

## Research Article

# Particle Filter Algorithm for Object Tracking Based on Color Local Entropy

**Huan Wang, Qinglin Wang, Yuan Li, and Yaping Dai**

*School of Automation, Beijing Institute of Technology, Beijing 100081, China*

Correspondence should be addressed to Yaping Dai; [daiyaping@gmail.com](mailto:daiyaping@gmail.com)

Received 27 September 2013; Accepted 9 December 2013

Academic Editor: Fenyuan Wang

Copyright © 2013 Huan Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To achieve accurate visual object tracking and overcome the difficulties brought by the object deformation, occlusion, and illumination variations, a particle filter for object tracking algorithm based on color local entropy (CLE) is proposed. First we improved the traditional histogram weighted function by using a scale factor. Then, for the shortcoming that the color feature is sensitive to illumination and environmental interference, a color local entropy object observation model is constructed by mapping the object from color feature space to local entropy space. In addition, an adaptive updating strategy of the object template is designed and the number of particle is adjusted dynamically according to the tracking performance. The experimental results show that compared with several existing algorithms, the proposed algorithm is more effective and robust for the real-time object tracking under the situation of illumination variation, object occlusion, and nonlinear motion.

## 1. Introduction

The real-time object tracking as the key step of Intelligent Transportation System (ITS) has been paid more and more attention. It is the premise of the traffic behavior analysis and prediction. Because of the complexity of the traffic environment, tracking moving object within a video sequence is still a challenging task. The difficulties of object tracking are mainly including the variation of object itself, such as pose variation in the process of the object motion and nonrigid object shape deformation; the extrinsic environment interference, such as illumination variation, occlusion, and camera vibration; and the real-time restriction.

So far, many tracking approaches have been developed, such as Kalman filter [1], mean shift [2], and Particle filter (PF) [3, 4]. Among these methods, Particle filter can solve the problem of moving object tracking in nonlinear, non-Gaussian conditions. It has a wider range of applications than Kalman filter method. Since Isard and Blake [5] introduce it into the field of video based tracking, it has become an important research direction gradually. However, in the particle filter framework, the tracking performance depends greatly on the feature selection and expression. The color

feature is stable and insensitive to object deformation, rotation, and partial occlusion, and it has advantages of simple calculation and fast processing, so it has been widely used in object tracking. In [6], Perez et al. proposed the color-based particle filter algorithm for object tracking and extension to multiple objects. Nummiaro et al. [7] proposed weighted color histogram observation model in RGB space and the object model is updated adaptively. But, color histogram is sensitive to illumination and environmental interference. Therefore scholars put forward many improved algorithm. Reference [8–10] constructed the object observation model by using spatiogram, which is histogram augmented with spatial means and covariance to capture a richer description of the object. Spatiogram contains color information and spatial layout of these colors and has better robustness than color histogram. Reference [11] constructed the multiple color distributions model according to prior knowledge of the object; it has a good performance especially when the object colors changed drastically. These algorithms above improved the color feature performance in some degree but did not consider the influence of the illumination. For this reason, many scholars have proposed multiple cues tracking method, such as two features fusion: color fused with texture [12],

or edge [13, 14], sift [15], and motion [16], and more than two features fusion: color fused with edge, texture, regional gray feature, and so on [17, 18]. Multiple cues fusion method improves the robustness of object model and overcomes the instability brought by using a single measurement source, but the extraction of a variety of feature and the calculation of the fusion strategy can lead to large sample space, heavy calculation burden. So the multiple cues fusion method cannot meet the real-time requirement in the actual project.

On account of the problems above, this paper starts from improving the descriptive ability of traditional color feature model, introduces the local entropy into the object feature model, proposes an object tracking algorithm based on color local entropy particle filter (CLEPF). First, the weight function of the histogram is improved by using a scale factor. Second, aiming at the disadvantages of the color feature which is sensitive to illumination and susceptible to environmental interference, we map the object from color feature space to local entropy feature space, a color local entropy object observation model is constructed in Hue-Saturation-Value (HSV) color space. Additionally, an adaptive update strategy of the object template is designed, and the number of particle is adjusted dynamically according to the tracking performance. The experimental results show that the proposed algorithm is more effective and robust by comparison with several existing algorithms.

