

## A New Robust Object Tracking Algorithm Based on Multi-Feature Fusion

---

**Wanjun Xu<sup>1</sup>**

*Air Force Engineering University, Xi'an, 710077, China*

*Email: Xuwanjun901219@163.com*

**Zhiqiang Hou**

*Air Force Engineering University, Xi'an, 710077, China*

*Email: Xuwanjun901219@163.com*

**Shunyi Fan**

*Air Force Engineering University, Xi'an, 710077, China*

*Email: Xuwanjun901219@163.com*

**Bo Dai**

*Air Force Engineering University, Xi'an, 710077, China*

*Email: Xuwanjun901219@163.com*

The object tracking by single feature often leads to poor robustness. In this paper, an object tracking algorithm based on multi-features fusion is presented. An adaptive method of choosing object color histogram is presented and the histogram is background weighted in order to get an accurate color model of the object. At meanwhile, then spatiograms feature is applied to obtain spatial layout of these color information. These features are rationally fused in the framework of Particle filter. The uncertainty measurement method is then introduced into features fusion to adjust the relative contribution of different features adaptively. Experimental results indicate that the proposed method is robust and highlights good performance in complex scene.

*ISCC 2015*

*18-19, December, 2015*

*Guangzhou, China*

---

<sup>1</sup>This research was supported by National Natural Science Foundation of China (No.61175029, No.61473309)

## 1.Introduction

The object tracking is one of the most important components in a wide range of applications in computer vision [1,2,3]. It is still a challenge for trackers to adapt it to large appearance variation such as illumination, pose change and background clutters with single feature in complex scenes. It's a prevailing way to combine more than one sort of features to describe the target.

In recent years, there has been lots of literatures about this and they all show that the methods by several features usually outperform that with only a single one [4]. How to represent the features of target effectively is one of the keys of fusing multi-features algorithms [5]. Li et al. [6] proposed an improved mean shift tracking algorithm. Firstly, the object's color is analyzed by using a clustering algorithm. According to the clustering result, the color space of the object is partitioned into subspaces by matrix factorization and orthonormal transformation. The tracking algorithm by affine kernel transformations by using the color and boundary cues is introduced by Ido. L[7]. Ning et al. [8] introduced LBP into mean shift and proposed a mean shift tracking algorithm based on texture model. Gu [9] developed a feature uncertainty measurement method. The feature uncertainty is based on the distribution of particles.

In this paper, we propose an object tracking algorithm of multiple features fusion based on the adaptively background-weighted. Our main work is presenting a new robust adaptively color histogram and construct a accurate and effective target model. The color histogram is background-weighted as the color model, and the spatiograms as the second feature. At meanwhile, with the particle filter, color feature and spatiograms are regarded as two independent tracking clues, using the fusion strategy proposed by Gu et al. [9]. Experiments show that the proposed tracker is more robust to the background clutter.

## 2.Particle Filter and Uncertainty Measurement

### 2.1 Particle Filter

Particle filter [11], based on Bayesian filter, has proved to be very successful for prediction of posterior density. It includes two stages: prediction and update. The particle filter assumes that  $p(x_{k-1}/z_{1:k-1})$  represents the posterior density at time  $k-1$ . The state of a tracked object is described by the vector  $x_{k-1}$  while the vector  $z_{1:k-1}$  denotes all the observations up to time  $t-1$ . Formula (2.2) is the Bayesian principle. Index  $k$  represents the time. Our tracker is based on particle filter to get the posterior density.

