

# Particle Filter Tracking Algorithm based on Dynamic Feature Fusion

Heng Yang, Guojian Cheng, Hongchu Chen

School of Intelligence Science and Information Engineering

Xi'an Peihua University, Xi'an, China

87781151@qq.com, 1683939013@qq.com, 22432117@qq.com

**Keywords:** Object tracking; Particle filter; Dynamic feature fusion; Object model

**Abstract.** A particle filter object tracking algorithm based on dynamic feature fusion is proposed in this paper. The presented algorithm uses the complementary features, which are gray histogram and gradient histogram, to represent the object model. In the tracking procession, the confidence for each feature is adjusted according to the discrimination between the object and the background, and the object model is dynamic online established and updated. The presented method can improve the accuracy of the object modeling and furthermore improve the accuracy of the particle filter tracking algorithm. Experimental results have demonstrated the effectiveness of our approach.

## 1 Introduction

Object tracking has always been a research issues in the field of computer vision, and it has been widely applied in scene monitoring, intelligent robot, autonomous navigation of UAV, behavior recognition and other fields. In recent years, the particle filter tracking algorithm has gained more and more researchers attention[1-2].

A key problem in particle filter algorithm framework is the establishment of target state model. It is an important factor that directly affects the performance of tracker. If the selected feature description cannot distinguish the target feature from the background feature, it will cause the tracking failure. Collins et al. [3] obtained 49 image feature descriptions based on the linear combination of color components of R, G and B, and selected 5 color features with target and background having the largest segmentation to establish the target model. Raziperchikolaei et al. [4] used the clustering algorithm to subdivide the color space adaptively to get a more accurate representation of the target histogram model. However, a single feature is lack of adaptability to scene changes, and tracking accuracy can be reduced rapidly when it cannot effectively distinguish targets and backgrounds. A kind of natural thinking is to adopt the fusion of multiple features, so as to improve the accuracy of the target model. Zhang et al. [5] put forward an adaptive feature fusion tracking algorithm in particle filter framework. At the same time, color and shape feature model are used together to describe targets, so as to enhance the tracking performance. Hourali et al. [6] also used skin color and shape histogram features to describe the target model together and track the face under the particle filter framework. Niu et al. [7] used SURF features and color features are combined to describe the target model. In addition, literatures [8,9] also proposed a target model based on multiple feature fusion under the particle filter framework. However, the above feature fusion tracking algorithms have limitations in improving the accuracy of the target model, because they can not dynamically adjust the confidence degree of all kinds of features according to the scene.

## 2 Target Tracking Algorithm based on Dynamic Feature Fusion

In order to overcome the shortcomings of the above algorithms, a dynamic feature fusion algorithm is proposed under the particle filter framework. The core idea is to represent the target by using gray histogram feature and gradient histogram feature together and adjust their weights dynamically in the process of tracking according to the distinction between target and background. As a result, our algorithm can achieve adaptable ability to scene changes and the goal of improving the tracking accuracy.

### 2.1 Construction of Target Model

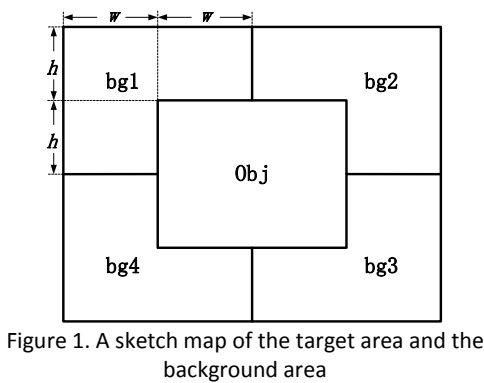
A rectangle in the image is used to describe the target state  $X$ , i.e.  $X = \{x_0, y_0, 2w + 1, 2h + 1\}$ ,  $x_0, y_0$  refers to the center of the target rectangle;  $w, h$  refers to the half width and half height of the target rectangle. Our algorithm uses the gray feature and gradient feature fused together to model the target. These two features are not only simple that can meet the real-time requirements of the system, but also have complementarity. Grayscale feature is a macroscopical characteristic, and gradient feature can describe the local details of the target.

The gray level features have a strong adaptability to the partial occlusion and the change of the appearance size. In our algorithm the gray level of the target is quantized into 128 grades, and the model is expressed by histogram  $p_I = \{p_i^{[u]}\}, u = 1, 2, \dots, 128$ . The gradient reflects the local gray change information in different directions. The target rectangle is divided into  $2 \times 2$  sub regions. For each sub region, the gradient magnitude value of each pixel is calculated, and mapping it to one of the 36 bins according to the gradient direction value. In this way, a gradient histogram of  $2 \times 2 \times 36 = 144$  bins is constructed as  $p_G = \{p_g^{[u]}\}, u = 1, 2, \dots, 144$ .