The organization of this paper is shown as follows. Section 2 explains the theory of particle filter algorithm briefly. Section 3 gives the definition of color local entropy (CLE) and constructs the color local entropy object observation model. Section 4 designs a particle filter algorithm based on color local entropy, including the selection of the motion model, the adaptive updating strategy of object template, the dynamical adjustment of particle number, and the detailed realization of the algorithm. The experimental data and results are analyzed and discussed in Section 5. Section 6 is the conclusion.

## 2. Particle Filter Algorithm

In the particle filter algorithm, the posterior density of the system state is described by a group of weighted particles  $\{(x_{k-1}^{(i)}, w_{k-1}^{(i)}), i = 1, 2, \dots, N\}$ , where  $x_{k-1}^{(i)}$  is the state of the  $i$ th sample at discrete time  $k$  and  $w_{k-1}^{(i)}$  is associated weight. In order to make the posterior density only described by the distribution of the particle in the state space, all particle weight values should be  $1/N$ . Particles should be random samples of posterior density, but in solving practical problems, taking samples from posterior density directly does not work. Usually, Samples must be taken in some suggested density  $q(x_k | x_{k-1}, Y_k)$  [19].  $Y_k$  denotes all available observations  $\{y_1, y_2, \dots, y_k\}$ . Then the system state estimation is calculated by particles and associated weights, and continuously updating particles can guarantee the algorithm valid. Generic Particle Filter provides that a template can be expressed as follows:

$$\{(x_k^{(i)}, w_k^{(i)})\}_{i=1}^N = \text{PF} \left[ \{(x_{k-1}^{(i)}, w_{k-1}^{(i)})\}_{i=1}^N, y_k \right]. \quad (1)$$

The standard particle filter algorithm procedure is as follows.

- (1) Particle set initialization:  $k = 0$ ; take  $N$  particle samples  $\{(x_0^{(i)}, 1/N)\}_{i=1}^N \sim p(x_0)$  from system prior density function. All the particle initial weights are  $1/N$ ; set  $k = 1$ .

For  $k = 1, 2, 3, \dots$ , perform the following loop to end.

- (2) Importance sampling: take new particle samples from importance probability density function  $\{x_k^{(i)}\}_{i=1}^N \sim q(x_k | x_{k-1}^{(i)}, y_k)$ . Calculate particle weights and normalize

$$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(y_k | x_k^{(i)}) p(x_k^{(i)} | x_{k-1}^{(i)})}{q(x_k^{(i)} | X_{k-1}^{(i)}, Y_k)}, \quad (2)$$

$$\bar{w}_k^{(i)} = \frac{w_k^{(i)}}{\sum_{i=1}^N w_k^{(i)}}.$$

Calculate system posterior probability density:

$$p(x_k | Y_k) \approx \hat{p}(x_k | Y_k) = \sum_{i=1}^N \bar{w}_k^{(i)} \delta(x_k - x_k^{(i)}). \quad (3)$$

- (3) Resample: count effective particle number  $\widehat{N}_{\text{eff}}$ ; set threshold as  $N_T$ ; if  $\widehat{N}_{\text{eff}} < N_T$ , resample the particles  $\{(x_k^{(j)}, w_k^{(j)})\}_{j=1}^N = \text{Resample}[\{(x_{k-1}^{(i)}, w_{k-1}^{(i)})\}_{i=1}^N]$ , (Specific method can be seen in [20]).
- (4) Output: system state estimation  $\hat{x}_k = \sum_{i=1}^N \bar{w}_k^{(i)} x_k^{(i)}$ .

End For.

## 3. Color Local Entropy Object Model

Feature selection and expression is the key step of object tracking. Good feature model should have good stability, intuition and can distinguish the object from its background. The color histogram is insensitive to object deformation, rotation, scaling, and partial occlusion and has the advantages of easy calculating and fast processing. But it is sensitive to illumination variation. For this question, the local entropy of the image reflects the amount of the image local information, which has advantages of well resisting radiation distortion, deformation, and illumination change; it has been successfully applied in the image matching and retrieval [21–23]. Therefore, this paper combines color weight histogram feature with local entropy and builds a new color local entropy object observation model.

