V^{th} = 0.25$, and the leakage factor of 1. For GAT and SpikingGAT, we adopt 4 hidden layers of 19 dimension with 8 attention heads and an MLP layer (152 neurons) for classification. For GCN and SpikingGCN, 4 hidden layers of 152 dimension and an MLP layer of 152 neurons.

As shown in Table 4, our SpikingGAT models outperform the GCN models in SBM CLUSTER and MNIST datasets and outperform the SpikingGCN models in all datasets. And in Fig. 3, we plotted the accuracy and loss curves of SpikingGAT and SpikingGCN during both the training and testing phases. The curves illustrate that SpikingGAT converges faster than SpikingGCN and achieves higher accuracy and lower loss during both the training and testing phases. The results demonstrate the effectiveness of incorporating attention mechanisms into Spiking Neural Networks to enhance their performance in different graph learning tasks, highlighting their generalization capabilities.

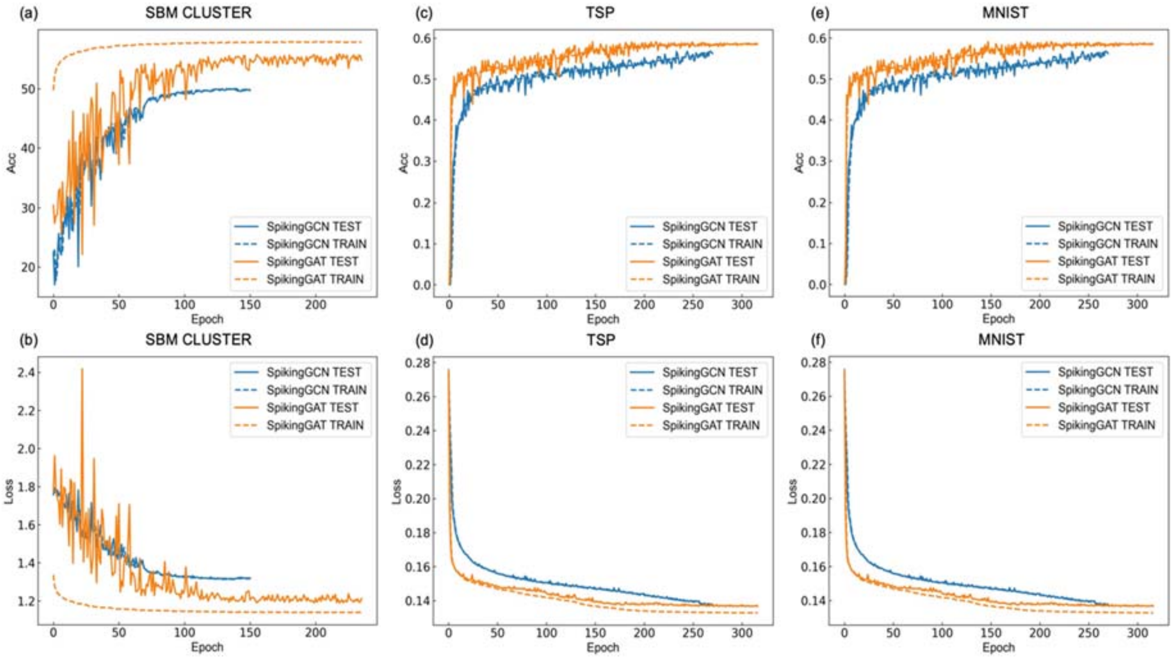


Fig. 3. (a)(c)(e) Accuracy variation of SpikingGAT and SpikingGCN with respect to the epoch on training and test dataset. (b)(d)(f) Loss variation of SpikingGAT and SpikingGCN with respect to the epoch on training and test dataset.

Besides, our SpikingGAT models can achieve comparable performance with the GAT models with a minor gap. It proves the feasibility and capability of spiking mechanism and spatial-temporal dynamics, which can work well on graph representation learning.

In this section, we have evaluated the performance of our proposed SpikingGAT model on a range of graph learning tasks, including node classification, edge classification, and graph classification. Our experiments on both single graph datasets and multi-graph datasets have demonstrated the SpikingGAT model's capability to learn effective representations and generalize to different types of graph tasks.

Table 4. Performance comparison on multi-graph datasets.

Method	SBM CLUSTER (ACC.±s.d.)	TSP (F1 SCORE ±s.d.)	MNIST (ACC.±s.d.)
GCN [9]	47.828 ±1.510	0.643 ±0.001	90.120 ±0.145
GAT [9]	57.732 ±0.323	0.671 ±0.002	95.535 ±0.205
Spiking GCN	50.181 ±1.284	0.568 ±0.005	92.318 ±0.005
Spiking GAT	**55.576 ±0.193**	**0.589 ±0.005**	**95.483 ±0.002**

4. Conclusion

This paper introduces a novel Spiking Graph Attention Network (SpikingGAT) model that effectively integrates attention mechanisms with spiking neural networks for graph representation learning. SpikingGAT can effectively deal with the spatiotemporal information in graph structures, which is a limitation of traditional neural networks. In addition, the integration of attention mechanisms allows the model to selectively focus on important nodes and features, resulting in improved performance in graph representation learning tasks.

We evaluate the performance of our SpikingGAT model on various benchmark datasets, including single graph datasets (Cora, Citeseer, and Pubmed) and multi-graph datasets (SBM CLUSTER, TSP, and MNIST). Our experiments demonstrate that SpikingGAT model achieves comparable performance to GCN and GAT models while maintaining better biological plausibility. Furthermore, our model exhibits better performance in graph representation learning, as evidenced by t-SNE visualizations.

These experiments validate the effectiveness and versatility of our SpikingGAT model, demonstrating its ability to generalize and adapt to different types of graph tasks. By combining efficient attention mechanisms with interpretable SNNs, our work opens up new possibilities for future research and applications in graph representation learning.

Acknowledgements

This work was supported by Science and Technology Innovation 2030 – New Generation of Artificial Intelligence, China project (2020AAA0109101), National Natural Science Foundation of China (No. 62106119, 62276151) and Zhejiang Lab’s International Talent Fund for Young Professionals.

References

[1]. M. Xu, Y. Wu, L. Deng, et al., Exploiting spiking dynamics with spatial-temporal feature normalization in graph learning, *arXiv Preprint*, 2021, arXiv:2107.06865.

[2]. W. L. Hamilton, R. Ying, J. Leskovec, Representation learning on graphs: Methods and applications, *arXiv Preprint*, 2017, arXiv:1709.05584.

[3]. T. N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, *arXiv Preprint*, 2016, arXiv:1609.02907.

[4]. P. Velickovic, G. Cucurull, A. Casanova, et al., Graph attention networks, *arXiv Preprint*, 2017, arXiv:1710.10903.

[5]. W. Gerstner, W. M. Kistler, R. Naud, et al., Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition, *Cambridge University Press*, 2014.

[6]. Y. Wu, L. Deng, G. Li, et al., Direct training for spiking neural networks: Faster, larger, in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, pp. 1311-1318.

[7]. D. P. Kingma, J. A. Ba, A method for stochastic optimization, *arXiv Preprint*, 2014, arXiv:1412.6980.

[8]. L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, *Journal of Machine Learning Research*, Vol. 9, Issue 11, 2008, pp. 2579-2605.

[9]. V. P. Dwivedi, C. K. Joshi, T. Laurent, et al., Benchmarking graph neural networks, *Journal of Machine Learning Research*, Vol. 23, 2022, pp. 1-48.

[10]. C. K. Joshi, Q. Cappart, L. M. Rousseau, et al., Learning the travelling salesperson problem requires rethinking generalization, *Constraints*, Vol. 27, Issues 1-2, 2022, pp. 70-98.

(074)

On-FPGA Neural Decoding with Spiking Neural Networks

G. Leone, L. Martis and P. Meloni
Università degli studi di Cagliari, piazza d'Armi, 09123, Cagliari, Sardinia (Italy)
E-mail: name.surname@unica.it

Summary: This work presents a spike decoding system implemented on a low-end Zynq-7010 FPGA, which includes a multiplier-less spike detection pipeline and a spiking-neural-network-based decoder mapped in the programmable logic. We tested the system on two publicly available datasets and achieved comparable results with state-of-the-art neural decoders that use more complex deep learning models. The system required 7.36 times fewer parameters than the smallest architecture tested on the same dataset. Moreover, by exploiting the spike sparsity property of the neural signal, the total amount of computations is reduced by about 90 % during a test carried out on real recorded data.

The low computational complexity of the chosen spike detection setup, combined with the power efficiency of spiking neural networks, makes this prototype a well-suited choice for low-power real-time neural decoding at the edge.

Keywords: Neural decoding, Spiking neural networks, FPGAs.

---

1. Introduction

Intracortical sensors have proven to be valuable instruments within the decoding of multiple motor functions from recorded neural activity. The decoding is often based on the processing of spikes, typical action potentials that can be identifiable in the neural track by means of a neural signal processing task, referred to as spike detection. The output of spike detection is called multi-unit activity (MUA) and can be effective in several decoding tasks when supported by reliable machine learning decoding algorithms, such as Long-Short Term Memory (LSTM) or Recurrent Neural Network (RNN) [1-3].

In this context, Spiking Neural Networks (SNNs), that, unlike other artificial neural networks, use biologically-plausible spiking neurons as their fundamental units, are a promising tool for low-power neural processing at the edge, intrinsically compatible with intracortical spikes.

This work presents a real-time spike-based neural decoding system integrated with a multiplier-less spike detector, designed to be implemented on an FPGA and connected to a 96-channel multi-electrode array (MEA) [4].

The results are demonstrated on a well-known benchmark, achieving state-of-the-art accuracy by using 7.36 times fewer parameters than the smallest alternative in literature. The system takes profit form typical sparsity in SNNs to save 90 % of the operations during inference.

To the best of our knowledge, in our work, a hardware SNN is used for the first time to solve a continuous neural signal decoding task in real-time.

2. System Architecture

The system architecture is represented in Fig. 1. The decoder is composed of two cascaded modules: a spike detector and an SNN-based decoder.

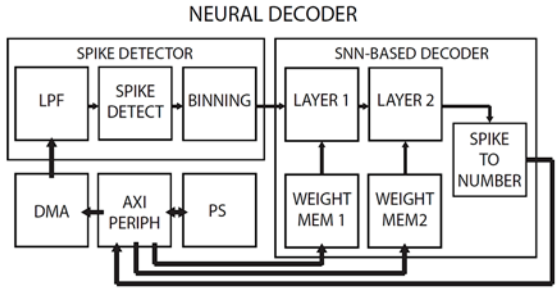


Fig. 1. General overview of the processing platform implemented on the FPGA device.

2.1. Spike Detector

The spike detection and the spike binning tasks are appointed to five pipelined modules that process the input channels in a time-multiplexed fashion.

The broadband neural signal is first filtered using a second-order moving average difference (MAD) filter. Afterwards, it is emphasized by means of a rectification. Then, the presence of a spike is detected when the resulting signal exceeds a mean-value-based threshold. Finally, a spike binning module counts the detected spikes per millisecond and forwards the bins to the SNN-based decoder.

2.2. SNN-based Decoder

The SNN used in this work is constituted of two dense layers of respectively 256 and 128 Loihi CUrrent BAsed Leaky Integrate and Fire (CUBA) Neuron. The Python implementation of the neuron model is derived by the PyTorch package SLAYER (Spike LAYer Error Reassignment), then extended and embedded as SLAYER 2.0 in the LAVA software framework [5] used in this work.

The hardware implementation of this algorithm relies on two layer modules, that compute the dynamics of the neurons, on a weight memory, and on a spike-to-number (S2N) converter, that translates the output spike train generated by the SNN's second layer into the target variable.

A *spike sparsity* stack allows taking advantage of the spike sparsity property of neural signal. It stores the addresses of the active spike sets permitting to involve in computation only active spikes and weights, to skip as much operations as possible for inactive inputs.

The neural decoder is connected to the Processing System (PS) through three AXI interfaces: 1) An AXI-stream interface is used to stream the neural samples in the PL; 2) An AXI-lite interface is controlled from the PS to set up the neural decoder; 3) An AXI-lite interface is used to transmit the decoding result.

The modules are implemented on a low-end Zynq 7010 FPGA (XC7Z010-1CLG400C).

3. Results

3.1. Accuracy

The accuracy achievable with our approach has been assessed using the dataset reported in [6], which comprises two intracortical recordings sampled using a 96-electrode Utah array during a delayed reach-to-grasp task. The used accuracy metric is the Pearson correlation between the output of the decoder and the dataset target. In Table 1 we compare with alternatives in literature. Our accuracy is comparable with the most accurate works while requiring 7.36 less parameters.

Table 1. Comparison in terms of accuracy and memory/computation requirements ([2] is the most accurate, [3] is the most accurate reporting on both N and L recordings).

	Decoder	Parameters	Accuracy N (CC)	Accuracy L (CC)
Ours	SNN	0.057M	0.83	0.78
[3]	QRNN	–	0.84	0.73
[2]	RNN	0.42M	0.91	–

Fig. 2 shows how the output of the decoder follows the variation of the target variable in the dataset.

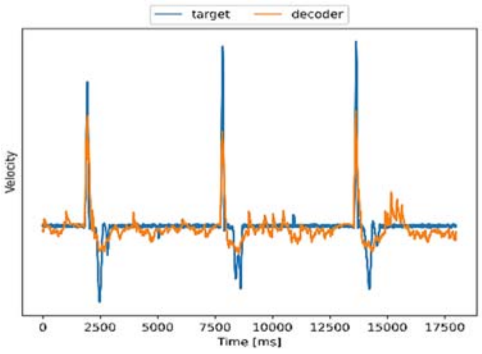


Fig. 2. Decoder output vs. Target variable.

3.2. Processing Performance and Sparsity

The design, when clocked at 2 MHz, can comfortably process the data in real time, ensuring a response in 1.8 ms on average, and in about 3 ms in the worst-case scenario. We also measured sparsity. The percentages of operation actually executed by the platform are respectively 12 % and 9 % for the two recordings in the dataset, thus the system is idle for around 90 % of the time, confirming the power efficiency of the proposed solution.

3.3. Hardware Utilization

The system, when deployed, uses only 32 BRAMs and 14 DSP slices, thus can be implemented using low-end devices to limit costs and power consumption. In order to save resources, the spike detection implementation is multiplierless, thus no DSP slices are used in the first processing stage. DSPs are used to accumulate synaptic weights during the synaptic current computation and for the integration in the Loihi neuron. Memory usage is limited by using 16-bit fixed point weight representation.

4. Conclusions

In conclusion, this study presents a spike decoding system implemented on a low-end FPGA, which achieves comparable results with state-of-the-art neural decoders using more complex deep learning models. The system requires significantly fewer parameters and reduces the total amount of computations exploiting sparsity. The low computational complexity and power efficiency of the system make it a suitable choice for low-power real-time neural decoding at the edge.

Acknowledgements

This work is funded by Regione Autonoma della Sardegna, program “P oC in Biomedicine” – POR-FESR 2014-2020 – Deep-ECGEE project

References

[1]. D. Sussillo, et al., A recurrent neural network for closed-loop intracortical brain–machine interface decoders, *Journal of Neural Engineering*, Vol. 9, Issue 2, 2012, 026027.

[2]. S.-H. Yang, et al., Selection of essential neural activity timesteps for intracortical brain-computer interface based on recurrent neural network, *Sensors*, Vol. 21, Issue 19, 2021, 6372.

[3]. N. Ahmadi, et al., Robust and accurate decoding of hand kinematics from entire spiking activity using deep learning, *Journal of Neural Engineering*, Vol. 18, Issue 2, 2021, 026011.

[4]. Utah Array, https://blackrockneurotech.com/

[5]. S. B. Shrestha, et al., SLAYER: Spike layer error reassignment in time, in Advances in Neural Information Processing Systems, Vol. 31, *Curran Associates, Inc.*, 2018, pp. 1419-1428.

[6]. T. Brochier, et al., Massively parallel recordings in macaque motor cortex during an instructed delayed reach-to-grasp task, *Scientific Data*, Vol. 5, Issue 1, 2018, pp. 1-23.

(075)

Spiking Neural Networks on Embedded Vector Processors

Felix Kreutz [1], **Daniel Scholz** [1], **Jiaxin Huang** [1], **Klaus Knobloch** [1] **and Christian Mayr** [2]
[1] Infineon Technologies Dresden GmbH & Co. KG, Königsbrücker Straße 180,
01099 Dresden, Germany
[2] Technische Universität Dresden, 01062 Dresden, Germany
E-mail: Felix.Kreutz@Infineon.com

Summary: Spiking Neural Networks (SNNs) are used in the domain of neuromorphic computing, they incorporate properties from the biological brain. The term spiking refers to an all-or-nothing activation of the neuron model that expresses as a binary state. The activation of neurons in biological neural networks only happens occasionally, such sparsity is one key concept in the domain of neuromorphic computing. With the rise in popularity of gradient-based optimization techniques for SNNs, the biologically inspired approach of SNNs moved closer to the methodologies of Artificial Neural Networks (ANNs). Such methodologies benefit from dedicated hardware that is optimized for the high throughput of densely packed data chunks. This work is providing an assessment of the potential benefit of sparse computations for an embedded vector processor. A main obstruction is the run time dependence that a sparse neuron activation introduces. The degree of sparsity in the network activations needed is quantified using a commercially available microcontroller.

Keywords: Spiking neural networks, Embedded computing, Single instruction multiple data.

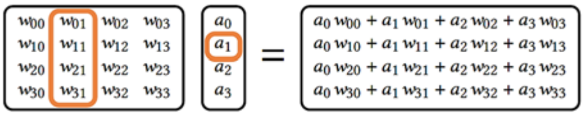
---

1. Introduction

The underlying work uses the methodologies described by the authors of [1]. Here a time discrete implementation of SNNs is used in combination with a binary activation function in a framework of recurrent neural networks. This makes the SNN comparable to other recurrent neural network structures like Long-Short-Term-Memory (LSTM) or Gated-Recurrent-Units (GRU). The introduction of the binary activation function prohibits gradient based optimization. This issue can be addressed by the definition of surrogate gradients as described by [2]. It is an established approach in neuromorphic computing to exploit the sparsity of spikes, where a spike is the activation of a neuron. Sparsity in this context means that a spiking neuron is creating spikes only occasionally. To the best of the authors knowledge, every dedicated neuromorphic hardware is making use of such activation sparsity in some way, for example by sparse message based communication between processing elements. Digital hardware following this scheme are TrueNorth [3], SpiNNaker2 [4] and Loihi [5]. The authors of [6] give the following summary "*In space, time, and connectivity, Loihi's architecture is optimized for sparse and nonbatched computation. [...] Memory access is always local and hypergranular, so data-dependent control flow is fast and efficient.*". This statement also holds for the other hardware examples mentioned. However, there are dedicated data-flow driven hardware designs requiring a data independent control flow to utilize their computational capabilities. Examples are the ARCv2 processor [7], the ARM cortex-m [8] or the RISC-V based Z-Scale [9]. All of which address the computation of embedded artificial neural networks. All are based on the Single- Instruction-Multiple-Data (SIMD) concept. This work takes an assessment of potential benefits of neural networks with sparse binary activation function like SNNs, in combination with hardware designed for embedded AI based on SIMD.

2. Data Dependent Computation

Let P_0 and P_1 be two neuron populations, in the scope of this work the connectivity between these two neuron populations is addressed. The activation of P_0 at any discrete point in time is given by the vector **a** and the input for every neuron in P_1 needs to be calculated based on the non-zero entries in **a** and a weight matrix $\mathbf{W}_{ij}$ using a vector matrix multiplication**.** Fig. 1 is viszualizing the operation, note that the values of **a** are binary, either 1 in case of a spike or 0 otherwise. Therefore the loading and computing of the column can be skipped dependend on the propability of an activation.



$$\begin{bmatrix} w_{00} & w_{01} & w_{02} & w_{03} \\ w_{10} & w_{11} & w_{12} & w_{13} \\ w_{20} & w_{21} & w_{22} & w_{23} \\ w_{30} & w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} a_0 \\ \boxed{a_1} \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} a_0 w_{00} + a_1 w_{01} + a_2 w_{02} + a_3 w_{03} \\ a_0 w_{10} + a_1 w_{11} + a_2 w_{12} + a_3 w_{13} \\ a_0 w_{20} + a_1 w_{21} + a_2 w_{22} + a_3 w_{23} \\ a_0 w_{30} + a_1 w_{31} + a_2 w_{32} + a_3 w_{33} \end{bmatrix}$$

Fig. 1. Vector-matrix multiplication, highlighting the dependence of a column to an entry of the input vector.

Three algorithms are implemented for this vector matrix multiplication, all are benchmarked on the ARCv2-processor. All vectors are located in the L1-cache and needed to be loaded to the vector registry to be processed by the ALUs. As reference algorithm the default approach is chosen, using the given multiply accumulate instruction of the processor and iterating over the input vector without introducing a dependency on the content of the vector. The second algorithm is using conditional branching based on the content of each element in the input vector. The third algorithm is operating on a compression of the input vector, and the iteration can be done only on the non-zero elements in the input vector. The vector compression instruction needs to be supported by the hardware, which is not the case for every SIMD-instruction set [10]. All three algorithms perform a vector matrix multiplication on a 64 by 64 weight matrix, therefore using an input vector of length 64. The probability for a non-zero activation in the input vector is changed between 0 and 1 while this probability is the same for each element in the input vector. Overall, 100 runs are performed per algorithm.

3. Results

The processing time for the vector matrix multiplication, as described previously, is given by Fig. 2.

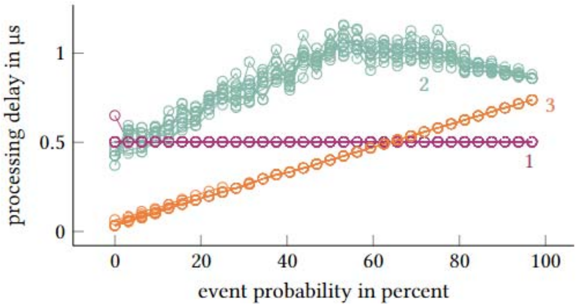


Fig. 2. Processing time over the probability for a non-zero entry in the input vector for the algorithms (1) default vector matrix multiplication, (2) data dependent conditional computation and (3) iteration over the compressed input vector. The benchmark was done on the Infineon TC4x-parallel processing unit with a 400 MHz clock.

The default data independent processing by algorithm (1) achieves to prove the complete weight matrix (4 Kbyte) in about 500 ns. The introduction of a conditional statement by algorithm 2 results in less memory access in the L1-cache dependent on the event/spike probability. However, the preloading needs to wait for the check of the vector element. The worst case (no successful branch prediction) can be found for a spike probability of 0.5. The proposed operating on a compressed vector with algorithm (3) makes the computation dependent on the spike probability. In this case a probability of 0.6 for a non-zero entry in the input vector achieves the highest data throughput.

4. Conclusions

This work shows that concepts of spiking neural networks like the sparse activation of neurons can also benefit SIMD architectures for embedded vector processing. The data dependent control flow of sparse computation heavily impacts the throughput of such systems. By using vector compression similar to the work of [10], the vector matrix multiplication can be accelerated by reducing the number of memory load accesses. The enabling factor for SNNs is the binary output space of spiking neurons in case of time discrete computations. The authors of [11] introduce a sparsity regularization for SNNs reducing the spike probability. Networks with an initial worst case spike probability of 0.5 are used. Therefore, this work shows that even SNNs not optimized for activation sparsity are benefitting from the proposed acceleration.

Acknowledgements

This work was funded by the German Federal Ministry of Education and Research (BMBF) within the KI-ASIC project (16ES0993). We thank Infineon Technologies AG for supporting this research.

References

[1]. G. Bellec, et al., A solution to the learning dilemma for recurrent networks of spiking neurons, *Nature Communications*, Vol. 11, Issue 1, 2020, 3625.

[2]. E. O. Neftci, et al., Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks, *IEEE Signal Processing Magazine*, Vol. 36, Issue 6, 2019, pp. 51-63.

[3]. F. Akopyan, et al., Truenorth: Design and tool flow of a 65 mw 1 million neuron programmable neurosynaptic chip, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 34, Issue 10, 2015, pp. 1537-1557.

[4]. C. Liu, et al., Memory-efficient deep learning on a SpiNNaker 2 prototype, *Frontiers in Neuroscience*, Vol. 12, 2018, 840.

[5]. M. Davies, et al., Loihi: A neuromorphic manycore processor with on-chip learning, *IEEE Micro*, Vol. 38, Issue 1, 2018, pp. 82-99.

[6]. M. Davies, et al., Advancing neuromorphic computing with Loihi: A survey of results and outlook, *Proceedings of the IEEE*, Vol. 109, Issue 5, 2021, pp. 911-934.

[7]. Synopsys, DesignWare ARC Processors, https://www.synopsys.com/designware-ip/processor-solutions/arc-processors.html) Accessed: 2022-11-02

[8]. L. Lai, et al., CMSIS-NN: Efficient neural network kernels for ARM Cortex-m CPUs, *arXiv Preprint*, 2018, arXiv:1801.06601.

[9]. Y. Lee, A. Ou, A. Magyar, Zscale: Tiny 32-bit RISC-V systems, in *Proceedings of the 2nd RISC-V Workshop*, 2015.

[10]. M. Kurtz, et al., Inducing and exploiting activation sparsity for fast inference on deep neural networks, in *Proceedings of the International Conference on Machine Learning (PMLR'20)*, 2020, pp. 5533-5543.

[11]. Y. Yan, et al., Backpropagation with sparsity regularization for spiking neural network learning, *Frontiers in Neuroscience*, Vol. 16, 2022.

(079)

Bayesian Deep Transfer Learning for Refinement of Stability Predictions in Milling

V. Ostad Ali Akbari, M. Kuffa and K. Wegener
Institute of Machine Tools and Manufacturing (IWF) – ETH Zurich,
Leonhardstrasse 21, 8092 Zurich, Switzerland
Tel.: +41 44 633 73 50, fax: +41 44 632 11 25
E-mail: oaavahid@iwf.mavt.ethz.ch

Summary: This paper presents a Bayesian deep transfer learning approach for updating stability predictions in milling. It uses pre-trained BDNNs based on existing physics-based stability models and a small experimental dataset to refine the network predictions for enhanced prediction accuracy versus experimentally observed stability states. The method requires fewer experimental samples and avoids extensive model parameter identification, making it suitable for industrial applications. Experimental validation shows acceptable prediction accuracy with quantification of chatter prediction probability.

Keywords: Chatter stability, Bayesian deep learning, Transfer learning, Receptance coupling, Physics-informed machine learning.

1. Introduction

Self-excited vibrations, also known as chatter, have been the main factor limiting the productivity of milling operations in manufacturing industries. Despite the extensive research focused on predicting unstable process conditions using physics-based models for chatter avoidance, the inadequate prediction accuracy of these models has hindered the development of a practical solution for industries. This flaw is mainly due to imprecise inputs fed into stability models. Stability Lobe Diagrams (SLDs) are commonly used to visualize the boundary between stable and unstable cuts based on spindle speed and depth of cut. The well-known Zero-order Approximation (ZOA) method, developed by Altintas and Budak [1], generates such SLDs using knowledge of structural dynamics at the tooltip and cutting forces.

Obtaining the required dynamics at the tooltip for generating SLDs is often computed through Receptance Coupling Substructure Analysis (RCSA) [2]. This approach couples experimentally measured machine tool dynamics with model-predicted dynamics of the tool-holder assembly, eliminating the complexities of modeling the entire machine tool structure as well as the excessive measurement efforts required for measuring tooltip dynamics for all machine-holder-tool combinations. However, precise estimation of tool-holder interface stiffness and damping is still necessary for modeling the tool-holder assembly, which are not known on a shop floor. Similarly, the estimation of cutting force coefficients, which relate the uncut chip thickness to the resulting forces, presents analogous challenges.

Recent attention has been given to data-driven methods to train a classifier on experimentally observed stability states. However, collecting a sufficient training dataset for a diverse range of machine and process configurations is still prohibitively expensive. An attempt in a pure machine learning approach can be found in a study by Cherukuri et al. [3].

Some researchers have combined physics knowledge and data-driven methods to refine unknown model parameters while reducing the required amount of experimental data. Postel et al. [4] pre-trained 200 Deep Neural Networks (DNNs) with simulated data and fine-tuned them with a smaller experimental data set to improve the prediction accuracy of DNN classifiers.

This paper follows a probabilistic perspective and employs Bayesian DNNs (BDNNs) for stability predictions. Following *Bayes by backprop* method, it allows for quantifying the uncertainty in the structure of a neural network while only doubling the number of tunable parameters, unlike the ensemble technique in [4], which required training of 200 DNNs. Furthermore, the probabilistic perspective can quantify the *epistemic uncertainty* due to lack of knowledge and consider that when generating stability maps. Section 2 discusses the physics-based modeling of milling stability to generate data for the pre-training of BDNNs. Section 3 further explains BDNNs and their training method. Section 4 presents validation cases and Section 5 concludes the paper.

2. Physics-based Stability Predictions

In order to obtain the required tooltip dynamics for ZOA through RCSA, the bending receptance of the machine tool at the spindle flange is measured once experimentally through the inverse RCSA approach by Namazi et al. [5]. The dynamics of tool-holder assembly are modeled using a finite-element implementation of Timoshenko beams as presented by Ostad Ali Akbari et al. [6].

