Sensors & Transducers, Vol. 177, Issue 8, August 2014, pp. 85-91

www.sensorsportal.com

Sensors & Transducers

http://www.sensorsportal.com

Improved Mean Shift for Robust Object Tracking

Hongying Zhang, Zheng Hu
Civil Aviation University of China, Tianjin, 300300, China
E-mail: carole_zhang0716@163.com, zhu_guyue@163.com

Received: 20 March 2014 /Accepted: 31 July 2014 /Published: 31 August 2014

Abstract: In this paper, we present an improved mean shift for robust object tracking in complex environment. Traditional RGB color model used in mean shift tracker is sensitive to interference from similar background. In order to solve this problem, a new saliency-color target model is proposed through using the state-of-the-art target representation and updated background-weighed method. In addition, traditional mean shift method using fixed tracking window may cause tracking errors when target becomes close to or far away from the camera. Therefore, tracking window with self-adjust scheme is proposed in this paper. The tracking region parameters are updated through affine transform of feature corner datasets between adjacent frames. Moreover, a new prediction strategy is utilized to track the target with fast motion and partial occlusion. Experiment results demonstrate the effectiveness of proposed method, which can track object robustly under similar background, size changing, partial occlusion, etc. *Copyright © 2014 IFSA Publishing, S. L.*

Keywords: Mean shift, Target model, Scale and orientation, Partial occlusion, Robust tracking.

1. Introduction

Robust object tracking has been a research focus in recent years. It is a critical issue in computer, and widely applied in robot grasp [1], human computer interface [2], intelligent surveillance system [3], visual navigation [4], etc. As can be seen in [5-8], lots of object tracking algorithms have been proposed by researchers and achieved significant progress in visual tracking field. Among different kinds of tracking methods, mean shift algorithm is the most popular one for its high efficiency, utility and convenience to being combined with other method. However, robust object tracking is still a challenging task in reality due to some complex factors such as similar background interference, changing target scale and orientation, fast motion and partial occlusion.

The mean shift is a non-parameter density estimation method, originally proposed by Fukunaga for cluster analysis [9], and now it has been applied in various fields such as object tracking [10], image segmentation [11], pattern classification [12], etc. Among various mean shift tracker, the kernel-based object tracking proposed by Comaniciu [13] is the most typical one. This method uses the kernel-weighted color histogram as the target model and candidate one, and the similarity between them is evaluated by Bhattacharyya coefficient. In general, kernel-based object tracking method can obtain satisfying performance. However, it is not robust enough in complex environment since only color information is used to represent the target. The interference from clutter and similar background may lead to inaccurate tracking. Later, Ning [14] presented an object model using color-texture histogram and applied it to the mean shift framework. He demonstrates that the proposed method greatly improved the tracking efficiency and accuracy, especially when the target color is similar to the background one. In Ning's method, the target model may not be precise enough for only a small number of pixels on the target are used. Recently, Tavakoli [15] presented a novel local saliency operator called local similarity number (LSN), and he used a joint saliency-color histogram to represent the target in a mean shift tracking framework. Experiment results show that the proposed tracking algorithm can create excellent tracking result and outperform Ning's method. The main contribution of Tavakoli's method [15] is in preserving the unity of the target better. However, the background information was neglected by Tavakoli, which may be one of the reasons why target is lost in Tavakoli's third experiment [15].