**3.1. Weight Color Histogram Feature.** Because the peripheral pixels may belong to the background or be affected by partial occlusion, the pixels which near center of region have more identification value than peripheral pixels. Therefore, to increase the reliability of the color distribution, pixels farther from the center are given less weight than pixels near

the center by employing a weighting function. In traditional method, weight is Gaussian function [7], which only gives one pixel that in the center of region maximum weight. In this paper we improve the traditional weighting function by defining a scale factor  $\delta$ . The proposed method can give pixels maximum weight within a certain range of the object center by adjusting the scale factor  $\delta$ ; it can reflect the difference of the pixels in different position and describe the object model accurately.

**Definition 1.** Let  $\{x_i^*\}_{i=1,\dots,n}$  be the pixel locations in the rectangular region with  $w$  long and  $h$  wide;  $y$  is the center of the region, and the weight color distribution  $p_u = \{p_u\}_{u=1,2,\dots,m}$  is defined as

$$k(r) = \begin{cases} 1, & 0 < r \leq \delta \\ 1 - \left( \frac{r-\delta}{1-\delta} \right)^2, & \delta < r \leq 1, \end{cases} \quad (4)$$

$$p_u = C \sum_{i=1}^n k \left( \left\| \frac{x_i^* - y}{d} \right\|^2 \right) \delta[b(x_i^*) - u],$$

where  $k(\cdot)$  is weight function;  $\delta$  is scale factor, which determines the range of the center vicinity area with the highest weight value (as depicted in Figure 1); the parameter  $d = 0.5\sqrt{w^2 + h^2}$ ;  $n$  denotes total pixels of the object region;  $m$  is color quantization level;  $b(x_i^*)$  is the corresponding characteristic level for the pixel  $x_i^*$ ;  $\delta[b(x_i^*) - u]$  is a Dirac function, which is used to determine whether the pixel color value of the object area belongs to the  $u$ th bin (equal to 1, otherwise to 0);  $C = [\sum_{i=1}^n k(\|(x_i^* - y)/d\|^2)]^{-1}$  is normalization factor.

**3.2. Image Local Entropy.** Image local entropy is widely used in image matching and retrieval for its advantages of resisting geometric distortion, easy calculation, and stability to illumination. Let an image which size is  $M \times N$ , and image function is  $f(x, y)$  and nonnegative; then

$$H_f = - \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} p_{ij} \log p_{ij}, \quad (5)$$

$$p_{ij} = \frac{f(i, j)}{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} f(i, j)} \quad (6)$$

$H_f$  is called the entropy of the image. If the entropy is defined in a certain local area of image, it is called image local entropy [21].

Local entropy is based on statistical methods, which reflects the amount of the local image information. Although in image matching and retrieval field local entropy has unique advantages, it can be seen from formula (6) that the definition of probability  $p_{ij}$  is a ratio of a single pixel gray value and the sum of local gray values. Therefore, when the background is bright and the object is dark, the value of  $p_{ij}$  will be too small and can lead to calculation error.

**3.3. Color Local Entropy Object Model.** This paper combines weight color histogram and the local entropy, complementary

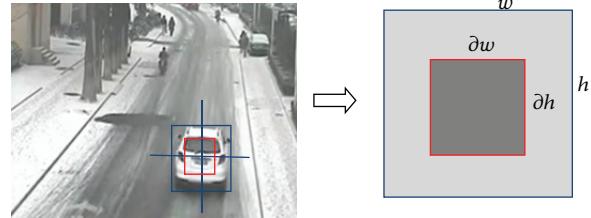


FIGURE 1: The object area schematic diagram.

advantage proposes color local entropy (Color Local Entropy, CLE) model.

**Definition 2.** In a rectangular region with  $w$  long and  $h$  wide, the image color local entropy  $H_c$  is defined as follows:

$$H_c = - \sum_{u=1}^m p_u \log p_u, \quad (7)$$

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

where  $m$  is color space quantization level;  $p_u$  refers to Section 3.1.

The CLE can describe and express the object better than traditional color feature model. On one hand, CLE not only retains the advantages of the color model which is stable to object deformation and partial occlusion, but also has good robustness to illumination variations; meanwhile the introduction of the weight function which represents space position information reduces the influence of the symmetry of entropy. On the other hand, analyzing from the view of human vision, the more concentrated the object color distributes in the space, the more the stimulation that human eyes get, and vice versa [24]. For CLE proposed, the value of the CLE indicates the decentralization or centralization of color distribution; thus compared with the traditional color model, CLE is more in accord with the characteristics of the human vision.

