



Anti-occlusion object tracking based on correlation filter

Jun Liu^{1,2} · Gang Xiao¹ · Xingchen Zhang¹ · Ping Ye¹ · Xingzhong Xiong² · Shengyun Peng^{1,3}

Received: 17 May 2019 / Revised: 26 September 2019 / Accepted: 11 November 2019 / Published online: 27 November 2019
© Springer-Verlag London Ltd., part of Springer Nature 2019

Abstract

Despite remarkable progress, visual object tracking is still a challenging task as objects usually suffer from significant appearance changes, fast motion, and serious occlusion. In this paper, we propose an anti-occlusion correlation filter-based tracking method (AO-CF) for robust visual tracking. We first propose an occlusion criterion based on continuous response values. Based on the criterion, objects are divided into four categories to adaptively identify the occlusion of objects. Then we propose a new detection condition for detecting proposals. When the occlusion criterion is triggered, the re-detection mechanism is executed and the tracker is commanded to stop, and then the re-detector selects the most reliable proposal to reinitialize the tracker. Experimental results show that our method outperforms other state-of-the-art trackers in terms of both precision rate and success rate on the widely used object tracking benchmark dataset. In addition, AO-CF is able to achieve real-time tracking speed.

Keywords Visual tracking · Circulant matrices · Correlation filter · Kernel methods · Occlusion

1 Introduction

Object tracking is one of the most popular fields in computer vision, with a wide range of applications in face recognition, behavior analysis and intelligent monitoring. Existing tracking algorithms can be roughly classified as generative and discriminative methods. The generative method seeks to consider tracking as a problem of finding the most similar region to the target. The target is represented as a template [1] or parameter model in feature space [2,3]. The similarity is measured in feature space or a low-dimensional subspace to describe the target and incrementally learn the subspace to adapt to appearance changes during tracking. The discriminative method formulates the tracking problem as a binary classification task whose goal is to discriminate the target from the background [4,5]. This classifier is trained online using sample patches of the target and the surrounding background.

Correlation filter-based method is known for accuracy and real-time (about 20 frame/s) tracking, so we use correlation filter as a basic tracker and improve it to handle object occlusion. The main contributions of this paper are as follows:

1. An occlusion criterion based on response values is proposed. Considering the different capabilities of different objects to resist environmental disturbances, we also propose the idea of classifying objects to further improve tracking performance.
2. In the re-detector, the EdgeBox [6] method is used to detect objects. To reduce computational complexity, the method of limiting the area is employed to filter the proposals. Besides, three conditions are set to select different detection thresholds to reduce the error detection probability of the re-detector.
3. Extensive experiments are carried out on the large-scale benchmark datasets [7,8] and the experimental results demonstrate the superiority of the proposed algorithm.

✉ Xingchen Zhang
xingchen@sjtu.edu.cn

¹ School of Aeronautics and Astronautics, Shanghai Jiao Tong University, Shanghai 200240, China

² School of Automation and Information Engineering, Sichuan University of Science and Engineering, Yibin 644000, China

³ College of Civil Engineering, Tongji University, Shanghai 20092, China

2 Related work

Since inception of the MOSSE tracker [10], several advances have made discriminative correlation filter the most widely used methodology in short-term tracking. Major boosts in

performance are due to introduction of kernels [9], multi-channel [11,12] and scale estimation [13,14]. In addition to hand-crafted features, correlation filtering has recently been combined with neural networks to exploit the potential of deep features. The ACFN [15] with attentional feature-based correlation filter network is proposed to track the objects with fast variation. Besides, the DCFNet [16] combines discriminant correlation filter and Siamese network to ensure the tracking speed while improving the accuracy.

There exist several algorithms that are able to handle drift and occlusion. Hare et al. [17] propose a structured output support vector machine to estimate the target's location directly. In [18], a template matching algorithm for object tracking is proposed, where the template is composed of several cells and has two layers to capture the target and background appearance, respectively. The mean image intensity difference between two consecutive frames and a threshold are used to detect occlusion in [19]. Kalal et al. [20] decomposes the long-term tracking into three components: tracking-learning-detection (TLD). Among the multiple classification results of the tracker, Zhang et al. [21] use a minimum entropy criterion to select the best tracking results to correct undesirable model updates and achieve the goal of solving drift problems. In addition, Zhang et al. [22] propose an output constraint transfer (OCT) method by modeling the distribution of correlation response in a Bayesian optimization framework to mitigate the drifting problem. To overcome the adverse effects of distorted data, Zhang et al. [23] further propose the latent constrained correlation filters (LCCF) and introduced a subspace alternating direction method of multipliers (ADMM) framework to solve the new learning model. Zhao et al. [24] propose an improved LCT tracker to handle occlusion.

3 The basic correlation filter tracking components

Suppose one-dimensional data $\mathbf{x} = [x_1, x_2, \dots, x_n]$, a cyclic shift of \mathbf{x} is $\mathbf{P}\mathbf{x} = [x_n, x_1, x_2, \dots, x_{n-1}]$. Therefore, all the cyclic shift visual samples, $\{\mathbf{P}^u\mathbf{x} | u = 0 \dots n-1\}$, are concatenated to form the data matrix $\mathbf{X} = C(\mathbf{x})$. To mitigate the boundary effect, the input image is multiplied by a cosine window. The circulant matrix has an interesting property, which is expressed as $\mathbf{X} = \mathbf{F}^H \text{diag}(\mathbf{F}\mathbf{x})\mathbf{F}$. The \mathbf{F} is known as the DFT matrix, which transforms the data into Fourier domain, and \mathbf{F}^H is the Hermitian transpose of \mathbf{F} . The objective function of linear ridge regression can be formulated as:

$$\min_{\mathbf{w}} \sum_i^n (f(\mathbf{x}_i) - y_i)^2 + \lambda \|\mathbf{w}\|^2 \quad (1)$$

where λ is a regularization parameter and function $f(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$ is used to minimize the squared error over samples \mathbf{x}_i and their regression targets y_i . The ridge regression has the closed-form solution, $\mathbf{w} = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y}$. Substituted by Eq. 1, we have the solution $\hat{\mathbf{w}}^* = \frac{\hat{\mathbf{x}}^* \odot \hat{\mathbf{y}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda}$, where $\hat{\mathbf{x}} = \mathbf{F}\mathbf{x}$ denotes the DFT of \mathbf{x} , and $\hat{\mathbf{x}}^*$ denotes the complex-conjugate of $\hat{\mathbf{x}}$. In the case of no-linear regression, kernel trick, $f(\mathbf{z}) = \mathbf{w}^T \mathbf{z} = \sum_{i=1}^n \alpha_i \kappa(\mathbf{z}, \mathbf{x}_i)$, is applied to allow more powerful classifier. For the most commonly used kernel functions, the circulant matrix trick can also be used [9]. The dual space coefficients α can be learnt as $\hat{\alpha}^* = \frac{\hat{\mathbf{y}}}{\hat{\mathbf{k}}^{\mathbf{x}\mathbf{x}} + \lambda}$, where $\hat{\mathbf{k}}^{\mathbf{x}\mathbf{x}}$ is defined as kernel correlation in [9]. In this paper, we adopt the Gaussian kernel in which the circulant matrix trick can be applied as below:

$$\hat{\mathbf{k}}^{\mathbf{x}\mathbf{x}'} = \exp\left(-\frac{1}{\delta^2} \left(\|\mathbf{x}\|^2 + \|\mathbf{x}'\|^2 - 2\mathbf{F}^{-1}(\hat{\mathbf{x}} \odot \hat{\mathbf{x}'})\right)\right) \quad (2)$$

where δ is the Gaussian kernel bandwidth. The circulant matrix trick can also be applied in detection to speed up the whole process. The patch \mathbf{z} at the same location in the next frame is treated as the base sample to compute the response in Fourier domain,

$$\hat{\mathbf{f}}(\mathbf{z}) = (\hat{\mathbf{k}}^{\tilde{\mathbf{x}}\mathbf{z}})^* \odot \hat{\alpha} \quad (3)$$

where $\tilde{\mathbf{x}}$ denotes the data to be learnt in the model. When we transform $\hat{\mathbf{f}}(\mathbf{z})$ back into the spatial domain, the translation with respect to the maximum response is considered as the movement of the tracked target.

For the scale variation of the objects, we apply the method of the scale pool [14]. In this method, the template size is fixed as $\mathbf{s}_T = (s_x, s_y)$, and the scaling pool is defined as $\mathbf{S} = \{t_1, t_2, \dots, t_k\}$. Suppose that the target window size is \mathbf{s}_t in the original image space. For the current frame, we sample k sizes in $\{t_i \mathbf{s}_t | t_i \in \mathbf{S}\}$ to find the proper target. The samples can be resized into the fixed template size \mathbf{s}_T by using bilinear-interpolation, and the final response is calculated by

$$\arg \max \mathbf{F}^{-1} \hat{\mathbf{f}}(\mathbf{z}^{t_i}) \quad (4)$$

where \mathbf{z}^{t_i} is the sample patch with the size of $t_i \mathbf{s}_t$, which is resized to \mathbf{s}_T . Since the response function obtains a vector, the max operation is employed to find its maximum scalar. As the target movement is implied in the response map, the final displacement needs to be tuned by t to get the real movement bias.

4 The occlusion criterion

In Table 1, we list the response value τ of the target in the second frame in the partial sequences and also give the ratio μ

Table 1 The response value of the second frame and the ratio of the object size in the first frame

Sequence	<i>Basketball</i>	<i>Bird1</i>	<i>Box</i>	<i>Boy</i>	<i>Couple</i>	<i>Cardark</i>	<i>David</i>	<i>Doll</i>	<i>Faceoccl</i>
τ	0.767	0.366	0.848	0.747	0.666	0.545	0.842	0.826	0.856
μ	0.011	0.004	0.029	0.004	0.020	0.009	0.065	0.019	0.182
Sequence	<i>Freeman4</i>	<i>Fish</i>	<i>Girl2</i>	<i>Human5</i>	<i>Human6</i>	<i>Human7</i>	<i>Jogging-1</i>	<i>Jumping</i>	<i>Lemming</i>
τ	0.634	0.802	0.913	0.496	0.683	0.776	0.725	0.567	0.950
μ	0.003	0.069	0.025	0.002	0.003	0.056	0.041	0.011	0.020
Sequence	<i>Liquor</i>	<i>Ironman</i>	<i>Mhyang</i>	<i>Shaking</i>	<i>Sylvester</i>	<i>Skiing</i>	<i>Subway</i>	<i>Panda</i>	<i>Toy</i>
τ	0.929	0.480	0.882	0.711	0.925	0.367	0.688	0.749	0.922
μ	0.049	0.013	0.057	0.019	0.041	0.003	0.010	0.009	0.035

