The Visual Computer (2021) 37:567–585
https://doi.org/10.1007/s00371-020-01824-6

ORIGINAL ARTICLE

A robust tracking algorithm with on online detector and high-confidence updating strategy

Enzeng Dong[1] · Mengtao Deng[1] · Zenghui Wang[2]

Published online: 28 February 2020

Abstract

The discriminative correlation filter-based tracking algorithms cannot correctly track the target if the target is occluded or out of view and reappears in the field of vision, and they cannot ensure the tracking model is updated correctly if the tracking information is not correct. In this paper, a robust correlation tracking algorithm is proposed. Here, a failure detection strategy, which is based on the maximal confidence score and peak-to-sidelobe ratio to detect or measure the reliability of the tracking result, is integrated into the tracker. Moreover, the redetection module based on the keypoints matching method for consensus voting is introduced into the proposed tracking algorithm to redetect objects in case of tracking failure. In addition, an adaptive high-confidence updating method is proposed to avoid error model information introduced into the tracker caused by occlusions, out-of-view or illumination changes, where the learning rate is determined by the change rate of the confidence map. The OTB-2015 dataset and VOT-2016 dataset are used to evaluate the performance of the proposed tracking algorithm. The experimental results show that the proposed tracking algorithm performs better than most of the state-of-the-art trackers, and it has higher accuracy and robustness than the DSST tracker.

Keywords Correlation filter · PSR · Confidence degree · Consensus voting · Keypoints matching

1 Introduction

Visual tracking is defined as the problem of finding the motion of a target given a sequence of images based on different frames in a video, and it is widely used in computer vision applications such as surveillance, security and motion analysis. Though many visual tracking algorithms have been proposed, there are many challenges in the practical applications to solve the problems caused by occlusion, out of view, deformation, illumination variation, fast motion, motion blurring, background clutters, out-of-plane rotation, in-plane rotation, and scale variation [1,2]. Therefore, it is desired to have a robust visual tracking method solving the above-mentioned problems.

Visual tracking algorithms can be generally classified into two categories: the generative methods [3–5] and discriminative methods [6–8]. For the generative methods, tracking is formulated as searching for the most similar region to the target within a neighborhood, for example, the scale-adaptive mean-shift tracking method of [5]. This kind of algorithm focuses on the description of the target itself, ignores the background information, and is prone to drift when the target is similar to the background color or occluded [9]. Different from generative trackers, the discriminative methods regard tracking problem as a classification problem, which aims at finding decision boundaries that can distinguish the target from background, and does not need to establish a complex model to describe the object. This kind of algorithm usually takes the target area as positive sample in the initial frame and the background area as negative sample to train the classifier. The next frame uses the trained classifier to find decision boundaries between the object and the background. The main difference between the discriminative and generative methods is that the classifier of the discriminative method adopts machine learning, and the discriminative model is trained to use background information. Since the classifier of the discriminative methods can distinguish fore-

Enzeng Dong and Mengtao Deng contributed equally to this work.

✉ Enzeng Dong
dongenzeng@163.com

[1] Tianjin Key Laboratory for Control Theory and Applications in Complicated Systems, Tianjin University of Technology, Tianjin 300384, China

[2] Department of Electrical and Mining Engineering, University of South Africa, Florida 1710, South Africa

ground and background accurately, its tracking performance using discriminative model is generally better than the generative method. Kalal et al. [7] proposed the TLD tracker that decomposes the tracking task into three sub-tasks: tracking, learning and detection, and the TLD tracker can be used to build a long-term tracking system based on the P-N learning. Hare et al. [8] considered the spatial distribution of samples within a search space and proposed a framework of adaptive visual object tracking method based on the structured output prediction algorithm that is used to predict the object location, and the algorithm was proved to have good performance. Zhang et al. [10] proposed a real-time compressive tracking method that formulated the task as a binary classification in the compressed domain.

Over the past 5 years, visual tracking algorithms based on correlation filter have become a research hotspot due to its fast speed and high accuracy. In 2010, Bolme et al. [11] firstly used the correlation filter for visual tracking, and designed a Minimum Output Sum of Squared Error (MOSSE) filter. Using the fast Fourier transformation (FFT), the MOSSE tracker becomes more efficient. Recently, many tracking algorithms based on correlation filters have been proposed, including circulant structure kernel (CSK) tracker [12], kernelized correlation filter (KCF) tracker [13], color name tracker (CN) [14], discriminative scale space tracker (DSST) [15], large margin correlation filter tracker (LMCF) [16], and spatially regularized correlation filters (SRDCF) tracker [17], and other improved algorithms [18,19] have been proposed, the tracking effects of which are getting better and better.

Although the correlation filter-based tracking algorithms are efficient, they are not robust enough, and the target cannot be tracked correctly when the target is occluded or out of view and reappears in the field of vision [2,11]. Moreover, the model cannot be properly updated online if the tracking information is not correct in these tracking algorithms. Hence, in the long-term tracking process, if the training tracker is updated with unreliable model information, some erroneous model information will be inevitably introduced into the tracker, causing the tracker to continuously accumulate errors, and eventually lead the tracking to fail.

To solve the problems of correlation filter-based tracking algorithms, a robust tracking algorithm based on redetection module and high-confidence updating is proposed. The main contributions of this study can be summarized as follows: (1) We proposed a robust tracking algorithm which absorbs the strong discriminative ability from DSST tracker; (2) to detect or measure the reliability of the tracking result, a failure detection module, which is based on the maximal confidence score and peak-to-sidelobe Ratio (PSR), is introduced to detect the confidence of the tracking results; (3) to prevent tracking failures, an online detector based on the keypoint matching method for consensus voting is used to relocate the target if the confidence of the tracking result is low; (4) to avoid erroneous model information being introduced into the tracker, a high-confidence updating method is proposed to select reliable target model. The experimental results on OTB-2015 and VOT-2016 datasets show that the proposed method performs better than most of the state-of-the-art trackers, and it can achieve promising performance on visual tracking especially there are occlusion and out of view in the video sequences.

The rest of the paper is arranged as follows. The related work is given in Sect. 2. Section 3 introduces the proposed algorithm in details. We give the experimental results and analysis of the proposed method in Sect. 4. Conclusions are made in Sect. 5.

