Long Time Target Tracking Algorithm Based on Multi Feature Fusion and Correlation Filtering

Junsuo Qu*
Xi'an University of Posts and Telecommunications, China
qujunsuo@xupt.edu.cn

Yuan Zhang*
Xi'an University of Posts and Telecommunications, China
976704160@qq.com

Kai Zhou
Xi'an University of Posts and Telecommunications, China
1092479804@qq.com

Abolfazl Razi
Northern Arizona University
Abolfazl.Razi@nau.edu

ABSTRACT

This paper considers the problem of long-term target tracking in complex scenes when tracking failures are unavoidable due to illumination change, target deformation, scale change, motion blur, and other factors. More specifically, we propose a target tracking algorithm, called Re-detection Multi-feature Fusion (RDMF), based on the fusion of Scale-adaptive kernel correlation filtering and re-detection. The target tracking algorithm trains three kernel correlation filters based on HOG, CN and LBP features, and then obtains the fusion weight of response graphs corresponding to different features based on APCE criterion, and uses weighted Average to complete the position estimation of the tracked target. In order to deal with the problem that the target is occluded and disappears in the tracking process, a random fern classifier is trained to perform re-detection when the target is occluded. After comparing the OTB-50 target tracking data set, the RDMF algorithm improves the range accuracy by 10.1% compared with SAMF algorithm, and is better than KCF, DSST, CN and other algorithms.

CCS CONCEPTS

• **Computer systems organization**; • **Real-time system architecture**;

KEYWORDS

Multi-feature fusion, Random fern classifier, RDMF algorithm

ACM Reference Format:
Junsuo Qu*, Yuan Zhang*, Kai Zhou, and Abolfazl Razi. 2021. Long Time Target Tracking Algorithm Based on Multi Feature Fusion and Correlation Filtering. In *2021 4th International Conference on Artificial Intelligence and Pattern Recognition (AIPR 2021), September 24–26, 2021, Xiamen, China.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3488933.3488996

1 INTRODUCTION

In recent years, due to the fast-computing characteristics of correlation filtering in the frequency domain, the computing speed of target tracking algorithm has reached hundreds of frames per second, which make the correlation-filtering target tracking algorithm more popular than before [1-3]. A good correlation filter can produce correlation peak at the target position and low response at the background position [4]. Mosse (minimum output sum of squared error) algorithm proposed by Bolme is the founder of this kind of tracking algorithm. Henriques improved the Mosse algorithm and proposed the CSK (circular structure with kernels) algorithm. Henriques also proposed the KCF (kernelized correlation filters) algorithm to improve the CSK algorithm [5-6]. Danelljan et al. Proposed the color name CN (color name) algorithm [7-8].

In recent years, with the continuous development of computer vision technology, various target tracking algorithms have been put forward, and the performance of the tracker has been significantly improved. However, in engineering applications, it often faces various complex scenes, such as illumination change, target occlusion, target deformation and scale change. To address these challenges, based on the summary of various target tracking algorithms at home and abroad [9-10], a long-time target tracking algorithm based on multi feature fusion and scale adaptive transformation is proposed in this paper. The correlation filtering algorithm is improved from three aspects: target appearance feature description, scale estimation and target relocation after occlusion [11-13].

2 LONG TIME TARGET TRACKING BASED ON MULTI-FEATURE FUSION

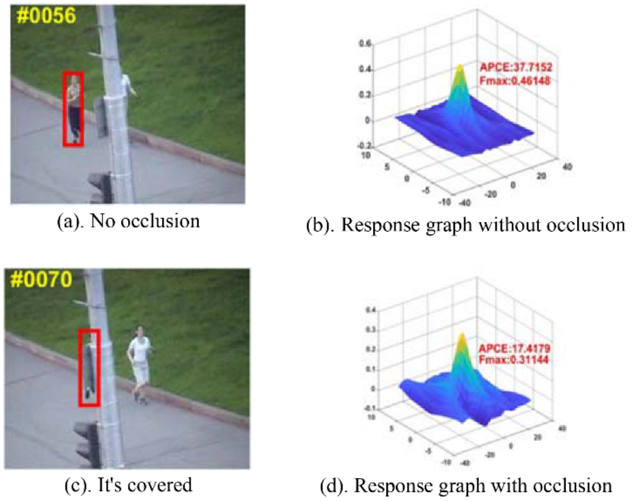
Robust representation of the target is an essential part of the target tracking task. Target tracking based on a single feature is not suitable for all scenarios [14]. Therefore, we use the fusion of multiple feature domains along with the target re-detection for target tracking in this paper. More specifically, the robust model representation of the target is established by multi-feature fusion [15], when a target is occluded, the re-detector is activated to prevent target loss.