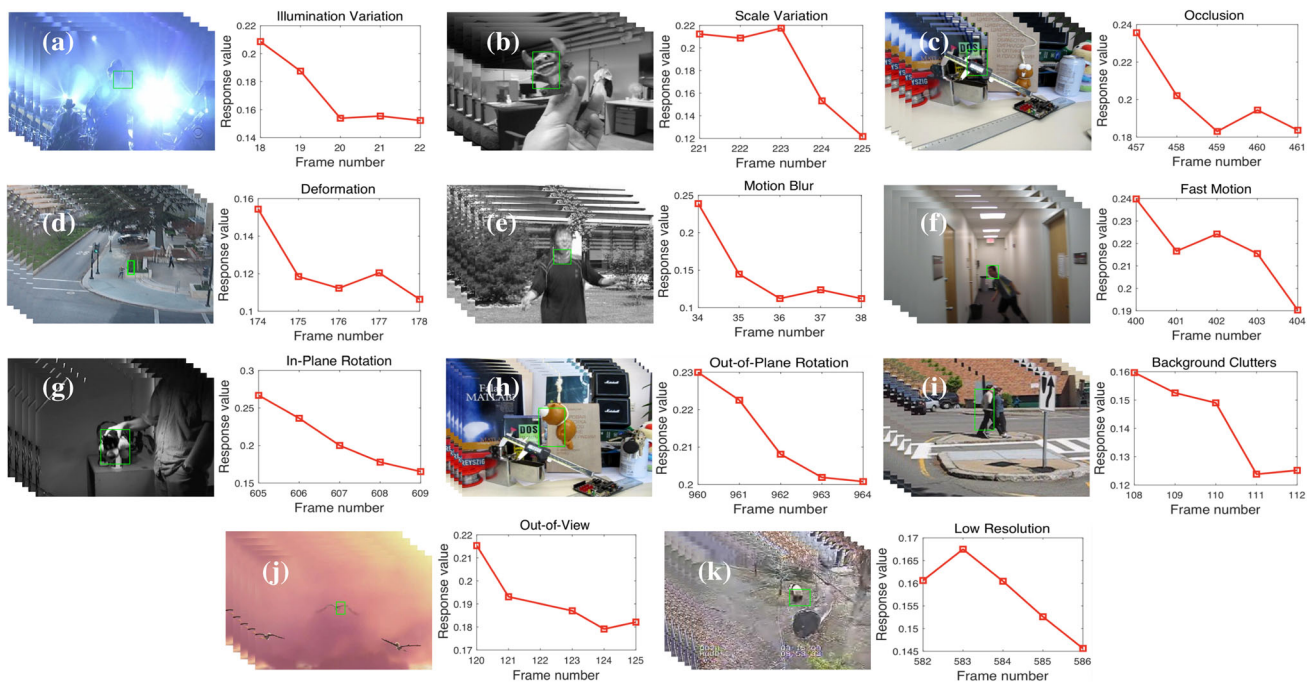


Fig. 1 The trend of response curve in 11 challenging scenarios. **a** Illumination variation (IR); **b** scale variation (SV); **c** occlusion (OCC); **d** deformation (DEF); **e** motion blur (MB); **f** fast motion (FM); **g** in-

plane rotation (IPR); **h** out-of-plane rotation (OPR); **i** background clutter (BC); **j** out-of-view (OV); **k** low resolution (LR)

of the object area to the entire image area. Figure 1 shows the trend of response values in 5 continuous frames with 11 challenging attributes. As can be seen from Fig. 1, the response values in the 11 challenge scenarios are drastically reduced compared with the response values of the second frame. In addition, we noticed that objects with larger τ have stronger ability to resist interference than objects with smaller τ . For example, for the objects *Toy*, *Lemming*, and *Sylvester* which are with very large τ and μ , even when they are affected by disturbances, the tracker still works well. In contrast, in the case of *Shaking*, *Jumping*, *Couple* and *Panda*, the tracker has experienced varying degrees of drift. As for the objects *Human5*, *Bird1*, *Boy* with μ less than 0.005, the tracker drifts

in the tracking of the first two objects, while the last one works well. It should be noted that the response value of the object *Boy* is much higher than the objects *Human5* and *Bird1*. In Table 1, the objects with smaller μ generally have lower response values, while the objects with larger μ generally have higher response values. This phenomenon also indicates that the large size objects with high response values in the initial environment are more resistant to interference than small size objects with low response values.

In view of the above analysis, we first divide the objects into four categories based on the response value τ and the area ratio μ .

$$\begin{cases} a_1 \leq \tau \wedge b_1 \leq \mu \rightarrow \text{general objects} \\ \tau < a_1 \wedge b_1 \leq \mu \rightarrow \text{general objects(weak)} \\ a_1 \leq \tau \wedge \mu < b_1 \rightarrow \text{small objects} \\ \tau < a_1 \wedge \mu < b_1 \rightarrow \text{small objects(weak)} \end{cases} \quad (5)$$

As can be seen from Fig. 1, when the response value drops to 0.25, the tracker has a tendency to fail. Further, the tracker may have failed when the response value falls below 0.18. According to the classification, each category will be assigned a threshold $d_i |_{i=1...4}$, mainly to keep $d_i \cdot \tau$ at around 0.25. Taking into account the small objects are susceptible to interference, the d_i of small objects will be appropriately increased to improve the accuracy of the criterion.

$$\begin{cases} \text{general objects} & \rightarrow d_1 \\ \text{general objects(weak)} & \rightarrow d_2 \\ \text{small objects} & \rightarrow d_3 \\ \text{small objects(weak)} & \rightarrow d_4 \end{cases} \quad (6)$$

The occlusion criterion consists of two phases. The first phase is to find a situation where the response values of the object drop dramatically over a period of time. And the task of the second phase is to find more severely degraded response values in the first phase. Here we consider the response values of five consecutive frames.