Prediction:

$$p(x_k / z_{1:k-1}) = p(x_k / x_{k-1})p(x_{k-1} / z_{0:k-1})dx_{k-1} \quad (2.1)$$

Update:

$$p(x_k / z_k) = \frac{p(x_k / x_k)p(x_k / z_{k-1})}{p(x_k / x_k)p(x_k / z_{k-1})dx_k} \quad (2.2)$$

For the non-linear and non-Gaussian estimation problems, the integration of Formula (2.3) and (2.4) is unsolved [12]; therefore, the particle filter translates the posterior density as follow:

$$p(x_k / z_{1:k}) = \sum_{i=1}^N \tilde{\omega}_k^{(i)} \delta(x_k - x^{(i)}_k) \quad (2.3)$$

$$\tilde{\omega}_k^{(i)} = \frac{\omega_k^{(i)}}{\sum_{i=1}^N \omega_k^{(i)}} \quad (2.4)$$

Where,

$$a_k^i = a_{k-1}^i \frac{p(z_k / x_k^i)p(x_k^i / x_{k-1}^i)}{q(x_k^i / x_{k-1}^i, z_{1:k})} \quad (2.5)$$

Formula(2.5) represents the importance sampling density, and the posterior density can be rewritten as:

$$p(x_k / z_{1:k}) \approx \sum_{i=1}^N \omega_k^i \delta(x_k - x_{k-1}) \quad (2.6)$$

### 2.2 Uncertainty Measurement

In the object tracking, each feature's accuracy of estimation of the target's location is different at different scenes, for which, Gu et al. [10] proposed a new method of target feature uncertainty measurement:

$$U_{k+1}^i = \text{var}_k H(p_k^i) \quad (2.7)$$

Where,  $U_{k+1}^i$  is the uncertainty of feature  $i$  at time  $k+1$ ;  $\text{var}_k$  is the variance of the particles' location.  $\text{var}_k = \text{tr}(\Sigma)$ , where  $\Sigma$  is the covariance matrix of particle's coordinates. The bigger  $\text{var}$  is, the disperse particles' distribution will be;  $p_i$  is the likelihood function, namely, the observation, which represents the  $i^{\text{th}}$  feature's similarity of two target template and candidate template;  $H(p_k^i)$  is the  $i^{\text{th}}$  feature's entropy of all particles' observation, which represents the degree of scatter of the observations. In the information theory, entropy is the measurement of uncertainty [13]:

$$H(p_i) = -\sum_{j=1}^N (p(z^i / x_j) \log_2 p(z^i / x_j)) \quad (2.8)$$

Where, the smaller  $H(p_i)$  is, the better accuracy of  $i^{\text{th}}$  feature's estimation of target's location will be.

### 3.Features Extraction and Fusion

In this context, the extraction of features is presented, as well as the computation of similarity of the target template and candidate template. Finally, the features' fusion strategy is presented.

#### 3.1 Features Extraction

##### 3.1.1 Color Feature

The traditional color histograms only compute the number of certain color value of the image, just reflecting the global color information of target; therefore, a novel method of adaptive color histogram is presented. This method, which adopts the idea of histogram equalization [13], can adaptively divide the color histogram on the base of target's color distribution in order to achieve accurate color model.

As shown in Fig.1(a),  $L$  is a cumulative histogram curve. Axis  $y$  represents the normalized and uniform color graduation value of pixels. Axis  $x$  represents the color graduation value of pixels that are divided not evenly. Here, axis  $y$  is evenly divided into four intervals  $[0, n_1]$ ,  $[n_1, n_2]$ ,  $[n_2, n_3]$  and  $[n_3, n_4]$  firstly. Then coordinates  $n_1, n_2, n_3$  and  $n_4$  are reflected to be  $m_1, m_2, m_3, m_4$  in axis  $x$  by  $L$ , the expected graduation value divided by adaptive histograms.