2.1 Multi-feature fusion based on APCE

We propose to evaluate the tracking performance of the tracker based on the APCE criterion [16]. The APCE value of the response graph reflects the flatness of the response graph. As shown in Fig.1(a) and Fig.1(b), when the target is correctly tracked, the maximum value and the APCE value of the response graph are 0.46 and 37.71 respectively, and there is only one sharp peak in the response graph and the rest of the response graph is flat. When the tracking quality of the target is degraded due to occlusion, as shown in Fig.1 (c) and Fig.1 (d), the maximum and APCE values of the response

AIPR 2021, September 24–26, 2021, Xiamen, China

ACM ISBN 978-1-4503-8408-7/21/09…$15.00
https://doi.org/10.1145/3488933.3488996



(a). No occlusion

(b). Response graph without occlusion

(c). It's covered

(d). Response graph with occlusion

Figure 1: Diagram of APCE value and maximum response under different tracking quality.

graph are 0.31 and 17.41 respectively. At this time, there is a peak value in the response graph, but the rest of the response graph is not flat and includes considerable fluctuations.

The APCE criterion is calculated as:

$$APCE = \frac{|F_{max} - F_{min}|^2}{mean\left(\sum_{w,h}\left(F_{w,h} - F_{min}\right)^2\right)} \tag{1}$$

where, F_{max}, and F_{min} represent the maximum and minimum values in the response graph respectively, and $F_{w,h}$ represents the response value at the position of column h in row w of the response graph.

After getting the corresponding response graphs of HOG, CN and LBP, we can get the final response graph by weighted averaging the corresponding response graphs of each feature according to the APCE criterion, and select the position with the largest response value in the final response graph as our estimated target in the current frame position. The formula is:

$$res = \left(A_{hog}res_{hog} + A_{cn}res_{cn} + \ A_{lbp}res_{lbp}\right)/\lambda \tag{2}$$

where, $\lambda = A_{hog} + A_{cn} + A_{lbp}$, A_{hog}, A_{cn}, A_{lbp} respectively represent the APCE value of the response graph corresponding to each feature, res_{hog},res_{cn},res_{lbp} respectively represent the response graph corresponding to each feature.

2.2 Scale adaptive strategy

The training process of the scale filter is as follows: at the beginning of the tracking task, an initial tracking box is calibrated around the target under tracking, and then training samples with different scales are generated around the center of the target as the input of the scale filter [17], Then, S image blocks with different scales (here we have $S = 33$) are extracted. The scale size of training samples follows formula$a^n M \times a^n N$, where, $n \in \{-\frac{S-1}{2}, \ldots \frac{S-1}{2}\}$ and $a = 1.02$ represent the scale factor. We extract HOG features from the training samples of different scales obtained, in this way.

2.3 Construction of re-detector

In this paper, a re-detector is trained to deal with the problem of that the target being disappear due to occlusion, which eventually leads to the tracking failure [18]. In the training phase, the positive samples are generated by selecting 10 image frames closest to the target. Then, we use data augmentation by applying scaling, rotation, and affine transformation to each frame to obtain more positive samples. The image frames far away from the target are selected as negative samples [19].

Let the training sample set be $X = \{x_1, x_2, \ldots x_K\}$, and the samples are divided into positive and negative samples, representing the target and non-target respectively, which are represented by class labels $C = \{c_1, c_2\}$. F_j stands for the jth fern tree, where, $j = 1, 2, \ldots 10$. The training process of random fern cluster classifier is as follows:

Step 1: Input positive or negative samples x_j as training samples into fern tree F_j. Each layer of the fern tree corresponds to the 2-bit LBP feature at the corresponding position of training samples, i.e., $f_d = \{00, 01, 10, 11\}$,where, $d = 1, 2, \ldots 8$ corresponds to layer d of the fern tree

Step 2: The eigenvalues of all layers of fern tree F_j are counted to get the eigenvalue $B_j = f_8 LBP$, where, $B_j \in \{0, 1, 2, \ldots 4^8\}$.