2 Related work

MOSSE (Minimum Output Sum of Squared Error filter) algorithm is the first one using correlation filtering in target tracking. Since it uses gray features, it is much faster than other algorithms, but the accuracy is lower. There are some variants of correlation filter-based tracking algorithms in the studies. CSK tracker introduces the concepts of cyclic matrix and core on the basis of MOSSE to improve tracking accuracy. The main purpose of CSK tracker is to solve the problem of sample redundancy caused by sparse sampling in traditional algorithm. The CN tracker extends the CSK tracker with color attributes and uses PCA to reduce the dimension of Color Name with less redundant information. Henriques et al. applied HOG (histogram of oriented gradient) feature to nuclear correlation filters on the basis of MOSSE. A KCF (kernelized correlation filter) tracker was proposed in 2014, and it can effectively solve the problem of redundancy of training data using cyclic matrix and discrete Fourier transform, which can greatly reduce its computational complexity and improve the tracking performance. DSST tracker uses HOG feature to learn adaptive multi-scale correlation filter to deal with the scale variation of target object. Scale-adaptive with multiple features tracker [20] (SAMF) integrates HOG feature and CN feature on the basis of KCF and uses scale pool technology to obtain the optimal scale of target in scale variation.

Spatially regularized discriminative correlation filters (SRDCFs) based on KCF adds penalties to the loss function, improves the edge effect, and achieves a breakthrough in the effect, but its computation cost is high. CSRDCF [21] is a tracking method that combines filtering and color probability; it can achieve good tracking precision, but its computation cost is relatively high. Moreover, the spatial and channel reliability were proposed in [21]. DeepSRDCF tracker [22] replaces HOG feature with CNN feature on the basis of SRDCF algorithm, which greatly improves the tracking performance, but introduces CNN feature that increases the computation cost and reduces the tracking speed. Continuous-convolution operator tracker (C-COT) [23] uses depth neural network VGGNet to map feature maps with different resolution to continuous space domain through cubic interpolation algorithm, and obtains sub-pixel precision target location. Although the tracking accuracy of these methods is improved, their computation cost is also increased and cannot be applied to real-world scenarios.

Moreover, these visual tracking algorithms are prone to drift when the target is occluded or beyond the field of view, and the target cannot be continuously tracked when the target appears again in the field of view. In addition, these tracking algorithms update tracking models at each frame without considering whether the tracking result is accurate or not. If the target is severely occluded or out of view, the tracking result will be unreliable, and it may cause the tracking to fail. To void the problem of tracking failure, Chao Ma et al [24] introduced an online random fern classifier to redetect objects in case of tracking failure, and the proposed algorithm performs well. To avoid the problem of erroneous samples for online model update, Wang et al [16] proposed a model update strategy to avoid model corruption by the high-confidence selection from tracking results, which can effectively avoid the model corruption problem. To suppress the effects of background clutters, Chenglong, Li et al [25] proposed a tracking method of via dynamic graph learning, which decomposes the problem of target state estimation into the three sub-problems of Patch-based graph learning, structured SVM tracking, and Model update.

Based on the above analysis, a new robust tracking is proposed in next section. We proposed the tracking method that mainly includes four parts: discriminant DSST tracking, failure detection, redetection and high-confidence updating strategy. Compared with [16] and [25], although the framework of the proposed method looks similar, the methods of each component are different.

3 The proposed method

To solve the mentioned problems in Sect. 2, a new robust visual tracking method is proposed. Firstly, this new method estimates the target position by using the discriminant DSST tracker. Secondly, a failure detection scheme is used to find the confidence of the tracking result of the current frame, and the detector is started to relocate the target if the confidence of the tracking result is low. Moreover, in order to reduce the error information accumulated during the tracking process, an adaptive high-confidence updating strategy is proposed to improve the robustness of the tracker. Finally, OTB-2015 dataset [2] and VOT-2016 dataset [26] are used to evaluate the comprehensive performance of the proposed method. Hence, the new method mainly includes four parts: correlation filter tracker, failure detection module, redetection module, and high-confidence updating module. The flowchart of the algorithm is shown in Fig. 1.

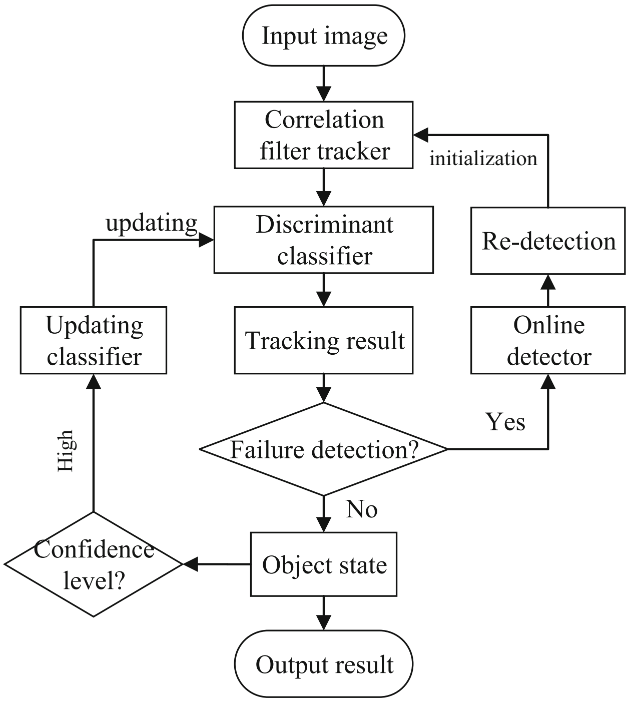


Fig. 1 The flowchart of the proposed method

3.1 Correlation filter tracking

As the basis of the proposed tracker, the DSST filter is used to predict the target location. Let f be a rectangular patch of the target, extracted from the feature map. The target sample f consists of a d-dimensional feature vector $f(n) \in R^d$, at each location n in a rectangular domain. We denote the feature dimension number $l \in \{1, 2, \ldots, d\}$ of f by f^l. The objective is to find an optimal correlation filter h, consisting of one filter h^l per feature dimension. This can be achieved by minimizing the cost function:

$$\varepsilon = \left\| \sum_{l=1}^{d} h^l * f^l - g \right\|^2 + \lambda \sum_{l=1}^{d} \left\| h^l \right\|^2 \tag{1}$$