Along with the background interference, change on scale and orientation also presents a challenge to mean shift tracker with fixed window. As shown in [16], when the target moves away from the camera, a fixed tracking window will cause lots of interference information with the target becoming small, inaccurate even lost tracks may occur. In [13], Comaniciu performs the mean shift procedure at previous scale, 10 % larger scale and 10 % smaller scale respectively, the best scale is selected in terms of the largest Bhattacharyya coefficient. This method is only suitable for the target with decreasing size, and it may become invalid when the target size is enlarging. In [17], Collins presented a scale selection method based on Lindeberg's scale space theory which shows a good adaptability to changing size. However, this method requires recomputing Gaussian kernels at each mean shift iteration, which is time consuming. Besides, it cannot deal with target rotation as same as A. Dulai's method [18]. Recently, feature points based mean shift tracker attracts researcher's attention. In [19, 20], the famous SIFT (scale invariant feature transform) keypoint information is used to correct the changed scale and orientation of mean shift tracking window. SIFT [21] shows good performance in many respects, including high repeatability with changing scale and orientation, robustness to illumination variation and noise interference. However, SIFT's calculation is complex, and it may be invalid on the target with uniform color distribution. Lately, some researchers employ the Harris corner matching method to calculate the affine structure between adjacent frames [22], then the scale and orientation variations are obtained based on the affine structure. In fact, only three pairs of matched points are used to calculate affine structure, which is not accurate enough. Besides, mathematical optimization techniques such as least square method can also be used to calculate the affine parameters, which are time-consuming.

According to above issues, we combine the saliency-color target model with a corrected background weighting method [23] which can effectively reduce background's interference. During the mean shift iteration phase, a novel tracking window with self-adaptive scale and orientation is utilized to track the object more tightly and accurately. Our self-adaptive scheme takes full advantage of location changes of corresponding points between adjacent frames. Furthermore, in order to track target more robustly under fast motion and partial occlusion, our method use a dynamically switching prediction scheme. In general, we use Kalman filter [24] to predict target location, and a linear prediction method is utilized when partial occlusion happens. Finally, we will verify that our improved mean shift algorithm could track target robustly under complex environment as mentioned above.

2. Target Representation

In this section, we introduce a state-of-the-art target representation method, and then we modify it through a corrected background weighting strategy which is expected more robust to similar background interference.

2.1. Saliency-Color Target Representation

In [15], Tavakoli defined a new visual descriptor called local saliency operator which count the number of similar pixels in a neighborhood and he named this operator local similarity number (LSN):

$$LSN_{P,R}^{d} = \sum_{i=0}^{P-1} f\left(g_i - g_c, d\right), \tag{1}$$

where $g_{i|i=0,\ldots,P-1}$ correspond to the intensities of P neighbouring pixels lying on a circle of radius R, d is the tolerant of intensity difference, and

$$f\left(x,d\right) = \begin{cases} 1, & \left|x\right| \le d \\ 0, & \left|x\right| > d \end{cases}, \tag{2}$$

Fig. 1 shows the nine saliency degrees of center pixel for 8-connected neighborhood.

In Tavakoli's color-LSN target representation, he modified $LSN_{8,1}^{d}$ to preserve the unity of the target as well as its edges, lines, and corners through LSN mask as follows:

$$mLTN_{8,1}^{d} = \begin{cases} 1 + \sum_{i=0}^{7} f\left(g_i - g_c, d\right), & \sum_{i=0}^{7} f\left(g_i - g_c, d\right) \in \left\{0,1,2,3,4\right\} \\ 0, & others. \end{cases} \tag{3}$$

The saliency-color representation histogram can be achieved by 8-bin quantized color RGB channels and 5-bin saliency information from mLSN, whose size is $8 \times 8 \times 8 \times 5 = 2560$. Then the target model can be represented through the probability distribution of saliency-color feature $\{q_u\}_{u=1,\ldots,m}$:

$$q_u = C\sum_{i=1}^{n} k(\left\| \frac{x_i^* - x_0}{h} \right\|^2)\delta\left[b(x_i^*) - u\right], \qquad (4)$$

where $\{x_i^*\}_{i=1,\ldots,n}$ are the coordinates of target pixels whose center is x_0, $k(x)$ is an isotropic kernel profile with bandwidth h which assigns smaller weights to pixels farther from the center, and $\delta(x)$ is the Kronecker delta function, C is a normalization constant which makes $\sum_{u=1}^{n} q_u = 1$, so

$$C = 1 \Big/ \sum_{i=1}^{n} k(\left\| \frac{x_i^* - x_0}{h} \right\|^2), \qquad (5)$$