Step 3: For the sample c_i whose sample category is x_i, we can get the eigenvalues of the sample in the last layer of the fern tree, we can get the histogram of the eigenvalues corresponding to a single fern by counting the eigenvalues of all training samples. The abscissa is the eigenvalue, and the ordinate represents the number of times that the eigenvalue appears in the training process.

Step 4: The Step 3 completes the histogram statistics of the eigenvalues of a single fern [20]. Step 1-3 is executed 10 times to complete the histogram statistics of the eigenvalues of the training samples.

If S is the test sample and the sample category is represented by C, the process of classifying the test sample to determine whether or not it is a candidate sample is as follows:

Step 1: Input the sample into the random fern classifier k_j to get the 2-bit-lbp eigenvalue k_j of the test sample.

Step 2: Find the maximum likelihood estimation of k_j, using the following equation:

$$p_j^{c_t} = p(C = c_t|k_j) = \frac{N_j^{k_j c_t}}{N_j^{c_t}}, t = 1, 2 \tag{3}$$

Here, $N_j^{c_t}$ represents the sum of the histogram frequencies corresponding to the jth fern tree of category c_t, and $N_j^{k_j c_t}$represents the occurrence frequency of the characteristic value k_j in the histogram.

Step 3: If the final output meets $\frac{1}{n}\sum_{j=1}^{n} p(C = c_t|k_j) > 0.6$, for $n = 10$, then the test sample belongs to class c_t.

Through the above three steps, we can classify the test samples, but after executing the random fern classifier, we will still get multiple candidate samples. Therefore, we need to select the best sample from these candidate samples as our re-detection target. Here, we calculate the correlation between the candidate samples and the sample set based on a KNN classifier, i.e., the Euclidean distance between the selected and the candidate samples. Then,

Table 1: Video Properties

Attribute classification	Property description
IV (Illumination Variation)	Light change
SV (Scale Variation)	Scale change
OCC(Occlusion)	Occlusion
DEF(Deformation)	Deformation
MB (Motion Blur)	Motion blur
FM (Fast Motion)	Fast motion
IPR (In-Plane Rotation)	Plane rotation
OPR (Out-of-Plane Rotation)	Solid rotation
OV(Out-of-View)	The target is out of view of the camera
BC (Background Clutters)	The background of the target is complex
LR (Low Resolution)	low resolution

according to the category of these five training samples, we decide whether or not the candidate samples are the targets we want to detect.

2.4 Algorithm flow

The algorithm flow is as follows:

Input: the initial position $P_0 = (x_0, y_0, s_0)$ of the target, where, (x_0, y_0) represents the center position coordinate of the target and s_0 represents the target scale.

Output: estimated target position $P_t = (x_t, y_t, s_t)$.

For $t = 1$ to T (T is the total count of frames of the input video frames)

Location estimation:

Extract an image block in the 2nd frame. The center of the image block is the location of the target in the previous frame, i.e. (x_{t-1}, y_{t-1}), and the length and width of the image block are 2.5 times of the length and width of the target in the previous frame. The HOG, CN and LBP features of the image block and its virtual samples generated by translation are extracted respectively.

Calculate response matrices $f_{hog}, f_{cn},\ f_{lbp}$ for corresponding to each feature map based on the corresponding filter, where, the response matrices $f_{hog},\ f_{cn},\ f_{lbp}$ are corresponding to the response graphs $res_{hog},\ res_{cn}, res_{lbp}$.

Assign the weight of each response graph based on the APCE criterion to get the final response graph. The position of the maximum response value in the final response graph is the estimated target position (x_t, y_t).

Scale estimation:

Taking (x_t, y_t) as the center, cut out the image blocks of different scales around the center, and extract their HOG features. The image blocks of different scales are correlated with the scale filter, and the image block size with the largest response value is selected as the size s_t of the current frame target.

Determine the target location:

Calculate the APCE value for each feature according to (1). When A_{hog}, A_{cn}, A_{lbp} are all less than 0.45 times of the historical mean value, we conclude that the performance of the tracker is poor due to the occlusion of the target. Therefore, the re-detector is activated to re-locate the target position $P_t^{'} = (x_t^{'}, y_t^{'}, s_t^{'})$ and $P_t = P_t^{'}$; otherwise, the re-detector is not activated, and P_t is the result of the position and scale estimations.

Model update:

When $\mathrm{A_{hog}}, \mathrm{A_{cn}}, \mathrm{A_{lbp}}$ are greater than or equal to 0.45 times of the historical mean value, we conclude that the target position is more accurate. At this time, we use the results of the tracker to train the position filter and scale filter. Also, positive and negative samples are collected around the target as training samples of the re-detector. We update the re-detector every time we collect 200 training samples.