$$\mathbf{Y} = [y(1), y(2), y(3), y(4), y(5)] < d_i \cdot \tau \quad (7)$$

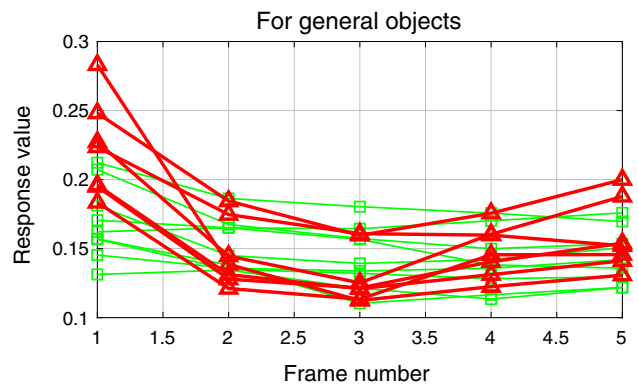
$$\text{sum}(\mathbf{Y} < \theta \cdot d_i \cdot \tau) \geq 2, \theta < 1 \quad (8)$$

where $y(i) |_{i=1,2...5}$ is the response value and $y(i)$ is the element of \mathbf{Y} . θ is a coefficient mainly to keep $\theta \cdot d_i \cdot \tau$ at around 0.18. The operator $\text{sum}(\cdot)$ is used to count the number of $y(i) < \theta \cdot d_i \cdot \tau$ in the set \mathbf{Y} .

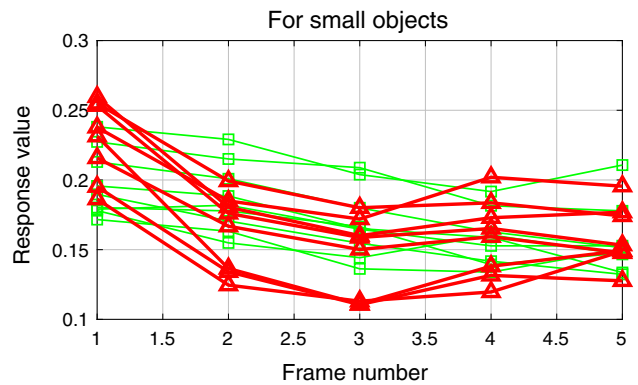
5 Re-detection mechanism

Based on the sliding window, the EdgeBox algorithm cleverly designs a scoring function which uses the number of edge segments completely contained in the sliding window as a measure. If the score is large, the area is likely to contain objects. Then, a series of proposals are generated based on the score. The EdgeBox algorithm typically takes only about 0.35 seconds to process a 480×720 image. Given the advantages of EdgeBox in terms of run-time and recall, we use this method to detect objects. Meanwhile, the threshold is set to accept only the top k bounding boxes. Considering the scale variation of the object, our screening strategy is described below:

$$1.2^{-1} \cdot b_w^{\text{occ}} \cdot b_h^{\text{occ}} < b_w \cdot b_h < 1.2 \cdot b_w^{\text{occ}} \cdot b_h^{\text{occ}} \quad (9)$$



(a) The response value curves for general objects.



(b) The response value curves for small objects.

Fig. 2 The response value curves for five consecutive frames (color figure online)

where b_w and b_h are the width and height of the proposals. b_w^{occ} and b_h^{occ} are the width and height of the object bounding box.

In the re-detector, the correlation operation is implemented on the filtered proposals and the maximum response value is taken. Then, the detection result is used to reinitialize the tracker if the maximum response value reaches the threshold $v\tau$. It should be noted that when the coefficient v is set too large, it will cause no object to be detected, and if v is too small, it may lead to error detection. In general, v takes a value of 0.5, such as [24]. However, when environment disturbance and object itself change dramatically, the response value corresponding to the proposal is relatively small, and vice versa. In view of the above concerns, how to properly set the coefficient v is very important.

The response value curves for five consecutive frames after the criterion is triggered are shown in Fig. 2, where each curve represents a sequence. It is clear that the fluctuation trend of the red curves is more intense than the green curves. In a certain sense, the fluctuation trend of the response value reflects the severity of the environmental disturbance and the change of the object itself. The more intense the fluctuation of the response curve, the greater the difficulty the

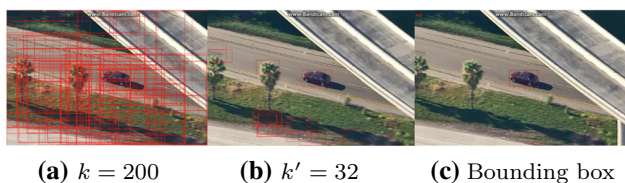


Fig. 3 The object extracted from k proposals

re-detector is in detecting potential objects. Therefore, we adopt a relatively small threshold coefficient for the objects represented by the red curves, while choosing a larger threshold coefficient for green curves. Since the fluctuation trend of the response value is mainly reflected in the first three frames, the method of separating the objects represented by the red and green curves is as follows:

$$\tau_{occ3} < \tau_{occ2} < \tau_{occ1} \tag{10}$$

$$w < \tau_{occ1} - \tau_{occ3} \tag{11}$$

$$z < \frac{\tau_{occ1} - \tau_{occ2}}{\tau_{occ1} - \tau_{occ3}} = r \tag{12}$$

where τ_{occ1} , τ_{occ2} and τ_{occ3} are the response values of the first 3 frames after the criterion is triggered, respectively. w represents the minimum difference in response value between the first frame and third frame. r is used to measure the degree of decline of the response value of the second frame. Subsequently, the separated curves will be assigned different threshold coefficients $v = [v_1, v_2, v_3, v_4]$. Compared with general object detection, small objects are easily interfered by similar objects in the background and feature representations are inaccurate, this will cause the re-detector to capture the wrong target. Therefore, it is necessary to appropriately increase the v_i of the small objects to reduce the probability of the error detection. Figure 3 shows the entire process of detection, screening and determination. In addition, the whole flowchart of proposed method is shown in Fig. 4.