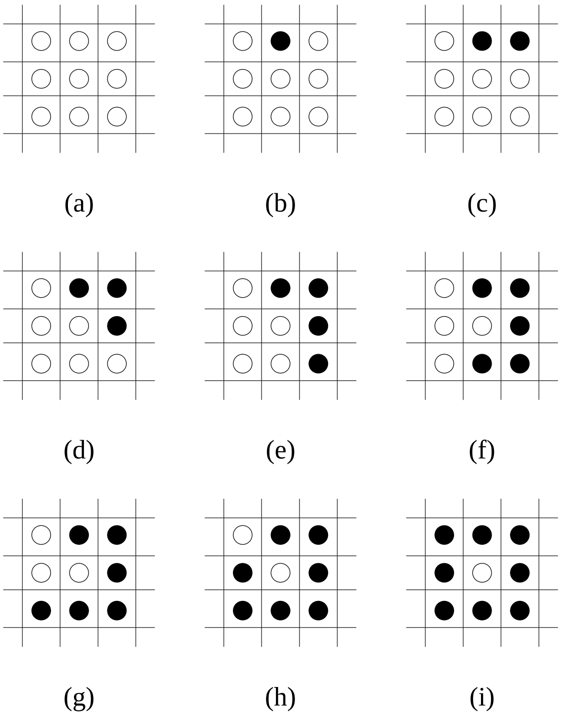


Fig. 1. The nine saliency degrees of $LSN_{8,1}$. Black circles mean pixels similar to the center.

Similarly, the target candidate model can be represented as follows:

$$p_u(y) = C_h \sum_{i=1}^{n_h} k(\left\| \frac{x_i - y}{h} \right\|^2)\delta\left[b(x_i) - u\right], \qquad (6)$$

and

$$C_h = 1 \Big/ \sum_{i=1}^{n_h} k(\left\| \frac{x_i - y}{h} \right\|^2), \qquad (7)$$

As shown in Fig. 1, any of the 256 patterns is a member of $LSN_{8,1}$ groups, so more useful information could be used to model the target. Compared with Ning's color-texture representation method utilizing only the nine uniform LBP texture patterns, LSN is expected to model the target more accurately.

2.2. Background Weighting

Although saliency-color model could represent the target well, some background information in the target region still affects tracking accuracy, especially when the target color is similar to the background. Comaniciu [13] introduced the background-weighted coefficient to make the target model more robust to background interference. However, as declared by Ning in [23], traditional background-weighted histogram [13] actually does not enhance mean shift tracking performance by modifying the representation of target model and candidate model at the same time. This is mainly because mean shift iteration formula is invariant to scale changes of the weights. In this paper, we utilize the background information only on the target model but not the candidate one [23] to decrease the weights of prominent background features.

As shown in Fig. 2, the target of interest is included in the red rectangle, and background region is around the target with the same size. Suppose the background is represented as $\{\hat{o}_u\}_{u=1,\ldots,m}$ (with $\sum_{u=1}^{m}\hat{o}_u = 1$), and $\hat{o}^*$ is the smallest nonzero value. The weight coefficient is

$$v_u = \left\{ \min\left(\frac{\hat{o}^*}{\hat{o}_u}, 1 \right) \right\}_{u=1,\ldots,m}, \qquad (8)$$

From (8), we can see that the bigger the background intensity is, the smaller weight coefficient it has.

Fig. 2. Foreground and background region of the target.

Then the target model can be redefined as follows:

$$\hat{q}_u = C v_u \sum_{i=1}^{n} k(\left\| \frac{x_i^* - x_0}{h} \right\|^2) \delta\left[b(x_i^*) - u \right], \quad (9)$$

And the normalization const is

$$C = 1 \Big/ \sum_{i=1}^{n} k(\left\| \frac{x_i^* - x_0}{h} \right\|^2) \sum_{u=1}^{m} v_u \delta\left(b\left(x_i^*\right) - u \right), \quad (10)$$

3. Target Localization through Mean Shift Iteration

To track the target in the current frame, we should find out candidate region which is the most similar to target model. Bhattacharyya coefficient is used to measure the similarity between target model and candidate model:

$$\rho\left(p(y), \hat{q} \right) = \sum_{u=1}^{m} \sqrt{p_u(y) \hat{q}_u}, \quad (11)$$