Taking the logarithm operation in (7) into account, discarding the higher order term of the Taylor series expansion, the following approximate formula is obtained:

$$H_c \approx - \sum_{u=1}^m p_u (p_u - 1). \quad (8)$$

In the tracking process, let  $p_u$  and  $H_c$  denote the color distribution and color local entropy of current object state, and let  $p_{ref}$  and  $H_{ref}$  denote the color distribution and local entropy of the object reference model. The similarity measure between the object reference model and the current state can be expressed by Euclidean distance:  $d_t^2 = \|H_{ref} - H_c\|^2$ . In the particle filter framework, the observation probability of the particle  $x^i$  is

$$p(y_k | x_k^i) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left( -\frac{d_t^2}{2\sigma^2} \right). \quad (9)$$

## 4. Object Tracking Algorithm Based on CLEPF

*4.1. Object Motion Model.* This paper uses external rectangle to describe the state of the moving object. Let  $X(k) = [a_x(k), a_y(k), v_x(k), v_y(k), w(k), h(k)]^T$  denote the object state vector, where  $(a_x(k), a_y(k))$  denotes the coordinates of center point of rectangle,  $(v_x(k), v_y(k))$  denotes the speed of the object, and  $(w(k), h(k))$  denotes the size of the rectangle.

Since the time of the frame interval is very short, it is assumed that the moving object is in uniform velocity within a frame interval. Therefore, this paper chooses the first-order autoregressive model. The state motion model is defined as  $X(k+1) = AX(k) + w(k)$ , where,  $A$  is the state transition matrix of system and  $w(k)$  is the zero-mean Gaussian measurement noise.

*4.2. Object Template Updating and the Determination of Particle Number.* During the tracking process, because of the object deformation and object occlusion, the fixed object reference template cannot adapt to the change of object and environment. With error continuous accumulating, the tracking accuracy will decline, so it is necessary to update the object observation model to ensure the tracking accuracy and stability.

Let  $H_{\text{ref}}$ ,  $H_c$  denote object reference template and current CLE, and let  $w^{(i)}$  denote the weight of particle. The adaptive updating strategy of object model as follows:

- (1) If  $\max(w^{(i)}) \geq \lambda_1$ , updating the template by  $H_{\text{ref},t+1} = (1 - a)H_{\text{ref},t} + aH_{u,t}$ ,  $a$  is forgetting factor.
- (2) If  $\max(w^{(i)}) < \lambda_2$ , that means the object is interfered by complex environment. Stop updating the template. Increase the particle number. Set the particle number to be  $N + K$  in next frame till no longer meets this condition, and then set the number to be  $N$ .  $N_{\text{max}}$  is the maximum number of particles.  $\lambda_1, \lambda_2$  are threshold according to the experiment.

*4.3. CLEPF Tracking Algorithm Steps.* The complete CLEPF algorithm for object tracking is summarized as follows.

- (1) Initialize particle: set = 0;  $N$  is the number of particles; select initial object manually, calculate weight color distribution  $q_u$  and CLE  $H_u$ , then distribute particles randomly around the initial state of the object  $\{(x_0^{(i)}, 1/N)\}_{i=1}^N \sim p(x_0^{(i)})$ , and set  $k = 1$ .
- (2) Object state prediction: predict the state of  $x_k^{(i)}$  according to state motion model and particle  $x_{k-1}^{(i)}$ .
- (3) Calculate particles weight: weights of the particles are calculated and normalized by formula (2).
- (4) Update object model and set the particle number: update object reference model and adjust particle number using the updating strategy in Section 4.2.
- (5) Output the tracking result: let the particle with the highest weight denote the tracking result and then acquire the tracking area.

- (6) Particle resample: resample the particle according to importance principle [19]. Set  $k = k + 1$ , jump to Step (2), and track next frame.