6 Experimental results and discussion

6.1 Overall performance

The performance of proposed tracker is evaluated on OTB dataset using the one-pass evaluation (OPE) protocols. In AO-CF, we use the fusion feature of raw pixel, hog, and color label, and the scale pool is selected as [1, 0.99, 1.01]. The other parameters of the AO-CF tracker use the default parameters of KCF. In the occlusion criterion, $a_1 = 0.6$, $b_1 = 0.005$, $d_1 = 0.3$, $d_2 = 0.5$, $d_3 = 0.4$, $d_4 = 0.6$, $\theta = 0.7$. In the re-detection mechanism, $k = 200$, $w = 0.05$, $z_1 = z_2 = 0.6$, $v_1 = 0.7$, $v_2 = 0.5$, $v_3 = 0.8$, $v_4 = 0.6$. We compare AO-CF with 11 state-of-the-art trackers including

KCF [9], DSST [13], LCT [25], ACFN [15], DCFNet [16], MEEM [21], OCTKCF [22], LCKCF [23], SAMF [14], DLSSVM [26] and Staple [27]. In order to better demonstrate the effectiveness of our proposed method, we also make corresponding improvements to KCF, named IKCF. The experimental environment is Intel Core i5 2.3 GHz CPU with 8.00 GB RAM, MATLAB 2017b.

As shown in Fig. 5, AO-CF performs favorably against the other twelve state-of-the-art methods. With the aid of occlusion criterion and re-detection mechanism, the proposed AO-CF significantly improves the performance in both distance precision (5.9%) and overlap success (3.6%) when compared to the foundational SAMF on the OTB-100 dataset. For the OTB-50 dataset, AO-CF gains 8.7% in distance precision and 6.0% in overlap success. Compared to KCF, the IKCF has also greatly improved the precision and overlap rate. The tracking speeds of AO-CF and IKCF are 51.2 frame/s and 93.3 frame/s, respectively.

6.2 Ablation study

Several modifications of the proposed method are tested to expose the contributions of different parts in our architecture. Two variants use the proposed detection threshold setting method in the detector with peak to sidelobe ratio (CF-PSR) [28] and median flow (CF-MF) [29], and one variant uses only 0.5 (CF-(0.5)) as the detection threshold. Figures 6 and 7 show the tracking results and the speed comparisons. Compared to CF-(0.5) with a fixed detection threshold, our method increases the overlap rate by 1.9%. In addition, our tracker is also significantly better than CF-PSR and CF-MF, which can be attributed to the fact that the proposed criterion is more accurate than PSR and MF in the evaluation of tracking failure. From the perspective of tracking speed, our method still achieves good real-time performance even if EdgeBox increases the amount of calculation. Compared with other methods (KCF, Staple, etc), although the proposed tracker is not superior in speed, our approach is better than them in terms of overall performance. It is worth noting that the speed of CF-MF is seriously degraded due to the use of a computationally intensive MF method.

6.3 Attribute-based evaluation

To further reveal the performance of the tracker, we also evaluate the proposed method using 11 annotated attributes in the OTB-100 dataset. In the distance precision comparison of Fig. 8, AO-CF ranks first in IV, OPR, SV, OCC, IPR, BC, and ranked second in DEF, LR. In the comparison of the overlap success of Fig. 9, AO-CF ranks first in OPR, IPR, BC, and ranked second in IV, OCC, DEF, MB, OV, LR. Regarding the OCC, AO-CF has achieved considerable performance in distance precision and overlap rate. Since other attributes also

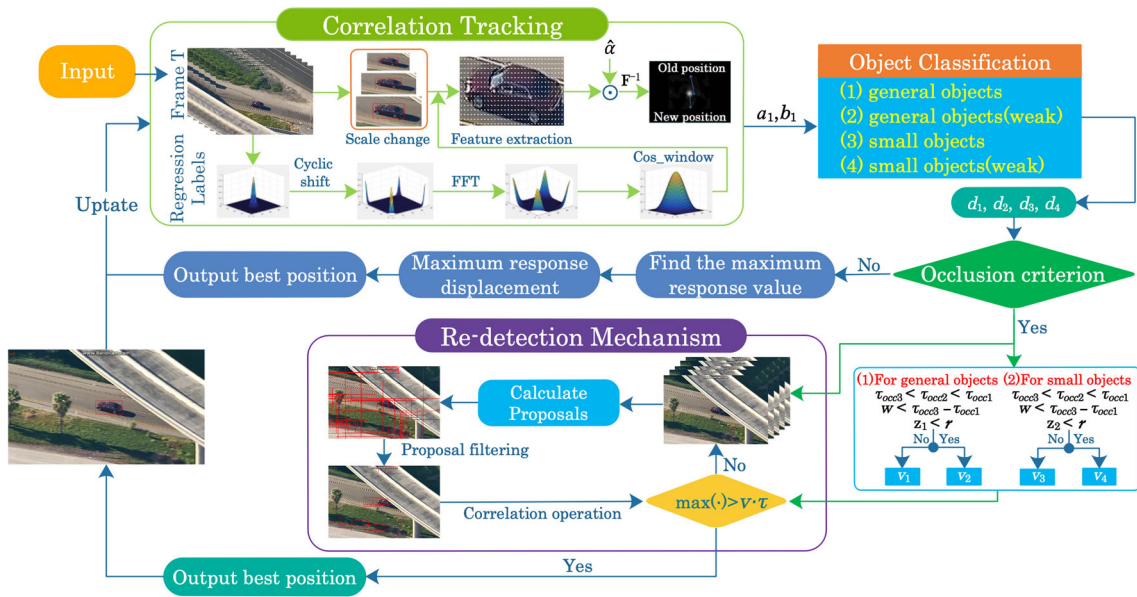
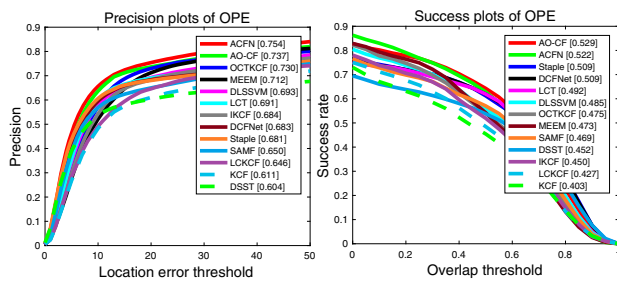
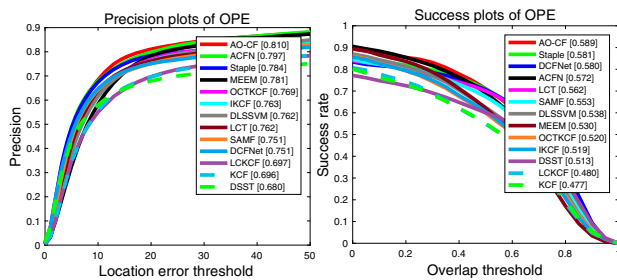