The target location can be obtained when Bhattacharyya coefficient achieves maximal value. Using Taylor expansion around the starting point y_0 (target tracked in the previous frame), Bhattacharyya coefficient can be approximated as

$$\rho\left(p(y), \hat{q} \right) \approx \frac{1}{2} \sum_{u=1}^{m} \sqrt{p_u(y_0) \hat{q}_u} + \frac{C_h}{2} \sum_{i=1}^{n_h} w_i k\left(\left\| \frac{x_i - y}{h} \right\|^2 \right). \quad (12)$$

where

$$w_i = \sum_{u=1}^{m} \sqrt{\frac{\hat{q}_u}{p_u(y_0)}} \delta\left[b(x_i - u) \right], \quad (13)$$

In (12), the first term is a const, so the maximal Bhattacharyya coefficient is achieved only when the second term is maximized. w_i represents the kernel based probability density distribution, and its physical meaning is the likelihood of current pixels belonging to the target. Then through mean shift iteration [25], we can find out optimal candidate region similar to the target model, among which each iteration moves the current candidate region center from y_0 to new position y_1:

$$y_1 = \frac{\sum_{i=1}^{n_h} x_i w_i g\left(\left\| \frac{x_i - y_0}{h} \right\|^2 \right)}{\sum_{i=1}^{n_h} w_i g\left(\left\| \frac{x_i - y_0}{h} \right\|^2 \right)}, \quad (14)$$

where $g(x) = -k'(x)$.

From the analysis above, we can see that the accurate probability density distribution w_i is a critical foundation for mean shift iteration, and formula (11) demonstrates that w_i directly depends on target representation method. So our mean shift tracker is expected to track target more robustly compared with traditional algorithm, especially when there existing interferences from background.

4. Tracking Window Self-adjust and Prediction

During the tracking phase, the target size and orientation may be changed, the fixed tracking window usually causes tracking errors. To track target more tightly and precisely, in this section, we present a self-adaptive tracking window strategy based on corner dataset. Firstly, we extract feature points on the target and match them between adjacent frames. Considering that there are only small extent of scale and orientation changes of target between adjacent frames, we utilize FAST (Features from Accelerated Segment Test) [26-27]. As an efficient corner detection algorithm, FAST has low computation cost and high performance in repeatability compared with SUSAN, Harris and DoG. FAST compares the intensity of a pixel p with its 16 neighborhood pixels (a circle of radius 3), if there are more than N contiguous pixels brighter or darker than p, then p will be considered as a feature point. Rosten [27] suggest that the best FAST detector is achieved when N takes 9.

Suppose that there are N pairs of matched feature corners on the target between adjacent frames, the corresponding corner datasets are named as $M_t = \{x_{t,i}, y_{t,i}\}_{i=1,\ldots,N}$ and $M_{t+1} = \{x_{t+1,i}, y_{t+1,i}\}_{i=1,\ldots,N}$ respectively, and the target center are $C_t = \{x_{t,c}, y_{t,c}\}$ and $C_{t+1} = \{x_{t+1,c}, y_{t+1,c}\}$, as illustrated in Fig. 3.

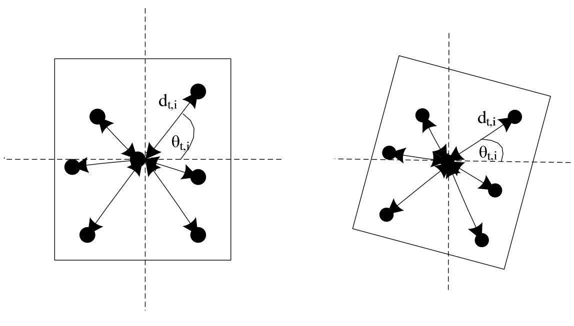


Fig. 3. The matched feature corner datasets on the target between adjacent frames.