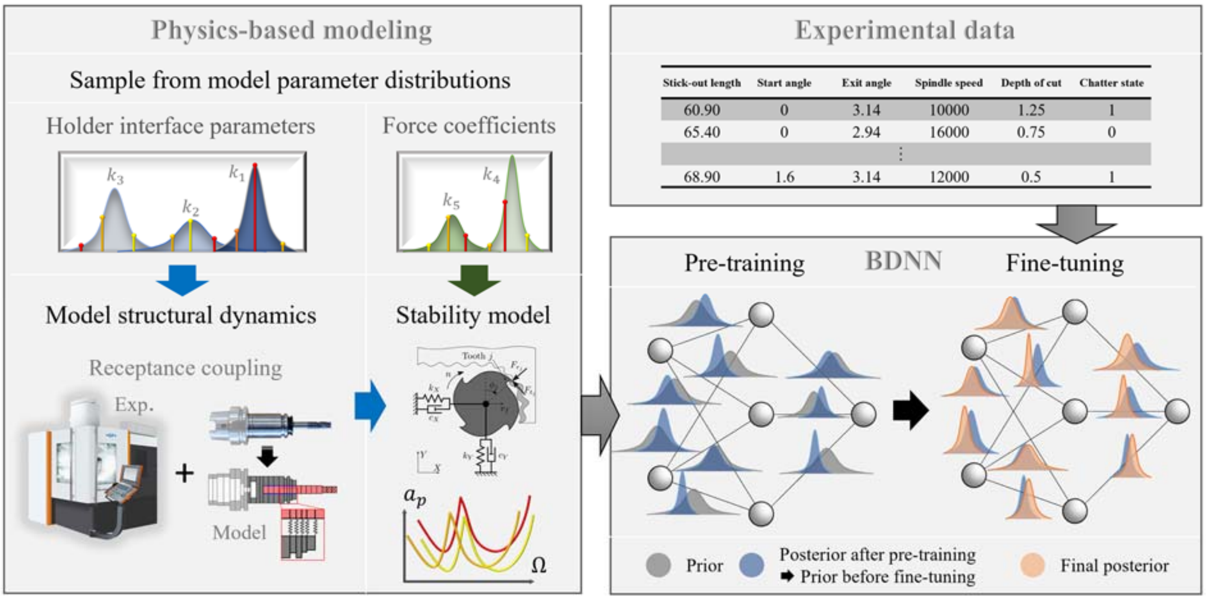


Fig. 1. Bayesian deep transfer learning strategy. Samples from physics-based mode parameters (Table 1) are used to generate multiple SLD samples. The stability information from these SLDs is used to pre-train a BDNN. The posterior distributions of network weights are considered as priors in the fine-tuning stage using experimental data.

Fig. 2 illustrates the coupling scenario and the tooltip receptance is computed using the following equation:

$$H_{tooltip} = H_{T1T1} - H_{T1T2}(H_{S1S1} + H_{T2T2})^{-1}H_{T2T1}, \quad (1)$$

where H_{ij} represents the receptance matrix between degrees of freedom i and j. The interface parameter values could be influenced by unaccounted factors such as the clamping pressure or dimensional tolerances, leading to a possible variability in these parameters. A similar argument applies to the tangential and radial cutting force coefficients as well. These coefficients are influenced by various factors such as the material properties of the workpiece, coating, wear state, cutting-edge geometry, and feed per tooth, resulting in variations around their mean values.

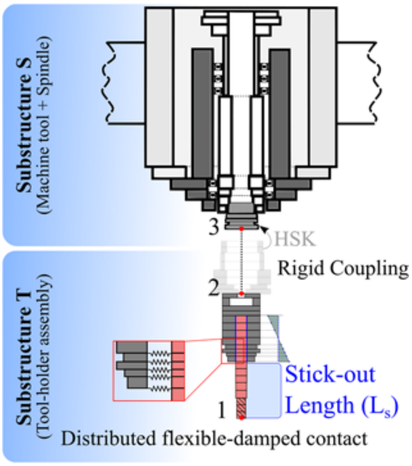


Fig. 2. Substructuring scheme for computation of tooltip dynamics through RCA.

Table 1. Parameters for physics-based stability predictions and their corresponding expected distributions.

Description	Unit	Expression	Distribution
Translational stiffness	$\left[\frac{N/m}{mm}\right]$	$k_{tt} = 10^{k_1}$	$k_1: \mu = 8.8, \sigma = 1.0$
Translational damping	$\left[\frac{1}{mm}\right]$	$\eta_{tt} = 10^{-k_2}$	$k_2: \mu = 4.0, \sigma = 1.0$
Rotational stiffness	$\left[\frac{Nm}{mm}\right]$	$k_{rr} = 10^{k_1 k_3}$	$k_3: \mu = 0.3, \sigma = 0.1$
Rotational damping	$\left[\frac{1}{mm}\right]$	$\eta_{rr} = 10^{k_1}k_3$	
Tangential force coefficient	$[MPa]$	$K_{tc} = 100k_4$	$k_4: \mu = 6.0, \sigma = 3.0$
Radial force coefficient	$[MPa]$	$K_{rc} = k_5 K_{tc}$	$k_5: \mu = 0.3, \sigma = 0.1$

A BDNN will be trained for each scenario that is defined by a combination of:

- Machine tool;
- Holder;
- Tool;
- Workpiece material.

The following information determines a process condition whose stability state is to be validated through either the physics-based or BDNN model, also will be considered as input vector x to BDNN in Fig. 3:

- Tool stick-out length L_s;
- Immersion angles: start $\emptyset_{st}$ and exit $\emptyset_{ex}$ angles for computation of *directional dynamic milling force coefficients* in ZOA solution;
- Spindle speed Ω and axial depth of cut a_p.

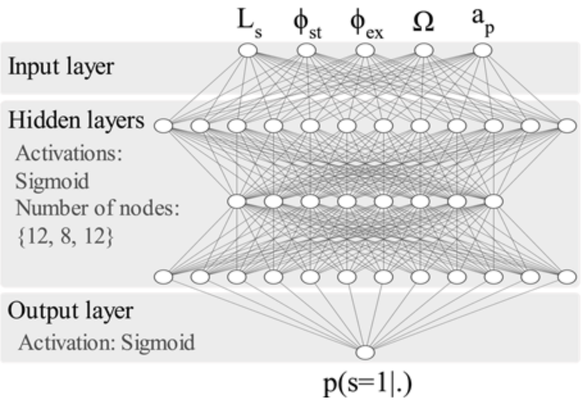


Fig. 3. Structure of the BDNN, mapping the process parameters into chatter state probability.

3. Bayesian Deep Learning

The primary difference between standard deterministic neural networks and Bayesian neural networks lies in their perspective in defining the network parameters. While standard DNNs use deterministic values for weights, Bayesian neural networks employ a distribution that expresses the probability of weight values.

By defining parameter distributions, BDNNs are capable of incorporating epistemic uncertainty, when there is limited training data and incomplete knowledge about the underlying phenomena. As more training data becomes available, this uncertainty can be reduced. During Bayesian learning, the distribution of model parameters w is updated from a prior $p(w)$ belief to a posterior $p(w|\mathcal{D})$ distribution in accordance with the training data $\mathcal{D}$, using the Bayes rule:

$$p(w \mid \mathcal{D}) = \frac{1}{Z} p(w) p(\mathcal{D} \mid w) \tag{2}$$

Computing the closed-form solution of the posterior distribution from the Bayes rule is often impossible due to the intractable computation of the scaling factor Z. Hence, approximation techniques such as Markov Chain Monte Carlo (MCMC) or variational inference are used to approximate the posterior of network weights. In this study, *Bayes by Backprop* approach, explained in [7], optimizes the means and standard deviations of network weights directly and is effective as it only doubles the number of optimizing parameters. In BDNNs, the distributions of weights (including biases) are approximated by a family of independent Gaussians $q(w|\lambda) \cong \mathcal{N}(w; \mu, \Sigma)$ where μ denotes the vector of mean values and Σ is the covariance matrix $\Sigma = diag(\log(1 + \exp(\rho)))$ with only positive diagonal terms. The optimization tunes the parameters of q, denoted by $\lambda = \{\mu, \rho\}$, so that the deviation between the approximated posterior $q(w|\lambda)$ and true posterior $p(w|\mathcal{D})$ is minimized:

$$\mathcal{L}(\lambda) = KL[q(w \mid \lambda) \,||\, p(w \mid \mathcal{D})] \tag{3}$$

The Kullback-Leibler (KL) divergence quantifies the difference between the two distributions and can be written by substituting the Bayes rule:

$$\mathcal{L}(\lambda) = \int q(w \mid \lambda) \log \frac{q(w \mid \lambda)}{p(w) p(\mathcal{D} \mid w)} dw \tag{4}$$

By rearranging the terms in the following way, it demonstrates the Bayesian learning objective that aims for maximizing the expected data likelihood (expectation is indicated by $\mathbb{E}$) and minimizing deviations from the prior, also called complexity cost:

$$\begin{aligned} \mathcal{L}(\lambda) = {} & KL[q(w \mid \lambda) \,||\, p(w)] \\ & - \mathrm{E}_{w \sim q(w|\lambda)}[\log p(\mathcal{D} \mid w)] \end{aligned} \tag{5}$$

The training is performed by stochastic gradient descent which requires the gradient of the loss $\mathcal{L}(\lambda)$:

$$\nabla_{\lambda} \mathcal{L}(\lambda) = \nabla_{\lambda} \mathrm{E}_{w \sim q(w|\lambda)}[\mathcal{F}(w, \mathcal{D})] \tag{6}$$

Considering the deterministic function $\mathcal{F}(w, \mathcal{D})$:

$$\begin{aligned} \mathcal{F}(w, \mathcal{D}) = {} & \log q(w \mid \lambda) - \\ & - \log p(w) - \log p(\mathcal{D} \mid w) \end{aligned} \tag{7}$$

A direct computation of the gradient for an expectation is prohibitive. A trick known as reparameterization can be used in such cases where the scholastic part is separated from the optimizing parameters:

$$w = t(\lambda, \varepsilon) = \mu + \log(1 + \exp(\rho)) \circ \varepsilon, \tag{8}$$

where $\varepsilon \sim \mathcal{N}(0, I)$ and $\circ$ stands for point-wise multiplication. As a result, the gradient of expectation becomes equal to the expectation of gradients:

$$\nabla_{\lambda} \mathcal{L}(\lambda) = \mathrm{E}_{\varepsilon \sim \mathcal{N}(0,I)}[\nabla_{\lambda} \mathcal{F}(t(\lambda, \varepsilon), \mathcal{D})] \tag{9}$$

This expectation is approximated using Monte Carlo:

$$\nabla_{\lambda}\mathcal{L}(\lambda) = \frac{1}{m}\sum_{j=1}^{m}\nabla_{\lambda}\mathcal{F}(t(\lambda,\varepsilon^{(j)}),\mathcal{D}) \qquad (10)$$

After the network is trained, the inference of the process stability state can be obtained through a marginal probability of the chatter state over all possible values of the network weights:

$$\begin{aligned} p(s^{*}=1 \mid x^{*},\mathcal{D}) &= \int p(s^{*}=1 \mid x^{*},w)\,p(w \mid \mathcal{D})\,dw \\ &= \mathrm{E}_{w\sim q(w\mid\mathcal{D})}[p(s^{*}=1 \mid x^{*},w)] \\ &\approx \frac{1}{m}\sum_{j=1}^{m} p(s^{*}=1 \mid x^{*},w^{(j)}), \end{aligned} \qquad (11)$$

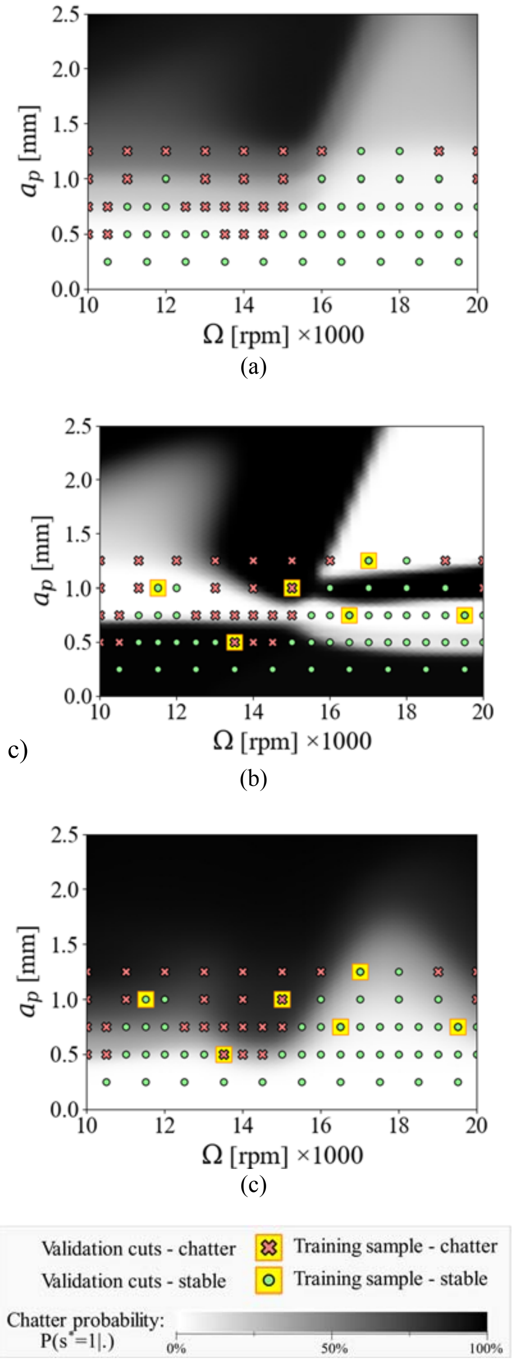
where the expectation is approximated by Monte Carlo, averaging chatter probabilities, that can be assessed over a grid of spindle speed-depth of cut to generate the full probabilistic stability maps that are shown in Section 4.

4. Experimental Verifications

This section investigates the effectiveness of the proposed approach versus experimental validation cuts. The validation cuts are performed on a GF Mikron HPM800U 5-axis milling machine with a holder type RegoFix PG25x100H and a 4-flute flat end mill of Voha 032456120 (D = 12 mm, L = 110 mm). The feed per tooth is set to 0.05 mm/tooth. The workpiece material is AL6082. Fig. 4 compares the prediction of model versus validation cuts in different training cases.

5. Conclusions

This paper presented a Bayesian deep transfer learning method for refining stability predictions in milling operations. By utilizing pre-trained BDNNs based on existing physics-based stability models and a small experimental dataset, the proposed approach significantly improved the prediction accuracy for stability states compared to experimentally observed states. The transfer learning technique enables leveraging the knowledge of machine tool structural dynamics and stability analysis from physics, which reduced the required amount of training data. Moreover, the fine-tuning step further adjusted the distributions of network parameters to correct the inaccuracies of the physics-based model based on limited experimental data, without the need for extensive model parameter identification, making it a viable option for industrial applications. The experimental validation demonstrates the effectiveness of the proposed method, providing acceptable prediction accuracy and quantification of chatter prediction probability. As a future work, the input features of the BDNN can be expanded to broaden the applicability of a single BDNN.



(a)

(b)

(c)

Fig. 4. Effectiveness of transfer learning for stability predictions using BDNN. a) predictions of a pre-trained model without fine tuning b) only fine-tuning step without pre-training c) proposed approach to pre-train on simulated data and fine-tune on a small set of experimental data.

Acknowledgments

The authors express their gratitude to Innosuisse, the Swiss Innovation Agency, for the financial support received for this work (Grant 32334.1 IP-ICT).

References

[1]. Y. Altintaş, E. Budak, Analytical prediction of stability lobes in milling, *CIRP Annals*, Vol. 44, Issue 1, 1995, pp. 357-362.

[2]. G. Duncan, M. Tummond, T. Schmitz, An investigation of the dynamic absorber effect in high-speed machining, *Int. J. Mach. Tools Manuf.*, Vol. 45, Issues 4-5, 2005, pp. 497-507.

[3]. H. Cherukuri, E. Perez-Bernabeu, M. Selles, T. Schmitz, Machining chatter prediction using a data learning model, *Journal of Manufacturing and Materials Processing,* Vol. 3, Issue 2, 2019, 45.

[4]. M. Postel, B. Bugdayci, K. Wegener, Ensemble transfer learning for refining stability predictions in milling using experimental stability states, *The International Journal of Advanced Manufacturing Technology,* Vol. 107, 2020, pp. 4123-4139.

[5]. M. Namazi, Y. Altintas, T. Abe, N. Rajapakse, Modeling and identification of tool holder-spindle interface dynamics, *International Journal of Machine Tools and Manufacture*, Vol. 47, 2007, pp. 1333-1341.

[6]. V. Ostad Ali Akbari, M. Postel, M. Kuffa, K. Wegener, Improving stability predictions in milling by incorporation of toolholder sound emissions, *CIRP J. Manuf. Sci. Technol.*, Vol. 37, 2022, pp. 359-369.

[7]. C. Blundell, J. Cornebise, K. Kavukcuoglu, D. Wierstra, Weight uncertainty in neural network, in *Proceedings of the International Conference on Machine Learning (ICML'15)*, 2015, pp. 1613-1622.

(080)

Revealing Patterns of Symptomatology in Parkinson’s Disease: A Latent Space Analysis with 3D Convolutional Autoencoders

E. Delgado de las Heras [1], **F. J. Martinez-Murcia** [1], **I. A. Illán** [1], **C. Jiménez-Mesa** [1],
D. Castillo-Barnes [2], **J. Ramírez** [1] **and J. M. Górriz** [1]
[1] Dept. of Signal Theory, Networking and Communications, University of Granada,
C/ Periodista Daniel Saucedo Aranda S/N 18071 Granada, Spain
[2] Dept. of Communications Engineering, University of Malaga, C/ Boulevard Louis Pasteur, 35,
29071 Malaga, Spain
Tel.: + 34 958241717
E-mail: fjesusmartinez@ugr.es

Summary: This work proposes the use of 3D convolutional variational autoencoders (CVAEs) to trace the changes and symptomatology produced by neurodegeneration in Parkinson’s disease (PD). In this work, we present a novel approach to detect and quantify changes in dopamine transporter (DaT) concentration and its spatial patterns using 3D CVAEs on Ioflupane (FPCIT) imaging. Our approach leverages the power of deep learning to learn a low-dimensional representation of the brain imaging data, which then is linked to different symptom categories using regression algorithms. We demonstrate the effectiveness of our approach on a dataset of PD patients and healthy controls, and show that general symptomatology (UPDRS) is linked to a d-dimensional decomposition via the CVAE with R2>0.25. Our work shows the potential of representation learning not only in early diagnosis but in understanding neurodegeneration processes and symptomatology.

Keywords: Neurodegeneration, Variational autoencoder, Latent space, Symptomatology, Regression, Longitudinal modeling.

---

1. Introduction

Parkinson’s disease (PD) is a progressive neurodegenerative disorder that affects more than 6.2 million people worldwide [1]. PD is characterized by a loss of dopamine-producing neurons in the brain, causing tremors, stiffness, and cognitive decline among other symptoms. FPCIT (ioflupane) SPECT is the most widely extended neuroimaging technique for the diagnosis of PD. FPCIT binds to the presynaptic dopamine transporters (DaTs), allowing to visualize and quantify the DaT concentration at the striata, that characterizes PD [2]. This modality may allow to tackle one of the biggest challenges for PD: the early detection and monitoring of disease progression [3].

Recent advances in deep learning, and in particular, convolutional neural networks (CNNs) have shown promising results in medical imaging applications such as segmentation, registration, and classification [4]. Autoencoders (AEs), a type of self-supervised neural network, can learn a low-dimensional representation of high-dimensional data, making them well-suited for image compression, denoising, and anomaly detection [5].

In this paper, we propose a novel approach to detect and quantify subtle changes in DaT concentration and distribution in the brain using 3D convolutional variational AEs (CVAEs). Our approach leverages the power of deep learning to learn a low-dimensional representation of brain imaging data, which enables us to longitudinally compare images and identify patterns of change that are indicative of neurodegeneration. We demonstrate the effectiveness of our approach on a dataset of PD patients and healthy controls, and show that latent spaces can capture the variability of individual symptom categories, as well as the overall disease stage.

2. Materials and Methods

2.1. Dataset

Data for this study were obtained from the Parkinson’s Progression Markers Initiative (PPMI) database. For the most recent information on the study, please visit www.ppmi-info.org. We used the standardized “Original Cohort BL to Year 5” that includes individuals initially diagnosed as either controls (CTL) or PD affected subjects with varying levels of severity. After removing subjects with no FPCIT scans, the CTL group comprises 101 males and 53 females, while the PD group comprises 284 males and 159 females, followed for up to 5 years, with a total 1399 sessions of available data for studying the progression of imaging biomarkers and their link to PD-specific progression indicators. Symptomatology is assessed via the MDS-UPDRS scale [6], either in its aggregated form (UPDRS -total-) or its 4 parts: 1) non-motor aspects of daily living; 2) motor aspects of daily living (tremor, walking, etc.); 3) motor examination (rigidity, posture, gait etc.) and 4) motor complications (dyskinesia, fluctuation items).

FPCIT scans labeled as “Reconstructed” were used, having all of them a consistent orientation and similar sizes (for a scan of shape). However, no spatial normalization is performed, as it was demonstrated to have small impact when using CNN architectures [4]. Intensity was normalized by subtracting the background average intensity, and then dividing all voxel intensities by a non-specific reference, defined as the average voxel intensity of the cerebellum and occipital lobe. Finally, to favor convergence of the CVAE training, the upper-end values of the intensity distribution were compressed using a sigmoid function.

2.2. 3D Convolutional Variational Autoencoders

An autoencoder is a neural network architecture that consists of an encoder network and a decoder network. In this work we used a 3D convolutional encoder-decoder architecture, capable of dealing with volumetric 3D images. The 3D encoder can capture spatial features at multiple scales, similarly to well-known 3D convolutional NNs [4]. Its output is a lower-dimensional representation which encodes the most salient features of the input data. Under the manifold hypothesis [5], this representation is often considered as coordinates in a "latent space" underlying the dataset. The decoder network performs the inverse operation.

Variational autoencoders consider that the latent space is indeed the parameter space of D Gaussian distributions, from which the input of the decoder is sampled [7], allowing for non-sparse sampling and a quantification of uncertainty in the point estimates. The β-VAE loss function [8] was used (Eq. (1)):

$$L_{VAE} = L_{recon} + \beta L_{KLD} \qquad (1)$$

This loss function allows controlling for the proportion of the reconstruction error and the Kullback-Leibler Divergence (KLD) (Eq. (3)) using the parameter $\beta = 1$. Reconstruction error was estimated assuming that the voxel distribution among patients was Gaussian, which means that we could use the volumetric Mean Squared Error (MSE) (Eq. (2)) between the original voxel intensities x_j and the output voxel of the decoder $\hat{x}_j$ for all j voxels in the images. Each subject's loss was added to conform the batch loss (reduction "sum"), which proved to improve training convergence.

$$L_{recon} = \sum_j \left(x_j^2 - \hat{x}_j^2\right), \qquad (2)$$

$$L_{KLD} = -\frac{1}{2}\sum_{i=0}^{D}\left(1 + log\sigma_i^2 - \mu_i^2 - \sigma_i^2\right) \qquad (3)$$

The specific architecture and parameters of the 3D-CVAE is shown at Table 1 and Table 2, including information about Kernel Size (KS), Stride (ST) and Zero Padding (ZP). Different latent dimensionality (3, 8 and 20) was tested for 400 epochs with Adam optimizer (lr = 1e-3).

2.3. Regression

The purpose of this work is to predict 4 categories of symptomatology using the latent distribution of the FPCIT image dataset. To perform such task, we explore the predictive power of two widely-used regression algorithms: Decision Trees (DT) and XGBoost. DTs are relatively simple to understand and have been widely used in medicine, because the resulting trees are directly interpretable [9]. XGboost is a scalable tree boosting system that uses approximate tree learning, and is widely extended in data science, achieving high performance in many challenges [10].

Table 1. Encoder architecture parameters.

Type	# in	#out	KS	ST	ZP
Conv3D (1)	1	32	3	2	1
Conv3D (2)	32	64	3	2	1
Conv3D (3)	64	128	3	2	1
Conv3D (4)	128	256	3	2	1
Linear	73728	512	–	–	–
2 × Linear (μ and σ)	512	D	–	–	–

Table 2. Decoder architecture parameters.

Type	# in	#out	KS	ST	ZP
Linear	D	73728	–	–	–
ConvTranspose3d (1)	256	128	3	2	1
ConvTranspose3d (2)	128	64	3	2	1
ConvTranspose3d (3)	64	32	3	2	1
ConvTranspose3d (4)	32	1	3	2	1

2.4. Experimental Setup

The experimental setup is defined by first training the dataset and using the parameters of the latent distribution as features. Additionally, K-Means features (KMF), which capture the distance to a K-means cluster center, were generated to capture the salient characteristics of the data. The number of clusters used to generate KMFs is established as the maximum value between 8 and the logarithm of the dimensionality times 8, as in [11]. Then, the two regression algorithms are trained with or without these KMFs to predict the different UPDRS categories. 10-Fold cross-validation is used to compute the performance values, including the Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and coefficient of determination between the real and predicted values for UPDRS.

To more fully understand the output of the pipeline, we applied SHapley Additive exPlanations (SHAP) [12] to the outputs of each system. SHAP yields metrics such as feature importance, and the contribution of each feature to the output of the algorithm, and thus can be used to track the most relevant characteristics on the data.

3. Results and Discussion

3.1. Regression Results

The performance of the proposed regression algorithms on the 3, 8 and 20-dimensional latent space of the CVAE is shown at Fig. 1. We obtained R2 > 0 for almost all target symptomatology categories and d-dimensional latent spaces. We can observe that, while low latent dimensionality (3, 8) is better for individual symptom categories, the details of the overall symptomatology score (UPDRS-total) is better accounted for by the 20-dimensional (20-D) CVAE, achieving a MAE = 12.21 and a R2 = 0.26. This speaks of the complexity of the composite, but also of its relationship to the FPCIT distribution patterns. Similarly, UPDRS 3, more dependent on motor symptoms, benefits from the 20-D space.

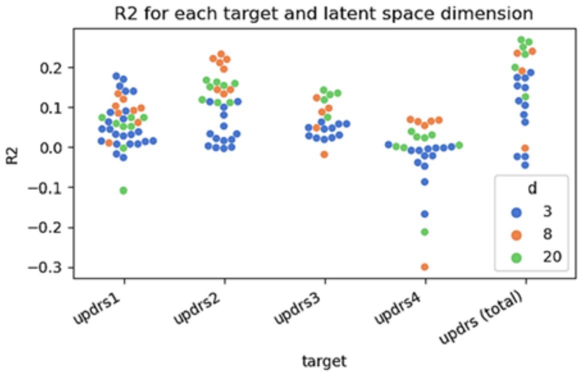


Fig. 1. R2 performance for the models trained with the five symptomatology scales and 3, 8 and 20 latent-dimension.

A detailed view of the best results obtained for each combination of target scale and dimensionality of the latent space *D*, including their MAE, RMSE and values is shown at Table 3.

Table 3. Encoder architecture parameters.

Target	D	Model	MAE	RMSE	R2
updrs1	3	DT	3.53	4.5	0.15
	8	DT (KMF)	3.58	4.55	0.13
	20	DT (KMF)	3.69	4.71	0.07
updrs2	3	XGB (KMF)	3.96	5.23	0.11
	8	XGB (KMF)	3.68	4.86	0.23
	20	XGB (KMF)	3.74	5.08	0.16
updrs3	3	XGB (KMF)	10.78	13.11	0.06
	8	XGB (KMF)	10.21	12.93	0.09
	20	XGB (KMF)	10.03	12.59	0.14
updrs4	3	DT	0.7	1.44	0
	8	XGB	0.62	1.39	0.07
	20	DT (KMF)	0.71	1.43	0.02
updrs (total)	3	XGB (KMF)	13.15	16.42	0.17
	8	XGB	12.72	15.75	0.24
	20	XGB	12.21	15.51	0.26

3.2. Interpretability of the Space

The regression results show that there exist a predictive power in the manifold representation of the FPCIT dataset. The interpretation of the effective patterns captured by latent variables can be, in consequence, of key importance for clinical validation. While DTs (best for UPDRS 1) are visual by nature, ensemble methods such as XGBoost are more difficult to interpret. Here, SHAP can pave the way to interpret the contribution of each of the latent features to the output variables.

In this regard, the prediction of total symptomatology (UPDRS – total) is of special interest. In both the 8- and 20-dimensional spaces XGBoost achieves the largest predictive ability in terms of. Consequently, the most important features -measured by SHAP- can capture the patterns that lead to this high performance. The SHAP analysis of the XGBoost regressor with the 20-dimensional space reveals the behavior shown at Fig. 2.

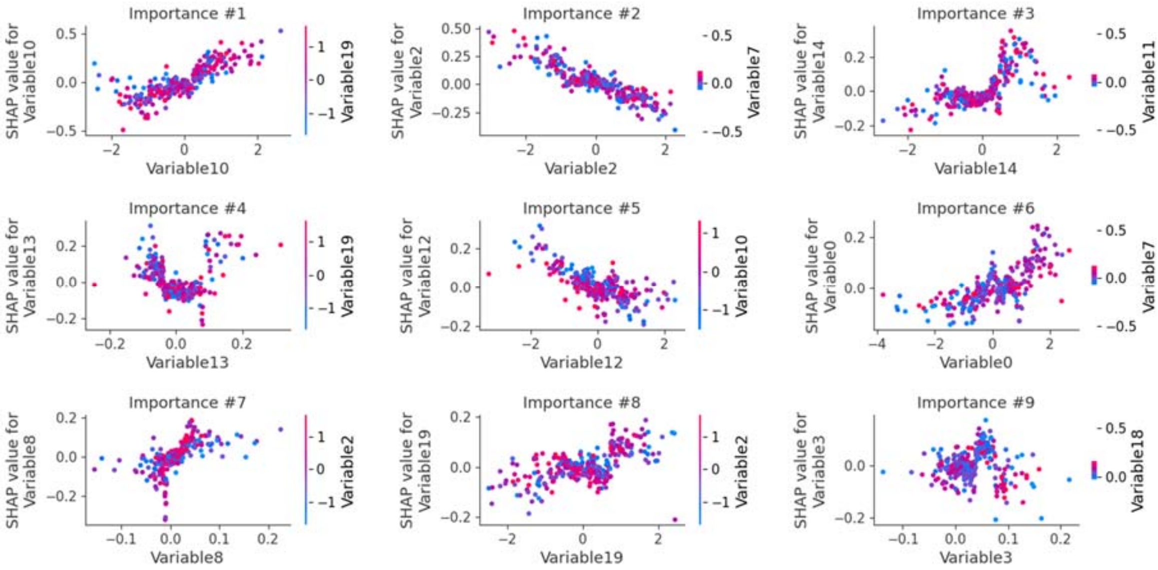


Fig. 2. SHAP dependence plot of the XGBoost regressor with the 20-dimensional space.