3 EXPERIMENTAL RESULTS AND ANALYSIS

The experimental conditions of RDMF algorithm are as follows:

- Hardware conditions: the CPU is Intel (R) core (TM) i5, the main frequency is 2.4GHz, and the memory is 8GB.
- Software conditions: Windows 7 operating system, the code environment is MATLAB 2016a.
- Test video: OTB-50 video test sequence in visual tracker benchmark library.

In order to evaluate the efficacy of the proposed tracking algorithm in a comprehensive way, the data in the video library are marked with attributes. There are 11 kinds of video attributes, as shown in Table 1

Each group of videos in the video set contains many of these 11 attributes, enabling us to evaluate the tracking algorithm objectively. The evaluation method includes both qualitative and quantitative evaluations.

Firstly, the algorithm is evaluated quantitatively. Since the algorithm tracks the target based on the multi-feature fusion strategy, considering the robustness of the HOG, LBP, and CN feature maps, respectively, to illumination, motion blur, and target deformation, combining these three complementary features is beneficial to deal with the illumination change, target deformation and motion blur altogether. OPE (One Pass Evaluation) represents the experimental results, which are drawn based on the tracking algorithm after one run on the data set.

In 2, we compare the RDMF algorithm with classical target tracking algorithms such as KCF, DSST, SAMF, TLD and CSK. It can be seen that due to the fusion of multiple features, the tracking accuracy of the proposed algorithm is among the best in the three

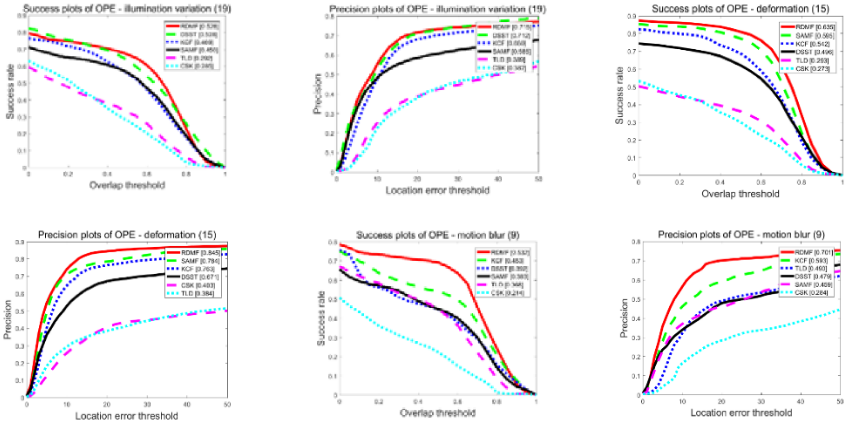


Figure 2: Comparison of IV, DEF and MB.

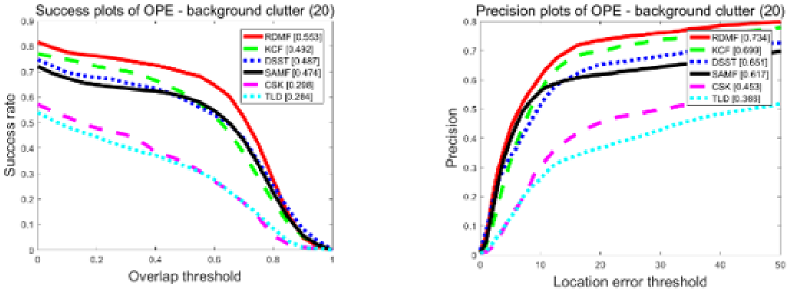


Figure 3: Comparison of the proposed RDMF algorithm with commonly-used target tracking methods in terms of dealing with complex backgrounds.

