

Adaptive Fusion Color and Haar-like Feature Object Tracking Based on Particle Filter

MA Xian-Bing, SUN Shui-Fa^{*1}, QIN Yin-Shi, Hu Song, LEI Bang-Jun

Abstract. The conventional kernel color histogram based particle filter (CHGPF) object tracking algorithm suffers from cluttered backgrounds and illumination changes and results in tracking lost easily. In this paper, a particle filter based object tracking algorithm fusing color and haar-like feature (CHAPF) is proposed. As haar-like features describe the object structure in multi-scale model, the haar-like feature based Semi-Supervised On-line Boosting (HSEMOB) tracking algorithm can well distinguish between the object and background, and be adaptive to changes of the object and environment by updating haar-like features. In our proposed algorithm, the results of HSEMOB and CHGPF are weighted adaptively to obtain a new particle filter weights. So, the color-based and haar-like based observations are fused into the framework of particle filter to make it adapt to the change of scene. Algorithm performance is tested with the environment complex videos. Experimental results show that the algorithm has better accuracy compared to CHGPF and HSEMOBPF for the scene of an object has similar appearance to the background or changes of illumination.

Keywords: Particle filter • observation model • haar-like • confidence

1 Introduction

Object tracking has a wide range of applications in many fields , such as the scene monitoring, motion capture, robotics autonomous navigation, vehicle autopilot system, video editing and compression, human computer interaction. In the actual tracking process, various types of features are used to characterize the object, including color, edge, optical flow, texture, motion and SIFT feature point. Many successful and accurate object tracking approaches have been proposed in recent years [1]. Such as the tracking algorithm of kernel color histogram based particle filter [2] was proposed by Katja Nummiaro in 2003, which is used widely for its robustness to partial occlusion, rotation and scale change. However, the use of only color feature results in weak robustness and tracking failure, especially

¹ *Corresponding Author: SUN Shui-Fa(✉), e-mail: watersun@ctgu.edu.cn

MA Xian-Bing, SUN Shui-Fa, QIN Yin-Shi, HU Song, LEI Bang-Jun
Institute of Intelligent Vision and Image Information, China Three Gorges University, China

when an object has similar appearance to the background or in unstable illumination case. So, it is necessary to fuse a variety of features to solve this problem.

In recent years, a number of multi-feature fusion based tracking algorithms were proposed [3-5], whose main difference is the strategy of feature extraction and fusion. However, previous algorithms of feature extraction are modeled on the object itself, with little considering the role of the background. Collins and Liu [6] proposed that object tracking can be regarded as classification between object and background, the classification results of the current frame are obtained by the classifier through learning the previous frames. On this basis, many tracking algorithms have been developed. In 2006, Grabner et al. designed an on-line boosting classifier which segments an object out of the background through adaptive feature selection [7]. However, there is a key issue for on-line adaption: the updating of the tracker may introduce an error accumulation which, finally lead to tracking failure (drifting). The drifting problem is significantly alleviated by the Semi-Supervised On-line Boosting method [8] which also can stay adaptive to appearance changes. Feature fusion strategy is mainly divided into two categories: multiplicative fusion [9] and additive fusion [10]. Literature [11] is a typical example of multiplicative fusion, it assumes independence between features in order to easily construct a multi-feature joint likelihood and apply it to object tracking. On the other hand, the additive fusion method also occupies a very important position by using the additive fusion to approximate joint likelihood.

In this paper, an algorithm of fusing color and haar-like feature based particle filter (CHAPF) is proposed: at first we set strong classifier, kernel color histogram as observation models of haar-like feature and color feature respectively, and get the particle weight of each feature by the observation models. Then using the fusion strategy to adjust the weights of each feature in order to obtain the new particle weight and estimate the object position. On one hand the algorithm overcomes the problem caused by the similar appearance and the illumination change, on the other hand it achieves more accurate and stable tracking results compared to CHGPF and HSEMOBPF.

2 The tracking algorithm based HSEMOB

HSEMOB [8] is regarded as classification between object and background. The classifiers of HSEMOB are continuously updated by positive and negative samples in order to achieve robust tracking. HSEMOB is made up of weak classifier, selector and strong classifier. Each weak classifier consists of one Haar-like feature based basic classifier whose recognition rate is higher than 50%. Each selector consists of one weak classifier and all selectors share one feature pool, namely Haar-like feature pool. The algorithm is mainly divided into two stages of initialization and tracking. (1) initialization: HSEMOB generates a large number

of Haar-like features (each feature corresponds to a weak classifier) randomly according to the object size, then updates these classifier by training positive and negative samples, and chooses the best classifier from each selector to form a strong classifier by linear combination.

$$h^{strong}(x) = \text{sign}\left(\sum_{n=1}^N \alpha_n \cdot h_n^{sel}(x)\right) \quad (1)$$

where $h_n^{sel}(x)$ is the best weak classifier in the n^{th} selector. α_n is the n^{th} selector's weight. $\text{sign}(\cdot)$ represents the sign function. (2) tracking: As given an initial position of the object, the stronger classifier is evaluated at many possible positions in a surrounding search region and build the confidence map which is analyzed in order to estimate the most probable position.