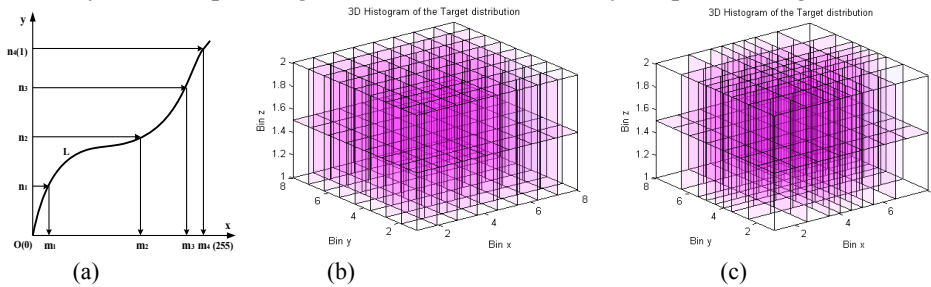


Figure 1: Two Methods of Distributing Color Space

Fig.1(b) represents the division of color space by traditional color histograms and Fig.1 (c) represents that of adaptive color histograms. The former one evenly divides the color space. While the latter one divides more ‘panes’ for the region of relatively concentrated color distribution but less ‘panes’ for the region of relatively disperse region. After video input, determine graduation value of histograms in the first frame by using the above method. The size of interval indicates the distribution of color. In the follow-up tracking, compute the histograms of target template according to the obtained graduation value, then the adaptive color histograms are acquired.

In the object tracking, the background information is usually included in the detected target region. To reduce this interference, CBWH is proposed [9]. The background is represented as  $\{\hat{o}_u\}_{u=1,2,\dots,N}$  (with  $\sum_{u=1}^N \hat{o}_u = 1$ ). Denote by  $\hat{o}^*$  the minimal non-zero value in  $\{\hat{o}_u\}_{u=1,2,\dots,N}$ . The corresponding weight of background histogram is:

$$v_u = \min\left\{\frac{\hat{o}^*}{\hat{o}_u}, 1\right\} \quad (2.9)$$

Therefore, the target template can be expressed:

$$\hat{q}_u = q_u \cdot v_u \quad (2.10)$$

### 3.1.2spatiogram

The spatiogram is defined as follows [15]: an image is a two-dimensional mapping  $I: x \rightarrow v$  from pixel  $x=[x,y]^T$  to values  $v$ . The spatiogram of an image can be defined as follows [15]

$$h = \langle n_b, \mu_b, \Sigma_b \rangle, b = 1, 2, \dots, B \quad (2.11)$$

$$\mu_b = [\mu_{bx}, \mu_{by}]^T \quad (2.12)$$

$$\Sigma_b = \frac{1}{n_b} \sum_{i=1}^{n_b} (s_i - \mu_b)^T (s_i - \mu_b) \quad (2.13)$$

Where:  $n_b$  is the number of pixels of the  $b^{\text{th}}$  bin,  $\mu_b$  and  $\Sigma_b$  are the mean vector and covariance matrices, respectively, of the coordinates of those pixels.  $B$  is the number of bins in the spatiograms.

### 3.2The Measurement of Similarity of Histograms

The color model and the spatiogram model are constructed respectively to reduce the amount of calculation [9].

Measurement of similarity of the color histogram

The target template and candidate template of BWH can be defined as follows:

Target template:  $Q = \{\hat{q}_u\}_{u=1,\dots,N}$ , Candidate template:  $P = \{\hat{p}_u\}_{u=1,\dots,N}$

A popular measurement between two distributions  $\hat{p}_u$  and  $\hat{q}_u$  is the Bhattacharyya, a coefficient [10]

$$\rho[P, Q] = \sum_{u=1}^N \sqrt{p_u q_u} \quad (3.1)$$

The larger  $\rho$  is, the more similar the distributions will be. The Bhattacharyya distance is defined as

$$d = \sqrt{1 - \rho[P, Q]} \quad (3.2)$$

Measurement of similarity of the spatiograms

The similarity between two spatiograms can be computed as follows [15]:

$$\rho(h, h^i) = \sum_{b=1}^B \psi_b \rho_n(n_b, n_b^i) \quad (3.3)$$

We set  $\psi_b$  to the probability that  $\bar{x}_b$  was drawn from a Gaussian distribution described by  $(\bar{x}_b, \hat{\Sigma}_b)$ :

$$\psi_b = \eta \exp\left\{-\frac{1}{2}(\mu_b - \mu_b^i)^T \hat{\Sigma}_b^{-1}(\mu_b - \mu_b^i)\right\} \quad (3.4)$$

The  $\eta$  is Gaussian normalization constant  $\eta^{-1} = (\sum_b^{-1} + (\sum_b^i)^{-1})$  and . The similarity between histograms is :

$$\rho_n(n_b, n_b^i) = \frac{\sqrt{n_b n_b^i}}{\sqrt{(\sum_{j=1}^B n_j)(\sum_{j=1}^B n_j^i)}} \quad (3.5)$$