SHAP dependence plots show how the SHAP value of importance depends on a given feature, ordered from highest to lowest importance. In Fig. 2 we see that the top-3 features that contribute to the output of the algorithm are variables 2, 10 and 0. Color show the values of a second feature that may have an interaction effect with the feature we are plotting. Remember that these are the mean parameters of a Gaussian distribution that encodes the pattern for each subject. In this regard, we see an almost linear dependency between importance to the algorithm output and values of the variables, which indicates that there exist a clear relationship between them. To visualize the patterns captured by features 2 and 10, we generated brain images sampling these two features with the decoder and setting 0-mean to all other features. The result is shown at Fig. 3.

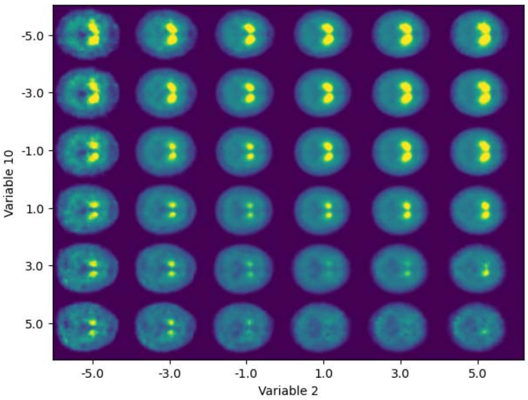


Fig. 3. Latent manifold for variables 10 and 2 of the 20-dimensional latent space of the 3D CVAE.

We observe that the composition of the two variables account for two relevant characteristics of FPCIT: the overall intensity of the striata, the separation between them and the ratio between uptake at the anterior and posterior parts of the striata. General intensity and separation patterns seem to be encoded by our variable 10, whereas the striata anterior-posterior length is generally encoded by variable 2. In this regard, the overall intensity (uptake) at the striata is indeed the objective of FPCIT as a biomarker, and therefore expected. As for the anterior-posterior striata patterns, many works that used FDOPA, a presynaptic dopaminergic marker, reported a distinct anterior–posterior gradient of uptake as PD progresses [13]. This could be then a relevant marker for assessing the progression of the disease. Since SHAP importance is higher for this variable 2, we can assume that its contribution to the modeling of the progression is higher, perhaps showing a smoother transition as the symptoms progress, in contrast to overall drug uptake, more related to late neurodegeneration.

4. Conclusions

We hypothesize that the non-linear self-supervised latent space of a 3D Convolutional Variational Autoencoder (CVAE) is linked to the symptoms shown by affected subjects. The latent features of a trained CVAEs were related to different aspects of the MDS-UPDRS scale with R2 > 0.20, proving a link between FPCIT spatial patterns and Parkinson’s Disease symptomatology. The most relevant variables for the predictive algorithms reveal that the CVAE captures patterns related to overall intensity, striata separation and differences between anterior-posterior parts of the striata, and that this last feature is more relevant for assessing the progression of symptoms than the former. This comprehensive approach may help to better understand the complex relationship between clinical disease and imaging biomarkers, beyond the obvious relationship between the average binding-ratio and the disease.

Acknowledgements

This work was partly supported by the MINECO/FEDER under the RTI2018-098913-B-I00 projects, and in by the Consejería de Economía, Innovación, Ciencia y Empleo (Junta de Andalucía) and FEDER under the P20-00525 project and PPJIA2021-17. Work by F.J.M.M. was supported by the MICINN “Ramón y Cajal” RYC2021-030875-I. Work by C.J.M. is supported by Ministerio de Universidades under the FPU18/04902 grant.

References

[1]. V. L. Feigin, A. A. Abajobir, K. H. Abate, et al., Global, regional, and national burden of neurological disorders during 1990-2015: A systematic analysis for the global burden of disease study 2015, *The Lancet Neurology*, Vol. 16, Issue 11, 2017, pp. 877-897.

[2]. F. Segovia, J. M. Górriz, J. Ramírez, F. J. Martínez-Murcia, D. Salas-Gonzalez, Preprocessing of 18F-DMFP-PET data based on hidden Markov random fields and the gaussian distribution, *Frontiers in Aging Neuroscience*, Vol. 9, 2017, 326.

[3]. E. Tolosa, A. Garrido, S. W. Scholz, W. Poewe, Challenges in the diagnosis of Parkinson’s disease, *The Lancet Neurology*, Vol. 20, Issue 5, 2021, pp. 385-397.

[4]. F. J. Martinez-Murcia, J. M. Górriz, J. Ramírez, A. Ortiz, Convolutional neural networks for neuroimaging in Parkinson’s disease: Is preprocessing needed?, *International Journal of Neural Systems*, Vol. 28, Issue 10, 2018, 1850035.

[5]. F. J. Martinez-Murcia, A. Ortiz, J.-M. Gorriz, J. Ramirez, D. Castillo-Barnes, Studying the manifold structure of Alzheimer’s disease: A deep learning approach using convolutional autoencoders, *IEEE Journal of Biomedical and Health Informatics*, Vol. 24, Issue 1, 2020, pp. 17-26.

[6]. C. G. Goetz, B. C. Tilley, S. R. Shaftman, et al., Movement disorder society-sponsored revision of the unified Parkinson’s disease rating scale (MDS-UPDRS): Scale presentation and clinimetric testing results, *Movement Disorders: Official Journal of the Movement Disorder Society*, Vol. 23, Issue 15, 2008, pp. 2129-2170.

[7]. D. P. Kingma, M. Welling, Auto-encoding variational bayes, *arXiv Preprint*, 2013, arXiv:1312.6114.

[8]. I. Higgins, L. Matthey, A. Pal, et al., Beta-VAE: Learning basic visual concepts with a constrained variational framework, in *Proceedings of the International Conference on Learning Representations (ICLR'17)*, 2017.

[9]. J. E. Arco, A. Ortiz, J. Ramírez, F. J. Martínez-Murcia, Y.-D. Zhang, J. M. Górriz, Uncertainty-driven ensembles of multi-scale deep architectures for image classification, *Information Fusion*, Vol. 89, 2023, pp. 53-65.

[10]. T. Chen, C. Guestrin, XGboost: A scalable tree boosting system, in *Proceedings of the* 22nd *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785-794.

[11]. C. Boutsidis, A. Zouzias, P. Drineas, Random projections for K-means clustering, *Advances in Neural Information Processing Systems*, Vol. 23, 2010, pp. 1-9.

[12]. S. M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in Advances in Neural Information Processing Systems (I. Guyon, U. V. Luxburg, S. Bengio, et al., Eds.), *Curran Associates Inc.*, 2017.

[13]. V. Dhawan, Y. Ma, V. Pillai, et al., Comparative analysis of striatal FDOPA uptake in Parkinson's disease: Ratio method versus graphical approach, *Journal of Nuclear Medicine*, Vol. 43, Issue 10, 2002, pp. 1324-1330.

(081)

Sensors Fusion using RSSI and IMU Data for IoT Devices Indoor Localization

Wafa Njima, Xun Zhang, Hongxiu Zhao and Masood Jan
ISEP, Institut Supérieur d'Electronique de Paris, 75006 Paris, France
E-mails: wafa.njima@isep.fr, xun.zhang@isep.fr, hongxiu.zhao@ext.isep.fr, masood.jan@ext.isep.fr

Summary: The EU's 6G projects anticipate billions of IoT devices by 2030, requiring high-resolution radio-light localization services. To improve indoor localization accuracy, researchers propose data fusion using sensor, technology, and signal parameter fusion. Machine Learning techniques have been widely used recently in the field of localization to guarantee real-time location estimation even when dealing with large sensor networks since such techniques allow to build the indoor localization models optimized offline and just applied online. In this paper, we present two data fusion schemes using deep neural networks to improve indoor localization accuracy by combining received signal strength indicator (RSSI) and inertial measurement unit (IMU) data. Both proposed schemes have been tested based on the publicly available database IMUWiFine. Obtained results show that the integration of IMU data to refine the estimated coordinates does not necessarily improve the localization accuracy obtained when applying the conventional localization scheme using only RSSI data. Consequently, future work will determine which IMU parameter to combine with RSSI, depending on the available measurements and the considered indoor area, to improve the localization performance.

Keywords: Deep neural network (DNN), Fingerprinting method, Indoor localization, Inertial Measurement Unit (IMU), Received Signal Strength Indicator (RSSI).

1. Introduction

The European Union (EU) has initiated the first group of 6G projects in 2021. With billions of connected Internet of Things (IoT) devices expected in the future 6G network, high-resolution radio-light localization services will be necessary. By 2025, the number of connected devices is projected to reach 35 billion, growing to 50 billion by 2030, with more than half of all connected devices being machine-to-machine (M2M) devices from 2023 [1]. The 2030s will be remembered as the age of societal automation, with accurate localization being a crucial component for the billions of IoT devices, along with secure storage and management of the localization data for application access [2][3]. Indoor localization, exploring signal parameters (Received Signal Strength Indicator (RSSI), Time of Arrival (ToA), Angle of Arrival (AoA), Time Difference of Arrival (TDoA) and Channel State Information (CSI)) received from different technologies (WiFi, Bluetooth, Visual Light Communication (VLC), mmWave, Ultra WideBand (UWB), etc.) collected by several users moving in a considered area, is conducted mainly by proximity-based methods, geometric-based methods (i.e., multilateration, triangulation, etc.) [4] or fingerprints-based methods [5]. However, such classical methods (i) are heavily affected by harsh indoor propagation conditions (geometric methods) which degrades significantly the localization accuracy, or (ii) do not guarantee real-time location estimation since large fingerprints databases are scanned online for similarity evaluation conducted during the localization process (fingerprinting methods).

In order to guarantee online real-time indoor localization even in large sensor networks, machine learning (ML) techniques especially deep learning (DL) techniques have been widely used in recent years achieving promising performance which contributes to an improvement of the quality of offered services and applications [1]. Using such methods based on offline-collected fingerprint databases, a localization model can be built and optimized iteratively offline. This model is then applied online, taking data collected by the target to be localized as input and providing an estimated location as output. Many works have been proposed based on these methods using different types of DL methods mainly convolutional neural networks (CNNs) [6] and deep neural networks (DNNs) [7]. The CNNs which are a specific class of neural networks (NNs) have shown promising performance in the field of images and text classification. Consequently, authors in [8] formulate the problem of localization as 3D radio-images classification predicting the zone containing the target. Such a method achieves good localization accuracy; however, it is more complex compared to DNN-based methods [9] and the predictions of targets' positions still need refinements since the output of such systems is a zone and not specific coordinates. Thus, DNNs have been considered in many works as the privileged DL solution applied for localization aims integrating also a denoising process of user data collected during the site survey step conducted offline as mentioned above.

These DNN-based methods require a large amount of collected various and heterogeneous data to reflect real indoor propagation conditions and thus achieve satisfying localization accuracy. Data collection is a highly consuming task in terms of time, energy, and human effort. Thus, researchers propose three levels of data fusion to combine data collected by each user in order to address this issue maintaining a good localization performance, including (i) sensors fusion, (ii) technologies fusion, and (iii) signal parameters fusion. When moving from a reference point 'training position' to another, signal measurements can be collected by users with the associated inertial measurement units (IMU) corresponding to different sensors existing in the used mobile device (i.e., accelerometer and magnetometer). Thus, the constructed database, which is composed of the radio signal and IMU measurements, requires no extra human effort, time, or energy compared to the simple database containing only received signal parameter measurements, as both types of measurements are collected simultaneously.

Based on this motivation, we propose in this paper two localization schemes based on DNN applying sensor data fusion combining the received signal strength indicator (RSSI) as a signal parameter, for its simple acquisition without extra cost or the need for synchronization, and inertial measurement unit (IMU) sensor data. The remainder of this paper is organized as follows: In Section 2, we formulate the problem, and we describe the proposed system model. Obtained localization results are presented and discussed in Section 3. Finally, we conclude in Section 4.

2. Problem Formulation and Proposed System Model

We consider an indoor environment where *N* training and validation nodes are placed. Each node is associated with an appropriate coordinates *C* with *C = 3* in this paper since we consider 3D coordinates. We note that the proposed schemes are still available for 2D. *M* anchor nodes (i.e., access points (APs) when working with WiFi) are placed in known locations. We consider a scenario where *N* sensor nodes communicate with *M* APs according to a centralized architecture i.e., each sensor node receives signals from all APs placed in the studied area and registers the associated IMU data. Thus, at node $i = 1: N$, *M* RSSI values are received from all existing APs and IMU data are collected with I = 6 i.e., 3 accelerometer values of IMU and 3 magnetometer values of IMU. Once collected, data are organized in order to construct a radio map called the 'fingerprinting database' which is used online for localization with an offline trained deep learning network (DNN). The localization is associated with a data fusion process that allows combining RSSI data with IMU data: RSSI+acceleration and RSSI+magnetometer data. For this, we propose two different approaches for data fusion and localization based on DNN as mentioned in Fig. 1.

- **First fusion method:** we consider a DNN model which takes as input vectors composed of RSSI and IMU data with size (*M*+3, *N*) and gives as output the estimated coordinates (*C*, *N*).
- **Second fusion method:** we consider two different DNNs. The first DNN uses RSSI to predict the initially estimated coordinates. These coordinates will be combined with IMU data to construct N vectors. Each vector contains C+3 elements, which are used to refine the initially estimated coordinates and obtain the final estimated coordinates.

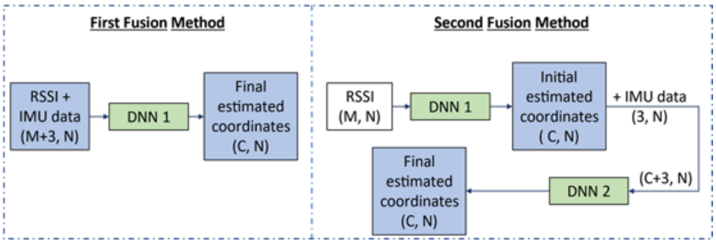


Fig. 1. Different steps of the two proposed methods for data fusion.

3. Evaluation of Localization Performance

In this Section, we present the obtained localization error when testing our proposed models based on the publicly available database IMUWiFine [10]. Each fingerprint contained in this database is composed of different fields as mentioned below:

- (1:*M)* RSSI values of APs expressed on dBm;
- *(M*+4: *M*+6) Accelerometer values of IMU;
- (*M*+7: *M*+9) Magnetometer values of IMU;
- (*M*+11: *M*+13): *x, y,* and *z* coordinates expressed in meters (m).

For model evaluation and test, we consider 2 sub-databases with *N* = 391 and *N* = 193, respectively and we choose randomly 500 positions which will be divided between training data (70 %) and validation data (30 %). Different used DNNs contain 2 hidden layers with 30 neurons and 10 neurons, respectively. In Fig. 2, we present a random distribution of 500 positions used for model evaluation.

We can easily notice from Table 1 that the second data fusion method is not efficient since it does not improve the localization accuracy compared to the use of RSSI only. However, the data fusion according to the first fusion method improves the localization accuracy by almost 15 % and 13 % when applied on Sub-database 1 and Sub-database 2 and considering the magnetometer data with collected RSSI. This improvement is equal to 9.33 % and 17.18 % when RSSIs are combined with acceleration. We can notice that the combination of RSSI with magnetometer data performs better when considering Sub-database 1, however, the combination of RSSI with acceleration is better when considering Sub-database 2. Thus, the choice of which parameter to combine with RSSI data is a simulation process, it changes from one set of data to another and it is related to the nature of the input database.

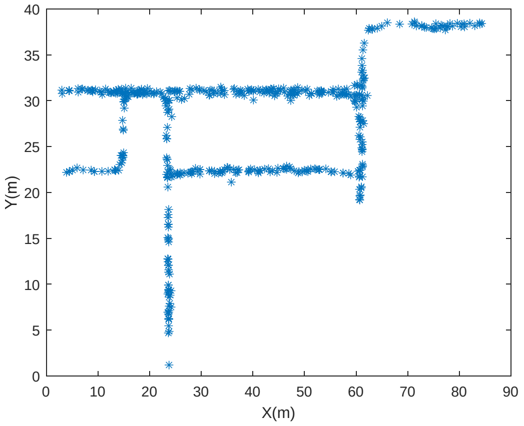


Fig. 2. Training and validation positions distribution over the studied indoor environment.

4. Conclusions

In this paper, we present two schemes for data fusion used for localization aiming to improve the obtained positioning accuracy. For this, we propose localization based on DNN taking as inputs collected RSSI data combined with IMU data (magnetometer data / acceleration data) and giving the estimated coordinates as outputs. Based on results obtained using the IMUWiFine database, we observed that the second fusion method, which ensures the localization based on two steps i.e., refining initially estimated coordinates, is not efficient since it is associated with a decrease in the localization accuracy compared to the conventional localization method using RSSI only. In future works, we intend to study how to select the IMU parameter to combine with RSSI (acceleration or magnetometer data).

Acknowledgments

The authors gratefully acknowledge the EU Horizon 2020 program towards the 6G BRAINS project H2020-ICT 101017226.

Table 1. Obtained localization error using data from the IMUWiFine database.

Parameters used for localization		**Sub-database 1**		**Sub-database 2**	
		Mean localization error (m)	**Improvement Vs RSSI only**	**Mean localization error (m)**	**Improvement Vs RSSI only**
RSSI + magnetometer data	Method 1	0.172	**15.68 %**	0.111	**13.28 %**
	Method 2	0.279	-36.76 %	0.128	0 %
RSSI + Acceleration	Method 1	0.187	**9.33 %**	0.106	**17.18 %**
	Method 2	0.258	-27.47 %	0.145	-13.28 %
RSSI only		0.204	–	0.128	–

References

[1]. J. A. Del Peral-Rosado, G. S. Granados, R. Raulefs, *et al.*, Whitepaper on new localization methods for 5G wireless systems and the Internet-of-things, in White Paper of the COST Action CA15104, *IRACON* , 2018, pp. 1-27.

[2]. J. Schiller, A. Voisard (Eds.), Location-Based Services, *Elsevier*, 2004.

[3]. R. Lea, M. Blackstock, Smart cities: An IoT-centric approach, in *Proceedings of the International Workshop on Web Intelligence and Smart Sensing*, 2014, pp. 1-2.

[4]. Z. Yang, Y. Liu, Quality of trilateration: Confidence-based iterative localization, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 21, Issue 5, 2009, pp. 631-640.

[5]. S. He, S.-H. G. Chan, Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons, *IEEE Communications Surveys & Tutorials*, Vol. 18, Issue 1, 2015, pp. 466-490.

[6]. M. Ibrahim, M. Torki, M. Elnainay, CNN based indoor localization using RSS time-series, in *Proceedings of the IEEE Symposium on Computers and Communications (ISCC'18)*, 2018, pp. 01044-01049.

[7]. A. B. Adege, H.-P. Lin, G. B. Tarekegn, *et al.*, Applying deep neural network (DNN) for robust indoor localization in multi-building environment, *Applied Sciences*, Vol. 8, Issue 7, 2018, 1062.

[8]. W. Njima, I. Ahriz, R. Zayani, et al., Deep CNN for indoor localization in IoT-sensor systems, *Sensors*, Vol. 19, Issue 14, 2019, 3127.

[9]. W. Njima, M. Chafii, A. Nimr, *et al.*, Deep learning based data recovery for localization, *IEEE Access*, Vol. 8, 2020, pp. 175741-175752.

[10]. M. Nurpeiissov, A. Kuzdeuov, A. Assylkhanov, et al., End-to-end sequential indoor localization using smartphone inertial sensors and WiFi, in *Proceedings of the IEEE/SICE International Symposium on System Integration (SII'22),* 2022, pp. 566-571.

(082)

Synthetic Data Generator based on Principal Curves

F. E. M. Borges, V. D. Reis and D. D. Ferreira
Federal University of Lavras, Department of Automation. Lavras, Minas Gerais, Brazil
Tel.: + 55 35999758275
E-mails: fernandoelias.mb@gmail.com, victor.daniel.reis10@gmail.com, danton@ufla.br

---

Summary: Nowadays, a large amount of data from different sources containing varied types of information are generated every day. However, there are reports of uncommon events that are rarer than others and are hidden in the data. Finding this information and treating it using machine learning is crucial to identify rare information and discovery knowledge. The generation of synthetic data of uncommon events to help the machine learning algorithms to treat unbalanced datasets is not trivial. This paper proposes a new method of synthetic data generation using Principal Curves. The method exploits the non-linear properties and relationships of data captured by the Principal Curve. To evaluate the algorithm, simulated experimental datasets were used. The statistical results showed the new method is suitable for future real-world applications.

Keywords: Machine learning, Pattern recognition, Principal curves, Synthetic data generation, Unbalanced learning.

---

1. Introduction

A large amount of data is generated every day from varied applications like healthcare, physics, finance, etc. In these applications, rare events can be hidden in the ordinary information. Examples of these events are rare diseases, financial frauds, risk management and others [1].

Since these uncommon events have low data, the machine learning algorithms may generate poor performances in their predictions. In a classification task, the classifier can become biased to the most numerous class [2].

One way to mitigate this problem is the oversampling of the minority class. The oversampling algorithms seek to generate new data from the original data, preserving its properties. There are several methods to generate new data from the original minority class database. Examples of oversampling methods are replication of the original data by random sampling; generate new data by Synthetic Minority Oversampling Technique (SMOTE) [3]; Adversarial Generative Networks (GAN) [4], and others.

Several applications of synthetic data generation can be observed in the literature. One of the most recent ones is that shown by Imakura et al. [5], who applied SMOTE to generate synthetic data for collaborative data analysis. The authors can generate new data with analytics capability and preserve their privacy.

In the work reported by Pu et al. [6], researchers investigated the use of a GAN variation to improve fault detection in industrial robots. The authors obtained good results with the application of oversampling, improving the classification hits.

The aforementioned examples show the good capacity of oversampling techniques for data augmentation, improving results in classification tasks.

In this paper, it is proposed a new method to generate synthetic data based on Principal Curves (PC) [7]. Due to the good capacity of PC to represent data in high dimensional spaces, it has been exploited for nonlinear data classification and clustering [8], in which promising results have been found. There are some algorithms to extract a PC, among them, the k-segments has been the most used [9]. The work reported in [8] applied the K-segments algorithm for clustering tasks, in which representative clusters for different data shapes were found.

2. Proposed Method

The proposed method may be summarized in two steps: (i) Principal Curve construction, and (ii) data generation.

2.1. Principal Curves Construction

K-segments is an algorithm to extract Principal Curves by an incremental method. It extracts the curves by straight-line segments and connects them by using another straight-line segment.

This algorithm takes advantages of being less bias to local minima and having practical convergence guaranteed. The k-segments functionality can be explained in three steps:

(i) Insertion of the first segment: in this step, the first segment is inserted in the direction of the first principal component and has a length of 3/2 of the standard deviation of the data;

(ii) Insertion of other segments and recalculation of their length: after insertion of the first segment, other segments will be inserted. For this, clusters based on Voronoi regions were found. Segments whose clusters have changed will have their lengths recalculated;

(iii) Convergence test: to verify if the k-segments converge, two tests are performed: if the maximum number of segments has been reached or if the Voronoi region has less than three points. If none of these conditions are satisfied, the k segments return to the second step.

Thus, the PC of the minority class is built based on the available data. After the PC construction, the k-segments algorithm returns the segment vertices and the points representing the data projected into de PC.

2.2. Data Generation

This step consists of obtaining the Euclidean distance from the original data to the projected points and generating new data. Firstly, the Euclidean distance from the original data to the projected points are calculated for each dimension. After that, a noise controlled by a hyperparameter called dispersion factor is added to the distances of each dimension. Thus, for a *n*-dimensional data, a modified (new) distance is obtained according to equation (1).

$$dx_i' = dx_i + r_i, \tag{1}$$

where dx_i' is the i-th axis modified distance, being $i = 1, 2, \ldots, n$; dx_i is the original distance from the original data to its projection in the PC, in i-th axis, and r_i is the noise added in the i-th axis.

From the modified distance, the synthetic data for the i-th dimension can be found as equation (2).

$$x_i' = x_{p,i} + dx_i', \tag{2}$$

where $x_{p,i}$ is the projected point in the i-th axis and x_i' is the synthetic data value in i-th axis. Finally, the synthetic data is represented by equation (3).

$$\mathbf{x}' = [x_1' \, x_2' \, \ldots x_n'] \tag{3}$$

Fig. 1 presents an illustration of the functionality of the proposed method. It is possible to note that the new data has a control region (represented by a circle); this region avoids instability of the synthetic data generation. The size of the synthetic data projection boundary increases as the dispersion factor increases.

Therefore, the method has main two parameters to adjust: the number of segments (k) used to build the PC, and the dispersion factor (r_i), which controls the region of new data generation.

To illustrate the complete functionality of the algorithm, a flowchart is presented in the Fig. 2.

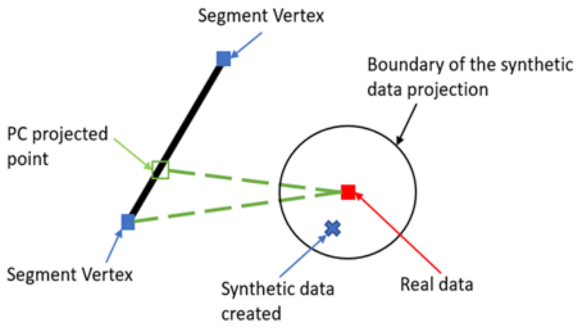


Fig. 1. Example of the method functionality.

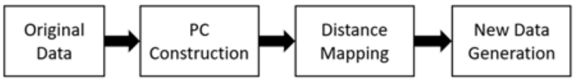


Fig. 2. Proposed method flowchart.

The proposed method was implemented in MATLAB software, using a computer with Intel® Core™ i5-1135G7 processor and 20 GB of RAM.

3. Results and Discussion

For performance evaluation of the proposed method, we used simulated datasets, and compared the generated synthetic data and the original one. We carried out four tests, by using ring, spiral, moon-shaped and 3-D spiral formats. For better visualization, the rate of synthetic data was fixed to 100 %. So, the size of synthetic data is the same as the original one.

We exploited visual analysis and applied the t-test and chi-square test (at 5 % of significance level) to compare the data distribution of the synthetic data and the original one.

Figs. 3-6 show the data distribution of both original and generated data (synthetic) for the ring, spiral, moon-shaped and 3-D spiral data sets, respectively. The statistical tests are presented in Tables 1-4, for the ring, spiral, moon-shaped, and 3-D spiral data sets, respectively.

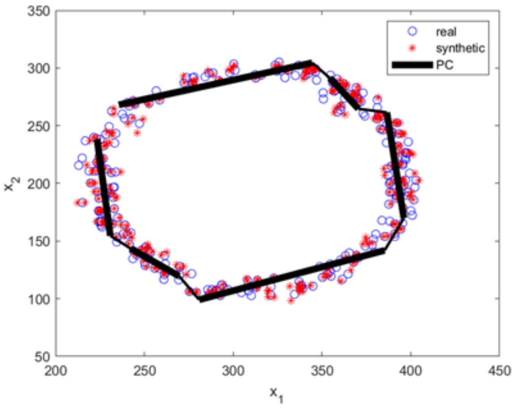


Fig. 3. Graphic distribution in ring data.

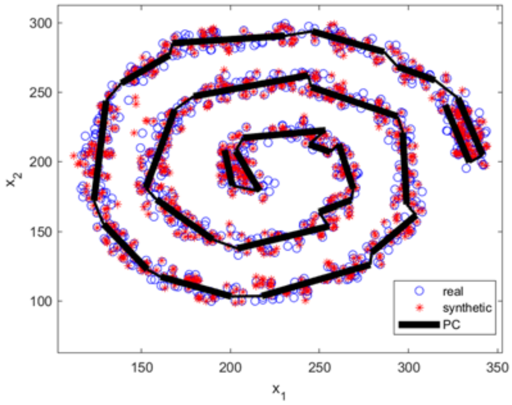


Fig. 4. Graphic distribution in spiral data.

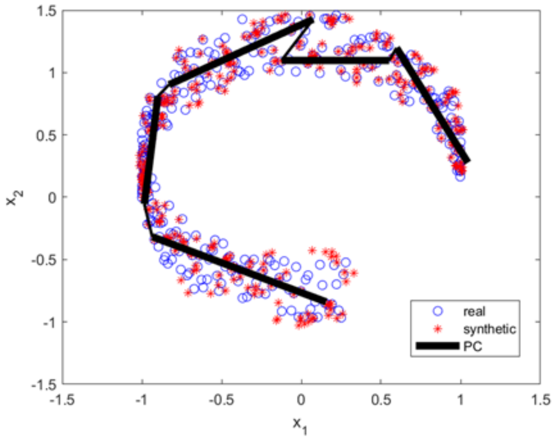


Fig. 5. Graphic distribution in moon-shaped data.