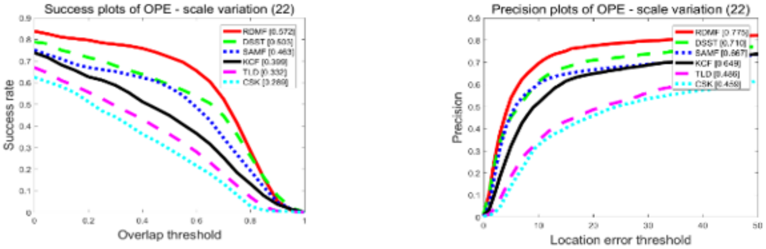


Figure 4: Comparison of the proposed RDMF algorithm with commonly-used target tracking methods in terms of dealing with scale change

challenging scenarios of illumination change, target deformation and motion blur. Due to using the multi-feature fusion, the proposed algorithm exhibits a good tracking accuracy in the case of complex target background, as shown in Fig.4:

As can be seen in 3, when dealing with a complex background, the overlapping success rate of the RDMF algorithm is 6.1% higher than that of KCF, and the range accuracy is 3.5% higher than that of KCF. The main reason is that the KCF only uses HOG features to describe the target, while our algorithm uses HOG, CN, and LBP features to describe the target. The advantage of this algorithm is not easy to introduce background noise when the object is similar. 4 presents the performance of the algorithms when subjected to the target scale change. It can be seen that, the overlapping success rate of the proposed RDMF algorithm is 6.9% higher than DSST algorithm equipped with scale estimation. Also, the range accuracy is 6.5% higher than the DSST algorithm. The justification is that the center position filter of the RDMF algorithm is trained based on three features, while the DSST algorithm only uses HOG features, therefore, the accuracy of the RDMF algorithm is higher in determining the center position of the target. The subsequent scale estimation is based on the center position of the image multi-scale

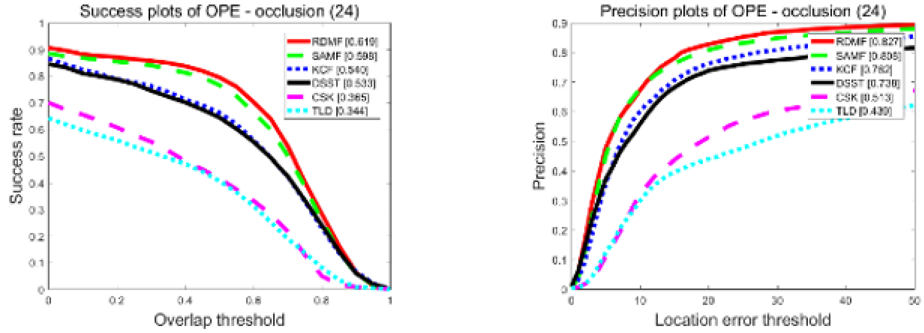


Figure 5: Comparison of the proposed RDMF algorithm with commonly-used target tracking methods in terms of coping with occlusion effect

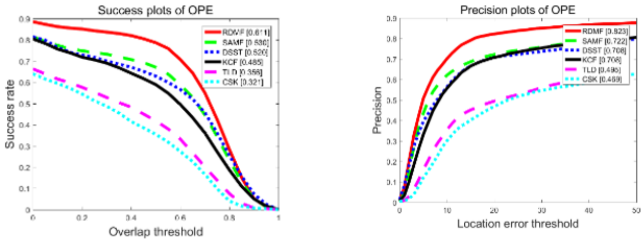


Figure 6: Overall effect comparison.

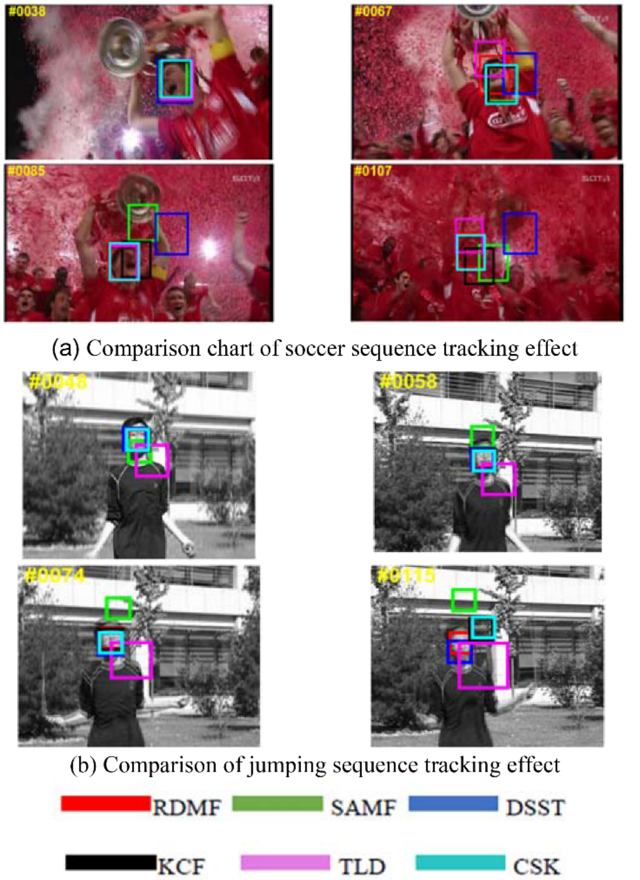
sampling, selecting the sample block with the best scale as the best scale of the target.