### 2.2 Calculation of the Feature Confidence

The key point of using multiple features to construct target model is to assign the weights dynamically to each feature according to the distinction between the target and background. That means if one feature lose effectiveness in distinguishing between target and background, it should be assigned small weight while other features should be assigned large weight accordingly.

Due to continuity of video sequences, we only need to consider the region around the target within a certain area as Fig. 1 shows. The size of the target area is  $(2w+1) \times (2h+1)$ , the other region within the rectangle in Fig. 1 is denoted as the background area. We define the kernel function of target and background as formula (1) and formula (2) respectively. For the target region, the pixels that are closer to the center of the target give the higher weight, while the pixels far away from the center of the target give lower weights. For the background region, the closer to the center the distance is, the smaller the weight is, and the pixels far away from the center give higher weight.



$$K_{obj}(x, y) = \exp\left(-\frac{(x-x_0)^2}{2w^2} - \frac{(y-y_0)^2}{2h^2}\right) \quad (1)$$

$$K_{bg}(x, y) = \begin{cases} \exp\left(-\frac{(-2w-x)^2 + (2h-y)^2}{2w^2 + 2h^2}\right) & \text{if } (x, y) \in bg1 \\ \exp\left(-\frac{(2w-x)^2 + (2h-y)^2}{2w^2 + 2h^2}\right) & \text{if } (x, y) \in bg2 \\ \exp\left(-\frac{(2w-x)^2 + (-2h-y)^2}{2w^2 + 2h^2}\right) & \text{if } (x, y) \in bg3 \\ \exp\left(-\frac{(-2w-x)^2 + (-2h-y)^2}{2w^2 + 2h^2}\right) & \text{if } (x, y) \in bg4 \end{cases} \quad (2)$$

The features of gray histogram and gradient histogram of target and background region are weighted by target and background kernel function respectively, such as  $p_{obj,f}^{[u]} = \sum_{(x,y) \in obj} K_{obj}(x, y) p_f^{[u]}$  and  $p_{bg,f}^{[u]} = \sum_{(x,y) \in bg} K_{bg}(x, y) p_f^{[u]}$ , in which  $f \in \{I, G\}$  denote the gray feature and gradient feature respectively. At a moment  $k$ , the discrimination of each feature to the target and the background is denoted as  $D_f(p_{obj,k}, p_{bg,k})$  and calculated by the Bhattacharyya distance.

According to Bayesian rule, the weight of the  $i$ th particle can be calculated as formula (3), in which  $p(Z_k | X_k^{(i)}, f) \propto \exp\left(-\frac{D_f(p_{obj,f}, p_{f,k}^{(i)})}{2\delta_f^2}\right)$  and  $p(f) \propto \frac{D_f(p_{obj,f}, p_{bg,f,k-1})}{\sum_{f \in \{I, G\}} D_f(p_{obj,f}, p_{bg,f,k-1})}$  represent the likelihood function and confidence degree of the feature  $f$  respectively.

$$w_k^{(i)} \propto p\left(Z_k | X_k^{(i)}\right) = \sum_{f \in \{I, G\}} p(Z_k | X_k^{(i)}, f) p(f) \quad (3)$$

$\delta_f^2$  is the noise parameter and is given by  $\delta_f^2=0.02$  and  $\delta_G^2=0.0158$  according to experimental experience. In this way, in the process of tracking, the target model is constantly revised, so that the current feature model can effectively distinguish between the target and the background to realize the reliable target tracking.

### 2.3 Details of the Algorithm

In the first frame, the target area and background area are calibrated manually. For the subsequent video frame  $k$  ( $k=1, 2, \dots$ ), target tracking is performed according to algorithm 1 until the end of video. It is worth noting that this algorithm only updates the confidence of the feature only when the feature template are updated, instead of updating the feature confidence in each frame, which can save a large amount of computing time.

---

#### Algorithm 1. Particle Filter Tracking Algorithm based on Dynamic Feature Fusion

---

##### Step1 state propagation:

For the particles  $\{X_{k-1}^{(i)}, 1/N\}_{i=1, \dots, N}$ , do one step propagation using first order motion model as  $X_k^{(i)} = AX_{k-1}^{(i)} + v_k^{(i)}$ , in which A is a first order motion model matrix and  $v_k$  is a Gauss random variable.

##### Step2 observation:

- (a) calculate the gray histogram  $p_{I,k}^{(i)}$  and gradient histogram  $p_{G,k}^{(i)}$  of each particle region in the particle set.
- (b) calculate the weight of each particle  $\omega_k^{(i)}$  according to formula (3) and normalized as  $\omega_k^{(i)} = \omega_k^{(i)} / \sum_{i=1}^N \omega_k^{(i)}$ . Therefore, a group of new particles is obtained as  $\{X_k^{(i)}, \omega_k^{(i)}\}_{i=1, \dots, N}$ .