3 The tracking algorithm of CHAPF

3.1 Observation model

Observation model is the key factor to affect the tracking performance of particle filter. Because the weight of the particle is calculated according to the observation model, and the result of tracking depends on the weight of the particles.

3.1.1 Color-based observation model

Color feature is often used for object tracking due to its robustness to partial occlusion, rotation and scale invariant. We use the kernel color histogram in HSV color space as the color distribution of object whose center is y : $p(y) = \{p_u(y)\}_{u=1, \dots, m}$.

$$p_u(y) = C \sum_{i=1}^M k\left(\frac{\|y - x_i\|}{a}\right) \delta[b(x_i) - u] \quad (2)$$

C is a normalized constant, a is used to adapt the size of the region, M represents the total number of object region, u represents color histogram index, δ is the Kronecker delta function, $k(\cdot)$ is the Epanechnikov kernel function.

3.1.2 Haar-like based observation model

Haar feature is used for learning and training the classifier by many learning algorithms, because of describing the object structure in multi-scale model.

HSEMOB is one, it introduces in semi-supervised work which use the initial samples to train a prior classifier and an online classifier at the initialization frame. Prior classification plays a supervisory role, it determines the properties of a sample with online classifier together and it is never updated during tracking process. However, online classifier will be updated with new sample entry. Our paper sets Haar-like based strong classification as observation model and confidence as observation value. Confidence level is associated with object similarity. The greater the confidence level is, the bigger the similarity is. The formula of calculating confidence levels is described by the following equation:

$$conf(x) = \frac{\sum_{n=1}^N v_n \langle h_n^{sel}(x) \rangle}{\sum_{n=1}^N v_n} \quad (3)$$

3.2 Adaptive fusion strategy

In this paper, we use the additive method [10] to implement the adaptive features fusion. Assuming in the case of multiple features are independent, observation probability value of each feature is given an adaptive weight, and particle weights are calculated by the distance between the observation value of feature and model.

$$w_i = \varepsilon_1 w_i^1 + \varepsilon_2 w_i^2 \quad (4)$$

$$w_i^1 = C_1 (1 / \sqrt{2\pi} \sigma_1) e^{-\frac{d_1^2}{2\sigma_1^2}}, \quad w_i^2 = C_2 (1 / \sqrt{2\pi} \sigma_2) e^{-\frac{d_2^2}{2\sigma_2^2}} \quad (5)$$

$$d_1^2 = 1 - \rho(q, p), d_2^2 = 1 - (1 + conf(x)) / 2 \quad (6)$$

$$\bar{\varepsilon}_l = 1 / (\min(w_i^l))^2, l = 1, 2 \quad (7)$$

$$\varepsilon_l = \bar{\varepsilon}_l / \left(\sum_{l=1}^2 \bar{\varepsilon}_l \right), l = 1, 2 \quad (8)$$

1,2 represents color feature and Haar-like feature respectively, w_i^1 、 w_i^2 represent the feature weight, C_1 、 C_2 are normalization factor, ρ refers to the similarity between particle and target by using Butterworth Acharya coefficient to calculate.

3.3 Algorithm

Based on the above discussion, the procedure of the proposed algorithm of fusing color and haar-like feature based particle filter is as follows:

Step 1, Initialization: set the initial object state x_0 , establish color-based and haar-like-based observation model respectively and the particle weights are initialized to $1/N$.

Step 2, Propagation: particles Propagate through the dynamic model of system and get a new set of particles $S^{(i)}k$.

Step 3, Observation: calculate observation probability of each particle by its observation model and do normalized. Then calculate the feature weight of color and haar-like respectively.

Step 4, Output: obtain the weight of new particles according to (4), estimate the current state of the target $\bar{x} = \sum_{i=1}^N \bar{w}_t^i x_t^i$

Step 5, Update: update the observation model for each feature according to update strategy.

Step 6, Resample: Resample N particles from the set $S^{(i)}k$, return to step 2.

4 Experimental Results and Analysis

In this section, in order to make a comparison, we also realize the algorithm of Semi-Supervised On-line Boosting based particle filter (HSEMOBPF). All of them are based on particle filter techniques. We use the same particle filter implementation for the three approaches and change only the observation model. In order to highlight the advantages of the proposed algorithm, the algorithms were tested with videos of complex conditions.