Here, g is the desired correlation output associated with the training example f; $*$ denotes the circular correlation; and λ denotes regularization parameters ($\lambda > 0$) that controls the impact of the regularization term. The solution to (1) is:

$$H^l = \frac{\overline{G} F^l}{\sum_{k=1}^{d} \overline{F^k} F^k + \lambda}, l = 1, \ldots, d. \tag{2}$$

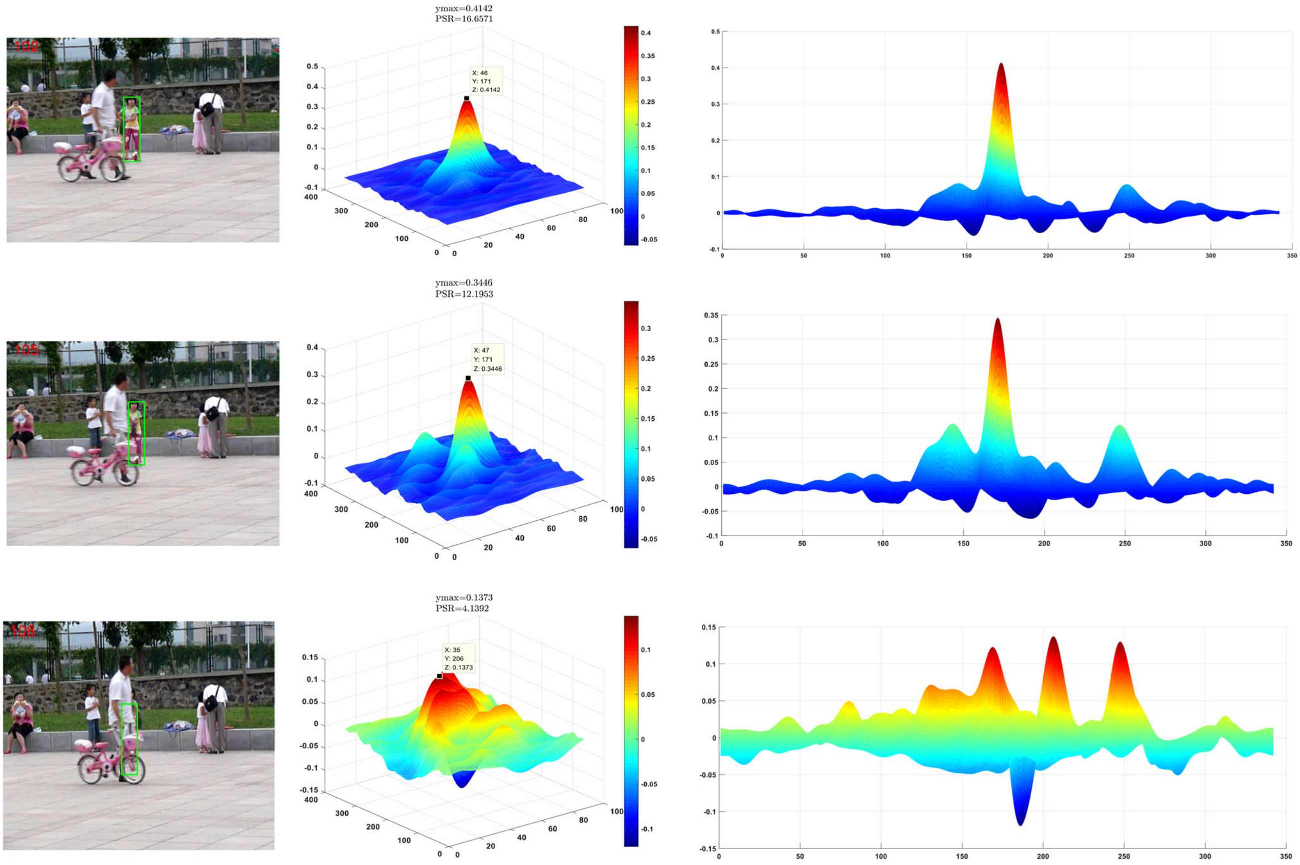


Fig. 2 The confidence map for the "Girl2" image sequences

Here, H^l, F^l and G denote the discrete Fourier transforms (DFT) h^l, f^l and g, respectively; $\overline{G}$ represents the complex conjugations of G.

To obtain a highly robust tracker, we update the numerator A_t^l and denominator B^t of the correlation filter of (2) as:

$$A_t^l = (1 - \eta)\, A_{t-1}^l + \eta \overline{G_t} F_t{}^l \tag{3}$$

$$B^t = (1 - \eta)\, B^t + \eta \sum_{k=1}^{d} \overline{F_t{}^k} F_t{}^k \tag{4}$$

Here, the scalar η is a learning rate parameter.

Let z_t correspond to an image patch centered around the predicted target location, and Z_t^l denotes the discrete Fourier transforms z_t. The new target state is then found by maximizing the score y_t.

$$y_t = \mathcal{F}^{-1}\left\{ \frac{\sum_{l=1}^{d} \overline{A_t^l} Z_t^l}{B_t + \lambda} \right\} \tag{5}$$

To cope with the scale variation, we follow the scheme of DSST. The scale filter is trained and then used to estimate the scale. For each $n \in \left\{ \left[-\frac{S-1}{2} \right], \ldots, \left[\frac{S-1}{2} \right] \right\}$, we extract an image patch I_n of size $a^n P \times a^n R$ centered around the target. Here, $P \times R$ denotes the target size in the current frame, S is the size of the scale filter, and a is the scale factor. More information can be found in the DSST tracker [8].

3.2 Tracking failure detection scheme

For many existing correlation filter trackers, it is very difficult to detect or measure the reliability of the tracking result. However, for trackers, it is important to detect the confidence degree of the tracking result or to determine whether the target is occluded or totally missing in the current frame. When the tracking failure is detected, the update of the target model and the training of the classifier should be stopped, and if the target appears again in the field of view, the redetection module is used to relocate the target. In view of the above-mentioned problems, a novel failure detection method is proposed to detect the confidence degree of the tracking result and then determine whether the tracking result is accurate.