## 5. Experiment Results and Analysis

To verify the accuracy and robustness of the proposed algorithm (CLEPF), considering object occlusion, illumination variation, small camera vibration, and some other factors, three typical comparison experiments have been designed to test the proposed algorithm and compared with three representative tracking method: color histogram [7], spatiogram [8], color and edge histogram [14]. In the end, we evaluate and compare the algorithms' quantitative performance and empirically demonstrate the accuracy of the proposed method.

The experimental environment is based on the PC with Intel Core 2 processor, XP operation system, and 1G RAM. Software development environment is VC++2010. The initial object model was selected manually. HSV color space, which is simplified form of Munsell color space, is in accordance with human's perception, and its three components are relative independent. Therefore H-S histogram ( $16 \times 4$ ) is constructed in the HSV color space. All the resolution of experimental videos is  $320 \times 240$ . The parameters are set as follows:  $N = 150$ ,  $N_{\text{max}} = 410$ ,  $K = 10$ ,  $a = 0.03$ , and  $\lambda_1 = \lambda_2 = 0.75$ .

*5.1. Vehicle Tracking.* The first experimental video is taken from video surveillance system in Beijing Institute of Technology. The environmental background contains building, pedestrians, and other moving vehicles. The vehicle tracking results are shown in Figure 2. From Figure 2(a) we can see that the color-based particle filter can keep tracking the white car, but the estimated object area deflect from the real object center. The tracking results of CLEPF are shown in Figure 2(b), it is clear that CLEPF has a much better performance in the tracking accuracy as well as adaptability of the object deformation. Overall, both algorithms can accomplish the tracking task under uncomplicated environment.

*5.2. Object Tracking under Object Occlusion.* The second set of experiment is used to verify the robustness of our algorithm under the condition of object occlusion. In this experiment, the object is a pedestrian, which had elastic motion and morphological change, moved from right to left. In the process of walking, the pedestrian was occluded by a quickly passed cyclist, a lamppost, and a tree. The tracking results are shown in Figure 3. From Figures 3(a) and 3(b), we can see that the color histogram method lost the object, and spatiogram method also cannot work very well; the tracking window gradually becomes larger during the tracing process. The pedestrian has been occluded three times, so color information and edge information are all disturbed in some degree. Therefore, as shown in Figure 3(c), the fusion algorithm performance is not very well. Because of using the proper model updating strategy and adaptive particle number method, from Figure 3(d), it can be seen that the proposed method could track the object correctly even if another object occludes the object partially or fully.



(a) Tracking results using color histogram



(b) Tracking results using CLEPF

FIGURE 2: Comparison of tracking results for experiment 1 (the number in the upper left corner is the frame number).



(a) Tracking results using color histogram



(b) Tracking results using spatiogram



(c) Tracking results using color and edge histogram



(d) Tracking results using CLEPF

FIGURE 3: Comparison of tracking results for experiment 2 (the number in the upper left corner is the frame number).