4.1 Test of effectiveness and robustness

4.1.1 Test of effectiveness

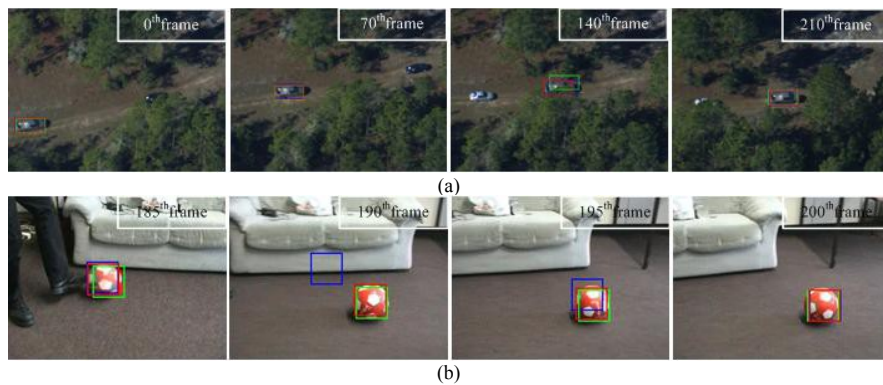


Fig. 1 Tracking results on test video. (a) (b) are the tracking results of video Egtest05 and Ball , respectively. Where the green box represents tracking results of CHGPF, blue represents

SEMOBPF, red represents CHAPF.

Experimental results show that Tracking lost is occurred in ball video for SEMOBPF. Tracking drift is occurred in Egtest05 for CHGPF. However, CHAPF have achieved effective tracking in two test videos. In Egtest05 video, as CHGPF utilizes color histogram as the object feature, it is difficult to distinguish the difference among the particles and results in tracking drift when the object is partially obscured and cause a very serious lack of color information. But the Haar-like-based HSEMOBPF is capable of multi-scale describing of the object structure, so it can distinguish the difference among the particles and inhibit tracking drift when the object is partially obscured in 85th frame. In Ball video, football is rolling too fast and the appearance model is changing in every moment, but the overall red and white based color information is still not changed. So, CHGPF which utilizes color histogram as the object feature can achieved effective tracking. HSEMOBPF which regards strong classifier based on haar-like features as observation model may cause tracking lost. As CHAPF combines color and Haar-like features, both of which play complementary roles, it can achieve stable tracking.

4.1.2 Test of Robustness

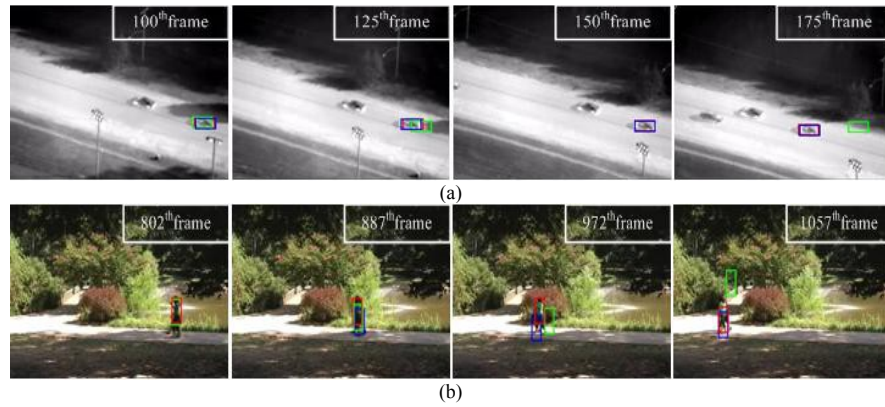


Fig. 2 Tracking results on test video. (a) and (b) are the tracking results of video Pktest02 and Park, respectively. Where the green box represents tracking results of CHGPF, blue represents SEMOBPF, red represents CHAPF.

Experimental results show that CHAPF outperforms CHGPF and HSEMOBPF when the object is similar to the background or in the mutable illumination scenarios. Taking video Pktest02 as an example, as color histogram is adopted as the object feature for CHGPF, it is difficult to distinguish the difference among the particles and results in tracking lost when the color of the object is similar to background. The proposed algorithm regards strong classifier based on haar-like

features as one observation model, it can select the characteristics which can distinguish the object and background most accurately by training positive and negative samples many times. Thus CHAPF can still distinguish when the object and background are similar and achieve effective tracking.