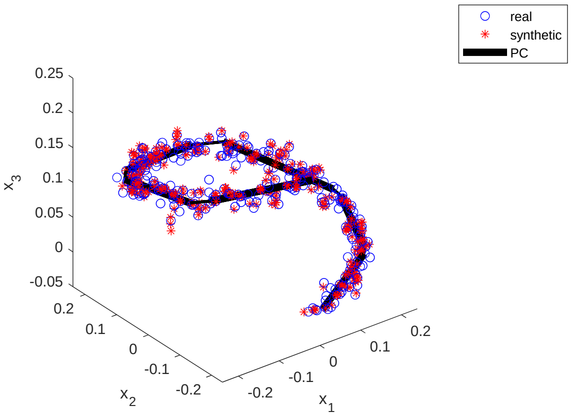


Fig. 6. Graphic distribution in 3-D spiral data.

Table 1. Statistical results for the ring data.

Dim.	χ^2	*p*-value	t-test
$x_1/\widehat{x_1}$	$7.74 * 10^4$	0.25	0
$x_2/\widehat{x_2}$	$7.74 * 10^4$	0.25	0

Table 2. Statistical results for the spiral data.

Dim.	χ^2	*p*-value	t-test
$x_1/\widehat{x_1}$	$9.46 * 10^5$	0.24	0
$x_2/\widehat{x_2}$	$9.46 * 10^5$	0.24	0

Table 3. Statistical results for moon-shaped data

Dim.	χ^2	*p*-value	t-test
$x_1/\widehat{x_1}$	$1.11 * 10^5$	0.24	0
$x_2/\widehat{x_2}$	$1.11 * 10^5$	0.24	0

Table 4. Statistical results for the 3-D spiral data.

Dim.	χ^2	*p*-value	t-test
$x_1/\widehat{x_1}$	$8.97 * 10^4$	0.24	0
$x_2/\widehat{x_2}$	$8.97 * 10^4$	0.24	0
$x_3/\widehat{x_3}$	$8.97 * 10^4$	0.24	0

By observing the graphical results, presented in Figs. 3-6, it is possible to notice that the proposed method can generate synthetic data without copying the original data. Furthermore, the shape of the new data has visual similarity in its shape, showing that the algorithm can preserve the shape of the original data.

The numerical results presented by Tables 1-4 show a maintenance of the statistical properties of the original data in the synthetic data. This is important for future applications in real-world scenarios, where statistical properties should be preserved to ensure that machine learning models trained on new data are able to generalize to real data during operation. Another scenario where maintaining the statistical properties of the data is crucial can be seen in [5], where only synthetic data will be distributed to maintain the privacy of the original data.

4. Conclusions

This paper aimed to provide a new algorithm of oversampling data. The simulated results show a proof of concept of the proposed method and present a potential for future applications of the algorithm in real world scenarios for classification and regression tasks.

For future works, the method will be applied to real unbalanced datasets to evaluate the impact of synthetic data on the performance of machine learning algorithms. Furthermore, the results generated by oversampling using PC will be compared with other methods present in the literature to compare their impacts on the machine learning model, computational cost and preservation of the original properties.

Acknowledgements

The authors would like to thank FAPEMIG, CNPq and CAPES, under Process number 88881.708828/2022-01, for supporting this work.

References

[1]. A. N. Tarekegn, M. Giacobini, K. Michalak, A review of methods for imbalanced multi-label classification. *Pattern Recognition*, Vol. 118, 2021, 107965.

[2]. E. Kaya, S. Korkmaz, M. A. Sahman, A. C. Cinar, DEBOHID: A differential evolution based oversampling approach for highly imbalanced datasets. *Expert Systems with Applications*, Vol. 169, 2021, 114482.

[3]. N. V. Chawla, et al., SMOTE: synthetic minority over-sampling technique, *Journal of Artificial Intelligence Research*, Vol. 16, 2002, pp. 321-357.

[4]. I. Goodfellow, et al., Generative Adversarial Nets, in *Proceedings of the International Conference on Neural Information Processing Systems (NIPS'14)*, 2014, Vol. 27, pp. 2672-2680.

[5]. A. Imakura, et al., Another use of SMOTE for interpretable data collaboration analysis, *Expert Systems with Applications*, Vol. 228, 2023, 120385.

[6]. P. U. Ziqiang, et al., Sliced Wasserstein cycle consistency generative adversarial networks for fault

data augmentation of an industrial robot, *Expert Systems with Applications*, Vol. 222, 2023, 119754.

[7]. T. Hastie, W. Stuetzle, Principal curves, *Journal of the American Statistical Association*, Vol. 84, Issue 406, 1989, pp. 502-516.

[8]. E. C. C. Moraes, D. D. Ferreira, G. B. Vitor, B. H. G. Barbosa, Data clustering based on principal curves, *Advances in Data Analysis and Classification*, Vol. 14, 2020, pp. 77-96.

[9]. J. J. Verbeek, N. Vlassis, B. Kröse, A k-segments algorithm for finding principal curves, *Pattern Recognition Letters*, Vol. 23, Issue 8, 2002, pp. 1009-1017.

(083)

What Does the MSW Industry Need: Data Technologies or More Waste?

R. Hussein
State University of New York, Syracuse, NY, USA
Tel.: + 315-695-2340
E-mail: ezpsc@yahoo.com

Summary: Massive amounts of intricate, complex, and unpredictable data are gathered by the municipal solid waste (MSW) industry (MSWI) for analysis and the making of management and operational decisions. In a world that is unpredictably changing, we depict the MSWI's crisis as follows: First, can long-term choices be made using the large amounts of seasonally collected MSW data from the modern world? Secondly, can the MSWI continue operating normally in the current, financially precarious MSWI and in a society that relies on live streaming technologies? Thirdly, how can the goals of the MSW frameworks, plans, acts, and other schemes be achieved using simple conventional numerical data manipulation? The primary objective of this article is to give the MSWI an overview of the cutting-edge concept of a data-analytics-driven, robust DDMSWD in order to capitalize on the companies that have already implemented data-driven technologies and modernize their processes.

Keywords: Data analytics, Data science, Municipal waste, Waste management, Resource recovery, Recycling, STEM.

---

1. Introduction

This paper introduces the novel idea of data-driven municipal solid waste (MSW) decision making, or DDMSWD. The world strives to reduce waste and increase reuse, recycling, and recovery to protect the built environment, including water, soil, and energy resources. The distinction between DDMSWD and conventional sustainable materials management, or CSMM, is that the latter aims at wastes destined for landfills or municipal combustors, while the former is a fresh approach.

Massive amounts of intricate, complex, and unpredictable data are gathered by the municipal solid waste (MSW) industry (MSWI) for analysis and the making of management and operational decisions. We depict the crisis briefly as follows: First, modern society is undergoing an unpredictably quick transformation. Who could have foreseen the current state of the economy, epidemics, conflicts, and international waste-related regulations, to name a few examples? The US Centers for Disease Control and Prevention (CDC) gathered and aggregated the COVID-19 data at the national [1], state [2], and county [3] levels on a weekly basis because of the difficulty with predictability. Therefore, the obvious question is: Can long-term choices be made using the large amounts of seasonally collected MSW data from the modern world? It is obvious that the MSWI's strategy for data mining needs to be carefully examined, from its roots in data to where it is now. Secondly, it appears that the MSWI is interested in "understanding" many aspects of waste, such as composition, characterization, rates, generation, and differences, based on the aforementioned quantitative assessments and obtained MSW data. On the one hand, the invaluable trove of findings derived from the MSW data collection [4, 7-13] amply illustrates the data's extraordinarily complex and variable hereditary features. The same treasure, however, shows how to examine that kind of data using simple numerical standard processes. When the two hands are put together, what significance do the numbers released for the industrial SMW operations have? Can the MSWI continue operating normally in the current, financially precarious MSWI [4] and in a society that relies on live streaming technologies? Thirdly, the MSWI is keeping an eye on trash-related objectives like zero waste, efficient garbage collection, waste reduction, recycling, and resource recovery, according to the same treasure. How can the goals of the Municipal Solid Waste (MSW) frameworks, plans, acts, and other schemes [6-8] and the Federal Resource Conservation Act (RCRA) of the United States [5] be achieved using simple conventional numerical data manipulation?

Even if it cannot predict the unforeseen, the MSW must continually be prepared to address both ongoing and new challenges. Whether or not the collected data and generated statistics indicate a good trend [7-12], the MSWI needs to adjust its analytical strategy to take into account modern technology and the globe as opposed to that of a bygone era. As unforeseen obstacles arise, this vision ensures that wise judgments are made.

The primary objective of this article is to give the MSWI as a whole an overview of the cutting-edge concept of a data-analytics-driven, robust DDMSWD in order to capitalize on the companies that have already implemented data-driven technologies and modernize their processes. The terms management and operation, MSW industry and MSW organizations, waste and municipal waste, strategy, plan, and framework were all used interchangeably throughout this text. Furthermore, while having every reason to be proud of its leadership, the State of New York's MSWI is solely utilized as an example in this report.

2. Background

The "Beyond Waste" plan by the New York State Department of Environmental Conservation (NYSDEC) [8] established seventeen strategic objectives for "sustainable materials management". Based on the data gathered, the Onondaga County Resource Recovery Agent (OCRRA) recently released a wealth of insightful statistics; however, it appears that they were produced using basic analytic methods [9-13]. Because no one could have even imagined the state of modern data analytics when those statistics were being developed (DA), one might be interested in the hidden information contained in such reports. This observation necessitates the immediate adoption of remedial measures to extend data-related waste practices to over 60 % of the top corporations operating on DA for viability and sustainability in the USA. There are many names on the list, including [14, 15] Google, IBM, Facebook, Instagram, Apple, Amazon, Starbucks, Tesla, Nintendo, Netflix, Bristol-Myers-Squibb, and McDonald's. The MSWI will be able to accomplish its objectives by finding opportunities for improvement in the current frameworks since municipalities aim to maximize the extraction of value from waste. If it weren't for its reliance on straightforward, traditional analytical techniques, the MSWI wouldn't have fallen behind other firms. Isn't it about time the MSWI updated its data-related tools?

The MSWI likely understands the advantages of data-driven decision-making processes, but it doesn't appear that they currently have a known strategy, structure, or plan to put the most well-known modern technology, DA, into practice. The literature that is now available [9-13] suggests adequately large waste-related data sets in terms of amount, diversity, and complexity, but their quantification techniques are not in line with contemporary methods. Before contemplating future stages like minimization or maximization, raising or reducing, establishing or advancing, engaging or encouraging the identified results, the MSW authority needs to concentrate on the first step any organization takes, i.e., the analysis of data. As was already said, this research offers a new perspective on the MSWI as a whole. The following sections explore various facets of DA, challenges, and opportunities for implementing the technologies in the MSWI, which has a long way to go.

3. Data Science and Analytics in Perspective

Data analytics includes both qualitative and quantitative methods for deriving insights from large, complicated data collections. Unquestionably, it has also revolutionized all industries. The characteristics of recorded data greatly differ in heterogeneity, real-time gathering, and integration of information. These characteristics make more traditional, fundamental data analysis methods inadequate for the MSW. As was already said, the ability of DAs to navigate the data tsunami is clearly essential to the survival of the top organizations. We realized that the two relevant approaches for future advances in DDMSWD are data identification and prediction as well as data analytics for performance and assessment.

Mathematical, statistical, computer and information science, machine learning, and other disciplines are generally used in the DA's execution. For the DDMSWD and nearly all other businesses, the DA is a broad and deep field.

4. A Perspective for Future DA-driven MSWI

The literature that is now available reveals the real root of MSWI's predicament. The industry gathers enormous amounts of complex data on the one hand while using fundamentally standard analytical techniques on the other. This finding might help to explain why the MSW dilemma is rising globally. The MSWI needs to move quickly to resolve its quandary using the DA's potent powers and learn from the major industries that moved from business as usual to modernization. This commonsense viewpoint is more in line with current methods than outdated ones. Via his demonstrated national and international academic and industrial research and development, including advanced computer development [16-22], this author advocates changing the current MSWI, beginning with capacity building of the future MSWI manpower. This requires the creation of special training on the applications of DA for MSW, including computational technologies, DA fundamentals and execution, etc. The existing MSW workforce must also be included to revamp its skills and prepare it to gradually move toward modern technologies in everyday practice and making decisions. Last but not least, encourage and assist the schoolers who are the seeds of the next generation of industrial leaders, to consider DA for acquiring lifelong industrial credentials. The New York Times, the Washington Post, the Guardian, and other sources attest to the importance of data science and analytics for tackling modern challenges. In this regard, data science has massive amounts of data in common with DA. Data analytics is essentially a special application of data science in which the datasets are massive and require overcoming logistical challenges to deal with them. Science, Technology, Engineering, and Mathematics, or STEM, form the basis of this phase of future workforce preparation for data analytics. We envisioned [23] a trifold DA-STEM-MSW incubator for training a workforce that could transform vast amounts of MSW data into useful outcomes. Python is a suitable coding language for the trifold technique due to its widespread use in DA. Being open-source and free, simple to learn intuitively, having a great online community, integrating well with other programs, and being quicker than comparable tools like R and MATLAB are just a few of Python's numerous benefits. Internships, tuition reimbursement, and joint projects with schools to promote hands-on learning should also be taken into account for further initiatives.

5. Conclusions

As long as there are living things, there will be SW. The same coin has two faces: data and waste. How can the MSWI create effective strategies with outdated technologies in a world where data is everywhere? To answer this question, this article examined the MSWI's predicament and envisioned a comprehensive, ground-breaking framework to transform its practice, DDMSWD, that revolves around data analytics.

References

[1]. Centers for Disease Control and Prevention, COVID data Tracker, https://covid.cdc.gov/covid-data-tracker/ #datatracker-home

[2]. Coronavirus in the U. S., https://www.nytimes.com/ interactive/2021/us/covid-cases.html

[3]. Centers for Disease Control and Prevention, COVID-19 Integrated County View, https://covid.cdc.gov/ covid-data-tracker/#county-view?list_select_state = all_states&list_select_county = all_counties&data-type = CommunityLevels

[4]. Onondaga County Resource Recovery Agency (OCRRA), 2019 Annual Report on Recyclables Recovered, https://ocrra.org/wp-content/uploads/1/ 2020/03/recycling_2019.pdf

[5]. US Environmental Protection Agency (EPA), Resource Conservation and Recovery Act Laws and Regulations, https://www.epa.gov/rcra

[6]. New York Department of Environmental Conservation, Municipal Solid Waste (MSW) Landfills in New York State, https://www.dec.ny.gov/chemical/23682.html

[7]. Onondaga County Resource Recovery Agency (OCRRA), Onondaga County Comprehensive Solid Waste Management Plan Update for 2021-2022, https://ocrra.org/wp-content/uploads/1/2022/03/ CSWMP-Update-2021-COMPILED-FINAL-Rev1-reduced.pdf

[8]. New York State Department of Environmental Conservation (NYSDEC), Beyond Waste – A Sustainable Materials Management Strategy for New York State, https://www.dec.ny.gov/docs/ materials_minerals_pdf /frptbeyondwaste.pdf

[9]. Onondaga County Resource Recovery Agency (OCRRA), 2019 Waste Characterization Study, https://ocrra.org/wp- content/uploads/1/2020/05/ WasteQC_2019_Final_K RL_050720.pdf

[10]. Onondaga County Resource Recovery Agency (OCRRA), 2020 Annual Report on Recyclables Recovered, https://ocrra.org/wp-content/uploads/1/ 2021/04/recycling_2020.pdf

[11]. Onondaga County Resource Recovery Agency, Comprehensive Recycling and Market Analysis, https://ocrra.org/wp-content/uploads/1/2020/08/ CRA_2019_082420.pdf

[12]. Onondaga County Resource Recovery Agency, Compliance Report for 2019 2020 Plan Update for 2021 2022, https://ocrra.org/wp-content/uploads/1/ 2022/03/CSWMP-Update-2021-COMPILED-FINAL-Rev1-reduced.pdf

[13]. Onondaga County Resource Recovery Agency (OCRRA), 2020 Annual report, https://ocrra.org/wp-content/uploads/1/2021/03/AnnualReport_2020_KRL _031921-1.pdf

[14]. Alyssa Schroer, 75 Big Data Companies to Know, March 2023, https://builtin.com/big-data/big-data-companies-roundup

[15]. ICAS, 10 Companies that Uses Big Data, 2022, https://www.icas.com/news/10-companies-using-big-data

[16]. R. Hussein, An analytic solution for the mechanics performance of packaging corrugated cardboards containers with non-rigid bonding and under edgewise loads, *Journal of Packaging Technology and Research*, Vol. 4, 2020, pp. 85-94.

[17]. R. Hussein, A First of Its Kind Data Analytics Driven Best Practice Decision Making Approach for Sustainable MSW, *New York State Department of Environmental Conservation Leadership*, 2022.

[18]. R. Hussein, Be Kind and Keep No Waste Left Behind in Mind, *Onondaga County Resource Recovery Agency Leadership*, 2022.

[19]. R. Hussein, R&D Knowledge-Based & Expert System Technologies for Built & Natural Environments, https://aitechm.wixsite.com/8kbextech04182021tmp

[20]. R. Hussein, AI-based tools for performance and monitoring of sustainable built and natural environments, and the climate, Chapter 7, in Advances in Artificial Intelligence (S. Yurish, Ed.), Vol. 1, *IFSA Publishing*, 2019, pp. 185-203.

[21]. R. Hussein, Knowledge-based tools for monitoring and management, and design of the engineered infrastructure construction systems, Chapter 5, in Advances in Computers and Software Engineering (S. Yurish, Ed.), Vol. 2, *IFSA Publishing*, 2019, pp. 199- 251.

[22]. R. Hussein, Treatise on sustainable infrastructure construction: green composites, cross laminated/mass timber, wood truss connectors, nondestructive technologies, health assessment and monitoring: utility poles and geofoam, Chapter 1, in Advances and Technologies in Building Construction and Structural Analysis (A. Kaboli, Ed.), *IntechOpen*, 2021, pp. 122-135.

[23]. R. Hussein, A novel python-based threefold data science, stem, climate change framework, in *Proceedings of the 3rd International Conference on Advances in Signal Processing and Artificial Intelligence*, 2021, pp. 28-30.

(084)

AI-based, Fast Moisture Measurement for Food Drying Application

N. Stoupas [1] and M. Maniadakis [1,2]
[1] MadeEasy SCE, I. Venezi 36, N. Alikarnassos, Herakleion, Crete, Greece
[2] Foundation for Research and Technology Hellas, N. Plastira 100,
Vasilika Vouton, Herakleion, Crete, Greece
E-mail: mmaniada@ics.forth.gr

Summary: Besides the popularity of food drying applications, they are rarely supported by digital systems providing information about the food state during dehydration. The moisture content of food is a key factor of food quality as it affects microbial growth and the quality of food storage, therefore being of particular interest for many food processing applications, especially for food drying. The present work provides a fast and low-cost solution for the direct estimation of food moisture. The proposed approach uses a mobile mini spectral sensor to collect information about the light reflection of the food containing valuable information about its content. A machine learning approach is then used to map the obtained spectral measurements to a food moisture prediction. The proposed approach achieves the accurate prediction of food moisture content. The latter makes the current work interesting for a wide range of food processing tasks.

Keywords: Machine learning, LSTM, Industrial application, Food processing.

---

1. Introduction

Food drying, refers to the removal of water by evaporation from a solid or liquid food, in order to obtain a “new product” with a fairly lower water content. To monitor the progress of drying it is necessary to frequently take food samples in order to measure its moisture content. This is a task that takes time to complete and therefore, alternative approaches are frequently considered [1].

Interestingly, Near Infrared (NIR) spectrometers consider the light absorbance characteristics of materials to observe and measure the material content of foods. What is particularly efficient with NIR spectroscopy is that it does not require any laborious and time-consuming preparation of the samples.

NIR spectroscopy has been well-known in material sensing and only recently it has been used for analysing food samples [2, 3]. The present work focuses particularly on the fast and accurate measurement of moisture in drying foods, here dried pulse puree is examined, to immediately assess if they are bellow a food-specific upper threshold.

The results of moisture measurements affect dried pulse puree performance in the food industry and marketing as moisture is a key parameter for the quality of food storage. The current work considers a handheld, easy to use Nirone Sensor X, that is provided by Spectral Engines, to collect information about the spectral signature of the food, The Nirone Sensor X is a single point scanner based on NIR InGaAs sensor measuring the spectrum from 1550 nm to 1950 nm with a step of 10 nms. By using this sensor, we can easily get the spectral signature of the studied material, which provides valuable information about the constituents of the food.

Then, we employ the Long Short-Term Memory (LSTM) neural networks [4] which can effectively balance between knowledge updating and forgetting, and have showed rather low sensitivity on the vanishing/exploding gradients problem, being particularly effective in learning long-term dependencies and dealing with signal processing. The LSTM neural network is in charge of processing the spectral signatures to predict the level of moisture in the food at the given moment. The proposed approach is used into a real food processing setup to facilitate food drying monitoring and save the waste of energy.

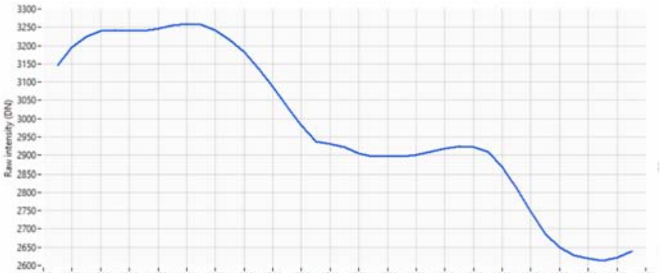
2. Methodology and Results

The technology developed in this work aims to support the production of dried pulse puree meals and is directly tested in the field. We use the Nirone Sensor X, which is a mini, handheld material analysing device, to observe and analyse the drying food.

To avoid direct contact of the sensor with the food, in the present work we use a transparent plastic bag as a container of the studied food samples. Two plots showing the spectral signature of the moisture-rich and the dried pulse puree, are provided in Fig 1. We can easily observe that the two signals have quite different characteristics which have been only affected by the presence of water.

We use an LSTM neural network to accomplish moisture forecasting based on the observed spectrum. More specifically, we adopt the stacked LSTM architecture that can effectively process associated value sequences to extract and retain for a long time their hidden interactions which are further utilized for prediction making. The LSTM takes as input 40 spectral measurements obtained by the Nirone Sensor X and provides as output a prediction of the percentage of the moisture contained in the food.

To train the LSTM we collected data that are described as pairs of (i) spectral measurements and (ii) the actual moisture of the food obtained by using a dedicated device, namely the PMB Moisture Analyser, which accurately determines the moisture content of food samples (this is a time-consuming process that takes approximately 4 minutes per sample to complete). Following this approach, we collected 180 spectrum-moisture content pairs that are used to tune the LSTM. From the available data, 150 randomly selected pairs are used for training the LSTM and the remaining 30 samples are used for testing the performance of the network in previously unseen samples.



(a) Moisture-rich food – 50% water

(b) Dried food – 8% water

Fig. 1. Indicative spectral measurements of food (a) without and (b) with water.

The mean squared error of the predicted moisture content for the training and the testing dataset have been 0.22 and 0.34, respectively. The accuracy of model predictions is graphically illustrated in Fig. 2 which provides a summary of the network performance when dealing separately with each sample. Interestingly, the error of the LSTM predictions is sufficiently low to allow its use for the direct and fast measurement of the moisture content of food samples. This is particularly important for food drying applications, which must ensure that the moisture content of food packaged and shipped to market is below a predetermined upper limit.

Besides the increased accuracy of the proposed approach, it is necessary to note that the LSTM model obtained after the training process is dedicated to the food that is considered in the given study and a new data collection and model training procedure is necessary to switch model performance to another food type.

3. Conclusions

The technology developed in this work aims to simplify and accelerate moisture estimation in food samples and at the same time improve the accuracy of the obtained predictions. Due to its simplicity and high performance, it may be of particular interest for a wide range of food processing applications.

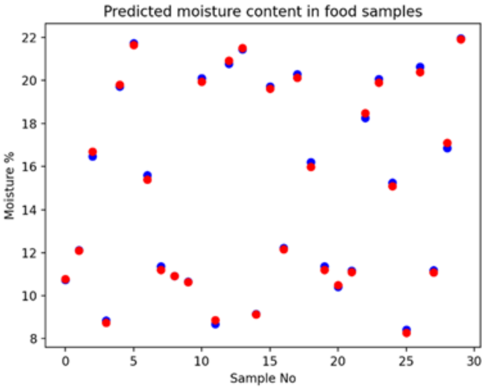


Fig. 2. Graphical illustration of LSTM-based food moisture predictions against the ground truth.

Acknowledgements

This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call

RESEARCH—CREATE—INNOVATE (project code: T2EDK- 03242, PRIME).

References

[1]. E. Neofotistou, N. Stoupas, M. Maniadakis, A low-cost AI-based soft sensor for food drying monitoring, in *Proceedings of the 4rd International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI'22)*, 2022.

[2]. M. V. Zambrano, B. Dutta, D. Mercer, H. MacLean, M. Touchie, Assessment of moisture content measurement methods of dried food products in small-scale operations in developing countries: A review, *Trends in Food Science & Tech.*, Vol. 88, 2019, pp. 484-496.

[3] K. B. Beć, J. Grabska, C. W. Huck, Miniaturized NIR Spectroscopy in Food Analysis and Quality Control: Promises, Challenges, and Perspectives, *Foods*, Vol. 11, 2022 May 18, Issue 10, 1465.

[4]. R. Staudemeyer, E. R. Morris, Understanding LSTM – a tutorial into Long Short-Term Memory Recurrent Neural Networks, *arXiv Preprint*, 2019, arXiv:1909.09586.

(085)

Force Signal and Superficial Electromyographic Signals Associated to Hand Movements: A General Mixed Effects Model

C. L. Sandoval-Rodriguez [1], D. M. Reyes-Bravo [2], A. D. Rincón-Quintero [1], O. Lengerke [1] and A. F. Jimenez-Quezada [1]
[1] Unidades Tecnológicas de Santander, Calle de los estudiantes 9-82, 680005, Bucaramanga, Colombia
[2] Universidad Autonoma de Bucaramanga, Av. 42 #48 - 11, 680002, Bucaramanga, Colombia
Tel.: (+57) 607 6917700
E-mail: csandoval@correo.uts.edu.co

Summary: Previous works have sought to achieve hand prostheses with natural movements. It has been employed tools to recognize patterns of surface electromyographic signals (sEMG) associated with each move. Although many successful studies classify some types of hand movements with high performance, the results show that the speed and strength must be analyzed so that the resulting moves are like those of a natural hand. This work evaluates sixteen healthy subjects at two different forces and six types of hand movements (pronation, supination, ulnar deviation, radial deviation, flexion, and extension -96 records of SEMG -Force). The objective was to obtain a general mixed effects model that would allow the relation of the force signal with the tone of the forearm SEMG signals. Fixed effects were associated with the force signal and random effects with the type of movement. The work results show models at the determination coefficient R^2 - median 0.9 and random effects with high incidence. The SEMG signal would explain the variation in force signal applied in each type of hand move.

Keywords: Forearm SEMG, Hand movements, Force signal, General mixed effects model.

---

1. Introduction

In previous works, the motivation of researches is focused on electromechanical systems with movements so close to naturals hands [1-5]. Also, for applications where force estimation is required to facilitate diagnosis of fatigue [6-8] and prediction the movement dynamics, like velocities and movement types [9-18]. Other works have used the electromyography signal to relate it to dynamic characteristics, such as strength, to establish the health condition of a subject with muscle fatigue and improving the accuracy of force prediction [19-21].

On the other hand, in [22], a model to relate force and sEMG surface electromyography was developed, divided into incremental and saturation zones. In each case, a different behavior was observed. Also, for each type of movement, the models were different. In this way, the move's type affects the model's parameters if it is to be general model. Therefore, in this work, we present a general mixed effects model, to explain the relationship between Force an sEMG signals.

2. Materials and Methods

2.1. Data Collection

We were studied six moves: ratio and ulnar deviation, flexo-extension, and pronation-supination according to [22, 23]. There are 96 records for six moves of 16 healthy subjects. We used Powerlab hardware and Bio Amp signal conditioner [22, 23] to data acquisition. Also we used software of adinstruments Labchart interface to observe the sEMG signals. We applied a combinate method for envelope extraction. First, a second-order low pass filter (Butterworth) was selected, and it use a two-sample moving averaging and a 20 ms window for obtaining the envelope. To force measurement, we used a previous validated procedure [22], and we will explain below in Fig. 1.

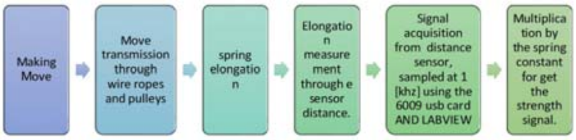


Fig. 1. Force measurement (procedure).

To continue with the explanation about the tests, the test was divided into three sessions: flexo-extension, prone-supination and radio-ulnar deviation.

Flexion-extension session: in this session the subject is positioned with flexion of the elbow at 90º, hand pronated and shoulder abduction at 45º. Here perform 2 repetitions of the flexion and extension movement.

Prone-supination session: in this session the same procedure was applied anterior, except for the location of the patient, who is seated with elbow and shoulder flexion at 90º and the elbow immobilized, with the anterior face of the hand oriented towards the center of the body (oriented towards a sagittal plane).

Radio-ulnar deviation session: in this session, the procedures are repeated above, but with elbow flexion to 90˚, shoulder abduction to 45º, hand pronated.

2.2. Modelling

We calculate sEMG envelope median, and median force signal for each move. Afterwards, a general linear model of mixed effects (GLME) was obtained, taking the average sEMG as fixed effects and the types of movement as random effects [24], see equation (1).

$$Force = est * sEMG + int + mtm, \qquad (1)$$

where *est* is the Value Estimated, *sEMG* is the Tone (sEMG envelope), *int* is the intercept, *mtm* is the move type model (random effects).