The variation of scale and orientation changes is

$$\begin{cases} \Delta s = \dfrac{\sum_{i=1}^{N} \dfrac{d_{t+1,i}}{d_{t,i}}}{N} \\ \Delta\theta = \dfrac{\sum_{i=1}^{N} \left(\theta_{t+1,i} - \theta_{t,i}\right)}{N} \end{cases}, \quad (15)$$

where

$$\begin{cases} \theta_{\bullet,i} = \arctan \dfrac{y_{\bullet,i} - y_{\bullet,c}}{x_{\bullet,i} - x_{\bullet,c}} \\ d_{\bullet,i} = \sqrt{\left(x_{\bullet,i} - x_{\bullet,c}\right)^2 + \left(y_{\bullet,i} - y_{\bullet,c}\right)^2} \end{cases}, \quad (16)$$

Then the scale and orientation of target on the current frame can be adjusted through

$$\begin{cases} s_{t+1} = s_t \times \Delta s \\ \theta_{t+1} = \theta_t + \Delta\theta \end{cases}, \quad (17)$$

Besides, we use a dynamically switched prediction method to forecast the target center in the next frame, which is essential for target with partial occlusion and fast motion. Bhattacharyya coefficient can be a good indicator of target occlusion. When the Bhattacharyya coefficient is larger than the threshold, the target is considered visible. We use Kalman filter [24] to predict the target center in the next frame, so target through fast motion can be tracked where traditional mean shift tracker will lost the target for no overlaps. Otherwise, when Bhattacharyya coefficient is smaller than the threshold, we deem that the target is obscured by something. Now, mean shift iteration based on the current target location and probability density distribution is not accurate any more, so we switched to linear prediction according to previous target center location until target visible again.

5. Experimental Results

In order to verify the performance of our improved mean shift tracking algorithm, we conducted several experiments and compared them with some other typical target tracking algorithm based on mean shift. All of our experiments are implemented using C++ and OpenCV 2.3.1 on Visual Studio 2010. The running computer is configured with a 2.70 GHz Pentium(R) Dual-Core and 2GB RAM, and the operating system is Windows 7.

5.1. Target Tracking Through Similar Background and Rotation

In the first experiment, Egtest01 sequence [28] is used to testify the performance of our improved mean shift tracker. The color of the car to be tracked is similar to the background in some extent, and the attitude of car changes gradually, so challenge from similar color interference and target rotation both exist in this sequence. Part of the tracking results (frame 15, 168, 210, and 314 respectively) using our improved mean shift tracker are shown in Fig. 4, we also listed out the corresponding tracking results using traditional mean shift tracker [13] and the saliency-color based state-of-the-art mean shift tracker [15].

(a) Traditional mean shift tracker

(b) The saliency-color based mean shift tracker

(c) Our tracker

Fig. 4. Part tracking result of the Egtest01 sequence.

As can be seen in above, traditional mean shift tracker is sensitive to similar background interference, and the tracking window cannot be adjusted when object's attitude changes. The saliency-color based state-of-the-art mean shift algorithm can track target more accurately than traditional mean shift.

However, the orientation self-adaptive problem still not be solved. Our improved mean shift algorithm can track the target well even under similar background interference and changing attitude.

This is mainly attributed to the corrected background weighting method and corner dataset based target orientation self-adaptive scheme in our mean shift tracker.

Quantitative assessments are shown in Table 1 (frame 1 to frame 350 are used), in which we summarized the specific tracking information including average error, standard deviation, average iterations and computation time per frame.

Table 1 shows that our improved mean shift tracker outperforms other two algorithms, although the computation time is slightly longer than them.

Table 1. Quantitative assessments of car sequence tracking result.

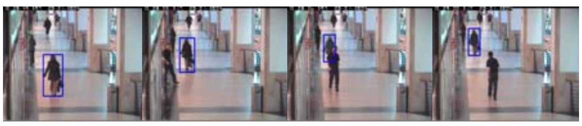
	Traditional tracker	**Saliency-color based tracker**	**Our tracker**
Average error	11.74	5.18	2.33
Standard deviation	10.62	4.85	1.71
Average iterations per frame	5.27	3.81	3.19
Average computation time per frame (ms)	8.35	12.58	14.46

5.2. Target Tracking Through Scale and Occlusion

The second experiment tries to follow a woman through scale and partial occlusion, and the experimental sequence can be available in CAVIAR data set [29]. In this sequence, the woman walks away from the camera, and the scale is becoming smaller and smaller. At the mean time, her trousers' color is similar to the floor. Beginning from the 160^{th} frame, a man comes out of a shop and walks across the woman, so partial occlusion occurs. Fig. 5 presents part of tracking results (frame 17, 160, 192 and 285) of the CAVIAR sequence using traditional mean shift tracker, Kang's scale adaptive mean shift tracking algorithm [22] and our improved mean shift tracking method respectively.