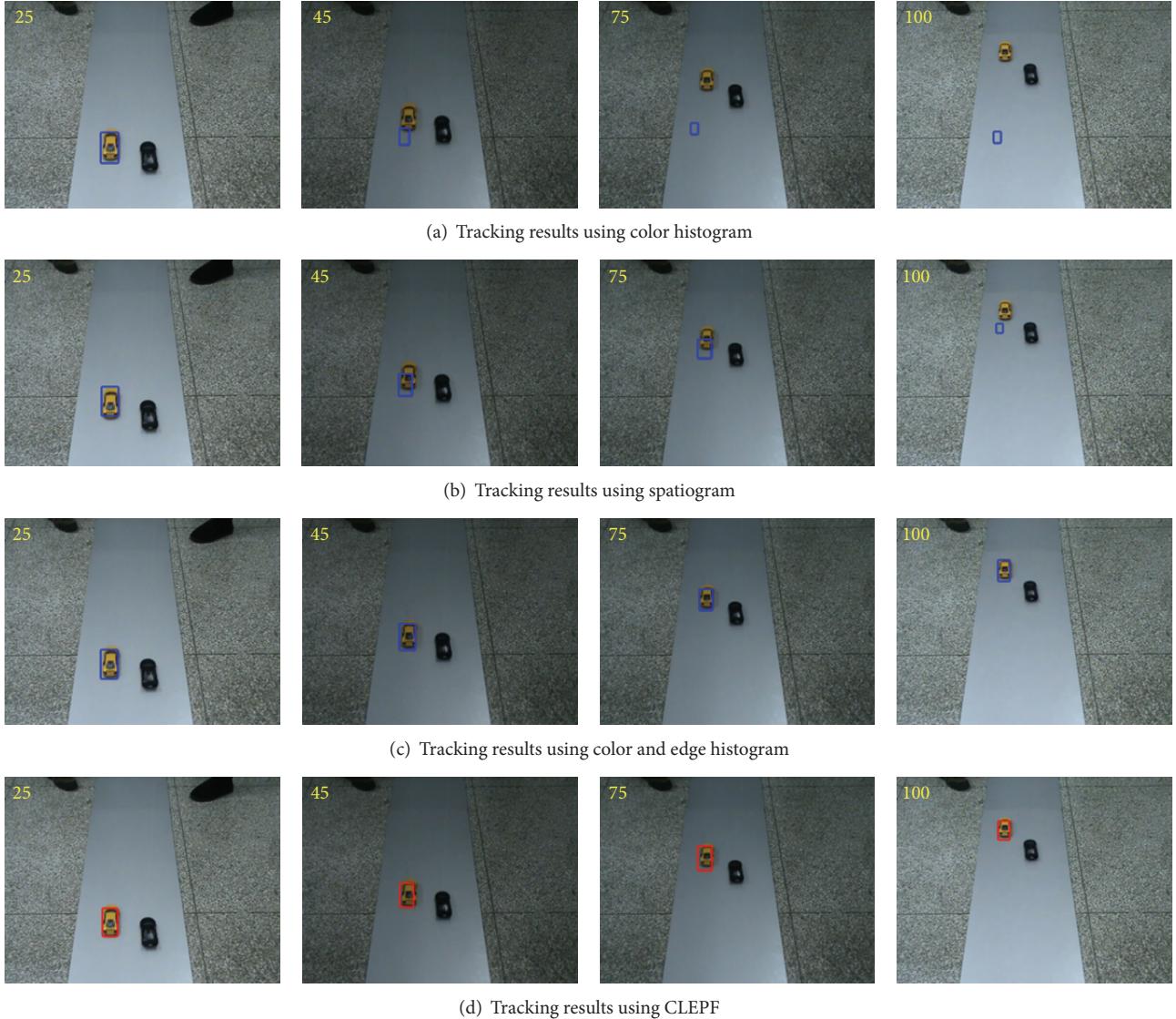


FIGURE 4: Comparison of tracking results for experiment 3 (the number in the upper left corner is the frame number).

**5.3. Object Tracking under Illumination Variation.** The third section of video sequence was captured by a hand-held camera in a typical laboratory environment. During the shoot, small camera dithering was unavoidable. This experiment is designed to verify the tracking robustness of our algorithm (CLEPF) under the condition of illumination change. The background illumination is changed by turning off and turning on the electric light repeatedly. As shown in Figure 4, the background illumination of Frame 45 is much darker than Frame 25 by turning off the light. Then, because of the camera autodimming function, the background brightness of Frame 75 is a little bit higher than Frame 45. Next, turn on the light again. According to the experimental results in Figure 4, although the background illumination changes again and again, CLEPF and the fusion algorithm work well and have a stable object tracking capability, color histogram, and spatiogram method lost the object.

In addition, tracking errors of those methods are computed and compared in Figure 5 for experiment 2 and experiment 3. Tracking error is the distance between the real object center position (obtained manually) and the centroid of object tracking window. As shown in Figure 5 our algorithm shows the good robustness and tracking accuracy. Tables 1 and 2 show the average runtime per frame and average error of four algorithms for experiment 2 and 3. From Tables 1 and 2, it is clear that the color and edge method take the longest time. That is because the extraction of multiple features and the calculation of the fusion strategy affect the computing speed, and other three methods take approximately the same time. Although the color histogram method takes the least time compared to other methods, it has the lowest accuracy. It is worth noticing that the proposed algorithm takes more time in experiment 2 than in experiment 3. That is because in experiment 2, three times