As shown in Fig. 2, when the tracking result is accurate, the confidence map should have only one sharp peak, which is similar to the ideal two-dimensional Gauss distribution, as shown in the first line of Fig. 2. In general, when the tar-

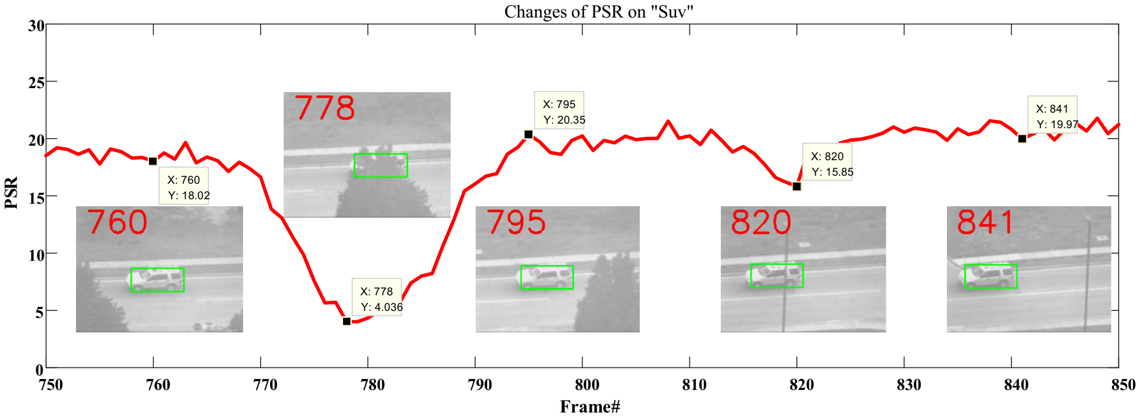


Fig. 3 The most challenging section of a video can be located by finding low points in the *PSR*

get is suddenly partially occluded, the confidence map still conforms to a two-dimensional Gaussian distribution and the maximal confidence score will suddenly decrease. However, there are multiple peaks in the confidence map, and the maximal confidence score can still represent the target as shown in the second row of Fig. 2. When tracking fails, especially when there are challenges such as occlusion, out of view, and background clutters, and so on, the confidence map will fluctuate intensely. At this time, the maximal confidence score is not for the target as shown in the third line of Fig. 2.

The maximal confidence score and the fluctuation of the response map can reflect the confidence degree about the tracking performance to some extent. The ideal response map should have only one sharp peak and be smooth in all other areas when the detected target is extremely matched to the correct target. The larger the maximal confidence score is, the more accurate the tracking result is. Otherwise, if the tracking result is inaccurate or tracking fails, the maximal confidence score will suddenly decrease, and the confidence map will fluctuate intensely. Therefore, we propose a new confidence degree detection method to detect whether the tracking result is accurate. The following two indicators are used to detect the confidence degree of the tracking result.

The first one is the maximal confidence score for the confidence map, which is defined as:

$$y_{\max} = \max \mathcal{F}^{-1}\left\{\frac{\sum_{l=1}^{d}\overline{A_t^l}Z_t^l}{B_t+\lambda}\right\} \tag{6}$$

In the process of tracking, if the maximal confidence score $y_{\max}^t$ of the tth frame is less than a threshold T_{th}, the tracking result of the current frame is considered unreliable.

The second indicator is called the peak-to-sidelobe ratio [11], which is a measure of peak strength. To compute the PSR the correlation output y is split into the peak, which is the maximum value and the sidelobe which is the rest of the pixels excluding an 11 × 11 window around the peak. The PSR is then defined as:

$$\text{PSR} = \frac{y_{\max} - \mu}{\sigma} \tag{7}$$

where $y_{\max}$ is the maximal peak value, μ and σ are the mean and standard deviation of the sidelobe, respectively.

PSR can reflect the fluctuation degree of the confidence map to some extent. As shown in Fig. 3, the change of PSR value is relatively stable if the target is visible in the detection range. Otherwise, the PSR will be significantly reduced if the target is occluded or tracking fails.

In this paper, we set a threshold T_{PSR}. When the PSR value of tth frame is less than T_{PSR}, the confidence degree of the tracking result of the current frame is considered to be low.

Therefore, we use two thresholds T_{th} and T_{PSR} to determine whether the tracker is failed to track the target based on the change in the maximal peak value $y_{\max}$ and PSR. If $y_{\max} > T_{\text{th}}$ and PSR $> T_{\text{PSR}}$, the tracking result is considered to be relatively reliable, that is, the tracking is accurate. Otherwise, the confidence degree of tracking result is low, that is, tracking fails.

3.3 Redetection module

When the target is long-term occluded or beyond the field of vision, many correlation filter trackers will not work properly. When the target appears in the field of vision again, the tracker cannot track the target continuously. Therefore, it is necessary to have the redetection module for a robust tracker. In this subsection, we design a detector for the redetection module based on consensus-based matching of keypoints. In the process of tracking, when the tracking failure is detected, the detector is used to relocate and track the target.

3.3.1 Keypoint matching

Firstly, a set of key points is set up based the target model.

$$O = \{(r_i, f_i)\}_{i=1}^{N^O} \tag{8}$$

where each keypoint denotes a location $r_i \in R^2$ in template coordinates and a descriptor f_i. We initialize O by detecting and describing key points in I_1 that are inside the initializing region I_1.

To match keypoints, the candidate keypoints in the image I_1 can be expressed as follows:

$$P = \{(a_i, m_i)\}_{i=1}^{N^{K_P}} \tag{9}$$

where a refers to the keypoint position in absolute image coordinates and is the index of the corresponding keypoint in O.

For each candidate keypoint, the Hamming distance between its descriptor and all the keypoints descriptors found in I_1, including the background keypoints, is calculated.

$$d\left(f^1, f^2\right) = \sum_{i=1}^{d} \text{XOR}\left(f_i^1, f_i^2\right) \tag{10}$$

According to the ratio ρ that the nearest neighbor must be larger than the second nearest neighbor, we match candidate keypoints in P to keypoints in I_1 [27]. The set of matched keypoints M consists of the subset of keypoint locations in P that match O. So the candidate keypoints match with background keypoints are removed from M [28].