5 presents the operation of different algorithms subject to target occlusion. It can be seen that the overlapping success rate of the RDMF algorithm is 2.1% higher than that of the SAMF algorithm, and the distance accuracy is 1.9% higher than that of the SAMF algorithm. This is due to the using a re-detector in the RDMF algorithm. When the target is occluded and lost, the re-detector will start to search for the target, and when the target reappears, the re-detector will remove the target from the target, that is why, the anti-occlusion ability of the algorithm is improved.

6 shows the overall quantitative evaluation effect comparison chart of each algorithm. We observe that the overlapping success rate of the RDMF algorithm is 8.1% higher than the SAMF algorithm, and the distance accuracy is 10.1% higher than the SAMF algorithm.

In addition to the presented quantitative results, we evaluate each algorithm's tracking effect in a more intuitive qualitative manner.

The two sequences in 7 present the soccer sequence and jumping sequence in OTB-50. In the soccer sequence, we see the results each tracking algorithm when the target encounters the motion blur and complex background. In the 85th frame of the soccer sequence, the target becomes blurred due to the person's jump, and in the 107th frame, the target suffers from the interference of background noise. We can see the effect of this algorithm on these two sequences, all the challenges have good robustness. In the jumping sequence, the tracking target is always suffering from the motion blur due to the continuous jumping of characters. We observe that the RDMF algorithm can perfectly trace the target.

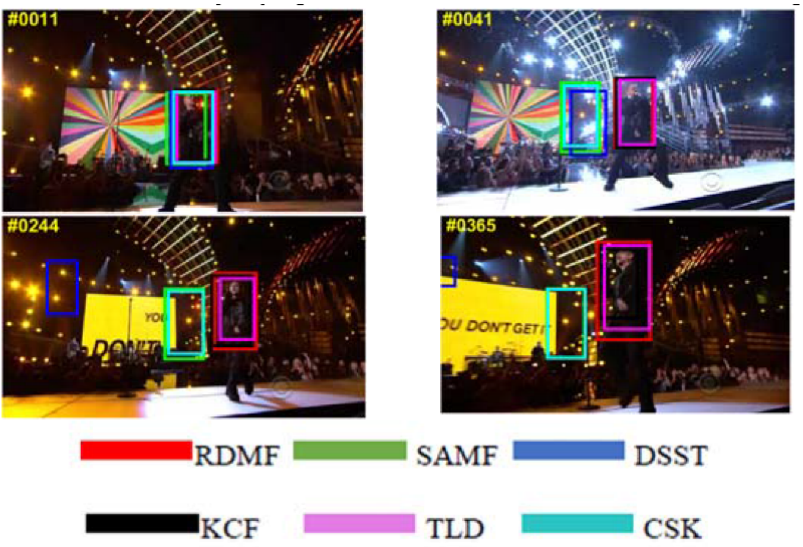


(a) Comparison chart of soccer sequence tracking effect

(b) Comparison of jumping sequence tracking effect

Figure 7: Comparison of effect of soccer sequence and jumping sequence.

The sequence in 8 is the singer2 sequence in OTB-50. It can be seen that the drastic change of illumination in the 41st frame leads to the failure of some trackers. In the 244th and 365th frames, it



8

Figure 8: Comparison of singer2 sequence effect

can be seen that the continuous movement of the target causes the constant deformation of the target. However, the RDMF algorithm can still follow the target robustly.

4 CONCLUSION

In this paper, we used the idea of fusing feature maps and invoking re-detection of targets to implement a robust long-term target tracking algorithm called RDMF. The results show that the proposed algorithm outperforms the most recently-developed target trackers, especially when the target is subject to scale change, extreme illumination variation, and occlusion. Compared with the SAMF algorithm, the overlap success rate and the range accuracy of the RDMF algorithm are improved by 8.1% and 10.1%, respectively. SAMF is a scale adaptive target tracking algorithm, which combines HOG and CN features. Compared with the SAMF, we also integrate LBP features to improve the proposed algorithm's robustness to target motion blur. Besides, we also offered to train a re-detector online to enhance the algorithm's robustness against occlusion.