3. Results

Tables 1 and 2 shows the results. Here it is observed that the random and fixed effects have high significance (p value < 0.05). While the intercept has low significance. Fig. 2 shows measured force as function of sEMG and random effects (mtm).

Table 1. features of GLME.

Features	Value
Number of observations	96
Fixed effects coefficients	1
Random Effects coefficients	6
Distribution	Normal

Table 2. results of GLME.

Name	Estimation	Square error (SE)	P value
Intercept	0.00012	0.095	0.9
sEMG	0.0075	0.044	<0.05
Random Effects coefficients **(mtm)**	Standard deviation = 1.16 $*10^{-3}$ P value <0.05		
Determination coefficient R^2	0.91		

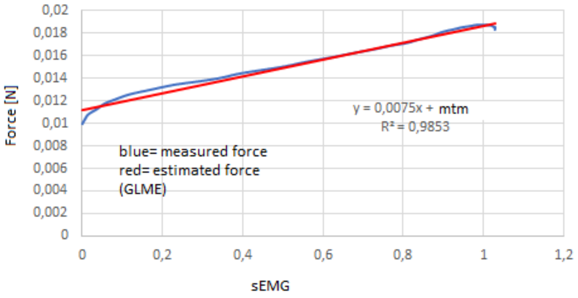


Fig. 2. Force Model GLME (without intercept) and measured force.

Additionally, Fig. 3 shows a comparison between measured force signal and estimated force signal. Here it observed that error (differences SE) is low.

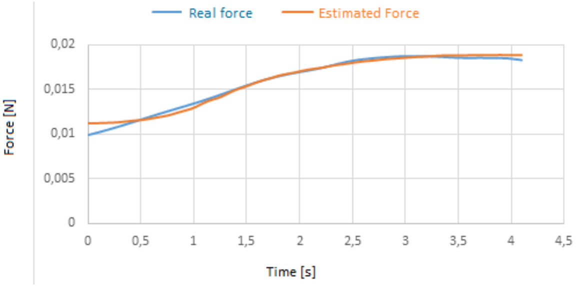


Fig. 3. Comparison between Force Model GLME (without intercept) and measured force.

4. Discussion

The dynamics of wrist movements are associated with sEMG activity in the forearm [23]. In many cases, sEMG has been used as a predictor of fatigue and other dynamic characteristics [3, 8, 25]. Previous work as [13] and [22] has evaluated the relationship between force and sEMG signals, with good results. However, the relationship is not linear in the whole domain. For example, in [22] the modeling was divided into two zones (increasing zone and saturation zone), and there are two distinct models with different coefficients. Also, procedure cannot be generalized. That is, this procedure must be applied to each subject and in each movement separately.

Therefore, at present work, we calculated the relationship between force and sEMG signals as GLME. We incorporated the effects of moving differently, because its significance, as random effects was high. The obtained model is general for all subjects (moves type) tested, and the determination coefficient was high (min 0.9). The Error (SE) was low, according to Figs. 3 and 4). Also, in Fig. 4 s It is seen that the maximum error occurs at the beginning (12 %) but always remains very low (4 %) afterwards (time > 0.2 s).

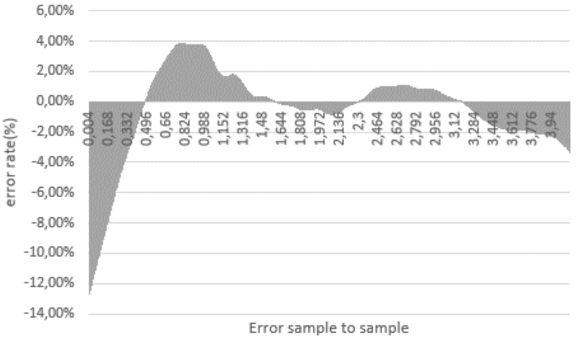


Fig. 4. Error (calculated sample to sample- 4 seconds for all subjects).

On the other hand, the possible variations parameters on model (force-sEMG) due to the velocity changes have not been evaluated. In this sense, an interesting test is to verify the variation of parameters in the models, when altering speed conditions in each case. That is, to evaluate the influence of velocity on the parameters of the Force-sEMG model, for each subject.

Finally, the relationship between force and sEMG is linear and general for all movements and subjects. Therefore, the shown model results could be used in future works, including hardware implementation.

5. Conclusion

The relationship between the Force and sEMG signals is linear when you enter the type of movement in the modeling (GLME). The random effects (type of movement) have a great importance in the model results. The coefficient of determination R^2 increases if the intercept in the model is neglected.

References

[1]. C. L. Sandoval-Rodriguez, E. Y. Veslin-Díaz, B. E. Tarazona-Romero, J. G. Ascanio-Villabona, C. G. Cárdenas-Arias, C. A. Angulo-Julio, Electromechanical Hand Prototype for the Simulation of the Opening and Closing Movement, *IOP Conf. Ser. Mater. Sci. Eng.*, Vol. 1154, Issue 1, 2021, 012035.

[2]. I. Chihi, L. Sidhom, E. N. Kamavuako, Hammerstein-Wiener Multimodel Approach for Fast and Efficient Muscle Force Estimation from EMG Signals, *Biosensors*, Vol. 12, Issue 2, 2022, 117.

[3]. J. Mouchoux, S. Carisi, S. Dosen, D. Farina, A. F. Schilling, M. Markovic, Artificial Perception and Semiautonomous Control in Myoelectric Hand Prostheses Increases Performance and Decreases Effort, *IEEE Trans. Robot.*, Vol. 37, Issue 4, 2021, pp. 1298-1312.

[4]. R. Kawama, H. Maemura, S. Tanigawa, Regional difference in the EMG-force relationship of the hamstring muscles during knee flexion, *J. Phys. Fit. Sport. Med.*, Vol. 10, Issue 5, 2021, pp. 233-241.

[5]. O. Fukuda, D. Sakaguchi, Y. He, N. Yamaguchi, H. Okumura, Bimodal Control of a Vision-Based Myoelectric Hand, *IEEE Access*, Vol. 9, 2021, pp. 98369-98380.

[6]. Q. Zhang, L. Fang, Q. Zhang, C. Xiong, Simultaneous estimation of joint angle and interaction force towards sEMG-driven human-robot interaction during constrained tasks, *Neurocomputing*, Vol. 484, 2022, pp. 38-45.

[7]. Y. Wu, S. Liang, T. Yan, J. Ao, Z. Zhou, X. Li, Classification and simulation of process of linear change for grip force at different grip speeds by using supervised learning based on sEMG, *Expert Syst. Appl.*, Vol. 206, 2022, 117785.

[8]. B. Fang *et al.*, Simultaneous sEMG Recognition of Gestures and Force Levels for Interaction with Prosthetic Hand, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 30, 2022, pp. 2426-2436.

[9]. C. L. Sandoval-Rodriguez, A. C. Pita-Mejia, Villamizar-Mejia R, B. E. Tarazona-Romero, O. Lengerke-Perez, Model to Relationship the Speed of Hand Movements with the sEMG Signals from the Forearm, *J. Phys.: Conf. Ser.*, Vol. 2224, 2022, 012094.

[10]. S. A. Raurale, J. McAllister, J. M. Del Rincon, EMG Biometric Systems Based on Different Wrist-Hand Movements, *IEEE Access*, Vol. 9, 2021, pp. 12256-12266.

[11]. H. Su, W. Qi, Z. Li, Z. Chen, G. Ferrigno, E. De Momi, Deep Neural Network Approach in EMG-Based Force Estimation for Human-Robot Interaction, *IEEE Trans. Artif. Intell.*, Vol. 2, Issue 5, 2021, pp. 404-412.

[12]. M. J. Islam, S. Ahmad, F. Haque, M. B. I. Reaz, M. A. S. Bhuiyan, M. R. Islam, A Novel Signal Normalization Approach to Improve the Force Invariant Myoelectric Pattern Recognition of Transradial Amputees, *IEEE Access*, Vol. 9, 2021, pp. 79853-79868.

[13]. C. Huang, M. Chen, Y. Zhang, S. Li, Zhou, Model-Based Analysis of Muscle Strength and EMG-Force Relation with respect to Different Patterns of Motor Unit Loss, *Neural Plast.*, Vol. 2021, 2021, 5513224.

[14]. R. Wen, K. Yuan, Q. Wang, S. Heng, Z. Li, Force-Guided High-Precision Grasping Control of Fragile and Deformable Objects Using sEMG-Based Force Prediction, *IEEE Robot. Autom. Lett.*, Vol. 5, Issue 2, 2020, pp. 2762-2769.

[15]. G. Beninati, V. Sanguineti, A dynamic model of hand movements for proportional myoelectric control of a hand prosthesis, in *Proceedings of the Annu. Int. Conference IEEE Eng. Med. Biol. Soc. (EMBS'19)*, 2019, pp. 6648-6651.

[16]. W. Seok, Y. Kim, C. Park, Pattern recognition of human arm movement using deep reinforcement learning, in *Proceedings of the Int. Conference Inf. Netw. (ICOIN'18)*, 2018, pp. 917-919.

[17]. A. Córdova, I. Nuin, D. Fernández-Lázaro, I. Latasa, J. Rodríguez-Falces, Actividad electromiográfica (EMG) durante el pedaleo, su utilidad en el diagnóstico de la fatiga en cidistas, *Arch. Med. del Deport.*, Vol. 34, Issue 4, 2017, pp. 217-223.

[18]. C. Sandoval Rodríguez, R. Villamizar Mejía, E. Delgado Velosa, A. Cordero Cardozo, Estimación de velocidad del movimiento de mano usando redes neuronales artificiales y mediciones electromiográfica, *Rev. Ing. Biomédica*, Vol. 4, Issue 8, 2010, pp. 41-56.

[19]. R. Ma, L. Zhang, G. Li, D. Jiang, S. Xu, D. Chen, Grasping force prediction based on sEMG signals, *Alexandria Eng. J.*, Vol. 59, Issue 3, 2020, pp. 1135-1147.

[20]. K. Wang, Y. Huang, X. Zhang, Estimation of handgrip force from nonlinear SEMG-force relationship during dynamic contraction tasks, *in Proceedings of the IEEE Int. Conference on Robot. and Biomimetics (ROBIO'17)*, 2018, pp. 412-417.

[21]. M. Nakatani, K. Murata, H. Kanehisa, Y. Takai, Force-velocity relationship profile of elbow flexors in male gymnasts, *PeerJ.*, Vol. 9, 2021, e10907.

[22]. C. L. S. Rodriguez, R. V. Mejia, B. E. T. Romero, A. D. R. Quintero, A. J. R. Nieves, Relationship between force signal and superficial electromyographic signals associated to hand movements, *Period. Eng. Nat. Sci.*, Vol. 11, Issue 1, 2023, pp. 64-73.

[23]. C. L. Sandoval-Rodriguez, Caracterización de la dinámica de los movimientos básicos de la mano a partir de la actividad electromiográfica del antebrazo, MD Thesis, *Universidad Industrial de Santander*, 2013.

[24]. B. T. West, K. B. Welch, A. T. Galecki, Linear Mixed Models: A Practical Guide Using Statistical Software, *CRC Press*, 2022.

[25]. M. Asefi, S. Moghimi, H. Kalani, A. Moghimi, Dynamic modeling of sEMG-force relation in the presence of muscle fatigue during isometric contractions, *Biomed. Signal Process. Control*, Vol. 28, 2016, pp. 41-49.

(086)

Reliable Learning-based Controllers and How Structured Simulation is a Path towards Them

K. Kušić [1], **R. Schumann** [2], **M. Gregurić** [1], **E. Ivanjko** [1] **and M. Šoštarić** [1]
[1] University of Zagreb Faculty of Transport and Traffic Sciences,
Department of Intelligent Transportation Systems, Zagreb, Croatia
[2] HES-SO Valais-Wallis, University of Applied Sciences Western Switzerland – Valais, Switzerland
kresimir.kusic@fpz.unizg.hr

Summary: New approaches to control stochastic non-linear time-variant processes include the application of machine learning techniques. One of the problems with learning-based controllers is their reliability in a wide area of process parameters as the controller is trained using a limited set of representative scenarios, either chosen by the designer or taken from historic records. Thus, reliable controller behavior can be guaranteed only in scenarios applied during controller training. Due to the very larger number of random variables and possible scenarios, not all variations can be applied in the controller training process using simulators to guarantee good controller behavior when applied in a real system. One case is traffic control (signal programs, variable speed limit, ramp metering) having large travel patterns variety. The concept of Structured Simulations Framework (SSF) can cover most probable learning scenarios. Thus, applying SSF enables a systematic controller training approach by complementing existing scenarios with synthesized ones that evoke or replicate substantial aspects of real traffic. Such training is necessary to ensure reliable learning-based controllers. This paper discusses the concept of applying SSF to ensure the reliability of learning-based controllers and proposes the application in traffic control for the case of variable speed limits on motorways.

Keywords: Learning-based controller, Controller reliability, Structured simulation, Variable speed limit.

1. Introduction

Today, there is a need to control stochastic non-linear time-variant processes. Classical feedback controllers designed using process modeling and linearization in characteristic working points cannot cope with such processes, especially regarding a wide area of process parameters. Different controller parameters are needed for particular working points. To overcome this problem, machine learning is applied and learning-based controllers can be designed [1]. The advantage of this approach is that the control law is learned as a mapping between the input measurements and control output for a wide area of different process parameters fulfilling a defined criteria function. The drawback is that a good control output can be ensured only for scenarios presented to the controller during the training phase. In the case of large processes with many random variables and possible scenarios, the generation of learning scenarios can very fast become unfeasible if one wants to guarantee good control output in every possible situation.

Traffic control is a good example of such a problematic process. Namely, today's traffic control centers manage larger networks of motorways or connected signalized intersections [2]. Change of traffic flows is under the influence of daily human behavior demanding repeating mobility patterns related to working days and weekends or holidays, the state of the transport infrastructure, and the weather. Thus, significant stochastic non-linear time variant behavior is present in the controlled process (signal programs for intersections, and variable speed limit or ramp metering rate on motorways) creating the need for an appropriate large set of structured learning scenarios [3]. Such a set of learning scenarios has to cover most often traffic behavior and behaviors that appear not often but are related to some potential regular events (bad weather influence, traffic incidents, holidays, etc.). A good starting point for the creation of such learning scenarios sets can be obtained by analyzing collected real world measurements as shown in [4].

To ensure that the controller learns how to resolve most of the possible scenarios, which can occur in real world situations, in feasible time, a structured approach of presenting the learning scenarios to the controller during the training process is needed. This is opposite to current training approaches where only a few representative learning scenarios are used and evaluated afterwards [5]. With a structured approach to creating training scenarios, it can be ensured that the controller can successfully resolve a wider area of scenarios that can appear in the real world increasing the reliability of such a learning-based controller when applied in real world applications. This describes the aim of this paper to define such a structured framework to increase the reliability of the learned controller with the use case of traffic control i.e., Variable Speed Limit (VSL) on motorways taken as a use case.

This paper is organized as follows. The second section elaborates on the structured simulation concept. The third section explains how learning-based controllers can be designed including open problems. The fourth section describes the possible application of the structured simulation concept for learning-based traffic controllers. The continuing fifth section concludes the paper.

2. Structured Simulation Concept

The main goal of the structured simulation approach is to automate and systematize a search process about the system's behavior by means of simulation or in a wider sense experimentation. It is assumed that system's behavior for whatever reasons cannot be described as an equation-based system. However, it is possible to analyze the behavior of the system in a controlled environment, e.g., in form of controlled experiments or computational simulations. Without loss of generality, we will address in the following that simulations are performed. For practical considerations, the total number of simulations needed to perform is relevant, as each simulation run requires resources, e.g., like computing time. Therefore, the idea of structured simulation is to maximize potential leanings about a system's behavior with a given number of simulations runs. We assume that the system's behavior can be computed as part of the simulation, and depends primarily on the input parameters of the simulation. Thus, each simulation run can provide an insight about the system. This knowledge discovery process happens in form of a guided search process, in which different states of the parameter space of the simulation system can be evaluated. The main idea of the structured search approach is that this search process needs to be a guided / informed search, taking advantage of potential domain knowledge to reduce the search time, i.e., a number of states in the parameter-space that needs to be investigated compared to an uninformed search in which the points in the state space would be visited in a purely randomized way, as it is the case in a Monte-Carlo like approach.

The main ideas of structured simulations have been implemented in the Structured Simulation Framework (SSF)[1]. The overall process is structured within three stages:

- Scenario generation in which the necessary states to be visited are computed;
- Simulation execution in which for each state the corresponding simulation is performed;
- Result handling in which results are collected and potentially analyzed.

This is visualized in Fig. 1. These stages can be executed partially overlapping. However, for the sake of simplicity we assume no parallel execution options, but a strict linear flow.

2.1. Scenario Generation

For the organization of the search process that needs to be performed, the scenario generation phase is most critical, and therefore will be detailed in the following. Depending on which phase of the controller design needs to be addressed, different search objectives, and therefore search strategies, need to be specified. The general idea that states, i.e., a particular combination of input parameters, need to be visited in a particular order to increase available information about the behavior of the system under investigation.

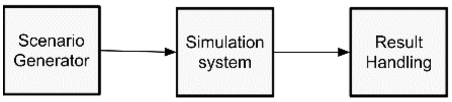


Fig. 1. Schematic structure of the Structured Simulation Framework modules [6].

For performing a state-based search approach state transitioning rules needs to be defined. These rules specify how the next state to visit can be derived. These state-transition rules are specified in form of Modifiers, which implement a particular type of state transition function. A set of modifiers can be provided, and given an initial state of the search, a number of states to evaluate can be derived. This allows for an ordering, and therefore enumerating the states to visit, which is a pre-requested for the structured search. The principle of this is shown in the following Fig. 2.

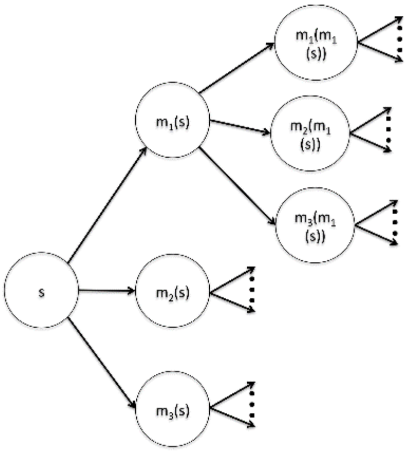


Fig. 2. Schematic representation of the state space enumeration.

The principles of structured simulation can be used in two different ways during the design phase of a learning-based controller. On the one hand, it can be used during the training phase of the controller, as it can be used to generate training data that can cover a wider range of the parameter-space, e.g., by implementing a novelty search strategy [7]. On the other hand, the framework can also be used in the validation phase of the controller, providing a systematic search for states in which the controller might not perform well, or to the contrary, provides empirical evidence that the controller is capable of

---

[1] http://silab.hevs.ch/structSim/structsim.html

handling various situations, it is likely to be confronted with during its operations [6]. So far, the SSF framework has been used with two different simulation environments, to ensure its general structure. A link to the commercial traffic simulator tool VISSIM, which is used by traffic researchers to validate traffic management approaches, taking advantage of learning-based traffic controllers is presented in [8]. The second simulator was a Game of life simulation [9]. In both cases, the evaluation has been done for each state in isolation. This can be considered as an alternative to the commonly used Monte-Carlo simulation approach. However, the foundation of SSF in VSL motorway analysis has not yet been explored in detail and a general concept needs to be established before the implementation of SSF in the simulations training process of the learning-based VSL controller can start.

3. Challenges in Design of Learning-based Controllers

The most used approach for learning-based controllers is the one which is based on the Reinforcement Learning (RL) approach. RL approaches can be divided into model-based and model-free methods. The model-free methods are currently most investigated since they do not require existing learning datasets. They are used in an online fashion for solving optimal control problems stated as Markov Decision Processes (MDPs). Most of the current control problems can be modeled by using the MDP approach since it enables modeling control tasks in discrete time. Thus, at each time step, the controller receives feedback from the controlled system in the form of a state signal and takes an action in response. As a result, the RL approach accounts for a change in the state signal that could lead to a change in the optimal control action. Thus, they can handle nonlinear and stochastic dynamics and nonquadratic reward functions [10]. Those features are ideal for controlling systems which are in their nature stochastic such as traffic flows. The common approach in designing the RL controllers is in the simulation loop. The system is modeled within a simulation environment and the RL controller performs learning in it until the end results converge to satisfactory values [11]. The core problem in their design is to generate enough representative states in the controlled system for desirable learning convergence of controller parameters. Particularly, this problem is related to the state-action exploitation-exploration ratio which governs when it is needed to stop the process of learning and continue just to use the learned control policies.

3.1. Latest Methodologies in Learning-based Control Design and Challenges

The model-free RL controllers can be divided into two wide categories with respect to design approaches. The first of them is value-based. They are based upon temporal difference learning in which the learn value function is computed. Typical representatives are Temporal Difference (TD), State-Action-Reward-State-Action (SARSA), and Q-learning (QL) algorithms. The second RL type of model-free controllers is related to directly learning an optimal policy or trying to approximate the optimal control policy if the true optimal policy is not attainable. The REINFORCE algorithm is the most prominent representative of that category. The policy-based controllers tend to directly optimize the control policy, which is the core goal of good control. Therefore, they are more stable and less prone to failure compared to value-based counterparts [5]. The value-based methods such as QL are less stable and suffer from poor convergence since they learn an action value function approximation usually called Q-values. Those Q-values are in their vanilla version stored in tabular format which is then used to find a corresponding policy. The advantage of value-based methods is their off-policy nature. Thus, in their operation work, they are much more sample efficient compared to policy-based methods since they exploit data from control knowledge repositories based on stored state-action function.

3.2. Open Problems

The convergence speed is problematic even for the most advanced learning-based controllers. It heavily depends on tuning the controller’s hyperparameters such as learning rates, discount factor in QL, regularization parameters of ANN models, etc. Furthermore, the learning convergence depends on the complexity, differentiation rate, and persistence of learning scenarios which can be understood as outliers. Those scenarios can significantly reduce the learning convergence or make convergence unstable [12]. This is especially the case with RL approaches which are based on Q-function approximation using ANN models. Thus, the coverage of sufficiently different learning scenarios and avoiding too extreme cases have the potential to stabilize learning convergence. Regarding these mentioned issues, it is needed to develop a framework that has the ability to generate scenarios that maintain increased convergence within a simulation environment to improve learning convergence. Moreover, it is imperative to assess the reliability of all possible scenarios. Thus, it is needed to avoid learning from scenarios that are not possible to happen outside the simulation environment. It is imperative to create constraints that prevent generating such scenarios. Those learning scenarios can have a negative effect on learning convergence as well.

However, several works applied RL techniques for VSL control on motorways. For example, in [3, 13], nonlinear function approximation techniques in RL were applied to improve control of an underlying nonlinear and nonstationary traffic flow on a motorway using RL based VSL. In [3], research on the importance of state description and simulation scenario generation on the learning process was presented. Particularly, the control policy of RL based VSL was further improved by enriching the agent's state variables with predictive information about the expected traffic (speeds and densities) of the controlled motorway segment by running parallel simulations.

4. Structured Simulations for Learning-based Traffic Controllers

In this section, we discuss the application of principles of structured simulation for the design of learning-based traffic controllers using the example of a VSL controller. Optimizing VSL on motorways is about choosing the right speed limits for the current traffic flow conditions [14]. In the event of a traffic jam, for example, the goal of the VSL system is to slow down and harmonize the traffic arriving into the downstream active bottleneck in order to relieve the congested sections and prevent further capacity drop and, if possible, re-stabilize the traffic flow (Fig. 3).

4.1. Motorway Traffic Process Characteristics

Traffic flow on a motorway can be described by the fundamental diagram relating three basic traffic parameters flow, speed, and density. Two main traffic characteristics can be distinguished: free-flow (stable) and congested (unstable) traffic flow (Fig. 6). Free flow condition is described by higher travel speeds and less dense traffic flow i.e., a lower number of vehicles per motorway segment. As the traffic volume increases, the traffic reaches the so-called critical point where the capacity is exceeded. From this point on, the traffic flow becomes unstable and significant interactions between vehicles occur characterized by higher traffic density and lower speed resulting also with lower flow rate. These three dependent variables are, thus, used either as a measure of state representation or to define objective functions for evaluating learning-based VSL controller behavior.

4.2. Learning-based VSL

Numerous approaches have been used for VSL control [15, 16]. However, nowadays, learning-based control techniques have shown great potential to improve VSL. Among them, the QL algorithm is widely studied for VSL control optimization. QL-based VSL (QL-VSL) controller (agent) perceives and interacts with its environment (motorway section) at each control time step by performing actions (speed limits) and receiving feedback signals called rewards (see Fig. 4). Thus, the QL-VSL agent learns to associate an action a with the expected long-term discounted rewards R for performing that particular action in a particular state s and by following an optimal policy π. Thus, the action-value function (1) expresses how good action is to be applied to a particular state of the environment for transition in the next improved state [17]. The QL algorithm can be expressed as:

$$q(s,a)_{\pi} = E[R_{t+1} + \gamma R_{t+2} + \\ +\gamma^2 R_{t+3} \cdots |S_t = s, A_t = a, A_{t+1:\infty} \sim \pi], \qquad (1)$$

where $q(s,a)$ represents an action-value function. R represents rewards that an agent received where discount factor γ controls the importance of future rewards. An action a is an executed action from an available set of actions A_t, s is a possible state of the controlled process from a set of states S_t, and optimal policy π denotes the mapping function from states to optimal actions.

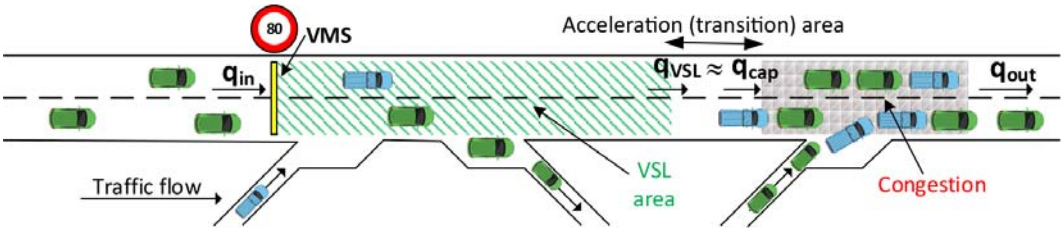


Fig. 3. Application of VSL for bottleneck control [15].

Therefore, the state variables that the VSL agent can use must be carefully selected to provide much information about the dynamics of the traffic flow, but uniquely. For example, densities uniquely define the traffic state on a motorway, but traffic volume does not (Fig. 6). The similar applies to the reward function when RL is used for modeling of a VSL controller [3], [5]. Fig. 4 visualizes the RL-VSL control process which can improve control policy during the operation. At every time step, the RL-VSL agent senses the world (traffic flow parameters on the controlled motorway section) and takes actions (speed limits) in it, and receives rewards or punishments based on the consequences of the taken actions and accordingly adjust the learning model in order to achieve better reward score.

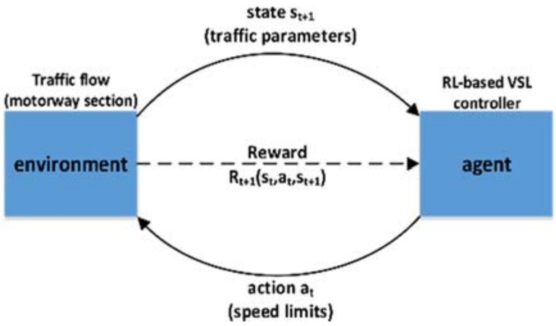


Fig. 4. Reinforcement learning framework for VSL control [15].

4.3. Structured Simulation-based VSL Controller Design

The important prerequisite for applying RL-VSL is appropriate training and evaluation processes in a traffic simulator. Thus, the quality of the learned control law strongly depends on the generated training dataset, which must provide relevant traffic scenarios for the motorway area where RL-VSL is to be used. For that, SSF can be used to create an appropriate training dataset complementing existing traffic scenarios with synthesized ones that evoke or replicate substantial aspects of real traffic.

1) Strategy for the training data generation: The overall training data has to be generated in a way that on the one hand, it provides enough data points so that the learning controller can actually generalize its behavior based on the training sample. This requires on the one hand a sufficient detailing of the parameter space in which the controller has to operate most of the time, and where the performance is critical, while on the other hand, the controller must have seen during its training phase a sufficient number of cases, which only have a low probability to occur in reality, to ensure that has obtained sufficient knowledge to react at least operative, even though not optimal in such situations. The search strategy therefore most balance between exploitation elements, here it focuses on certain areas (Fig. 5), and exploration elements in which it covers the typically larger part of the space, which will have a significantly lower probability of occurrence. In the following, we outline the strategy for computing a training set that aims to balance these two aspects. However, this approach requires an a-priori data analysis, e.g., of historic data, to provide insights into what parameter combinations can be considered typical, and to what probability they occur. For this, a theoretical model is proposed that relies on so-called modifiers that systematically change the states of the running simulation to reproduce the desired dataset for the training process. For simplicity, we allow the state domain contains two regions of interest, with each region assigned a modifier m1 and m2 computed by (2) and (3), respectively.

$$m1 = (max(\lceil e1(n) \rceil, \lceil p1(n) \rceil))k, \tag{2}$$

$$m2 = \\ = (\max(\lceil e2(M - n) \rceil, \lceil p2(M - n) \rceil))k, \tag{3}$$

Precisely, m1 is referred to as the modifier indicating the sufficient number of data points needed from the exploitation region [b, c] (Fig. 5), while m2 defines points in the exploration region [a, b) and (c, d]. The parameter δ defines the step size of the search strategy. Accordingly, $n = (c - d)/\delta$ and $M = (d - a)/\delta$ represent the number of discrete states within the exploitation and exploration regions, respectively. Similarly, the probabilities p1 and p2 define the probability of the occurrence of a state within the mentioned regions. The numbers e1 and e2 define the designed efficiency of the learning controller within the exploitation and exploration regions. If the controller has to deal with a large dataset (m1+m2 too large), one can use the scaling factor k = (0, 1]. By scaling the training dataset, the initial ratio between m1 and m2 remains the same, so there is no bias.