3.3.2 Consensus voting

In order to find the location the object center, each keypoint (a, m) in M casts a vote $h(a, m) \rightarrow R^2$ for the object center as follows:

$$V = \{h\,(a_i, m_i)\}_{i=1}^{K_P} \tag{11}$$

Here, the translation transformation of the target is considered.

$$h^T\,(a, m) = a - r_m \tag{12}$$

where r_m is the relative position of the corresponding keypoint in O.

No matter whether the coordinate a or the model index m is wrong, the voting result will not be the object center, but will be randomly pointed to a certain position in the image.

Before calculating the object center C, the outlier keypoints need to be identified and removed by looking for consensus in the voting behavior, as shown in Fig. 4. Finally, a hierarchical agglomerative clustering method [28,29], which uses Euclidean distance as similarity measure, is applied to cluster the voting results V. This clustering method organizes the data into a hierarchical structures according to a proximity matrix, which constitutes a dendrogram that is then cutoff at a certain threshold δ. Thus, V is partitioned into the disjoint subsets $V_1, \ldots, V_{m-1}$ and V_m. The subset V_c containing the largest number of elements is considered to be the consensus cluster [30].

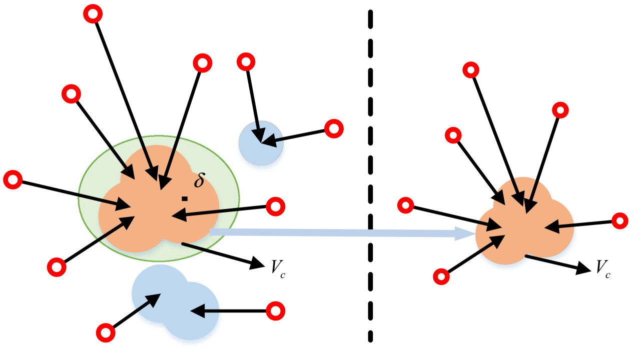


Fig. 4 Finding consensus in voting behavior

If V_c contains elements less than $\theta \cdot |O|$, we assume the object is not visible. Otherwise, we turn the votes in the consensus cluster into an estimate for the object center.

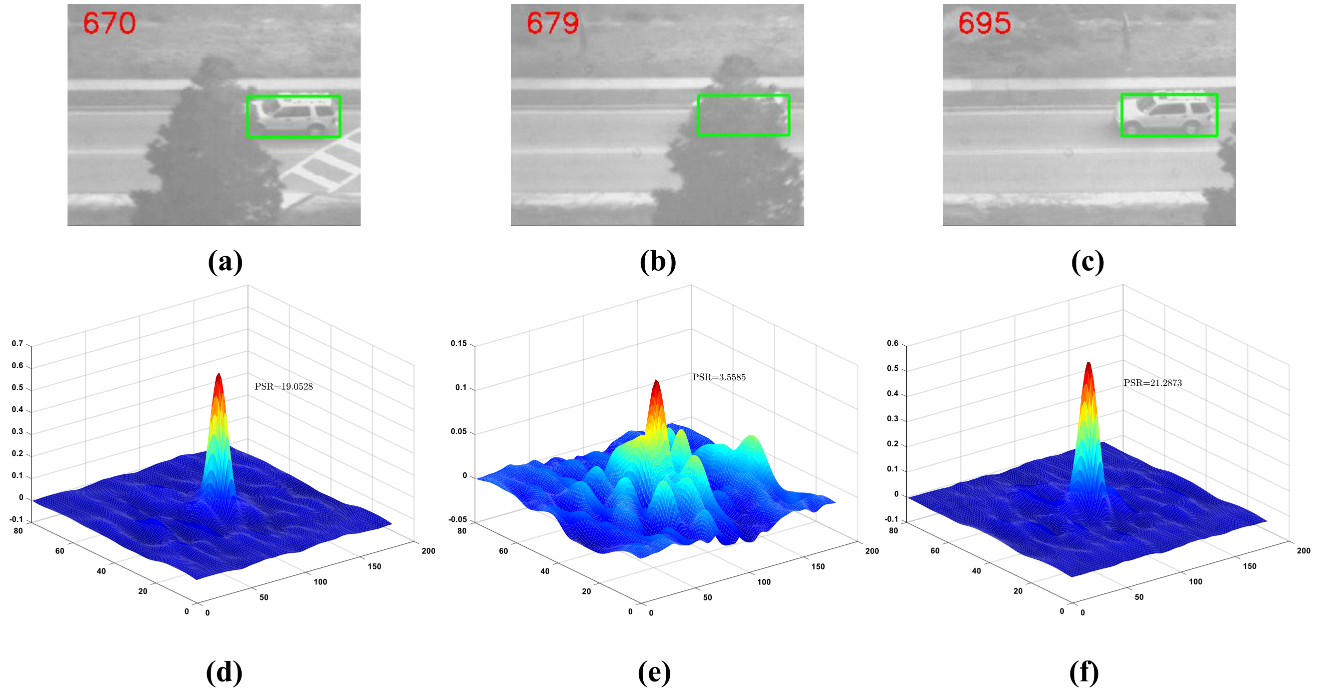
$$C = \frac{1}{n}\sum_{i=1}^{n} V_C^i \tag{13}$$

where $n = |V_c|$. It should be noted that the object center C and the scale s define the pose of the object of interest.

3.4 High-confidence update scheme

For highly robust trackers, high-confidence update is very important. Most existed trackers update tracking models [13,15,16,31] at each frame without considering whether the detection is accurate or not. Actually, the tracking result is unreliable when the target is severely occluded or out of view. To avoid error model information being introduced into tracker, a high-confidence model update method is introduced to control the frequency of model updates.

As mentioned in Sect. 3.2, the simple measurement of peak strength is called the peak-to-sidelobe ratio (PSR). The larger the PSR value is, the higher the confidence degree of the tracking result is. As shown in Fig. 5, in the "Suv" sequence, from the 670th to 695th frames, the moving vehicle is occluded by trees. It can be clearly seen that when the moving vehicle is completely occluded, the confidence map fluctuate intensely, and the PSR value decreases from 19.0528 to 3.5585. In this case, the tracking model is unreli-



(a) (b) (c)

(d) (e) (f)

Fig. 5 Changes in the confidence map of the Suv image sequence

able, and it is necessary to stop updating the tracking model during this period to avoid erroneous model information being introduced into the tracker. When the PSR value of a certain frame is greater than the historical average, the tracking result of the current frame is considered to be highly reliable.