4.2 Adaptive adjustment of feature weights

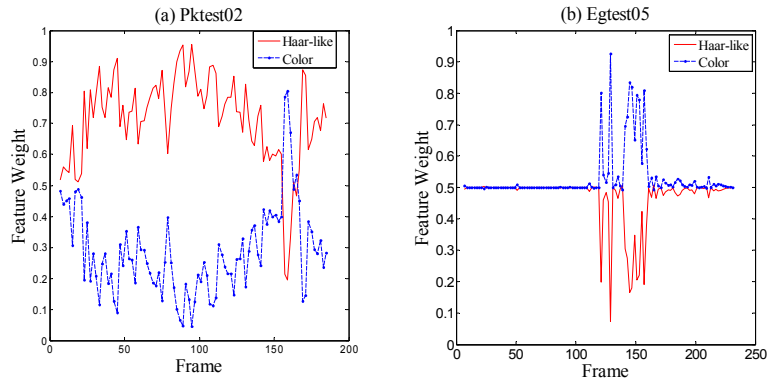


Fig. 3 Adjustment of feature weights. (a) and (b) are the feature weights of video Pktest02 and Egtest05, respectively.

From Fig.3 (a), we can see that the weight of the haar-like feature occupies a large proportion with an average value 0.7138, while color-based feature is an average value 0.2862 in whole tracking process of video Pktest02. Because the tracking results of Fig.2 (a) indicates that color-based observation cannot distinguish object and background well when the object share similar color with the background, resulting in tracking lost easily. But as haar-like based feature has the capable of multi-scale description of the object structure, it can well distinguish object from background with similar color. So in the process of tracking, a small proportion is adaptively allocated to the color feature, while a large proportion of allocation to haar-like feature. And Figure 3 (b) shows that in whole tracking process of video Egtest05, the proportions of two different observation models are both close to 0.5. However, weight percentage changes during the process of 120th frame to 170th frame, the average weight of color-based observation is 0.6592, while the average weight of haar-like based observation model is 0.3408. Because the main color information of car is not lost, when the car is obscured severely in 120th frame, so color-based observation model still can distinguish the difference among the particles, while the haar-like based observation model cannot. Therefore the system adaptively allocates a larger proportion to color-based observation model to ensure that stable tracking of system.

5 Conclusion and future work

In order to solve the robustness problem of tracking in the changing scene, a novel object tracking algorithm of fusing color and haar-like feature based particle filter is proposed. The weight of each feature is adjusted adaptively. Experiments have verified that the proposed CHAPF algorithm has stronger effectiveness and higher efficiency over CHGPF and HSEMOBPF. But a weak classifier pool generated randomly will produce a large number features with weak classification capability or redundancy, and affect the classification performance. Therefore, how to optimize the features is the key directions of future work. Furthermore, realizing Multi-object tracking and enhancing efficiency of multi-feature fusion algorithm are also a focus of future work.

Acknowledgments This study is funded by NSFC (No.61102155, No.61272237, No. 61272236), OYMI Research Team Plan of Hubei Province of China (No.T201002).

References

1. A. Yilmaz, O. Javed, M. Shah. Object tracking: A survey. *ACM Computing Surveys*, 2006, 38(4): 1-45.
2. K. Nummiaro, E. Koller-Meier, L. V. Gool. An adaptive color-based particle filter. *Image and Vision Computing*, 2003, 21(1): 99-110.
3. R. Bastos, J. M. S. Dias. Fully automated texture tracking based on natural features extraction and template matching. *Proceedings of the ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, 2005: 180-183.
4. W. Du, J. Piater. A probabilistic approach to integrating multiple cues in visual tracking. *Proceedings of the 10th Europe on Conference on Computer Vision*, 2008: 225-238.
5. P. Perez, J. Vermaak, A. Blake. Data fusion for visual tracking with particles. *Proceedings of the IEEE*, 2004, 92(3): 495-513.
6. R. T. Collins, Y. Liu, M. Leordeanu. Online selection of discriminative tracking features. *IEEE Transactions on PAMI*, 2005, 27(10): 1631-1643.
7. H. Grabner, H. Bischof. On-line boosting and vision. *Proceedings of 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006(1): 260-267.
8. H. Grabner, C. Leistner, H. Bischof. Semi-supervised on-line boosting for robust tracking. *Proceedings of 10th European Conference on Computer Vision*, 2008: 234-247.
9. C. H. Shen, A. V. D. Hengel, A. Dick. Probabilistic multiple cue integration for particle filter based tracking. *Proceedings of 7th Digital Image Computing: Techniques and Applications*, 2003, 399-408.
10. M. Spengler, B. Schiele. Towards robust multi-cue integration for visual tracking. *International Journal of Machine Vision and Applications*, 2003, 14(1): 50-58.
11. P. H. Li, F. Chaumette. Image cues fusion for object tracking based on particle filter. *Lecture Notes in Computer Science*, 2004, 3179: 99-110.