##### Step3 estimation: $E(X_k) = \sum_{i=1}^N \omega_k^{(i)} X_k^{(i)}$

##### Step4 update:

The weight of the target and template at the estimated position is calculated as  $\omega_k$ . If  $\omega_k$  is greater than the pre-set threshold T, then the gray histogram and the gradient histogram of target template are updated.

##### Step5 resampling: Resample particles according to their weights $\{X_k^{(i)}, \frac{1}{N}\} \sim \{X_k^{(i)}, \omega_k^{(i)}\}$

Goto Step1 to process the next video frame  $k=k+1$  and finish the algorithm until the whole video frames are processed.

---

## 3 Experimental Results and Analysis

In order to compare the tracking performance of the proposed algorithm, mean shift algorithm (MS), intensity based particle filter algorithm (I-PF), gradient based particle filter (G-PF), and the presented dynamic feature fusion based particle filter (DFF-PF) are implemented respectively. Two representative video data is used in this experiment, one is the pedestrian video under occlusion condition and the other one is the aerial vehicle video under low contrast condition. The experimental environment is Pentium(R) 4 3.00 GHz and 2G memory.

### 3.1 Pedestrian Tracking under Occlusion Condition

The experimental video sequence comes from the caviar database [10]. The database gives the real position data of the tracking target in each frame, and can be used as the reference for calculating the tracking error of various algorithms. The first 300 frames are selected from the video for experimental material.

Fig. 2 gives the tracking error curve of the particle filter algorithm which describes the target model with three different features, and the tracking error is measured by the Euclidean distance. It is obvious that the tracking error of DFF-PF algorithm is the smallest and the tracking accuracy is the highest. Fig. 3 shows that in the DFF-PF algorithm, the distinction degree between the two features varies with the frame number. It can be seen that the proportion of the two features is different in different frames. Before the 100th frame, the gradient feature has a higher distinction between the

target and the background than the gray feature, and the gray feature is more dominant after the 100th frame. It also indicates that the target description strategy based on dynamic feature fusion can adapt to the scene change better.

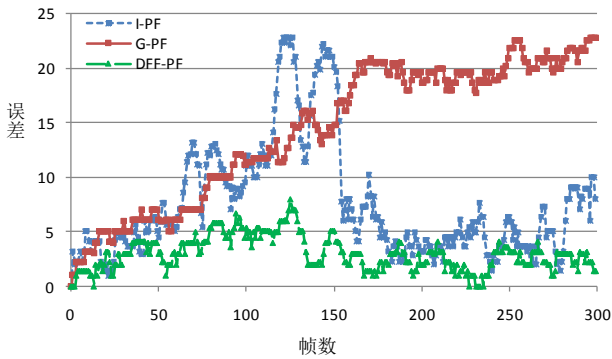


Figure 2. The tracking error curves of three kinds of algorithm with frame change

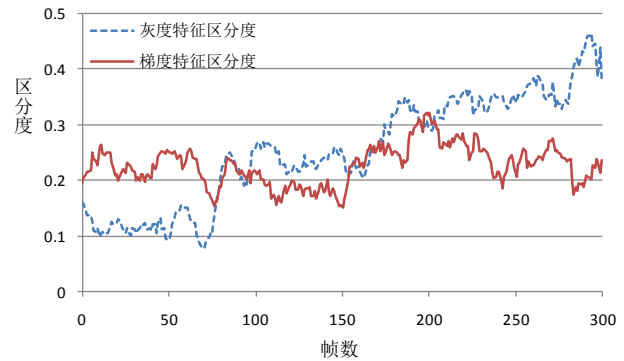


Figure 3. The discrimination degree curves of two features with frame change

### 3.2 Aerial Vehicle Tracking under Low Contrast Conditions

The video data of this experiment comes from the VSAM unmanned aerial vehicle (UAV) ground observation video database [11]. The whole video contrast is very low, and the tracking is difficult. Fig. 4 gives the tracking results of MS, I-PF, G-PF and DFF-PF algorithms. As the gray level of the tracking target and the background area is very similar, the MS algorithm and the I-PF algorithm quickly lose the target, and the gradient feature G-PF algorithm can track the target better during the vehicle running along the bridge. But when the target is under the bridge, the target apparent model changes, and the local details and background of the target are closer, which causes the gradient feature to decrease the discrimination degree of the target and the background, thus the tracking algorithm is invalid. As a result of the use of dynamic feature fusion strategy, the DFF-PF algorithm can accurately describe the target model at every moment, so it can track the target steadily and reliably from begin to end.