Fig. 4 The flowchart of AO-CF. The tracker is mainly composed of five components: correlation tracking module, object classification module, occlusion criterion module, re-detection mechanism module and detection threshold setting module



(a) Precision plot on OTB-50. **(b)** Success plot on OTB-50.



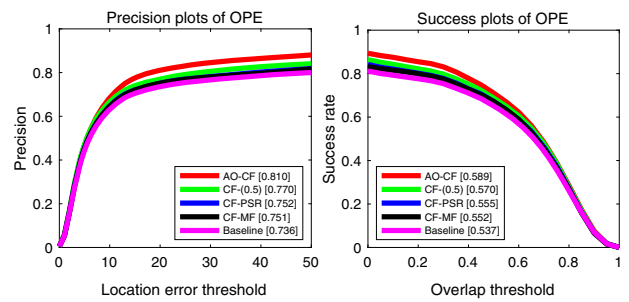
(c) Precision plot on OTB-100. **(d)** Success plot on OTB-100.

Fig. 5 Comparison of ten algorithms over OTB-50 and OTB-100 benchmark using OPE

make the response value drop and result in object loss, AO-CF also performs well under these attributes, such as IPR, OPR, IV, not just for OCC.

6.4 Qualitative evaluation

To intuitively present the superiority of our tracker, we visualize the tracking snapshots of top 10 trackers on eight



(a) Precision plot on OTB-100. **(b)** Success plot on OTB-100.

Fig. 6 Ablation study of AO-CF on OTB-100 dataset

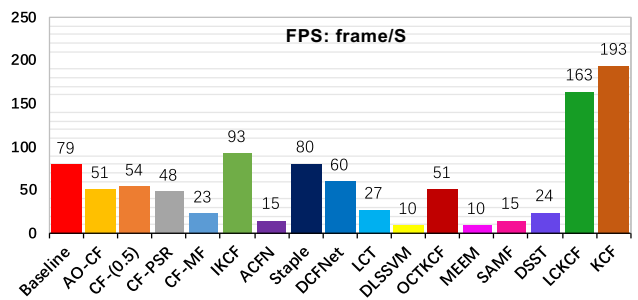


Fig. 7 Comparison of the tracking speed of 17 trackers

challenging sequences with OCC attribute. As can be seen from Fig. 10, the AO-CF is still tracking objects robustly when the objects undergo partially OCC or fully OCC. However, most trackers drift to the background after the objects are occluded. As for IKCF, it implements robust tracking on five sequences. Benefiting from the powerful