### 3.3 Fusion Strategy

The fusion strategy proposed by Gu et al. [10] is shown as follow:

$$p(z^1 \dots z^n / x) = \prod_{i=1}^n \left( \frac{p(z^i / x) + U^i U(x)}{1 + U^i} \right) \quad (3.6)$$

When  $n=2$ , the fusion strategy can be rewritten as follows:

$$p(z^1, z^2 / x) = \frac{1}{(1 + U^1)(1 + U^2)} \left( p(z^1 / x)p(z^2 / x) + U^1 U(x)p(z^2 / x) + U^2 U(x)p(z^1 / x) + U^1 U^2 (U(x))^2 \right) \quad (3.7)$$

Where,  $U^1$  and  $U^2$  are uncertainty respect to color and spatiograms features. The number of particles is  $N$ ,  $U(x)=1/N$ . Notice that, the observations are similarity between target template and candidate template.

### 3.4 Algorithm Steps

In summary, the main steps of the proposed algorithm are shown as follows:

**Step1 Initialization:** at the time  $k=0$ , extract color model of target, as well as the spatiograms. Then randomly select  $N$  initial particles  $\{x_0^i\}_{i=1}^N$  from prior distribution, with weight  $\{\omega_0^i\}_{i=1}^N = 1/N$ .

**Step2 State prediction:** at the time  $k>0$ , predict the state of  $x_k$ , according to  $x_k = Ax_{k-1} + N_k$ .

**Step3 Calculate particles' variance of space location  $\text{var}_k$  and entropy of two features:** calculate the  $\text{var}_k$  with the coordinates of particles. Get  $p(z^1 / x)$ ,  $p(z^2 / x)$  on the basis of Formula (3.1) and (3.3).

**Step4 Calculate the uncertainty of features:** calculate  $U^1, U^2$  with Formula (2.7).

**Step5 State update:** get the likelihood function  $p(z^1, z^2 / x)$  after features fusion with Formula (3.7), update and normalize the particles' weight in the

current frame with  $\omega_t^i = \omega_{t-1}^i \cdot p(z^1, z^2 / x)$ .

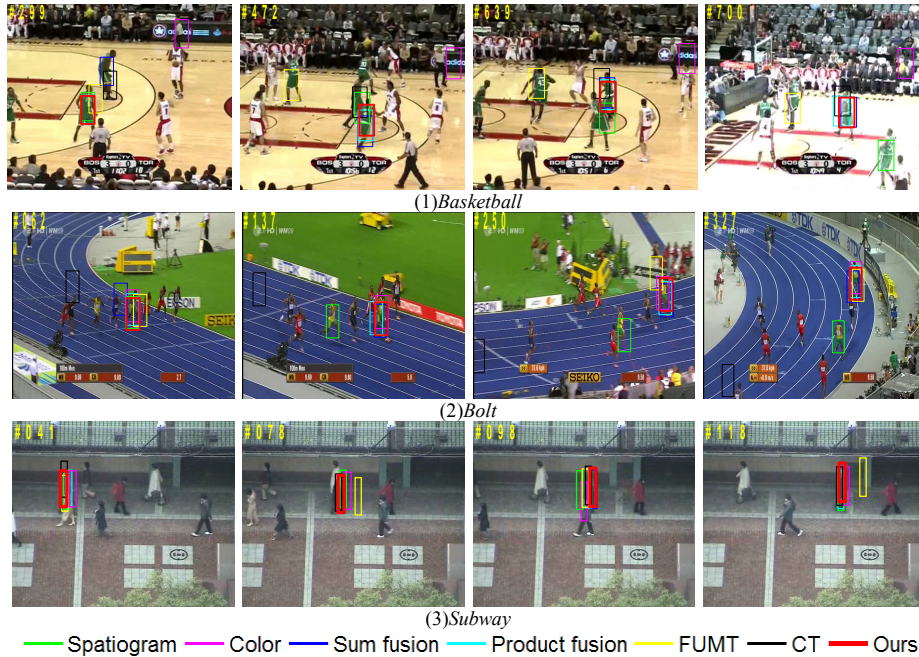
**Step6 Resampling:** resample particles **according** to its distribution.