However, PSR only reflects the confidence of current frame tracking result, so it cannot be used as the learning rate for model update. In this paper, we use the ratio of PSR between adjacent frames to represent the relative margin of confidence of tracking model. The learning rate is defined as:

$$\eta = \mathrm{b}\left(I_t^{\mathrm{PSR}}/I_{t-1}^{\mathrm{PSR}}\right) \tag{14}$$

Here, I_{t-1}^{PSR} denotes the PSR value of the $t-1$th frame, I_t^{PSR} denotes the PSR value of the tth frame, and b is a proportional parameter used to adjust the weight of $I_t^{\mathrm{PSR}}/I_{t-1}^{\mathrm{PSR}}$. Let $\beta = I_t^{\mathrm{PSR}}/I_{t-1}^{\mathrm{PSR}}$, usually β belongs to 0 to 2, which reflects the ratio of the confidence of the tracking result between adjacent frames. When β is greater than 1, the confidence of the tracking result is higher; otherwise, the confidence of the tracking result is lower.

4 Experiments

4.1 Experiment setup

In the experiments, the regularization parameter λ is set to 10^{-2}, the size of the translation estimation search window is set to 1.5 times the target size, the learning rate η ranges from 0 to 0.3, the scale series S is 33, the scale factor a is 1.02, and the parameter b is 0.15. For the detection and description of keypoints, we employ BRISK [32] with a dimensionality $d = 512$. For matching candidate keypoints to the model, the ratio threshold ρ is set to 0.8, the cutoff threshold $\delta = 20$, and the parameter θ is 0.1.

The experiments are carried out using Matlab 2016a and Visual Studio 2013+OpenCV 3.1.0 on a computer with Intel (R) Core (TM) i5-4590 CPU@3.30GHz+RAM 4GB whose operating system is Windows 10.

4.2 Experiment datasets

To evaluate our approach, we perform comprehensive experiments on two benchmark datasets: OTB-2015 [2], VOT-2016 [26]. In the following sections, the proposed tracker is denoted as “RHCT”.

4.2.1 OTB-2015 dataset

We evaluate the proposed tracking method on the OTB-2015 benchmark dataset and compare it with the state-of-the-art methods.

The OTB-2015 benchmark dataset provides 100 video sequences with ground-truth object locations and attributes for performance analysis. All these sequences are annotated with 11 attributes which cover various challenging factors, including scale variation (SV), occlusion (OCC), illumination variation (IV), motion blur (MB), deformation (DEF), fast motion (FM), out-of-plane rotation (OPR), background clutters (BC), out of view (OV), in-plane rotation (IPR) and low resolution (LR). Each sequence includes several attributes.

It uses precision and success plots to evaluate the performance of the algorithm. The success plot shows the fraction of frames with the overlap between the predicted and ground-truth bounding box greater than a threshold with respect to all threshold values. The precision plot shows similar statistics on the center error.

4.2.2 VOT-2016 dataset

For further validating the effectiveness of the proposed method, we also compare with other tracking approaches on the VOT-2016 challenge dataset.The VOT-2016 dataset consists of 60 challenging videos. The VOT-2016 benchmark contains results of 70 state-of-the-art trackers evaluated on 60 challenging sequences. For each sequence in the dataset, a tracker is evaluated by initializing it in the first frame and then restarting the tracker whenever the target is lost. The tracker is then initialized a few frames after the occurred failure. The overall performance is evaluated using expected average overlap (EAO) which accounts for both accuracy and robustness, and the equivalent filter operations (EFO) is used to evaluate the tracking speed of the algorithm. We refer to [26] for a detailed description of the VOT evaluation methodology.

Table 1 The tracking performance and processing speed of DSST, RHCT-v1, RHCT-v2, and RHCT on OTB-2015 dataset

Trackers	F&R	HCU	Mean DP	Mean OP	Mean FPS
DSST	NO	NO	71.2	60.6	25.1
RHCT-v1	NO	YES	76.8	68.4	**28.5**
RHCT-v2	YES	NO	84.5	73.7	20.1
RHCT	YES	YES	**88.2**	**78.6**	22.4

The Bold font denote the best results

4.3 Analyses of RHCT method

In this section, in order to analyze the contributions from failure detection and redetection module, high-confidence update strategy to the final tracking performance, the OTB-2015 and VOT-2016 datasets are used to evaluate the performance of the algorithm. We denote RHCT without failure detection and redetection (F&R) as RHCT-v1, without high-confidence update strategy (HCU) as RHCT-v2.

The tracking performance and processing speed on OTB-2015 dataset are shown in Table 1. As shown in Table 1, RHCT demonstrates the best tracking accuracy and the second fastest speed. Without both of F&R and HCU, DSST reaches the last one in distance precision(DP) and overlap precision(OP), but the tracking speed of DSST is relatively high. Without F&R, RHCT-v1 gets poor performance. That is because of tracking failure caused by occlusion or out of view, resulting in poor performance. But the tracking speed of RHCT-v1 is the best one. Without HCU, RHCT-v2 updates the tracking model in each frame, thus the tracking speed is the lowest. However, the failure detection and redetection module are introduced into the algorithm, so RHCT-v2 has high tracking accuracy. In addition,we also provide a comparison of RHCT tracker with DSST, RHCT-v1, and RHCT-v2 on VOT-2016 benchmark dataset. The tracking performance of overlap, accuracy, EAO, and EFO are shown in Table 2. As shown in Table 2, RHCT achieves the best performance of overlap, accuracy, EAO, and EFO is the second highest.

As shown in Tables 1 and 2, although the proposed method increases the amount of computation, the proposed method can achieve promising performance on visual tracking. The experimental results show the effectiveness of failure detection and redetection module, high-confidence update strategy.