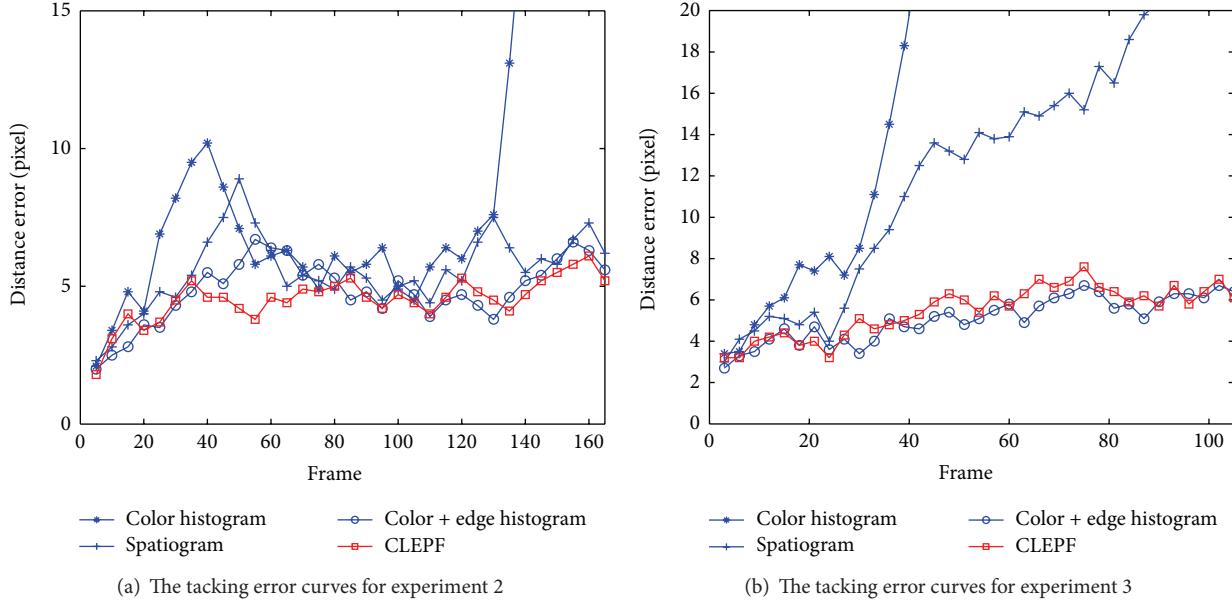


FIGURE 5: Comparison of the error curves.

TABLE 1: Performance comparison of four algorithms for experiment 2.

Algorithm	Frame rate (fps)	Parameter	Run time (ms)	Average error (pixel)
Color		18.2 (first 140 frame)	—	—
Spatiogram	25		43.6	7.41
Color and edge			163.1	5.92
CLEPF			57.7	4.53

TABLE 2: Performance comparison of four algorithms for experiment 3.

Algorithm	Frame rate (fps)	Parameter	Run time (ms)	Average error (pixel)
Color		—	—	—
Spatiogram	25	—	—	—
Color and edge			109.6	5.09
CLEPF			171	5.48

object occlusion caused the increase of particle number and then affected the speed of the algorithm. Overall, taking into consideration of the run time and accuracy factors, our algorithm has better performance than other algorithms and meets the requirement of real-time as well.

## 6. Conclusion

This paper proposed a robust and efficient object tracking algorithm. The contributions of this paper can be summarized as follows.

- (1) We improved the histogram weighted function.
- (2) A new color local entropy object observation model is constructed in HSV color space by mapping the object from color feature space to local entropy space.
- (3) An adaptive strategy of the object template is designed to update the appearance template online, and the number of particles is adjusted dynamically according to the tracking performance.

Compared with several existing methods, it is clear that the proposed algorithm is more effective and robust under the condition of object occlusion, illumination variation, and small camera dithering. Furthermore, the tracking speed of our algorithm is fast enough to meet the real-time requirement of practical engineering application.

Our algorithm can work well and have a good performance in object tracking, but there are still problems in it. Such as, when the illumination changes dramatically, the tracking error will increase, and object observation model is determined by global feature only. On the premise of meeting the real-time quality, how to integrate the local feature with global feature, further improving the robustness of the algorithm is what we need to do in future research.

## Acknowledgment

The authors acknowledge the support from the National Natural Science Foundation of China (Grant no. 61375100).