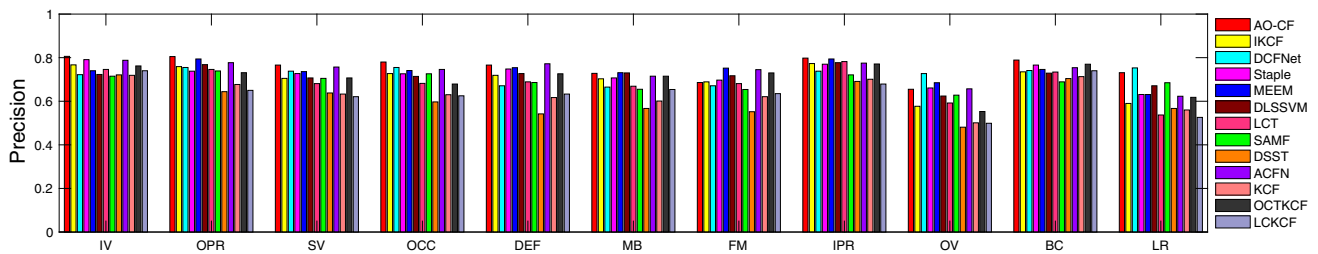


Fig. 8 Precision plots over 11 tracking challenges

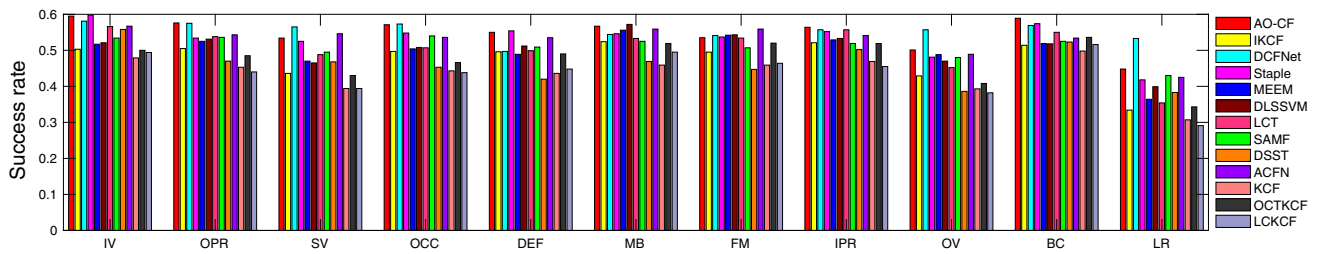


Fig. 9 Overlap success plots over 11 tracking challenges

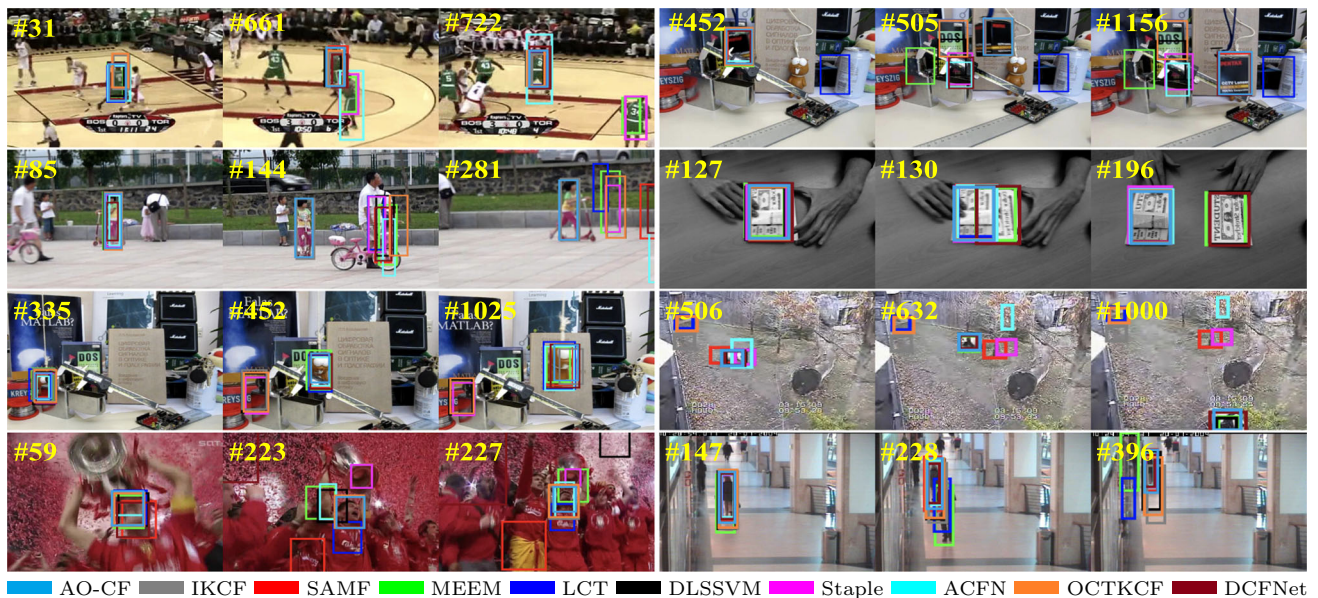


Fig. 10 Tracking results of top 10 methods on eight challenging sequences. From left to right and top to bottom, they are *Basketball*, *Box*, *Girl2*, *Coupon*, *Lemming*, *Panda*, *Soccer*, *Walking2*, respectively

convolution feature, DCFNet performs well in *Basketball*, *Lemming*, *Panda*, and *Walking2*, but fails in *Coupon* and *Soccer* when similar interferers appeared. In addition, MEEM and DLSSVM, which are based on SVM, also show inadequate ability to handle similar interferers in *Coupon* and *Soccer*. The ACFN using the attention mechanism is mainly suitable for FM, and the handling of the OCC is obviously weak. For correlation filtering algorithms based on hand-crafted features, namely, SAMF, LCT, Staple and OCTKCF, due to the limited search area caused by the boundary effect,

these trackers drift in *Girl2* and *Panda* when the object is severely occluded. However, with the help of model updating mechanism, some correlation filtering algorithms also show strong robustness in some OCC scenarios. For example, SAMF in *Box*, *Lemming*, *Walking2*, LCT in *Basketball* and *Lemming*. It is worth mentioning that OCTKCF, modeled on the Bayesian optimization framework, performs impressively in *Soccer* and *Basketball* sequences. In general, the occlusion criteria and the re-detector significantly



Fig. 11 Red boxes show our results and the green ones are ground truth. From left to right are *MotorRolling*, *Skiing*, *Jump*, respectively (color figure online)

improve the ability of the AO-CF and IKCF to handle OCC.

6.5 Failure cases

Apart from successful cases, we also discover a few failure cases as shown in Fig. 11. For the *MotorRolling* sequence, due to the serious in-plane rotation and scale variation of the object, the object bounding box contains a large amount of background information, which causes the proposed tracker to drift. For the *Skiing* sequence, although the fast motion of the object caused a sharp drop in the response value, the re-detector fails to activate because the time is too short and the conditions of the first phase in the occlusion criterion are not met. Regarding the *Jump* sequence, the object bounding box is almost at the position of the previous frame because the target template is not updated in time.