**Step7 Transfer to Step2:** continue to **track** the next frame.

## 4. Experimental Results and Discussions

In the framework of particle filter, six comparative experiments are introduced. These experiments include the feature uncertainty measurement tracker (FUMT) as proposed by Gu, particle filter trackers by using single cue (color feature and spatiogram feature) respectively, sum fusion, product fusion and the Real-time Compressive Tracking (CT) tracker [16]. All of them are implemented under the programming environment of MATLAB7.9.0. In the experiments, the number of particles is defined as  $N=200$ , and  $U(x)=1/N$ .

### 4.1 Qualitative Analysis



**Figure 3:** Results Comparison of Five Tracking Algorithms

#### (1) Basketball

Fig.8(1) states the experiment results obtained on sequence *Basketball* with 725 frames. In this sequence, the background has exerted interference on the target's color feature. And the reference of background is challenging. We can see that our method can track the target precisely throughout while other trackers lose the target successively.

#### (2) Bolt

In the sequence *Bolt*, the color change of background leads to low discrimination of the color feature, which decreases the weight of color feature. While the weight of spatiogram feature increases, then the background interference is overcome. The tracking results in frame 294 and 335 show sound performance of our method.

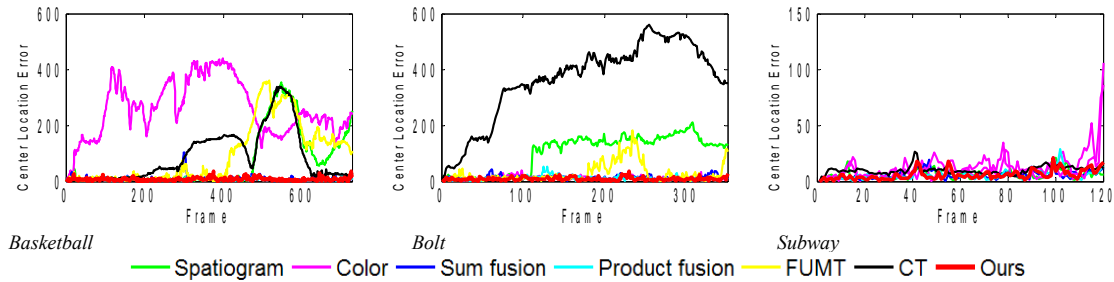
#### (3) Subway

Sequence *Subway* is aiming at tracking target walking in the subway passage. The target is the only one that walk at reverse direction with other people. During the tracking, target experiences occlusion and the background clutters several times. Our method, can avoid reference of the background and track the target precisely and robustly.

## 4.2 Quantitative Analysis

To further evaluate the proposed method and obtain the quantitative comparison to the state-of-art-works, we introduce the center location error (CLE) and real time performance.

The center location error measures the distance between the center location of the tracking results and the ground truths. The smaller center location error is, the better performance of tracking precision will be. Fig.4 shows the center location error curves of seven methods. Compared to other six methods, ours demonstrates a smaller center location error, indicating a robust performance.



**Figure 4:** Center Location Error Curves

Table1 shows the comparison of CLE of seven algorithms. Data indicates that the proposed method obtains better precision and has improved performance. Table.2 shows the comparison of real time performance of algorithms. From the data we can see that without any optimization, the proposed method can run at an average speed of 6.147 fps for *Subway* sequence. Of course, if optimized, the speed can be improved well. Of course, there maybe statistical error for the randomness of particle filter. These data are the average of ten sets of experimental data, which effectively avoid the reference of error or unreasonable data.