(a) Traditional mean shift tracker

(b) Kang's scale adaptive mean shift tracker



(c) Our tracker

Fig. 5. Part tracking result of the CAVIAR sequence.

From the tracking results above, we can see that traditional mean shift algorithm with fixed tracking window cannot track the target tightly, and it also has no mechanism for handling partial occlusion problem. As shown in Fig. 5(a), final tracking window mistakenly converged to the obstruction (the man). Kang's scale adaptive mean shift tracker calculates the affine structure between adjacent frames and target size thus is corrected through the affine parameters. Fig. 5(b) shows the effectiveness of this scheme. However, the anti-occlusion capability is still not so enough that the target is lost since the 168th frame. Fig. 5(c) demonstrates that our improved mean shift tracking algorithm has a good robustness to scale change and partial occlusion. This benefits from our dataset based scale adaptive mechanism and target location prediction method. The average tracking error is shown in Table 2.

Table 2. Average tracking error of CAVIAR sequence.

	Traditional tracker	**Saliency-color based tracker**	**Our tracker**
Average error of frame 1 to frame 160	17.66	8.93	5.84
Average error of frame 161 to frame 300	62.45	56.31	7.79

Table 2 shows that before occlusion (frame 1 to frame 160), all three algorithms can follow the target basically. But our method has less tracking errors than the two others for using corrected background weighting method and scale self-adaptive scheme. After frame 160, the woman gradually obscured by this man, both traditional mean shift tracker and Kang's scale adaptive mean shift tracker lost the target. However, good tracking results are achieved in our method by taking use of target location prediction based on a linear estimation method.

6. Conclusion

Traditional mean shift algorithm cannot track the target well in complex environments such as similar background interference, changing scale and orientation, and partial occlusion etc.. In this paper, we present a saliency-color model and corrected background weighting method to represent the target, then a tracking window self-adjust and prediction scheme is used to track target more tightly and precisely. Experiment results verify the performance of our proposed methods, it could track the object effectively and accurately even in complex environments mentioned above.

Acknowledgements

This paper is supported by Natural Science Foundation of Tianjin (No. 12JCQNJC00600).

References

[1]. Y. Yang and Q. Cao, A fast feature points-based object tracking method for robot grasp, *International*

Journal of Advanced Robotic Systems, Vol. 10, 2013, pp. 1-6.

[2]. Y. Ma, Z. H. Mao, W. Jia, et al, Magnetic hand tracking for human-computer interface, *IEEE Transactions on Magnetics*, Vol. 47, Issue 5, 2011, pp. 970-973.

[3]. N. Bellotto, B. Benfold, H. Harland, et al, Cognitive visual tracking and camera control, *Computer Vision and Image Understanding*, Vol. 116, Issue 3, 2012, pp. 457-471.

[4]. J. Fang, J. Yang, and H. Liu, Efficient and robust fragments-based multiple kernels tracking, *International Journal of Electronics and Communications*, Vol. 65, Issue 11, 2011, pp. 915-923.

[5]. D. Exner, E. Bruns, D. Kurz, et al, Fast and robust CAMShift tracking, in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 9-16.

[6]. I. Leichter, M. Lindenbaum, and E. Rivlin, Mean shift tracking with multiple reference color histograms, *Computer Vision and Image Understanding*, Vol. 114, Issue 3, 2010, pp. 400-408.

[7]. X. Wang and Z. Tang, Modified particle filter-based infrared pedestrian tracking, *Infrared Physics & Technology*, Vol. 53, Issue 4, 2010, pp. 280-287.