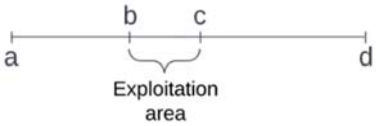


Fig. 5. State space domain.

For example, the low-frequency states (fewer probable events) are evident from the fundamental diagram in Fig. 6 (synthetic traffic data obtained from a microscopic road traffic simulator [18]). If we look at the x-axis (densities uniquely define traffic condition), we can roughly relate some density values to the boundaries (a, b, c, d) in the state space domain (Fig. 5).

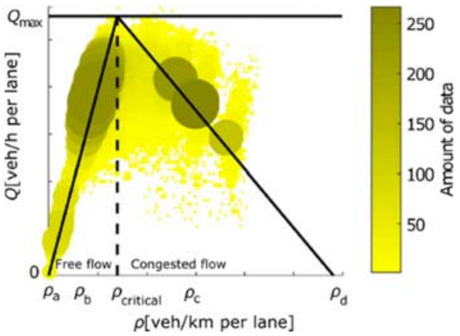


Fig. 6. Fundamental traffic flow diagram [18].

In reality, the fundamental diagram could have a different distribution of data, e.g., the most frequent data could be measurements that occurred during the night, which is characterized by low traffic intensity (longer period, more measurements received), while rush hours are less present. Thus, data density and the period during which the data are retrieved are correlated.

2) Technical integration of the SSF: Functional integration in the block diagram shown in Fig. 7 for automatic configuration of control strategies learned by RL-VSL consists of the SSF framework and the microscopic traffic simulator Simulation of Urban MObility (SUMO) [19].

At the heart of SSF is a scenario generator that is used to systematically change states to ensure a ratio of exploitation/exploration states, e.g. how much m1 and m2 data to present in the training process of RL-VSL. The output of the simulator is used to evaluate the controller itself; accordingly, some strategies where poor performances are detected need to be re-trained more frequently. For example, each simulation took T hours to cover all relevant scenarios. Of course, since RL-VSL does not require only one simulation for training, multiple runs are required. This is where the simulation controller comes in, whose job is to automate the training process; e.g. start the traffic simulation, stop the running simulation, insert a new value of the modifier into the running simulation, and finally collect the results of the running simulation for performance analysis and strategy evaluation. Therefore, there is in general a glue code between scenario generator, controller, and simulator which wiring blocks into SSF framework. Thus, in our case, we need to ensure communication between the SSF traffic scenario generator (that computes modifiers) and SUMO simulator together with the RL-VSL controller into one loop so that such a client-server architecture enables direct modification of simulated scenario and results analytics needed to compute modifiers for structure simulation scenario modification.

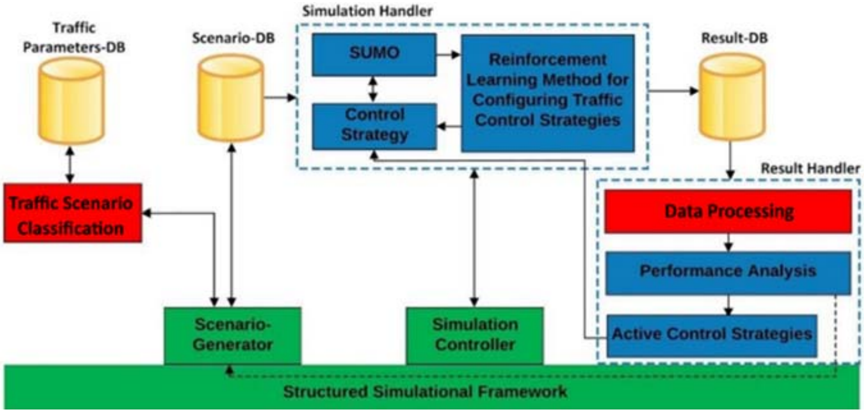


Fig. 7. Block scheme for automated configuration of control strategies learned by VSL learning controller.

5. Conclusion and Future Work

This paper presents a conceptual framework to obtain a representative dataset for training a learning-based VSL controller through the concept of applying SSF. Thus, the functional integration, i.e. closing the loop between SSF, RL-VSL controller, and simulator SUMO is explained. In general, the appropriate datasets used for training of RL-VSL should be non-biased, i.e., the data most likely to occur on the real motorway should be in a certain proportion to those less likely to occur. In this way, the required number of data points is minimized, so that the simulations and training require less computing power, but still ensure a reliable behavior of the controller in possible traffic cases, i.e., a satisfactory generalization. For this, we presented the detailed model within SSF used to calculate the state modifiers to systematically change/generate the states in the running simulation scenario in SUMO. Although the presented work is a rather conceptual study, it is based on some assumptions that should be addressed in future work. The theoretical model for calculating the modifiers, i.e., how often and in what direction the state space should be searched is presented in this paper. An important factor is the mapping from the fundamental diagram to the state space domain used in computing the modifiers. We have demonstrated the concept of mapping based on specific domain knowledge of the controlled process, i.e., from the fundamental traffic diagram by partitioning regions of the traffic densities according to traffic conditions: free flow, critical flow, and congested flow. However, this is not sufficient because it does not take into account the density of data (the occurrence of data), which, especially on motorways, depends on the time period in which the data are collected. Therefore, further analysis is needed to develop a suitable method for mapping the real traffic state space to the state space domain used in the SSF logic along with the simulation evaluation.

Acknowledgements

This work has been partly supported by the Croatian Science Foundation under the project IP-2020-02-5042, and by the European Regional Development Fund under the grant KK.01.1.1.01.0009 (DATACROSS). The author and Ph.D. student Krešimir Kušić received the 2021-2022 Swiss Government Excellence Scholarship to visit HES-SO Valais-Wallis, Switzerland. This research has been carried out within the activities of the Centre of Research Excellence for Data Science and Cooperative Systems supported by the Ministry of Science and Education of the Republic of Croatia.

References

[1]. M. Yu, S. Chai, A survey of direct learning control, in *Proceedings of the Chinese Control Conference (CCC'19)*, Guangzhou, China, 2019, pp. 2536-2541.

[2]. J. A. Calvo, I. Dusparic, Heterogeneous multi-agent deep reinforcement learning for traffic lights control, in *Proceedings of the 26th Irish Conference on Artificial Intelligence and Cognitive Science (AICS'18)*, 2018.

[3]. E. Walraven, M. T. Spaan, B. Bakker, Traffic flow optimization: A reinforcement learning approach, *Engineering Applications of Artificial Intelligence*, Vol. 52, 2016, pp. 203-212.

[4]. F. Vrbanić, M. Miletić, E. Ivanjko, Z. Majstorović, Creating representative urban motorway traffic scenarios: Initial observations, in *Proceedings of the 63rd International Symposium ELMAR-2021*, 2021, pp. 183-188.

[5]. M. Gregurić, K. Kušić, E. Ivanjko, Impact of deep reinforcement learning on variable speed limit strategies in connected vehicles environments, *Engineering Applications of Artificial Intelligence*, Vol. 112, 104850, 2022.

[6]. R. Schumann, C. Taramarcaz, Towards systematic testing of complex interacting systems, *CEUR Workshop Proceedings*, Vol. 2397, 2019, pp. 55-63.

[7]. M. Boussaa, O. Barais, G. Sunyé, B. Baudry, A novelty search approach for automatic test data generation, in *Proceedings of the* IEEE/ACM *8th International Workshop on Search-Based Software Testing*, 2015, pp. 40-43.

[8]. M. Gregurić, E. Ivanjko, R. Schumann, C. Taramarcaz, Structured simulation: A framework for the automated analysis of adaptive systems, in *Proceedings of the TUD 1102 COST ARTS Final Conference*, Bordeaux, France, 2015.

[9]. D. Kraft, Game of life simulation, Bachelor Thesis, *University of Applied Sciences*, Western Switzerland, 2017.

[10]. L. Busoniu, T. de Bruin, D. Tolić, J. Kober, I. Palunko, Reinforcement learning for control: Performance, stability, and deep approximators, *Annual Reviews in Control*, Vol. 46, 2018, pp. 8-28.

[11]. M. Gregurić, M. Vujić, C. Alexopoulos, M. Miletić, Application of deep reinforcement learning in traffic signal control: An overview and impact of open traffic data, *Applied Sciences*, Vol. 10, Issue 11, 2020, 4011.

[12]. A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, Y. LeCun, The loss surfaces of multilayer networks, in *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics (AISTATS'15)*, 2015, pp. 192-204.

[13]. E. Vinitsky, K. Parvate, A. Kreidieh, C. Wu, A. Bayen, Lagrangian control through Deep-RL: Applications to bottleneck decongestion, in *Proceedings of the 21st International Conference on Intelligent Transportation Systems (ITSC'18)*, 2018, pp. 759-765.

[14]. G. Iordanidou, C. Roncoli, I. Papamichail, M. Papageorgiou, Feedback-based mainstream traffic flow control for multiple bottlenecks on motorways, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, Issue 2, 2015, pp. 610-621.

[15]. K. Kušić, E. Ivanjko, M. Gregurić, M. Miletić, An overview of reinforcement learning methods for variable speed limit control, *Applied Sciences*, Vol. 10, Issue 14, 2020, 4917.

[16]. E. R. Müller, R. C. Carlson, W. Kraus, M. Papageorgiou, Microsimulation analysis of practical aspects of traffic control with variable speed limits, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, Issue 1, 2015, pp. 512-523.

[17]. R. S. Sutton, A. G. Barto, Reinforcement Learning: An Introduction, *The MIT Press*, 1998.

[18]. E. Ivanjko, K. Kušic, M. Greguric, Simulation analysis of two controllers for variable speed limit control, *Proceedings of the Institution of Civil Engineers – Transport*, Vol. 175, Issue 7, 2022, pp. 413-425.

[19]. D. Krajzewicz, J. Erdmann, M. Behrisch, L. Bieker, Recent development and applications of SUMO – Simulation of Urban MObility, *International Journal on Advances in Systems and Measurements*, Vol. 5, 2012, pp. 128-138.

(087)

EMDNet: A New DL Framework to Classify Motor Imagery-based Brain Activities in Both Healthy and Paraplegic Subjects

Niraj Bagh [1], **Fatemeh Shahlaei** [2], **M. Ramasubba Reddy** [1] **and M. S. Zambare** [2]
[1] Biomedical Engineering Group, Department of Applied Mechanics,
Indian Institute of Technology Madras, Chennai, India
[2] Department of Electronic Science, Savitri bai Phule University, Pune, India
E-mail: niraj.iitm17@gmail.com

Summary: In this research work, a new deep learning (DL) framework known as EMDNet is proposed to classify both left- and right-hand imaginations of the subjects. The EMDNet consists of two blocks: first is Empirical Mode Decomposition (EMD) and second is 1-D Convolutional Neural Networks (CNNs). The EMD block decomposed raw Electroencephalogram (EEG) signals into a set of Intrinsic Mode Functions (IMFs). Out of many IMFs, the one which is highly affected during both left- and right-hand imaginations is selected by the Power Spectral Density (PSD). The selected IMF is fed into the 1-D CNNs, which consist of convolutional layers, pooling layers and fully connected layers to classify both left- and right-hand imaginations. The effectiveness of the proposed network is tested on both benchmark dataset (BCI competition-II) and In-House Lab recorded dataset. The mean classification accuracy (% CA) of the proposed technique on both benchmark and Lab recorded datasets are found as 95.30 % and 91.83 %, respectively, which are higher than state-of-the-art techniques.

Keywords: Brain-computer interface, Convolutional neural network, Deep learning, Electroencephalogram, Empirical mode decomposition, Intrinsic mode function, Motor imagery.

1. Introduction

Brain-Computer Interface (BCI) is a control system, which converts neurophysiological signals produced by the human brain into control signals that can be used to operate wheelchairs, robotic arms, video games, etc., [1]. In the last few decades, several techniques like electroencephalogram (EEG), magnetoencephalogram (MEG), functional magnetic resonance imaging (fMRI), near-infrared spectroscopy (NIRS) have been used to measure and monitor brain activities. Among all these techniques, the electroencephalogram (EEG) is commonly used for BCI because of its high temporal resolution, non-invasive, portability and affordability [2-4]. Again, the EEG-based BCIs are categorized into four groups: steady state visual evoked potentials (SSVEP), P300 evoked potentials (P300), slow cortical potentials (SCPs) and motor imagery (MI) [5, 6].

This study mainly focuses on the motor imagery (MI) EEG-based BCI, and the motor imagery can be defined as the mental stimulation or rehearsal of any physical movements or passive observations of an action. As per the literature, the motor neurons present in the motor cortex region of the brain are synchronized during no specific mental task or rest state, resulting in an EEG signal with higher power in a specific frequency band. On the other hand, the motor neurons desynchronized and produced an EEG signal with lower power in a particular frequency band when doing some mental tasks. In general, the power of the sensorimotor frequency band (8-30 Hz) decreases in the contralateral hemisphere of the brain when a subject starts executing or imagining any movement, and this process is called event related desynchronization (ERD) [7, 8]. On the other hand, the power of the sensorimotor frequency band increases when a subject is in a resting state, and this process is known as event related synchronization (ERS). The ERD/ERS process plays a key role in designing a robust MI-based BCI system. The MI-based BCI has the following steps: (a) signal acquisition; (b) feature extraction; and (c) classification of the brain activities. Feature extraction is the most important step, and the extraction of notable features from the human brain is challenging when a subject imagines any movement.

Numerous feature extraction algorithms were proposed in the literature to address this issue, including band power (BP), inter-trial variance (IV), auto-regressive (AR) model, spectral decomposition, wavelet transform (WT), Hilbert transform (HT), filter bank common spatial patterns (FBCSP), and common spatial subspace decomposition (CSSD) [9, 10]. For example, Kim et al. implemented a Continuous Wavelet Transform (CWT) to raw EEG signals and computed the magnitude and phase features of the transformed signals. The obtained features were fed into a Convolutional Neural Network (CNN) model having one convolution layer, one max-pooling layer and one fully connected layer, and achieve a higher classification accuracy [11]. In order to further improve the classification accuracy, Zang et al. proposed a novel dynamic multi-scale network based on the Residual Network (ResNet). In this architecture, raw EEG signals were converted into 2-D images through a Short-time Fourier Transform (STFT) and fed into the ResNet. A new layer called dynamic multi-scale (DMS) is introduced in the RestNet to learn multi-scale features from different receptive fields at a granular level and enhance the classification accuracy [12]. A dual-stream convolutional neural network (DCNN) was suggested by Hung et al., which can employ both time-domain and frequency-domain signals as inputs. The extracted time-domain and frequency-domain features were fused by linear weighting to train a DCNN. The proposed technique was tested on the benchmark dataset and achieve a higher classification accuracy [13]. Although the performance of the MI-based BCI system is not up to the mark as those options in the literature. The lower performance of the models is due to the subject-specific EEG signals across the individual. To address this issue, in this paper, a new deep learning framework called Empirical Mode Decomposition Neural Network (EMDNet) is proposed to enhance the performance of the MI-based BCI system.

2. Materials and Methodology

2.1. Materials

To test the effectiveness of the proposed network, two MI-EEG datasets were used: one dataset is obtained from the BCI competition-II, Dataset-III, and another is an In-House Lab recorded dataset. The BCI completion-II dataset contains raw EEG signals from a healthy subject while imaging left and right-hand movements. On the other hand, the In-House Lab recorded dataset is the EEG signals recorded from 4 subjects (healthy subjects: 2 and paraplegic subjects: 2) as shown in Fig. 1.

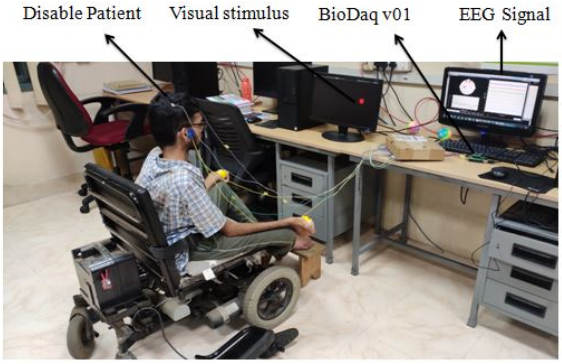


Fig. 1. EEG signals recorded from a paraplegic subject while imagining a ball grasping through left and right hands.

The experimental protocols and the time duration of both datasets are the same. In the experiment, the first seconds (t = 0-2 s) of each trial were silent or rest periods. After 2 s, a beep sound was provided, and a cross symbol was presented for 1s. At t = 3 s, a visual cue (random order) either pointing in the left or right direction was displayed. At the same instant, the subject was instructed to imagine either left hand or right-hand movements in the direction of a random visual cue until the arrow disappeared at t = 9 s. A break is provided to the subjects and after that, a new trial is started. The signals were recorded using three EEG channels (C3, C4 and Cz). The raw signals were band-pass filtered between 0.5 to 30 Hz and sampled at a rate of 128 Hz. The detailed description of the data recording and paradigms is explained in [14].

2.2. Methodology

The detailed description of the proposed technique is shown in Fig. 2. In this figure, raw EEG signals were extracted from both C3 and C4 channels while the subject imagined both left- and right-hand movements. The extracted signals were preprocessed and subjected to the Empircal Mode Decomposiion (EMD), and a set of Intrinsic Mode Functions (IMFs) was obtained. The significant IMFs, which are correlated with the hand movements were estimated from the fast Fourier transform (FFT), and the power spectral densities (PSDs) of the significant IMFs were computed. The significant IMFs were fed into the convolutional neural networks (CNN) for classifying both left hand (LH) and right hand (RH) imaginations.

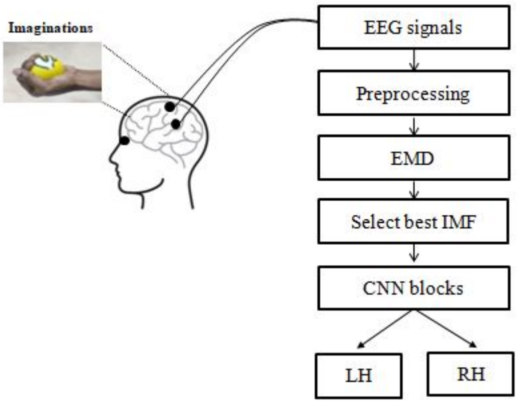


Fig. 2. Steps for classifying both left- and right-hand imaginations using the proposed technique.

2.2.1. Empirical Mode Decomposition

The EMD is a data-driven technique used to decompose any non-linear and nonstationary signal into amplitude modulation (AM) and frequency modulation (FM) components. These AM and FM components are known as Intrinsic Mode Function (IMF). The EMD generates a set of IMFs and each IMF has a distinct frequency band. The generated IMFs contain significant information about the signal and are used in many biomedical applications, such as the detection of seizures, cardiovascular diseases, etc. A detailed explanation of the EMD technique is presented in [15, 16].

2.2.2. Convolutional Neural Network

Convolutional Neural Networks (CNNs) are DL architectures commonly used for image or video classification [17], [18]. A standard CNN contains four processing modules: (a) Convolution blocks; (b) Activation functions; (c) sub-sampling (or pooling) operations; and (d) multi-layer fully connected (FC) neural networks. The convolutional layer consists of M filters that perform the dot product (i.e., convolution operation) with the input data. Each filter moves along the input with a specific step size (sharing the same weights), estimating M feature maps. The extracted feature maps are passed through the activation function specially Rectified Linear Unit (ReLu). The features extracted from the activation function are down-sampled through a max or average pooling layer, where a filter scans the input feature map and computes the maximum or mean of each sub-region under analysis. Finally, the network ends with one or more fully connected layers (as a standard multi-layer NN configuration) connected with a SoftMax activation function for classification or regression. The framework developed in this study is a combination of the EMD and 1-D CNN and is shown in Fig. 3.

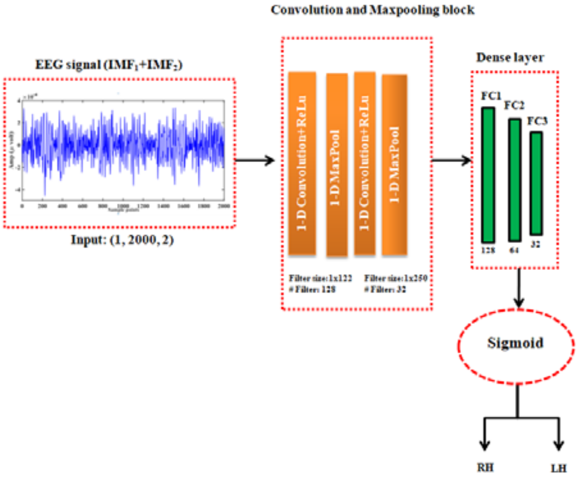


Fig. 3. Block diagram of the EMDNet for classifying left- and right-hand MI movements.

In this figure, raw EEG signals were decomposed into IMFs through the EMD technique, and the significant IMFs related to MI movements were identified. The significant IMFs were fed into 1D-CNN to produce high-level EEG features. Finally, the framework classifies both left and right-hand MI tasks of the subjects through the ”SoftMax” activation function. The input of the proposed framework is 122×2, where 122 is the number of temporal samples and 2 is the number of channels. The first 1-D CNN has 128 filters and the size of the kernel is 122. The activation function is set to ”ReLu” and padding is activated with 1 stride. Next, a 1-D Maxpool operation is performed with a kernel size of 4. The second 1-D CNN has 250 filters and the size of the kernel is 32. The activation function is set to ”ReLu,” and padding is activated with 1 stride. Next, a 1-D Maxpool operation is performed with a kernel size of 4. Next, it is followed by a flattened layer and three fully connected (FC) dense layers with neurons of 128, 64, and 32. In the end, the dense layer is connected with a “SoftMax” activation function to classify LH and RH imaginations. The hyperparameters of the model are represented in Table 1. The efficiency of the proposed EMDNet was evaluated by the classification accuracy (% CA), F1-score, Cohen’s Kappa coefficient (K) and area under the receiver operating characteristic (Auc).

Table 1. Parameters of the Proposed EMDNet.

Parameters	Type or value
Optimizer	Adam
Activation	ReLu, then, Softmax
Learning rate	0.001
Loss function	Binary cross-entropy
Batch size	64
Epoch	100

3. Results and Discussion

MATLAB (R-2016-a) and Python 2.7 (scikit learn-0.19.1) were employed for the preprocessing of the EEG signals and coding of the proposed framework. In order to remove power line interference, a 50 Hz notch filter is implemented on raw EEG signals. The decomposition of one trial EEG signal is shown in Fig. 4. The EMD generates sets of IMFs like IMF_1, IMF_2 and IMF_3 and are presented in Fig. 4(b), (c) and (d). Each IMF has a distinct frequency band and can be used for various biomedical applications. These IMFs are in the time-domain and in order to identify their frequency-domain, the PSD technique was applied to each IMF. Using the PSD technique, it is found that the IMF_2 has the best frequency band (12-30 Hz) to study the MI-based BCI. Fig. 5(a) and (b) shows the PSDs of IMF_2 in both C3 and C4 channels while the subject imagines both left- and right-hand movements. In Fig. 5(a), it is noticed that when the subject imagines left hand movement, the PSD of the C4 channel is lower than the C3 channel. Similarly, the PSD of the C3 channel is lower than the C4 channel, while the subject imagines right hand MI movements, as shown in Fig. 5(b).

The variation in PSD of IMF_2 can be clearly noticed in Fig. 5 in both C3 and C4 channels while the subject imagines both left- and right-hand MI movements, and therefore the IMF_2 signal can be used as an input to the 1-D CNN for classifying both left- and right-hand MI movements. The 1-D CNN model was implemented in “Google Colaboratory”, which is a free cloud service platform developed by Google. First, the IMF_2 dataset related to the left- and right-hand MI movements was extracted from the EMD and PSD techniques and uploaded to the “Google Colaboratory”. Next, the dataset was normalized, divided into timesteps and each part contained 125 samples and it was reshaped as (120, 2000, 2). Finally, the dataset is split randomly into the train and test sets, and the 1D CNN model is trained using train data and tested using test data. For each trial, a sliding window of size 250 moves along the time axis, and the model obtained from the training phase is tested section by section. Finally, the mean classification accuracy is obtained for the model with epochs of 400 and batch sizes of 64. The outcome of the proposed

EMDNet on both the benchmark and Lab recorded datasets is shown in Table 2.

The model metrics like classification accuracy (% CA), F1-score, Cohen's kappa coefficient (K) and area under the receiver operating characteristic (Auc) are calculated for both normal (N) and paraplegic (P) subjects. In case of the BCI competition-II dataset, the % CA, F1-score, K and Auc of the proposed model for subject N1 are found to be 95.30 %, 0.94, 0.91 and 0.97, respectively, and are shown in Table 2. The performance of the model for both normal and paraplegic subjects in Lab recorded dataset is presented in Table 2. From the table, it is observed that the highest and lowest performance of the model were found in subjects P1 and N2, respectively in Lab recorded dataset. The % CA, F1-score, K and Auc of the proposed model for subject P1 are 95.83 %, 0.94, 0.91 and 0.97, respectively. On the other hand, the % CA, F1-score, K and Auc of the proposed model for subject N2 are 87.50 %, 0.88, 0.75 and 0.91, respectively.

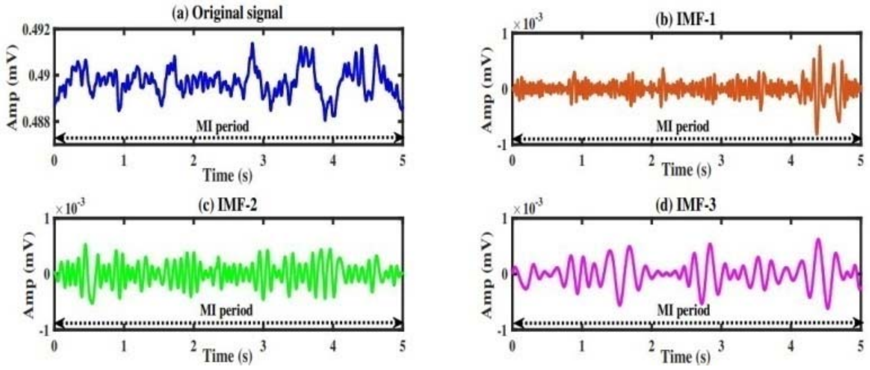


Fig. 4. (a) Empirical mode decomposition of EEG signal and its IMFs (b) IMF$_1$, (c) IMF$_2$, (d) IMF$_3$.

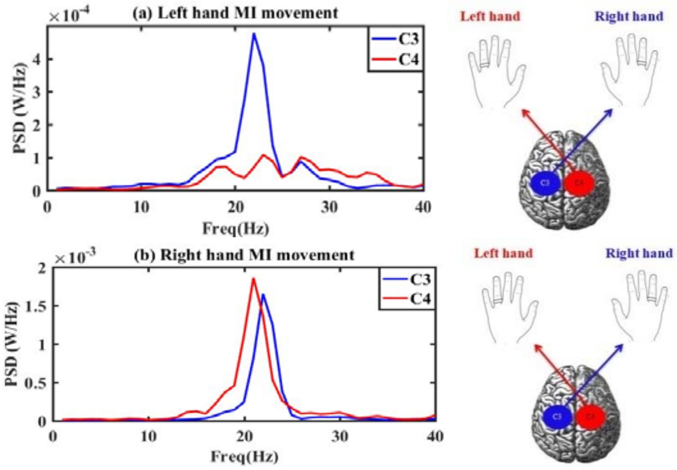


Fig. 5. PSD of IMF$_2$ in (a) left hand and (b) right hand MI movements.

Table 2. Performance of the proposed model on both benchmark and Lab recorded datasets.

Dataset	Subject	% CA	F1-score	K	Auc
BCI Comp-II	N1	95.30	0.94	0.91	0.97
Lab Recorded	N1	91.67	0.91	0.83	0.95
	N2	87.50	0.88	0.75	0.91
	P1	95.83	0.94	0.91	0.97
	P2	92.32	0.93	0.87	0.96

It is observed that the performance of two paraplegic subjects is superior to that of the two normal subjects in Table 2. The reason behind the higher performance of the paraplegic subjects is that they have a stronger imagination in comparison with normal subjects. It is assumed that they have imagined and practised moving their limbs many times, and thus they have a strong imagination. This is why they were highly alert during the experiment and properly imagined both left- and right-hand movements. Therefore, the proposed framework successfully detected their imaginations, and they had better results than the normal subjects. Table 3 shows the comparison between the proposed and state-of-the-art techniques on the same dataset (BCI competition-II, Dataset-III). The highest % CA reported in the literature is 90.71 % using the DCNN model. Using the same dataset, the % CA in the proposed EMDNet is found to be 95.30 %, which is higher than the state-of the-art techniques as presented in Table 3.

Additionally, the proposed framework was implemented on the Lab recorded dataset and found the mean % CA of four subjects to be 91.83 %. Table 3 shows that in both benchmark and Lab recorded datasets, the EMDNet framework performed well and had better results as compared to the conventional techniques.

Table 3. Comparison between the proposed technique and state-of-the-art techniques in terms of % CA.