Sequence	Basketball	Bolt	David3	Subway
Spatiogram	69.343	103.946	11.012	6.223
Color	259.432	12.682	29.376	11.624
Product fusion	6.929	11.062	7.117	5.646
Sum fusion	8.904	12.726	8.543	5.798
FUMT	93.781	27.938	30.055	11.571
CT	88.835	363.862	88.761	9.828
Ours	7.457	8.808	6.978	5.496

**Table1:** Average center location error

Sequence	Basketball	Bolt	David3	Subway
Spatiogram	4.635	5.320	3.180	6.037
Color	4.382	5.234	3.212	6.098
TA	2.502	1.630	1.768	2.055
FUMT	0.853	0.930	0.731	1.307
Ours	4.667	5.122	3.691	6.147

Note: TA (traditional algorithm, which fuses traditional color histogram and spatiogram feature)

**Table 2:** Average running speed

## 5.Conclusion

In conclusion, we proposed an algorithm based on multi-features fusion to overcome the poor robustness of algorithms based on single feature. In the framework of particle filter, two complementary features, BWH and spatiogram, are integrated to

track the target. Experimental results indicate that the proposed tracker has sound robustness for background clutters and occlusion. It is expected for combination of the multi-feature fusion and model updating, besides, our tracker highlights advantages over that based on single feature seemingly, which combines adaptive color histogram and spatiogram. In the future works, we will investigate how to overcome more complex problems, such as combining multi-feature with template updating, fusing more other features and so on.

## References

1. Wu, Y., Lim J, and Yang, M.H. *Online object tracking: a benchmark*: Proceedings of the Computer Vision and Pattern Recognition. Portland, United States. 2411-2418(2013)
2. Yang, F., Lu, H.C., and Yang, M.H.: *Robust superpixel tracking*. IEEE Transactions on Image Processing. 23(4), 1639-1651(2014)
3. Hou, Z.Q., Han, C.Z.: *A Survey of visual tracking* . Acta Automatica sinica.32(4),603-617(2006)
4. Yin P. P., Sun F. C., Wang C., *An adaptive feature fusion framework for multi-class classification based on SVM*. *Soft computing*, 2008, 12(5): 685-691.
5. Wang, Y.Z., Liang, Y., Zhao, C.H.: *Kernel-Based tracking based on adaptive fusion of multiple cues*. Acta Automatica Sinica.34(4), 393-399(2008)
6. Li, P.H.: *An Adaptive Binning Color Model for Mean Shift Tracking*. IEEE Transactions on Circuits and Systems for Video Technology. 18(9), 1293 -1299 (2008)
7. Ido L, Michael L, Ehud R. *Tracking by affine kernel transformations using color and boundary cues*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(1):164-171(2009)
8. Ning, J.F., Wu, C.K.: *A Mean Shift tracking algorithm based on texture model*. Pattern Recognition and Artificial Intelligence. 20(5), 612-618(2007)
9. Gu X., Wang H. T., Wang L. F., et al. *Fusing multiple features for object tracking based on uncertainty measurement*. *Acta Automatica Sinica*, 37(5): 550-559( 2011)
10. Katja N, Esther K M, and Luc V G.: *An adaptive color-based filter* .Image Vision Computing. 21(1),99-110(2003)
11. M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, et al.: *A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking*. IEEE Transactions on Signal Processing.50(2),174-188(2002)
12. Liu, Q., Tang, L.B., Zhao, B.J., et al.: *Infrared target tracking based on adaptive multiple features fusion and mean shift*. Journal of Electronics&Information Technology. (2012)
13. Han J. H., Yang S. J., Uk L. B., *A novel 3-D color histogram equalization method with uniform 1-D gray scale histogram*. *IEEE Transactions on image processing*, 20(2):506-512(2011)
14. Birchfield Stanley T., Rangarajan Sriram.: *Spatigrams versus histograms for region-based tracking*. Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, United States. 1158-1163(2005)
15. Kazuki T., Hiroshi T., Hiroshi M., *Robust tracking method by MeanShift using Spatiograms*. *Proceedings of SICE Annual Conference*, 1982-1988(2010)
16. Zhang K., et al. *Real-time Compressive Tracking*. *Proceedings of the 7th European Conference on Computer Vision* . Berlin, Germany, Springer-Verlag, 864-877(2012)