7 Conclusions

In this paper, an effective tracker is proposed to handle the occlusion of objects during tracking process. The proposed occlusion criterion and re-detection mechanism take the changes in tracking process into account comprehensively. The experimental results show that compared with the foundational SAMF on the OTB-100 dataset, the AO-CF has achieved significant improvements in distance precision (5.9%) and overlap success (3.6%). In the proposed tracker performs outstandingly for OCC in attribute-based evaluation. Moreover, our approach can obtain a real-time tracking speed.

Acknowledgements This paper is sponsored by National Program on Key Basic Research Project (2014CB744903), National Natural Science Foundation of China (61973212, 61673270), Shanghai Science and Technology Committee Research Project (17DZ1204304).

References

- Mei, X., Ling, H.: Robust visual tracking and vehicle classification via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(11), 2259–2272 (2011)
- Ross, D.A., Lim, J., Lin, R., Yang, M.: Incremental learning for robust visual tracking. *Int. J. Comput. Vis.* **77**(1), 125–141 (2008)
- Bai, B., Li, Y., Fan, J., Price, C., Shen, Q.: Object tracking based on incremental Bi-2DPCA learning with sparse structure. *Appl. Opt.* **54**(10), 2897–2907 (2015)
- Babenko, B., Yang, M., Belongie, S.: Robust object tracking with online multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(8), 1619–1632 (2011)
- Zhang, K., Zhang, L., Yang, M.H.: Fast compressive tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **36**(10), 2002–2015 (2014)
- Zitnick, C.L., Dollar, P.: Edge boxes: locating object proposals from edges. In: *European Conference on Computer Vision*, pp. 391–405 (2014)
- Wu, Y., Lim, J., Yang, M.: Online object tracking: a benchmark. In: *International Conference on Computer Vision and Pattern Recognition*, pp. 2411–2418 (2013)
- Wu, Y., Lim, J., Yang, M.: Object tracking benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(9), 1834–1848 (2015)
- Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(3), 583–596 (2015)
- Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: *International Conference on Computer Vision and Pattern Recognition*, pp. 2544–2550 (2010)
- Danelljan, M., Khan, F.S., Felsberg, M., van de Weijer, J.: Adaptive color attributes for real-time visual tracking, pp. 1090–1097 (2014)
- Galoogahi, H.K., Sim, T., Lucey, S.: Multi-channel correlation filters. In: *International Conference Computer Vision*, pp. 3072–3079 (2013)
- Danelljan, M., Häger, G., Khan, F., Felsberg, M.: Accurate scale estimation for robust visual tracking. In: *British Machine Vision Conference*, pp. 1–5 (2014)
- Li, Y., Zhu, J.: A scale adaptive Kernel correlation filter tracker with feature integration. In: *European Conference on Computer Vision*, pp. 254–265 (2014)
- Choi, J., Chang, H.J., Yun, S.: Attentional correlation filter network for adaptive visual tracking. In: *International Conference on Computer Vision and Pattern Recognition*, pp. 4828–4837 (2017)
- Wang, Q., Gao, J., Xing, J.: DCFNet: discriminant correlation filters network for visual tracking. *arXiv preprint arXiv:1704.04057* (2017)
- Hare, S., Golodetz, S., Saffari, A., Vineet, V., Cheng, M.M., Hicks, S.L., Torr, P.H.S.: Struck: structured output tracking with kernels. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(10), 2096–2109 (2015)
- Chen, D., Yuan, Z., Wu, Y., Zhang, G., Zheng, N.: Constructing adaptive complex cells for robust visual tracking. In: *IEEE International Conference on Computer Vision*, pp. 1113–1120 (2013)
- Holzer, S., Llic, S., Navab, N.: Multilayer adaptive linear predictors for real-time tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(1), 105–117 (2013)
- Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking–learning–detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(7), 1409–1422 (2012)
- Zhang, J., Ma, S., Sclaroff, S.: Meem: robust tracking via multiple experts using entropy minimization. In: *European Conference on Computer Vision*, pp. 188–203 (2014)
- Zhang, B., Li, Z., Cao, X.: Output constraint transfer for kernelized correlation filter in tracking. *IEEE Trans. Syst. Man Cybern.* **47**(4), 693–703 (2017)
- Zhang, B., Luan, S., Chen, C.: Latent constrained correlation filter. *IEEE Trans. Image Process.* **27**(3), 1038–1048 (2018)
- Zhao, J., Xiao, G., Zhang, X., Bavirisetti, D.: An improved long-term correlation tracking method with occlusion handling. *Chin. Opt. Lett.* (2019). <https://doi.org/10.3788/COL201917.031001>

25. Ma, C., Yang, X., Zhang, C.: Long-term correlation tracking. In: International Conference on Computer Vision and Pattern Recognition, pp. 5388–5396 (2015)
26. Ning, J., Yang, J., Jiang, S.: Object tracking via dual linear structured SVM and explicit feature map. In: International Conference on Computer Vision and Pattern Recognition, pp. 4266–4274 (2016)
27. Bertinetto, L., Valmadre, J., Golodetz, S.: Staple: complementary learners for real-time tracking. In: International Conference on Computer Vision and Pattern Recognition, pp. 1401–1409 (2016)
28. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: IEEE International Conference on Computer Vision, pp. 2544–2550 (2010)
29. Kalal, Z., Mikolajczyk, K., Matas, J.: Forward–backward error: automatic detection of tracking failures. In: IEEE International Conference on Pattern Recognition, pp. 2756–2759 (2010)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.