Dataset	Proposed by	Classifier	% CA
	Kim et al. [11]	CNN	90.00 %
	Zang et al. [12]	ResNet	90.47 %
BCI-II	Huang et al. [13]	DCNN	90.71 %
	Proposed	**EMDNet**	**95.30 %**
Lab Recorded	**Proposed**	**EMDNet**	**91.83 %**

4. Conclusion

In this research work, the EMDNet framework is proposed to detect both left- and right-hand imaginations of the subjects. The empirical mode decomposition decomposes raw EEG signals into a set of IMFs, and each IMF has a distinct frequency band. Out of many IMFs, IMF_2 is found to be best IMF for studying the motor imagery-based BCI. The power spectral density of the IMF_2 changes in both C3 and C4 channels while the subjects imagine both left- and right-hand movements. The variation in power spectral density of IMF_2 reveals that the relationship between the brain activities and left and right-hand MI movements is contralateral. Therefore, the IMF_2 signal is fed into the 1-D CNNs in order to distinguish both left- and right-hand MI movements of the subjects. The outcomes show that the EMDNet is an effective DL framework with higher classification accuracy and out-performed the state-of-the-art techniques. The higher outcomes concluded that the combination of EMD and 1-D CNNs is a powerful technique to identify the imagination of the subjects.

References

[1]. J. R. Wolpaw, *et al.*, Brain-computer interface technology: A review of the first international meeting, *IEEE Trans. Rehabil. Eng.*, Vol. 8, Issue 2, 2000, pp. 164-173.

[2]. B. Blankertz, G. Dornhege, M. Krauledat, K. R. Müller, G. Curio, The non-invasive Berlin Brain-Computer Interface: Fast acquisition of effective performance in untrained subjects, *Neuroimage*, Vol. 37, Issue 2, 2007, pp. 539-550.

[3]. A. Nijholt, *et al.*, Brain-Computer Interfacing for Intelligent Systems, *IEEE Intell. Syst.*, Vol. 23, Issue 3, 2008, pp. 72-79.

[4]. S. Amiri, R. Fazel-Rezai, V. Asadpour, A Review of Hybrid Brain-Computer Interface Systems, *Adv. Human-Computer Interact.*, Vol. 2013, 2013, 187024.

[5]. J. Vidal, Toward direct brain-computer communication, in Annual Review of Biophysics and Bioengineering, *Annu. Rev. Inc*, Vol. 2, 1973, pp. 157-180.

[6]. G. Pfurtscheller, A. Aranibar, Evaluation of event-related desynchronization (ERD) preceding and following voluntary self-paced movement, *Electroencephalogr. Clin. Neurophysiol.*, Vol. 46, Issue 2, 1979, pp. 138-146.

[7]. G. Pfurtscheller, Event-related synchronization (ERS): an electrophysiological correlate of cortical areas at rest, *Electroencephalogr. Clin. Neurophysiol.*, Vol. 83, Issue 1, 1992, pp. 62-69.

[8]. G. Pfurtscheller, F. H. Lopes Da Silva, Event-related EEG/MEG synchronization and desynchronization: Basic principles, *Clin. Neurophysiol.*, Vol. 110, Issue 11, 1999, pp. 1842-1857.

[9]. F. Shahlaei, N. Bagh, A. D. Shaligram, M. R. Reddy, M. S. Zambare, Classification of Motor Imagery Tasks Using Inter Trial Variance in the Brain Computer Interface, in *Proceedings of the IEEE International Symposium on Medical Measurements and Applications (MeMeA'18)*, 2018, pp. 1-6.

[10]. N. Bagh, M. R. Reddy, Hilbert transform-based event-related patterns for motor imagery brain computer interface, *Biomed. Signal Process. Control*, Vol. 62, 2020, 102020.

[11]. J. Kim, Y. Park, W. Chung, Transform based feature construction utilizing magnitude and phase for convolutional neural network in EEG signal classification, in *Proceedings of the 8th International Winter Conference on Brain-Computer Interface (BCI'20)*, 2020, pp. 1-4.

[12]. G. Zhang, *et al.*, A Dynamic Multi-Scale Network for EEG Signal Classification, *Front. Neurosci.*, Vol. 14, 2021.

[13]. E. Huang, X. Zheng, Y. Fang, Z. Zhang, Classification of Motor Imagery EEG Based on Time-Domain and Frequency-Domain Dual-Stream Convolutional Neural Network, *IRBM*, Vol. 43, Issue 2, 2022, pp. 107-113.

[14]. B. Blankertz, *et al.*, The BCI competition 2003: Progress and perspectives in detection and discrimination of EEG single trials, *IEEE Trans. Biomed. Eng.*, Vol. 51, Issue 6, 2004, pp. 1044-1051.

[15]. P. Flandrin, G. Rilling, P. Gonçalvés, Empirical mode decomposition as a filter bank, *IEEE Signal Process. Lett.*, Vol. 11, Issue 2, 2004, pp. 112-114.

[16]. N. E. Huang, *et al.*, The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis, *Proc. R. Soc. A*, Vol. 454, Issue 1971, 1996, pp. 903-995.

[17]. H.-J. Yoo, Deep Convolution Neural Networks in Computer Vision: a Review, *IEIE Trans. Smart Process. Comput.*, Vol. 4, Issue 1, 2015, pp. 35-43.

[18]. N. Y. Hammerla, S. Halloran, T. Plötz, Deep, convolutional, and recurrent models for human activity recognition using wearables, in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI'16)*, 2016, pp. 1533-1540.

(088)

Estimation of Height of a Shape a 2D Image from its Shadow using Neural Networks

Julián René Muñoz Burbano [1], **Pablo Emilio Jojoa Gómez** [2] **and Fausto Miguel Castro** [2]
[1] Corporación Universitaria Comfacauca Unicomfacauca, Group MIND.
Centro Histórico. Popayán, Colombia
Email: jburbano@unicomfacauca.edu.co
[2] Universidad del Cauca, Telecommunications Group I+D-GNTT,
Campus Universitario Tulcán. Popayán, Colombia
Tel: (57) 3108987728, (57) 3137063049
Emails: pjojoa@unicauca.edu.co, faustocastro@unicauca.edu.co

Summary: Determining the height of objects in a 2D image from their shadow and developing a model capable of estimating the height of uniform geometric shapes and objects of different sizes from the shadow they cast, is the fundamental part of this research, which proposes the use of convolutional neural networks (CNN) as a Machine Learning (ML) technique for pattern detection, feature extraction from shadows and shapes in images. For this purpose, a dataset was constructed with photographic images of shapes or objects, as well as their shadows cast from different angles and locations. The structure of the proposed dataset is characterized by the name of the shape, the name of the shadow, the length of the shadow, the height of the shape and the angle of the light source, which together allow to improve the accuracy of the model. In this sense, the research focuses on the analysis of lights and shadows of different geometric shapes or objects within a 2D image, where the projected shadow is the information to be used and with which it is intended to determine the height of the shapes or objects. The main reason for this research is oriented towards people who have visual impairment either total or partial, and that from the touch pretend to define or differentiate an object, becoming impossible if asked to indicate which of these are present in a 2D image (photograph or painting). Therefore, it is essential that from a 2D image is to highlight the objects to be represented in a 2.5 model, where the height of the selected objects will be the key to create the model. In this sense the project to be developed seeks to determine the height of predefined objects in a two-dimensional image from its shadow.

Keywords: Shadows, Shapes, Convolutional neural network, Dataset, 2D, Height.

---

1. Introduction

This Estimating the height of shapes or objects in a 2D image from their shadows is a challenging process when they are complex and irregular under uneven lighting conditions. This has presented its own challenges, such as the need for an accurate model that estimates the geometry of the object and the way light strikes it. Thus, the height of an object can provide relevant information about its shape, size and position. However, estimating the height of an object or shape in a 2D image from its shadow is a more complex problem, due to the lack of three-dimensional information about it. In this context, the use of neural networks has become a valuable tool to address this problem. In this research, a neural network-based solution for height estimation from its shadow is presented. This model uses a convolutional neural network architecture to extract relevant features from the image to perform height estimation.

To train and evaluate the model, a dataset was developed with photographs of different geometric shapes indicating a variety of shapes and shadows under different lighting conditions. The proposed model can accurately estimate the height of shapes under certain conditions, suggesting that it could offer a promising solution for estimating the height of shapes in 2D images from their shadow, which could have important applications in various fields, having the potential to significantly improve machine perception capabilities in various applications. The work will focus on evaluating the accuracy and efficiency of the proposed method, and comparing it with other existing algorithms for estimating the height of objects from their shadow.

The network is trained on a labeled dataset consisting of 2D images, their corresponding heights and their shadows, using the error backpropagation algorithm to minimize the difference between the estimated heights and the actual object heights. The approach proposed here was evaluated on a test data set and significant accuracy in estimating object heights was obtained compared to other existing methods. In addition, a sensitivity analysis was performed to evaluate the influence of different model parameters on the accuracy of height estimation. This approach is especially useful in situations where the image quality is low or there are ambiguities in the shape of objects. That is, the integration of cast shadow information into a learning model can be a promising approach, to solve the problem of estimating the height of a shape from its shadow in a 2D image.

By using these deep learning techniques, such as neural networks, large amounts of data can be processed and analyzed, allowing for greater accuracy in estimating the height of a shape from its shadow. In addition, the ability to integrate new geometric shapes and shadows within the set of images used in the learning model makes this technique highly adaptable and scalable for future research in which it is implemented with irregular shapes.

Similarly, in [9] (S. Mohajerani, P. Saeedi. 2018) addressed their research on automatic detection in an image of shadows using a segmentation method and the use of Deep Learning, where what they do is to identify the regions of shadows at the pixel level in an RGB image, whereby they manage to extract the features of the shadows, providing a Convolutional Neural Network (CNN) with knowledge to detect patterns of the shadows locally and globally and prior information of the illumination source and the dynamics of the objects in the image.

On the other hand [25] (Tao, M. W., Srinivasan, P. P., Hadap, et al. 2017) proposes an algorithm based on dense depth estimation, which combines blur and matching metrics. Furthermore, it defines an optimization framework that integrates photo coherence, depth coherence and shading coherence, for light field depth estimation in different scenarios. This algorithm incorporates the blur, correspondence and shading signals outperforming other more advanced algorithms.

1.1. The Shadow

Shadows can be defined as parts of the scene that are not directly illuminated by a light source due to an obstructing object or objects. A similar definition is posited in shadow detection and they likewise posit a typical shadow that could be divided into two different types. One type is denoted as self-shadow where the shadow region is on the object itself. The other type is the cast shadow for which the shadow region is on the background or on other objects. Here the cast shadow is usually divided into two parts, umbra and penumbra (Fig. 1).

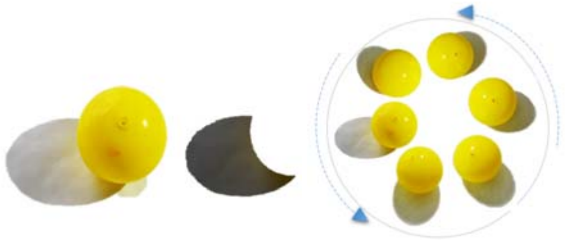


Fig. 1. Figure and corresponding shadow and Rotation of the light source.

In such a way that in [4] (Vicente, Samaras. 2014) in 2D digital images, different scenes can be found, which contain elements such as textures, edges, shapes, colors, shadows, etc., and shadows from which relevant information can be extracted and used in investigations by implementing shadow detection algorithms. This information is used to establish the relationship between the geometry of the object, the light source and the shadow area (Vicente, Samaras 2014).

Similarly, in [8], [6], (Kriegman, Belhumeur. 1998) and [11] (D. C. Knill, et al. 1997) it is also understood that shadows are a source of relevant information at the level of shapes of surfaces or objects, allowing to locate areas of interest in an image, direction of the illumination source, geometry of the shape, among other characteristics. Likewise, in [11] (Knill, Mamassian and Kersten. 1997) and in [17] (S. A. Shafer, T. Kanade. 1983), their research indicates that these geometric properties of shadows are of special interest, because they allow establishing perceivable relationships between shapes, shadows and illumination, according to the structure and height of the surface.

In many previous works in [24] (Hintze, Morse. 2019), the information that is extracted from the shadow, by means of techniques, methods and algorithms with purposes such as: determining the location of the shape or object, area of interest, direction of the illumination source among others, is usually oriented to its later elimination, where this makes its transcendence not so remarkable.

In [12] (Salvador, Cavallaro and Ebrahimi. 2004) shadows in a 2D image have relevant information about the scene, the location of the shape or object, the characteristics of the surface and the light source are obtained. He mentions techniques that are based on models that represent knowledge of the geometry of the scene, the objects or shapes, the light source, and techniques based on properties that identify shadows by using features such as geometry, brightness and color of the shadows. (See Fig. 2).

This provides several important elements where the shadow is one of the essential parts of the research, because it allows inferring shapes of objects, and when present in an image provides information that can be used to determine the shapes and orientations of objects.

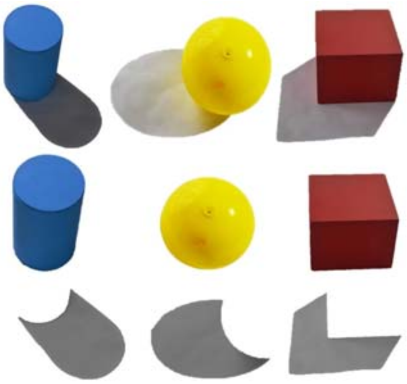


Fig. 2. Types of dataset figures.

Similarly, in [12] (Salvador, Cavallaro and Ebrahimi. 2004), address general aspects related to digital image processing, using convolutional neural networks (CNN's) with different data sets (datasets), where the latter have as main feature, images of geometric primitives and the shadows they cast according to an illumination source in a specific location. Similarly, in [2] (Panagopoulos, Hadap, Samaras and Dimitris. 2013) in their research employ synthetic images divided into small rectangular regions, and therefore, for each of these it is possible to capture what corresponds to the distribution of intensities and the ability to handle surfaces. At this point the geometry sections are combined, the intensity distribution is integrated into a dictionary that with all the sections can generate a hypothesis about that shape.

Using the learning model that processes the images of the training set, which with all the sections or small regions form a large dictionary, and where the decomposition of an image in shadow and reflectance images describe the perception of brightness for these images.

In [1], [13] (Hosseinzadeh, S., Shakeri, M. and Zhang, H. 2005) it is shown that the concept of sections corresponds to a collage of sections with a different reflectance and which are illuminated with an illumination source that varies slowly in intensity, and with shapes that are of a constant scale factor. This learning model with the data set to be generated would be used to determine the surface height and 3D generation of the object or shape, initially taking the information of the shadow it casts on a 2D image. In [12] (Salvador, Cavallaro, Ebrahimi, Touradj. 2004), [7] (G. Liasis, S. Stavrou. 2016) illumination, shadows and reflectance are an integral part of the different scenes, providing relevant information about the shape and appearance of the object.

In [2] (Panagopoulos, Hadap, and Samaras. 2013) propose the use of a dictionary use of a dictionary (Dictionary) that is confirmed by different patches (Patches) of geometry associated to the image, to a distribution of pixel intensities that can generate that geometry and with the use of larger image regions that allow capturing appearance data, it is possible with a data set and supported in neural networks it is possible to design a model that fits the proposed research.

In [22] (Varol, Shaji, Salzmann, Fua. 2011) has an approach based on deformable shape recovery that can work under complex illumination and on partially textured surfaces. They used an algorithm that performs a learned mapping of the intensity distribution and shape of local surface patches, which is focused on 3D surface shape estimation.

In [14] (D. S. Kim, M. Arsalan, K. R. Park. 2018) they employ a Convolutional Neural Network (CNN) for shadow detection using cameras and using the open database CAVIAR (Context-Aware Vision Using Image-Based Active Recognition). They indicate that in their research it is not computationally expensive both in training and validation and can be adapted to tasks such as image segmentation and has a level of accuracy of other methods.

1.2. The Dataset

The experimentation process is carried out under controlled conditions in terms of illumination, shapes and surface. Such conditions were used in the model allowing initially to test the model of the network and that it learned certain characteristics, which is why it was decided to use the different photographs of three geometric shapes on a uniform surface, with a dimmable LED lamp and the shadows corresponding to each shape (Fig. 3).

It is important to mention that the data set was oriented with the objective of capturing photographs under different lighting conditions to give an initial knowledge to the neural network.

The data set includes 800x600 color photographs (RBG) of various geometric shapes of various colors, taken from a distance of 90 cm from the edge of the surface. The surface where the shape or object is located is white, smooth and does not reflect light and is made of white matte adhesive vinyl. The surface has a size of 90 cm and a diameter of 180 cm, was designed circular to facilitate the displacement of the LED lamp, is made of MDF (Medium Density Board) wood of 9 mm (Fig. 4).

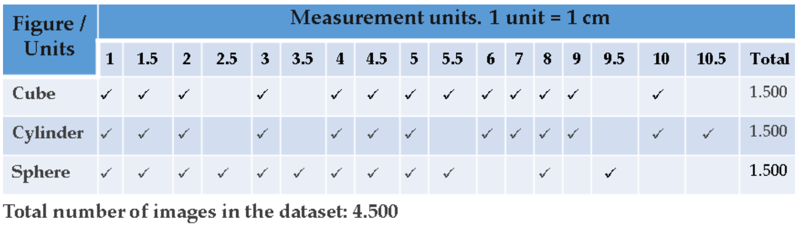


Figure / Units	Measurement units. 1 unit = 1 cm																	
	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	7	8	9	9.5	10	10.5	Total
Cube	✓	✓	✓		✓		✓	✓	✓	✓	✓	✓	✓	✓		✓		1.500
Cylinder	✓	✓	✓		✓		✓	✓	✓		✓	✓	✓	✓		✓	✓	1.500
Sphere	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓		✓			1.500

Total number of images in the dataset: 4.500

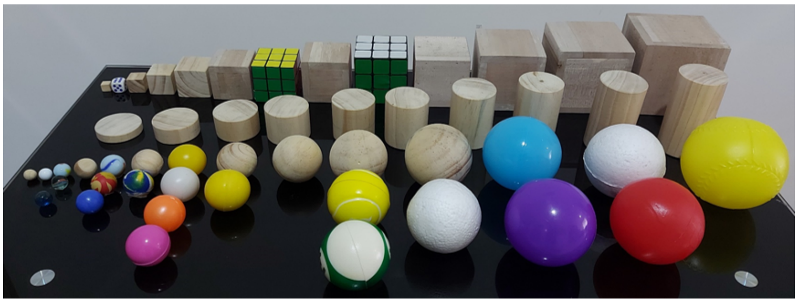


Fig. 3. Characteristics of the Dataset.

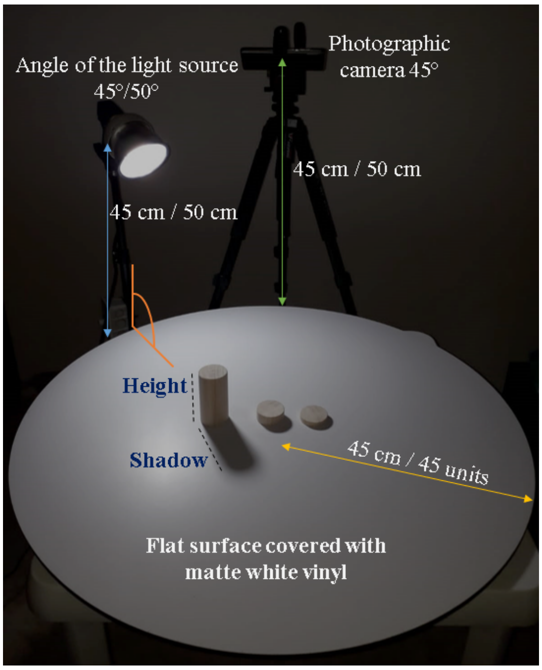
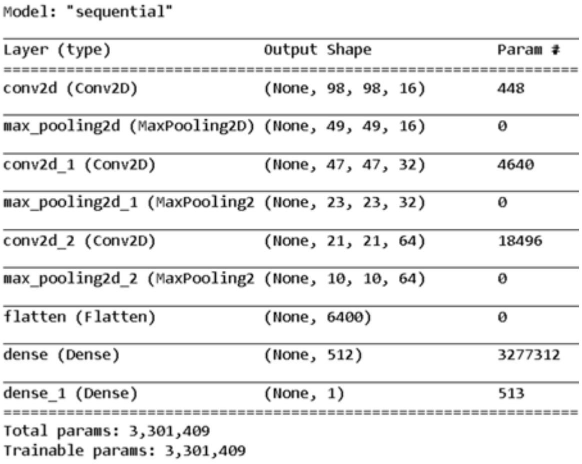


Fig. 4. Mock-up for taking photographs.

1.3. Model Description

With the research conducted, it is proposed to use and/or design a convolutional neural network, which through the training of a set of data will provide the necessary knowledge to predict the height of an object or shape with the projected shadow. With the learning process, it is expected that the network, by using as input information the shadow cast by the shape or object in a 2D image, will predict or determine the height of the object in question. The data set (Dataset) was designed and developed with objects and/or basic geometric shapes (sphere, cube, cylinder) to perform the experimentation in a controlled manner, since after this proposal and as a future work it is expected to increase the data set with more complex shapes and shadows. The total amount of images in the dataset is approximately 4500 color photographs and for each shape there are 1500 photographs. (See Table 1. Height Distribution of Geometric Shapes).

It is important to note that the proposed model (Figs. 5, Fig. 6) uses geometry of three-dimensional shapes of the objects or shapes with the corresponding shadow from different locations in the plane and generated by a single light source. The latter is rotated around the figure to obtain shadows from different angles. The light source is located 90 centimeters from the center of the object to the base of the lamp and the heights of the lamp vary from 45 centimeters (cm) to 50 centimeters (cm) so that the different shadows are observable on the surface under varying intensities of illumination (See Fig. 1, Rotation of the light source).



```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 98, 98, 16)        448
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 49, 49, 16)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 47, 47, 32)        4640
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 23, 23, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 21, 21, 64)        18496
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 10, 10, 64)        0
_________________________________________________________________
flatten (Flatten)            (None, 6400)              0
_________________________________________________________________
dense (Dense)                (None, 512)               3277312
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 513
=================================================================
Total params: 3,301,409
Trainable params: 3,301,409
```

Fig. 5. Model Architecture.

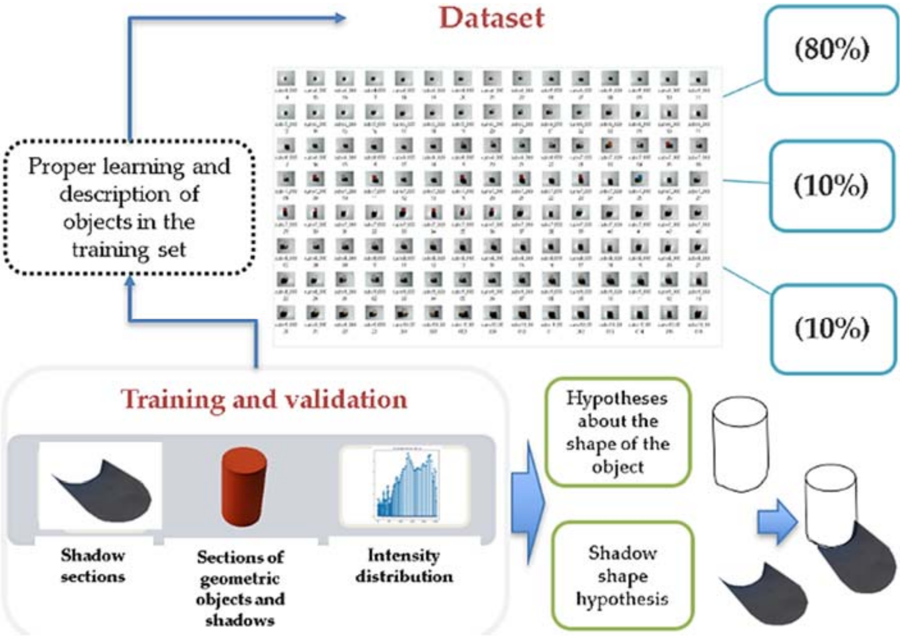


Fig. 6. General Scheme of the Proposed Model.

Table 1. Height Distribution of Geometric Shapes.

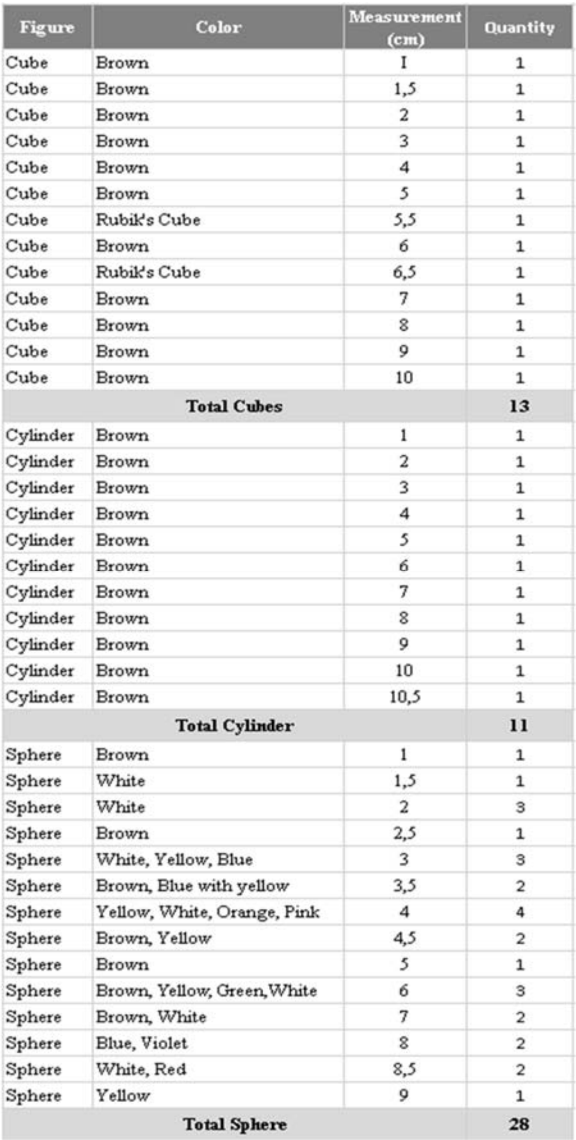


Figure	Color	Measurement (cm)	Quantity
Cube	Brown	I	1
Cube	Brown	1,5	1
Cube	Brown	2	1
Cube	Brown	3	1
Cube	Brown	4	1
Cube	Brown	5	1
Cube	Rubik's Cube	5,5	1
Cube	Brown	6	1
Cube	Rubik's Cube	6,5	1
Cube	Brown	7	1
Cube	Brown	8	1
Cube	Brown	9	1
Cube	Brown	10	1
Total Cubes			**13**
Cylinder	Brown	1	1
Cylinder	Brown	2	1
Cylinder	Brown	3	1
Cylinder	Brown	4	1
Cylinder	Brown	5	1
Cylinder	Brown	6	1
Cylinder	Brown	7	1
Cylinder	Brown	8	1
Cylinder	Brown	9	1
Cylinder	Brown	10	1
Cylinder	Brown	10,5	1
Total Cylinder			**11**
Sphere	Brown	1	1
Sphere	White	1,5	1
Sphere	White	2	3
Sphere	Brown	2,5	1
Sphere	White, Yellow, Blue	3	3
Sphere	Brown, Blue with yellow	3,5	2
Sphere	Yellow, White, Orange, Pink	4	4
Sphere	Brown, Yellow	4,5	2
Sphere	Brown	5	1
Sphere	Brown, Yellow, Green,White	6	3
Sphere	Brown, White	7	2
Sphere	Blue, Violet	8	2
Sphere	White, Red	8,5	2
Sphere	Yellow	9	1
Total Sphere			**28**

With this knowledge it is expected that the convolutional neural network will be able to predict the height of the shapes from their shadow. This height measurement is given in centimeters (units) using as input image only the projected shadow. Thus, it is initially expected to effectively estimate the height of the shape or object corresponding to the shadow, which is one of the challenges of the research. Moreover, there is additional information in the dataset that combined with appropriate 2D to 3D shape reconstruction algorithms it is possible to obtain the reconstruction of the shape according to the height established from the shadow.

In [21] (M. Daum, G. Dudek. 1998) the reconstruction of 3D surfaces from shadows is reported. Here, the movement of shadows that are cast is used to obtain information about the scene structure, based on the collection of images from a fixed point as an illumination source moves. As a result, a 3D scene reconstruction algorithm is obtained taking into account the trajectories of the light source.

On the other hand, in [26] (Huang, X., Gao, J., et al. 2007) their research addresses shape estimation from shading (SFS) which aim to solve a problem with few constraints to estimate the depth map from a single image, developing an example-based method to improve the accuracy of (SFS), improving the reconstruction quality from real images of different shapes by obtaining prior knowledge of their appearance for three-dimensional shape recognition.

Similarly in the proposal in [20] (E. Prados, O. Faugeras. 2006) with the shape from shading (SFS) and with a single image they estimate the 3D shape of a surface, where in diffuse surface the position of the illumination source is not known and where the reflectance map is not known.

In [15] (S. Savarese, et al. 2007), [16] (S. Savarese, et al. 2002) and [5] (R. Gouiaa, J. Meunier. 2014) they propose a new method to recover a shape from shadow (SFS). These projected shadows are relevant information about the shape of objects, discovering cavities that are not available from clues such as occluded boundaries. This method is called "occluding boundaries". According to the volume occupied by an object it is possible to identify and sculpt the regions of the volume that present inconsistencies with the pattern observed in the shadows. In summary, their proposal is a reconstruction system to recover the shape from silhouettes and shadow carving where shadow carving is used to carve the concavities and silhouettes are used to reconstruct the initial conservative estimate of the shape of the object.

On the other hand, in [3] (D. Forsyth, A. Zisserman. 1991), he shows how mutual illumination and image irradiance within an image give rise to more complex structures.