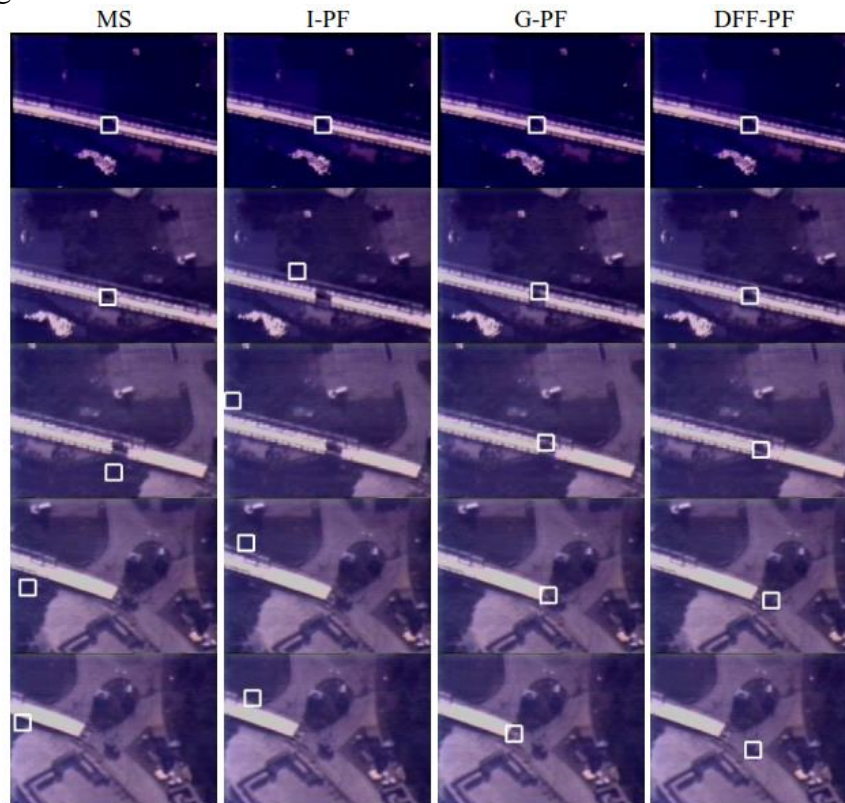


Figure 4. Four algorithms for tracking vehicles under low contrast conditions. Each row of images are the first, 81th, 224th, 450th, and 500th frame from left to right respectively.

## 4 Conclusion

A dynamic feature fusion modeling algorithm in particle filter frame which can be adaptive to scene change is proposed in this paper. The algorithm uses two simple and complementary feature models to describe the target model together. In the process of tracking, the algorithm can dynamically adjust the proportion of each feature in calculating the weights of the particles according to the discrimination degree of each feature. Experimental results verify the superiority of the proposed algorithm.

## References

- [1] Xuedong Wang, Tiancheng Li, Shudong Sun, Juan M. Corchado. 2017. A Survey of Recent Advances in Particle Filters and Remaining Challenges for Multitarget Tracking, *Sensors*, 17(12):1-21.
- [2] Rooji Jinana, Tara Raveendran. 2016. Particle Filters for Multiple Target Tracking, *International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST)*, 24(1): 980-987.
- [3] R.T. Collins, Y. Liu, M. Leordeanu. 2005. Online Selection of Discriminative Tracking Features. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 27(10), 1631-1643.
- [4] Ramin Raziperchikolaei, Mansour Jamzad. 2012. Visual Tracking using D2-Clustering and Particle Filter, *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 230-235.
- [5] B. Zhang, W. Tian and Z. Jin. 2007. Probabilistic Tracking of Objects with Adaptive Cue Fusion Mechanism. *Journal of Shanghai Jiaotong University*, 12(2):185-190.
- [6] F.Hourali, Dr. M. Sedaaghi. 2015. Robust and Real Time Face Tracking using Particle Filter based on Probabilistic Face Model, *International Journal of Research in Computer Applications and Robotics*, 3(2), pp. 71-78.
- [7] M. Niu, X. Mao, J. Liang and B. Niu. 2013. Object Tracking Based on Extended SURF and Particle Filter, *International Conference on Intelligent Computing*, pp. 649-657.
- [8] Zhimin Chen, Yuanxin Qu, Bing Liu, et. al. 2016. Improved Multiple Model Target Tracking Algorithm based on Particle Filter, *Journal of Aerospace Engineering*, 230(4): 747-759.
- [9] Dongsheng Ding, Zengru Jiang, Chengyuan Liu. 2016. Object Tracking Algorithm based on Particle Filter with Color and Texture Feature, *35th Chinese Control Conference*, pp. 4031-4036.
- [10] <http://groups.inf.ed.ac.uk/vision/CAVIAR/>
- [11] <http://www.vividevaluation.ri.cmu.edu/datasets/datasets.html>