ACKNOWLEDGMENTS

This research work was supported in part by grants from National Natural Science Foundation of China under Grant 51875457, International Cooperation and Exchange Program of Shaanxi Province under Grant 2018KW-026, Natural Science Foundation of Shaanxi Province under Grant (2018JM6120, 2019JM-606), and Xi'an Science and Technology Projects under Grant 201805040YD18CG24(6). Technicians at School of Automation in Xi'an University of Posts & Telecommunications, who produced devices and tools for the test and who helped to conduct the experiments are also acknowledged.

REFERENCES

[1] Fan H, Lin L, Yang F, *et al.* Lasot: A high-quality benchmark for large-scale single object tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 5374-5383.

[2] Danelljan M, Robinson A, Khan F S, *et al.* Beyond correlation filters: Learning continuous convolution operators for visual tracking[C]//European conference on computer vision. Springer, Cham, 2016: 472-488.

[3] Wu Y, Lim J, Yang M H. Object tracking benchmark[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1834-1848.

[4] Bolme D S, Beveridge J R, Draper B A, *et al.* Visual object tracking using adaptive correlation filters[C]. IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2010: 2544-2550.

[5] Henriques J F, Rui C, Martins P, et al. Exploiting the Circulant Structure of Tracking- by- Detection with Kernels[J]. Lecture Notes in Computer Science, 2012, 7575(1):702-715.

[6] Henriques J F, Caseiro R, Martins P, et al. High-Speed Tracking with Kernelized Correlation Filters[J]. IEEE Transactions on Pattern Analysis&Machine Intelligence, 2014, 37(3):583-596.

[7] Danelljan M, Khan F S, Felsberg M, *et al.* Adaptive color attributes for real-time visual tracking[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2014:1090-1097.

[8] Danelljan M, Häger G, Khan F, *et al.* Accurate scale estimation for robust visual tracking[C]//British Machine Vision Conference, Nottingham, September 1-5, 2014. BMVA Press, 2014.

[9] Li Y, Zhu J. A scale adaptive kernel correlation filter tracker with feature integration[C]//European conference on computer vision. Springer, Cham, 2014: 254-265.

[10] Ma C, Huang J B, Yang X, *et al.* Adaptive correlation filters with long-term and short-term memory for object tracking[J]. International Journal of Computer Vision, 2018, 126(8): 771-796.

[11] Wang M, Liu Y, Huang Z. Large margin object tracking with circulant feature maps[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 4021-4029.

[12] Danelljan M, Hager G, Shahbaz Khan F, *et al.* Learning spatially regularized correlation filters for visual tracking[C]//Proceedings of the IEEE international conference on computer vision. 2015: 4310-4318.

[13] Li F, Tian C, Zuo W, *et al.* Learning spatial-temporal regularized correlation filters for visual tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 4904-4913.

[14] Bolme D S, Beveridge J R, Draper B A, *et al.* Visual object tracking using adaptive correlation filters[C]. IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2010: 2544-2550.

[15] Saffari A, Leistner C, Santner J, *et al.* On-line random forests[C]//2009 ieee 12th international conference on computer vision workshops, iccv workshops. IEEE, 2009: 1393-1400.

[16] Henriques J F, Caseiro R, Martins P, et al. High-Speed Tracking with Kernelized Correlation Filters[J]. IEEE Transactions on Pattern Analysis&Machine Intelligence, 2014, 37(3):583-596.

[17] Ma C, Huang J B, Yang X, et al. Hierarchical convolutional features for visual tracking[C]. Computer Vision and Pattern Recognition(CVPR), 2015 IEEE Conference on. IEEE, 2015: 3074-3082.

[18] Ozuysal M, Calonder M, Lepetit V, *et al.* Fast keypoint recognition using random ferns[J]. IEEE transactions on pattern analysis and machine intelligence, 2009, 32(3): 448-461.

[19] Danelljan M, Hager G, Shahbaz Khan F, *et al.* Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 1430-1438.

[20] Cai Xiumei, Wang Yan, Bian Jingwei, Wu CHENGMAO. Overview of multi-target tracking data association algorithms [J]. Journal of Xi'an University of Posts and telecommunications, 2021,26 (02): 77-86.