## References

- [1] S.-K. Weng, C.-M. Kuo, and S.-K. Tu, “Video object tracking using adaptive Kalman filter,” *Journal of Visual Communication and Image Representation*, vol. 17, no. 6, pp. 1190–1208, 2006.

- [2] Y. Kang, W. Xie, and B. Hu, "A Scale adaptive mean-shift tracking algorithm for robot vision," *Advances in Mechanical Engineering*, vol. 2013, Article ID 601612, 11 pages, 2013.
- [3] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-gaussian Bayesian state estimation," *IEE Proceedings F*, vol. 140, no. 2, pp. 107–113, 1993.
- [4] P. Pan and D. Schonfeld, "Dynamic proposal variance and optimal particle allocation in particle filtering for video tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 9, pp. 1268–1279, 2008.
- [5] M. Isard and A. Blake, "Condensation—conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.
- [6] P. Perez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proceedings of the 7th European Conference on Computer Vision Part I*, (ECCV '02), pp. 661–675, Springer.
- [7] K. Nummiaro, E. Koller-Meier, and L. van Gool, "An adaptive color-based particle filter," *Image and Vision Computing*, vol. 21, no. 1, pp. 99–110, 2003.
- [8] S. T. Birchfield and S. Rangarajan, "Spatiograms versus histograms for region-based tracking," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 1158–1163, San Diego, Calif, USA, June 2005.
- [9] S. T. Birchfield and S. Rangarajan, "Spatial histograms for region-based tracking," *ETRI Journal*, vol. 29, no. 5, pp. 697–699, 2007.
- [10] C. R. del-Blanco, N. García, L. Salgado, and F. Jaureguizar, "Object tracking from unstabilized platforms by particle filtering with embedded camera ego motion," in *Proceedings of the 6th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS '09)*, pp. 400–405, September 2009.
- [11] I. Leichter, M. Lindenbaum, and E. Rivlin, "Mean Shift tracking with multiple reference color histograms," *Computer Vision and Image Understanding*, vol. 114, no. 3, pp. 400–408, 2010.
- [12] J. Ning, L. Zhang, D. Zhang, and C. Wu, "Robust object tracking using joint colour texture histogram," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 23, no. 7, pp. 1245–1263, 2009.
- [13] I. Leichter, M. Lindenbaum, and E. Rivlin, "Tracking by affine kernel transformations using color and boundary cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 1, pp. 164–171, 2009.
- [14] W. Li, J. Cao, and D. Wu, "Multi-feature fusion tracking based on a new particle filter," *Journal of Computers*, vol. 7, no. 12, pp. 2939–2947, 2012.
- [15] H. Zhou, Y. Yuan, and C. Shi, "Object tracking using SIFT features and mean shift," *Computer Vision and Image Understanding*, vol. 113, no. 3, pp. 345–352, 2009.
- [16] M. Kristan, J. Perš, S. Kovačič, and A. Leonardis, "A local-motion-based probabilistic model for visual tracking," *Pattern Recognition*, vol. 42, no. 9, pp. 2160–2168, 2009.
- [17] Z. Jiang, "Object modelling and tracking in videos via multidimensional features," *ISRN Signal Processing*, vol. 2011, Article ID 173176, 15 pages, 2011.
- [18] X. Qiu, S. Liu, and F. Liu, "An adaptive kernel-based target tracking method based on multiple features fusion," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 7, no. 1, pp. 91–97, 2012.
- [19] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.
- [20] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
- [21] J. Tian, K. Su, and J. Liu, "Image matching based on local entropy difference—algorithm and computer simulation," *Journal of Astronautics*, vol. 20, no. 1, pp. 28–32, 1999.
- [22] J. M. Zachary, *An information theoretic approach to content based image retrieval [Ph.D. thesis]*, Louisiana State University and Agricultural and Mechanical College, 2000.
- [23] J. Sun, Z. Xu, J. Liu, and Y. Yao, "An objective visual security assessment for cipher-images based on local entropy," *Multimedia Tools and Applications*, vol. 53, no. 1, pp. 75–95, 2011.
- [24] L. Itti and C. Koch, "Computational modelling of visual attention," *Nature Reviews Neuroscience*, vol. 2, no. 3, pp. 194–203, 2001.