[8]. X. Li, K. Wang, W. Wang, et al, A multiple object tracking method using Kalman filter, in *Proceedings of the IEEE International Conference on Information and Automation*, 2010, pp. 1862-1866.

[9]. K. Fukunaga and L. Hostetler, The estimation of the gradient of a density function, with applications in pattern recognition, *IEEE Transactions on Information Theory*, Vol. 21, Issue 1, 1975, pp. 32-40.

[10]. A. Dulai and T. Stathaki, Mean shift tracking through scale and occlusion, *IET Signal Processing*, Vol. 6, Issue 5, 2012, pp. 534-540.

[11]. H. Zhou, X. Li, G. Schaefer, et al, Mean shift based gradient vector flow for image segmentation, *Computer Vision and Image Understanding*, Vol. 117, Issue 9, 2013, pp. 1004-1016.

[12]. R. Du, Q. Wu, X. He, et al, Object categorization based on a supervised mean shift algorithm, *Computer Vision – ECCV 2012, Workshops and Demonstrations, Lecture Notes in Computer Science*, Vol. 7585, 2012, pp. 611-614.

[13]. D. Comaniciu, V. Ramesh and P. Meer, Kernel-based object tracking, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, Issue 5, 2003, pp. 564-577.

[14]. J. Ning, L. Zhang, D. Zhang, et al, Robust object tracking using joint color-texture histogram, *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 23, Issue 7, 2009, pp. 1245-1263.

[15]. H. R. Tavakoli, M. S. Moin and J. Heikkila, Local similarity number and its application to object tracking, *International Journal of Advanced Robotic Systems*, Vol. 10, Issue 184, 2013, pp. 1-7.

[16]. Y. Wang, C. Fengxiong and G. Hongxiang, Kernel spatial histogram target tracking based on template drift correction, *ACTA Automatica Sinica*, Vol. 38, Issue 3, 2012, pp. 430-436.

[17]. R. T. Collins, Mean shift blob tracking through scale space, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2003, pp. 234-240.

[18]. A. Dulai and T. Stathaki, Mean shift tracking through scale and occlusion, *IET Signal Processing*, Vol. 6, Issue 5, 2012, pp. 534-540.

[19]. R. Dong, L. Bo and C. Qimei, Object tracking algorithm based on dataset of local invariant feature points, *Chinese Journal of Scientific Instrument*, Vol. 33, Issue 9, 2012, pp. 2053-2060.

[20]. R Dong, L. Bo and C. Qimei, Multi-degree-of-freedom mean-shift tracking algorithm based on SIFT feature, *Control and Decision*, Vol. 27, Issue 3, 2012, pp. 399-407.

[21]. D. G. Lowe, Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision*, Vol. 60, Issue 2, 2004, pp. 91-110.

[22]. Y. Kang, W. Xie and B. Hu, A scale adaptive mean-shift tracking algorithm for robot vision, *Advances in Mechanical Engineering*, Vol. 2013, 2013, Article ID 601612.

[23]. J. Ning, L. Zhang, D. Zhang, et al, Robust mean-shift tracking with corrected background-weighted histogram, *IET Computer Vision*, Vol. 6, Issue 1, 2012, pp. 62-69.

[24]. A. H. Mazinan and A. Amir-Latifi, Applying mean shift, motion information and Kalman filtering approaches to object tracking, *ISA Transactions*, Vol. 51, Issue 3, 2012, pp. 485-497.

[25]. D. Comaniciu and P. Meer, Mean shift: A robust approach toward feature space analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, Issue 5, 2002, pp. 603-619.

[26]. E. Rosten and T. Drummond, Fusing points and lines for high performance tracking, in *Proceedings of the Tenth IEEE International Conference on Computer Vision*, 2005, pp. 1508-1515.

[27]. E. Rosten, R. Porter and T. Drummond, Faster and better: A machine learning approach to corner detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 32, Issue 1, 2010, pp. 105-119.

[28]. Datasets (http://vision.cse.psu.edu/data/vividEval/datasets/datasets.html).

[29]. CAVIAR Test Case Scenarios (http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/).