4.4 Experimental results and analysis

4.4.1 Experiment 1: threshold setting analysis

As mentioned before, the tracking result of each frame has its corresponding confidence score $y^t_{\max}$ (t denotes the image of the tth frame of the video sequence). In the case of successful tracking, the confidence score $y^t_{\max}$ can represents the

Table 2 The tracking performance of DSST, RHCT-v1, RHCT-v2, and RHCT on VOT-2016 dataset

Trackers	F&R	HCU	Overlap	Accuracy	EAO	EFO
DSST	NO	NO	0.5146	0.5037	0.1806	9.7141
RHCT-v1	NO	YES	0.5263	0.5172	0.2347	**10.4623**
RHCT-v2	YES	NO	0.5431	0.5283	0.2864	7.8345
RHCT	YES	YES	**0.5593**	**0.5385**	**0.3320**	8.2962

The bold font denotes the best results

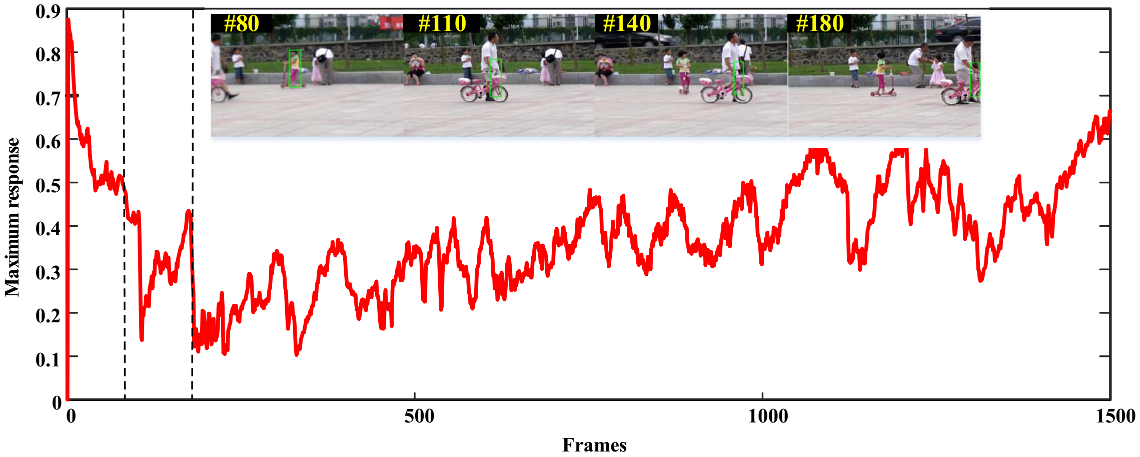


Fig. 6 The maximum confidence score curve of "Girl2" Sequence

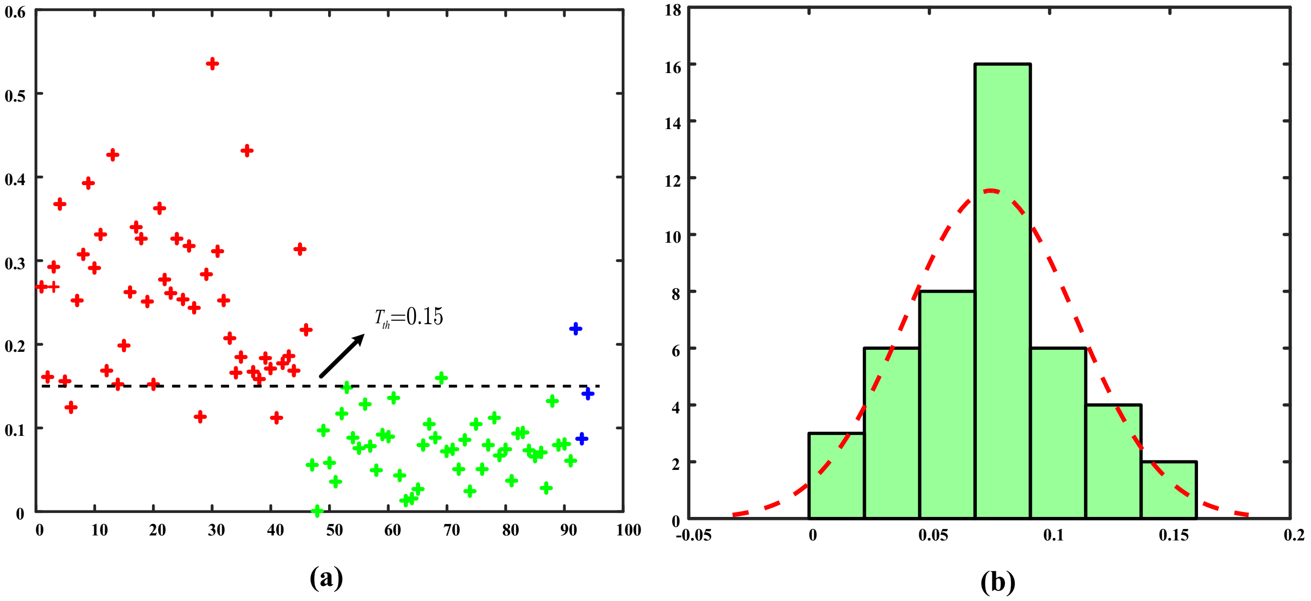


Fig. 7 Distribution of the minimum $y_{\max}$ for each video sequence in OTB-2015 dataset: **a** The minimum $y_{\max}$ distribution of each video sequence, **b** The Gauss distribution curve of the minimum $y_{\max}$ of the video sequence with failed tracking

credibility of tracking result to a certain extent. The larger $y_{\max}^t$ is, the more accurate the tracking result is. However, we need pay attention to some specific cased, for example, as shown in Fig. 6, when the target is occluded at the 110th frame, the tracking failure occurs, resulting in tracking an erroneous target object. Although the subsequent maximum confidence score will continue to increase, the maximum confidence score has no practical significance because the following tracking is an erroneous target object.