Similarly, in [27] (Atkinson, G. A., & Hancock, E. R. 2007), research presents a novel method for the reconstruction of 3D surfaces with polarization and shading information from two views. They use Fresnel theory in image processing to obtain estimates of the surface normal. Here they emphasize how the measured pixel brightnesses depend on the surface orientation combined with the refined estimates to determine the correspondence between two views of an object. In [19] (R. Liu, S. Menon, et al. 2022) they address 3D reconstruction when the object structure is partially or totally occluded. They introduce into the method projected shadows of an unobserved object, to perform the inference of the possible 3D volume to which it corresponds. The "inheritable" imaging model allows to jointly infer both the 3D shape of the object, its location and that of the illumination source.

Likewise, in [23] (D. Samaras, D. Metaxas. 2003) they present a method for the integration of nonlinear holonomic constraints in deformable models and its application to the problems of shape and direction estimation of the illumination source from hats. Their proposal indicates that it works when the direction of the light source is not known. They coupled the shape estimation method with a light estimation method where better shape estimation results in better light estimation and vice versa.

In [19] (R. Liu, S. Menon, et al. 2022) they address 3D reconstruction when the object structure is partially or totally occluded. They introduce into the method projected shadows of an unobserved object, to perform the inference of the possible 3D volume to which it corresponds. The "inheritable" imaging model allows to jointly infer both the 3D shape of the object, its location and that of the illumination source. The advantage of this approach is the estimation of multiple 3D scenes from shadows.

The data set is organized in folders and within each subfolder there are images corresponding to each geometric shape, the shadow, the height. It is important to point out that with the experimentation process, the length of the shadow and the angle of the light source were added to the Dataset already created, improving the accuracy of the model. (See Fig. 8. Dataset Folder Structure).

Some research such as in [10] (Sanin, C. Sanderson, B. C. Lovell. 2012) presents various methods and algorithms for detection, removal of moving shadows as of 2012, and places them in a feature-based taxonomy that is made up of four categories, chromaticity, physics, geometry, and textures. The use of tracking performance is proposed as an approach to determine shadow detection methods and algorithms. Of the most prominent methods are the method based on chromaticity is the fastest to implement and execute, but sensitive to noise and fails when the spectral properties of objects when these are similar to those of the background and the method based on textures of large and small regions where the results are more accurate, although its computational cost is high.

Likewise, in [18] (S. H. Khan, M. Bennamoun et al. 2014) they use multiple deep convolutional neural networks (ConvNets), where their architecture consists of alternating convolution and subsampling layers. This research is focused on shadow detection, having as input a database containing in various lighting conditions with sunny, cloudy and dark environments. For this they used a Conditional Random Field (CRF) model, which translates into predictions for each pixel of each test image, and are subsequently compared with the actual shadow masks. This enforces labeling consistency across the nodes of a gridded graph defined on the image by eliminating isolated labeling results.

3. Distribution and Structure of the Dataset

For the organization of the dataset, these were grouped according to the height measurements and their geometric shape. The dataset is made up of geometric figures such as spheres, cubes and cylinders of different heights.

The measurements were taken in centimeters, which are then handled in units, in order to facilitate the reuse of the dataset in any other measurement system. (See Fig. 7).

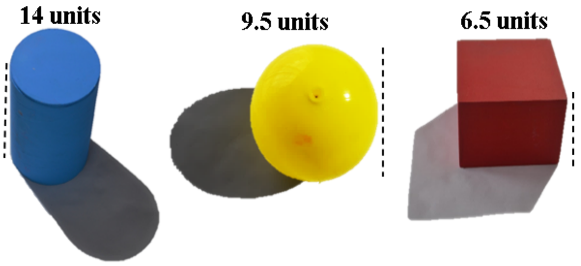
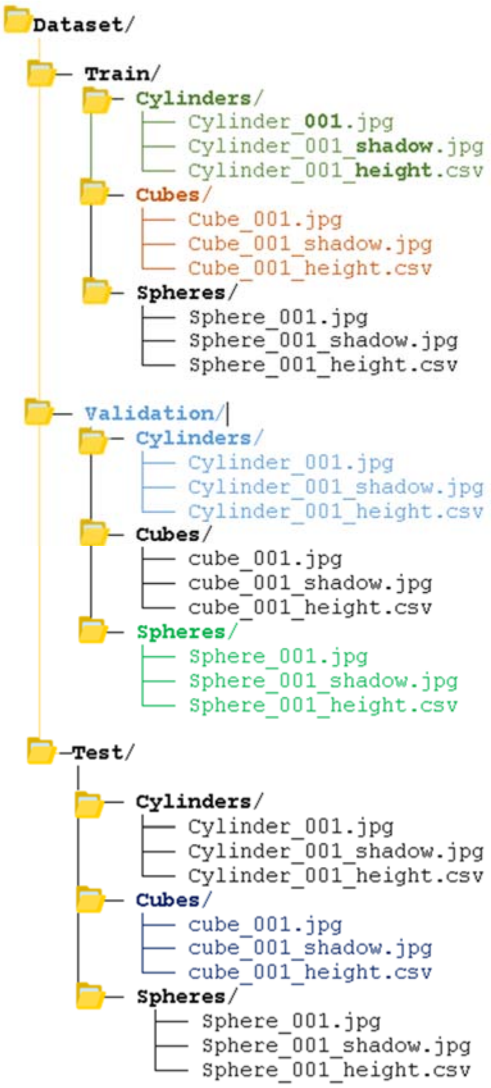


Fig. 7. Height Cylinder, Sphere and Cube.

Initially two variables (features) were defined in the dataset, the first being the type of figure or shape and the second the height of the 3D object. Subsequently, three more variables are added to the existing ones, these are the shadow, the shadow length and the angle of the light source. Therefore, a total of five more variables are included and implemented in the training, validation and test set. These variables are within the model and were used in the learning process to improve the accuracy of the model.

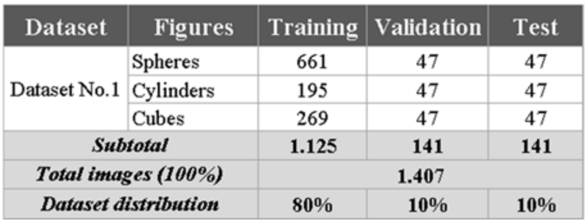


```
Dataset/
  ├─ Train/
  │   ├─ Cylinders/
  │   │   ├── Cylinder_001.jpg
  │   │   ├── Cylinder_001_shadow.jpg
  │   │   └── Cylinder_001_height.csv
  │   ├─ Cubes/
  │   │   ├── Cube_001.jpg
  │   │   ├── Cube_001_shadow.jpg
  │   │   └── Cube_001_height.csv
  │   └─ Spheres/
  │       ├── Sphere_001.jpg
  │       ├── Sphere_001_shadow.jpg
  │       └── Sphere_001_height.csv
  ├─ Validation/
  │   ├─ Cylinders/
  │   │   ├── Cylinder_001.jpg
  │   │   ├── Cylinder_001_shadow.jpg
  │   │   └── Cylinder_001_height.csv
  │   ├─ Cubes/
  │   │   ├── cube_001.jpg
  │   │   ├── cube_001_shadow.jpg
  │   │   └── cube_001_height.csv
  │   └─ Spheres/
  │       ├── Sphere_001.jpg
  │       ├── Sphere_001_shadow.jpg
  │       └── Sphere_001_height.csv
  └─Test/
      ├─ Cylinders/
      │   ├── Cylinder_001.jpg
      │   ├── Cylinder_001_shadow.jpg
      │   └── Cylinder_001_height.csv
      ├─ Cubes/
      │   ├── cube_001.jpg
      │   ├── cube_001_shadow.jpg
      │   └── cube_001_height.csv
      └─ Spheres/
          ├── Sphere_001.jpg
          ├── Sphere_001_shadow.jpg
          └── Sphere_001_height.csv
```

Fig. 8. Dataset Folder Structure.

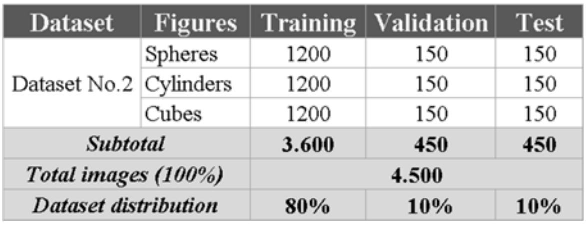
At the time of performing the network learning process, it is taken into account that it is distributed or divided into a training set (to build the model), a validation set (to test different parameters of the learning algorithm) and a test set (to evaluate the quality of the model). A suggested distribution of the data could be as follows, 80 % for the training set, 10 % for the validation set and 10 % for the test set. (Tables 2 and 3). Similarly, it could also be 70 %, 15 % and 15 % respectively.

Table 2. Dataset Distribution 1.



Dataset	Figures	Training	Validation	Test
Dataset No.1	Spheres	661	47	47
	Cylinders	195	47	47
	Cubes	269	47	47
Subtotal		**1.125**	**141**	**141**
Total images (100%)		**1.407**		
Dataset distribution		**80%**	**10%**	**10%**

Table 3. Dataset Distribution 2.



Dataset	Figures	Training	Validation	Test
Dataset No.2	Spheres	1200	150	150
	Cylinders	1200	150	150
	Cubes	1200	150	150
Subtotal		**3.600**	**450**	**450**
Total images (100%)		**4.500**		
Dataset distribution		**80%**	**10%**	**10%**

4. Conclusions

Regarding the scope and challenges of this research, one can indicate that related to the interpretation of 2D images by visually impaired people, since they cannot distinguish the objects in them from touch. However, 3D objects can be useful to provide a more complete understanding of the image. This aims to contribute to improve the accessibility and inclusion of visually impaired people in the interpretation of images and their environment. In this context, the present project aims to determine the height of predefined objects in a 2D image from their shadow. This method can be especially useful, allowing them to understand the visual information in an image. To achieve this goal, Machine Learning techniques are used, specifically convolutional neural networks (CNNs) that allow to detect relevant patterns and features in images. (See Fig. 5). Therefore, a dataset will be created to obtain images of objects and their shadows, which will be used to train, validate and test the convolutional neural network (CNN).

Experimental results show that the proposed method achieves significant accuracy in estimating the height of shapes from their shadows compared to other traditional computer vision methods. Additionally, the construction of the dataset is also an important contribution, since there are not many Datasets available that meet the specific requirements of this project. All in all, it is hoped that this research can be useful for future work.

References

[1]. S. Hosseinzadeh, M. Shakeri, H. Zhang, Fast Shadow Detection from a Single Image Using a Patched Convolutional Neural Network, *Natural Science and Engineering Research Council (NSERC) through the NSERC Canadian Field Robotics Network (NCFRN) and by Alberta Innovates Technology Future (AITF)*, 2018, pp. 3124-3129.

[2]. A. Panagopoulos, S. Hadap, D. Samaras, Reconstructing shape from dictionaries of shading primitives, *Lecture Notes in Computer Science*, Vol. 7727, 2013, pp. 80-94.

[3]. D. Forsyth, A. Zisserman, Reflections on Shading, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, Issue 7, 1991, pp. 671-679.

[4]. T. F. Y. Vicente, C.-P. Yu, D. Samaras, Single image shadow detection using multiple cues in a supermodular MRF, in *Proceedings of the British Machine Vision Conference*, 2013, pp. 126.1-126.11.

[5]. R. Gouiaa, J. Meunier, 3D reconstruction by fusioning shadow and silhouette information, in *Proceedings of the IEEE Conference on Computer and Robot Vision* (*CRV'14)*, 2014, pp. 378-384.

[6]. D. J. Kriegman, et al., 3D photography using shadows in dual-space geometry, *Lecture Notes in Computer Science*, Vol. 1407, March 1999, pp. 399-414.

[7]. G. Liasis, S. Stavrou, Satellite images analysis for shadow detection and building height estimation, *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 119, 2016, pp. 437-450.

[8]. D. J. Kriegman, P. N. Belhumeur, What shadows reveal about object structure, *Lecture Notes in Computer Science*, Vol. 1407, March 1998, pp. 399-414.

[9]. S. Mohajerani, P. Saeedi, CPNet: A context preserver convolutional neural network for detecting shadows in single RGB images, in *Proceedings of the IEEE 20th International Workshop on Multimedia Signal Processing (MMSP'18)*, 2018, pp. 1-5.

[10]. A. Sanin, C. Sanderson, B. C. Lovell, Shadow detection: A survey and comparative evaluation of recent methods, *Pattern Recognition*, Vol. 45, Issue 4, 2012, pp. 1684-1695.

[11]. D. C. Knill, P. Mamassian, D. Kersten, Geometry of shadows, *Journal of the Optical Society of America A*, Vol. 14, Issue 12, 1997, 3216.

[12]. E. Salvador, A. Cavallaro, T. Ebrahimi, Cast shadow segmentation using invariant color features, *Computer Vision and Image Understanding*, Vol. 95, Issue 2, 2004, pp. 238-259.

[13]. S. Hosseinzadeh, M. Shakeri, H. Zhang, Fast shadow detection from a single image using a patched convolutional neural network, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'18)*, 2018, pp. 3124-3129.

[14]. D. S. Kim, M. Arsalan, K. R. Park, Convolutional neural network-based shadow detection in images using visible light, *Sensors*, Vol. 18, Issue 4, March 2018, 960.

[15]. S. Savarese, et al., 3D reconstruction by shadow carving: Theory and practical evaluation, *International Journal of Computer Vision*, Vol. 71, Issue 3, 2007, pp. 305-306.

[16]. S. Savarese, H. E. Rushmeier, Implementation of a shadow carving system for shape capture, in *Proceedings of the First International Symposium on 3D Data Processing Visualization and Transmission*, June 2002, pp. 12-23.

[17]. S. A. Shafer, T. Kanade, Using shadows in finding surface orientations, *Computer Vision, Graphics and Image Processing*, Vol. 22, Issue 1, 1983, pp. 145-176.

[18]. S. H. Khan, M. Bennamoun, F. Sohel, R. Togneri, Automatic feature learning for robust shadow detection, in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'14)*, September 2014, pp. 1939-1946.

[19]. R. Liu, S. Menon, C. Mao, D. Park, S. Stent, C. Vondrick, Shadows shed light on 3D objects, *arXiv:2206.08990*.

[20]. E. Prados, O. Faugeras, Shape from shading, in Handbook of mathematical models in computer vision, Nikos Paragios, Yunmei Chen, Olivier Faugeras (Eds.), *Springer*, May 2006, pp. 375-388.

[21]. M. Daum, G. Dudek, On 3-D surface reconstruction using shape from shadows, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'98)*, 1998, pp. 461-468.

[22]. A. Varol, A. Shaji, M. Salzmann, P. Fua, Monocular 3D reconstruction of locally textured surfaces, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 34, Issue 6, 2012, pp. 1118-1130.

[23]. D. Samaras, D. Metaxas, Incorporating illumination constraints in deformable models for shape from shading and light direction estimation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, Issue 2, 2003, pp. 247-264.

[24]. Ryan Hintze, Bryan Morse, Shadow patching: guided image completion for shadow removal, in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019, pp. 1999–2008.

[25]. Tao, M. W., Srinivasan, P. P., Hadap, S., Rusinkiewicz, S., Malik, J., & Ramamoorthi, R., Shape Estimation from shading, defocus, and correspondence using light-field angular coherence, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, Issue 3, 2017, pp. 546–560.

[26]. Huang, X., Gao, J., Wang, L., & Yang, R., Examplar-based shape from shading, in *Proceedings of the 6th International Conference on 3-D Digital Imaging and Modeling (DIM 2007)*, Montreal, Canada, 2007, pp. 349-356.

[27]. Atkinson, G. A., & Hancock, E. R., Shape estimation using polarization and shading from two views, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, Issue 11, 2007, pp. 2001-2017.

(091)

A Coined Knowledge-Based Computational Toolkit for Biomassed-Based Sustainable Infrastructure: Python, VB, Matlab

R. Hussein
State University of New York, Syracuse, NY, USA
Tel.: + 315-695-2340
E-mail: ezpsc@yahoo.com

Summary: Knowledge-based (KB) systems are one of the subfields of AI that are focused on real-world applications. KB is focused on the acquisition and representation of knowledge in order to mimic expert human reasoning and address challenges that often call for human intervention, either independently or as a support to a human agent. One may consider KB to be a transfer of problem-solving human specialists' knowledge to a computer program that could process the same data and reach the same conclusion faster and with more options. This article coins a novel computational modular KB that is correct and unexplored to this point in the infrastructure industry, using synergetic client software. The modular approach provides a set of stand-alone algorithms with their own outputs, which the human expert may consider as justified and possible advice or recommendation. All the developments are based on the three most popular languages: Python, Matlab, and VB.

Keywords: Biomass, Biodegradable, CLT, Infrastructure, Knowledge-Based, lignin, Load bearing, Retaining systems.

1. Introduction

The infrastructure sector faces several global issues in the twenty-first century. In response to calls for innovative ideas and solutions and in order to help both technical and non-technical professionals navigate challenging processes and come to best-practice informed decisions, our research aimed at building a virtual brain. An integrated, multidisciplinary approach is the result of this concept's evolution. This article briefly describes a few selected modules, each of which is standalone software, can be used to explore a variety of user-defined what-if scenarios, employs a modular architecture to facilitate future expansion and adaptation to new challenges, is interactive so that users can operate it easily and see results immediately, is user-friendly, and needs no prior knowledge of computers. Some of the modules have built-in databases to avoid incorrect or impractical input data.

2. Biomass Biodegradable Load-Bearing

It is essential to fully grasp the potential of biomass in general and, in particular, lignin, which is weirdly neglected in sustainable infrastructure systems while being the second-most prevalent terrestrial polymer in the world [1]. Ligning is used to make by-products like fibers, foams, and inexpensive fuels. The development of modern biorefineries and the processing of biomass have highlighted the potential uses for lignin. The studies that were done showed that lignin improves rigidity, compressive strength, water resistance, and compressive modulus. There are very few studies on lignin-based rigid foam that can support construction loads. This author is not aware of any studies focusing on the use of lignin-based biodegradable rigid foams as load-bearing sustainable infrastructure construction components.

In an effort to replace the synthetic foam and fossil-based infrastructure components now in use, innovative software for biodegradable biomass-based load-bearing elements of sustainable infrastructure is coined in our KB [2]. Figure 1a depicts selected images from its dashboard. It takes into account various boundary conditions, loads, such as transverse, inplane, uniaxial, and biaxial loads, and instability situations, such as overall instability, shear, and local instability. Additionally, bonding with finite qualities rather than the traditional and unrealistic perfect bonding is used for the first time in the literature. This coined module doesn't require installation and provides a built-in database of material characteristics. All the input data and output results are engineering quantities, thus requiring no special efforts to use in real applications. To our knowledge, the software is one of a kind.

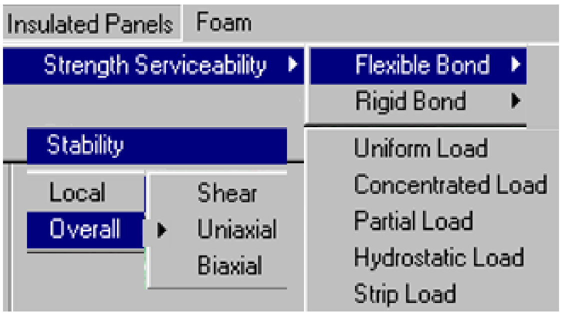
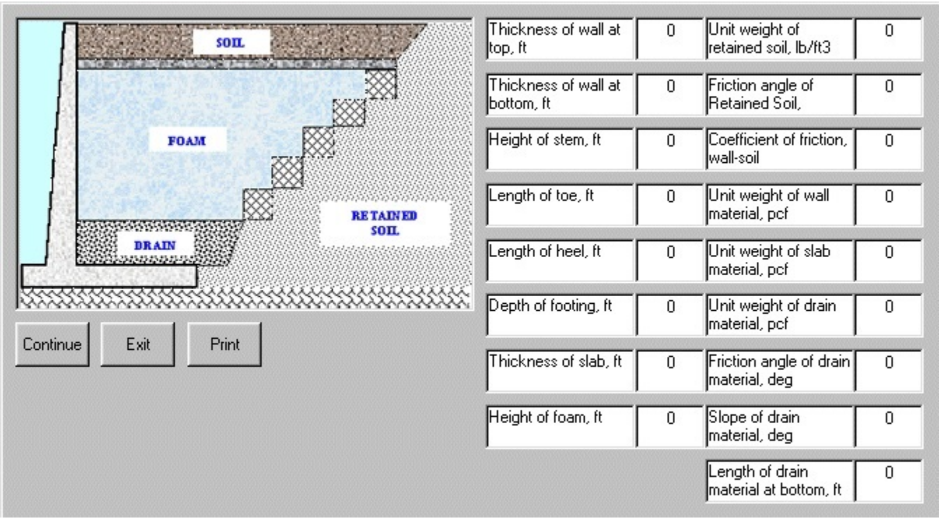


Fig. 1a. Illustrative screens for biomass-based biodegradable load-bearing client.



Thickness of wall at top, ft	0	Unit weight of retained soil, lb/ft3	0
Thickness of wall at bottom, ft	0	Friction angle of Retained Soil,	0
Height of stem, ft	0	Coefficient of friction, wall-soil	0
Length of toe, ft	0	Unit weight of wall material, pcf	0
Length of heel, ft	0	Unit weight of slab material, pcf	0
Depth of footing, ft	0	Unit weight of drain material, pcf	0
Thickness of slab, ft	0	Friction angle of drain material, deg	0
Height of foam, ft	0	Slope of drain material, deg	0
		Length of drain material at bottom, ft	0

Fig. 1b. Illustrative screen for biomass-based biodegradable retaining systems client.

Geofoam [2] is another use for the biomass-based load-bearing products, as seen in Fig. 1b.

This KB client is relevant to foundations, embankments, slope stability, stress distributions, net stresses, and cantilever and gravity retaining systems. As shown in Fig. 1c, it incorporates calculations for static, seismic, and spectral data.

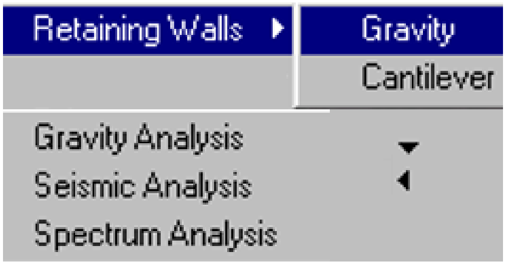


Fig. 1c. Illustrative screen for biomass-based biodegradable analysis client.

3. Biomass-based Composites

Cross-laminated timber (CLT) components are fabricated by adhering layers of timber with perpendicular grain orientations. The wide sides of the timber layers are where the adhesive is applied. Despite the fact that numerous studies on the design and construction of the CLT have been carried out, to the best of our knowledge, here is no toolkit or software, KB or not, for the engineering calculations of the CLT. This author fills this gap and developed, for the first time, a KB for the CLT [5].

By first creating the geometry, assigning numerical components (such as beams, columns, etc.), specifying the material properties, loads, and boundary conditions, the user is able to model, analyze, and design CLT systems. In reference 3, the governing formulas for the bending and compression design of CLT these are presented.

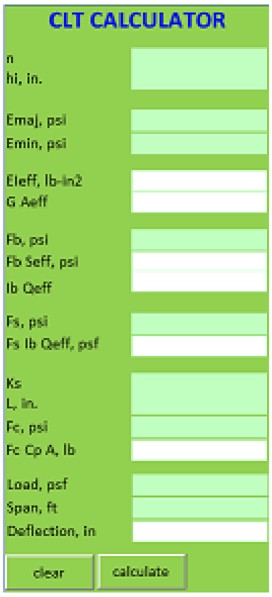


Fig. 2. CLT dashboard.

4. Why Python, Matlab, and VB?

According to the Tiobe index [4], Python was the most popular programming language on a global scale as of May 2023. Number 5 was VB, and number 15 was Matlab. Competent programmers, academic courses, software developers, and the global community are all taken into account when determining Tiobe's rating. The three specified coding languages were chosen by this author for his KB even though C ranks higher than VB in Tiobe's index. Instead of utilizing a language that produces the fewest number of lines in an algorithm, our aim was to execute computational calculations using ordinary computer knowledge and experience. There are linguistic distinctions that are outside the scope of this article. However, programmers can compare the various languages to decide which is most appropriate for the task at hands.

5. Conclusions

The use of biomass-based products has risen to the top of applications in sustainable infrastructure as a result of the scientific discoveries of modern materials and the tremendous advancements in manufacturing technology. Biomass-based products are widely available and offer unparalleled opportunities to address pressing problems, such as those related to the economy, the environment, energy, and the climate. However, the real-world applications lag behind the novel materials. What is the world missing, then? is the obvious question. The answer is provided in this article. It coins a kind of artificial intelligence to help experts navigate the intricate uses of the cutting-edge components of the failing infrastructure. This article described various toolkits for biomass-based composites as load-bearing components, cross-laminated timber, and retaining systems using modular computer architecture. In general, the method offered offers a new perspective on the future.

References

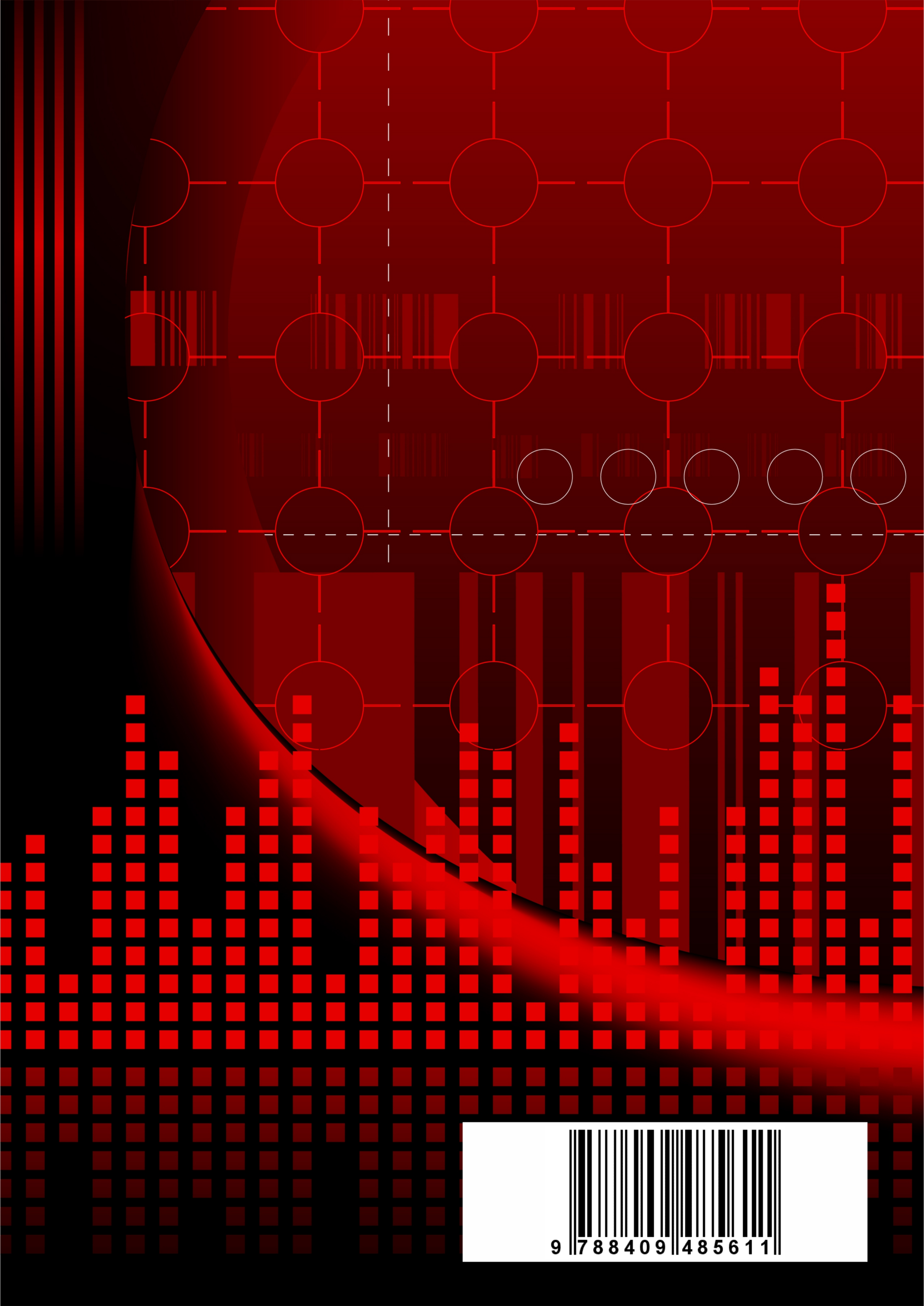
[1]. Tondi, G., et. al., Lignin-based foams: Production process and characterization, *BioResources Journal,* Vol. 11, No. 2, 2016, pp. 2972-2986.

[2]. Hussein, R., Sustainable Municipal Solid Waste Management Data Analytics Driven Perspective, *Acta Scientific Computer Sciences,* Vol. 4, Issue 10, 2022, pp. 30-33.

[3]. Erol Karacabeyli and Brad Douglas (Eds.), CLT handbook: cross-laminated timber, *FPInnovations and Binational Softwood Lumber Council,* 2013.

[4]. https://www.tiobe.com/tiobe-index/

[5]. Hussein, R., Treatise on Sustainable Infrastructure Construction: Green Composites, Cross Laminated/Mass Timber, Wood Truss Connectors, Nondestructive Technologies, Health Assessment and Monitoring: Utility Poles and Geofoam, Chapter, in Alireza Kaboli (ed.), Advances and Technologies Building Construction and Structural Analysis, *InTech Open*, 2021, p. 122-135.



9 788409 485611