In this section, the OTB-2015 dataset is used to evaluate the proposed method. As shown in Fig. 7, a confidence interval is put forward based on Eq.(15). When the confidence $\mathrm{TR}_{\mathrm{con}}^t$ of the tracking result of the $t$thframe belongs to the set (0, 0.15), the confidence of the tracking result is considered to be low.

$$\mathrm{TR}_{\mathrm{con}}^t = \begin{cases} \text{High} \;\; y_{\max}^t \geq 0.3 \\ \text{Low} \;\; 0 \leq y_{\max}^t \leq 0.15 \end{cases} \tag{15}$$

Figure 7a shows the minimum $y_{\max}$ distribution of each video sequence tested by the DSST tracker on the OTB-2015 dataset. The red asterisks indicate the minimum $y_{\max}$ distribution of the sequence that was successfully tracked. The green asterisks indicate the minimum $y_{\max}$ distribution of the video sequence with failed tracking, and the blue asterisks indicate the minimum $y_{\max}$ distribution of the sequence with poor tracking effect. Figure 7b shows the Gauss distribution curve of the minimum $y_{\max}$ of the video sequence with failed tracking. As shown in Fig. 6, when the maximal confidence score is less than the threshold $T_{\mathrm{th}} = 0.15$, the tracking result is considered unreliable.

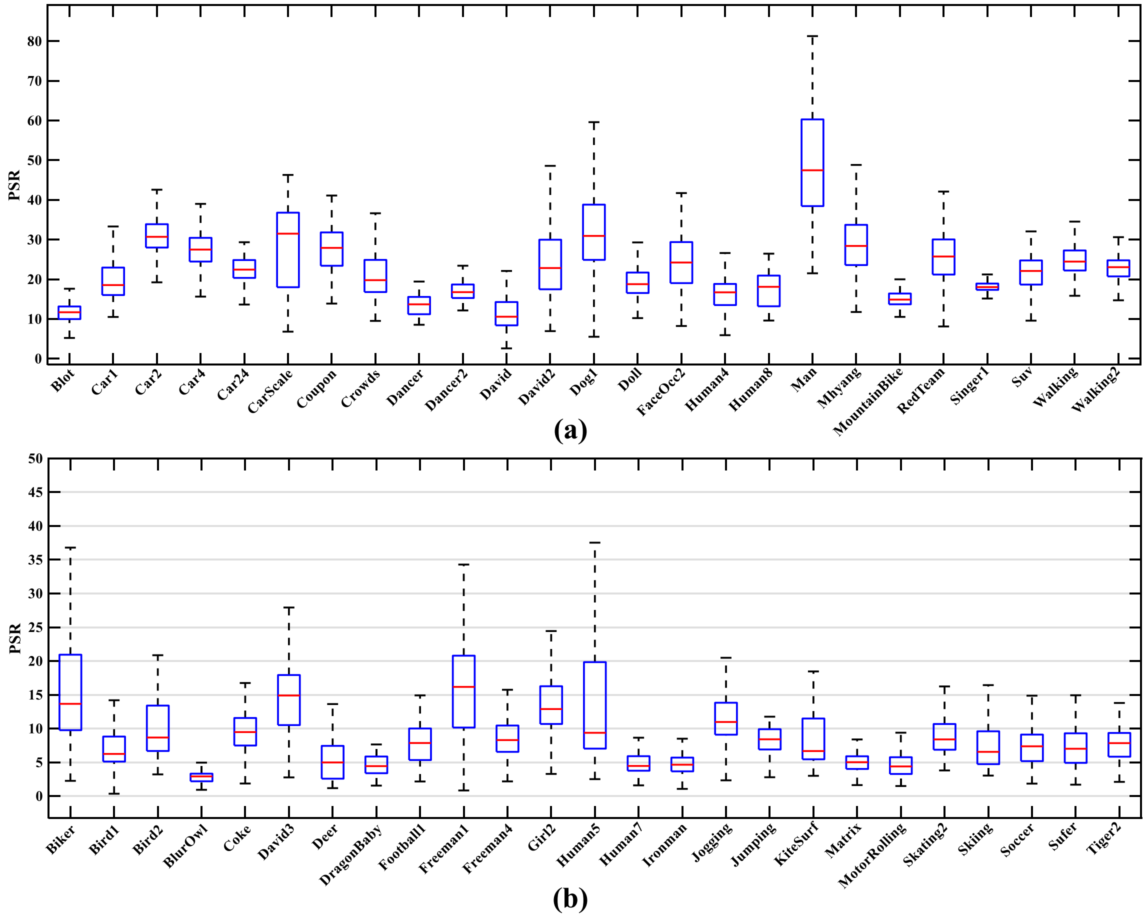


Fig. 8 Bar plots of 50 video sequence practical difference: **a** Bar plots of PSR of 25 video sequences with better tracking result, **b** bar plots of PSR of 25 video sequences with failed tracking

For the OTB-2015 dataset, Fig. 8a shows that under the condition of successful tracking, the PSR is usually between 10.0 and 50.0, which means that the tracking result is with high confidence. As shown in Fig. 8b, we find that the minimum PSR is less than 5.0 for sequences with failed tracking, and we can set the threshold $T_{\mathrm{PSR}} = 5.0$.

First, according to the change of $y^t_{\max}$, the reliability of the tracking result is initially determined. When $y^t_{\max}$ satisfies the unreliable confidence interval, then the PSR value is further calculated. If $T^t_{\max}$ is less than 5.0, the target tracking is considered to have failed. When the tracking failure occurs, the redetection module is activated to relocate and track the correct target.

4.4.2 Experiment 2: evaluation on out-of-view dataset

In the OTB-2015 dataset, the dataset that out of view includes 14 image sequences, among which have challenging characteristics such as occlusion, illumination variation, rotation, and deformation.

The precision and success plots of on out-of-view dataset are shown in Fig. 9. From Fig. 9, it can be seen that the proposed method performs favorably against the state-of-the-art trackers on out-of-view dataset.

In addition, we also provide further experimental evaluation on 14 out-of-view videos in the OTB-2015 dataset. In "Appendix" (as shown in Table 5), we give the average accuracy of the proposed method and the state-of-the-art trackers on out-of-view dataset. It is obvious to see that our method has achieved the best performance in OV.

As shown in Table 5, it shows the average overlap accuracy of the proposed method for each sequence and is compared to seven (7) state-of-the-art trackers. The best results are highlighted in bold. Compared with the existing trackers, the proposed method performs better, with an average overlap precision of 67.9% , which is 23.7% higher than the DSST algorithm. The Board, Box, and Liquor sequences mainly have occlusion challenges. The DSST tracker fails to track on Board, Box and Liquor. When the target appears in the field of vision again after the failed tracking, it cannot track the target continuously. The proposed method combines the

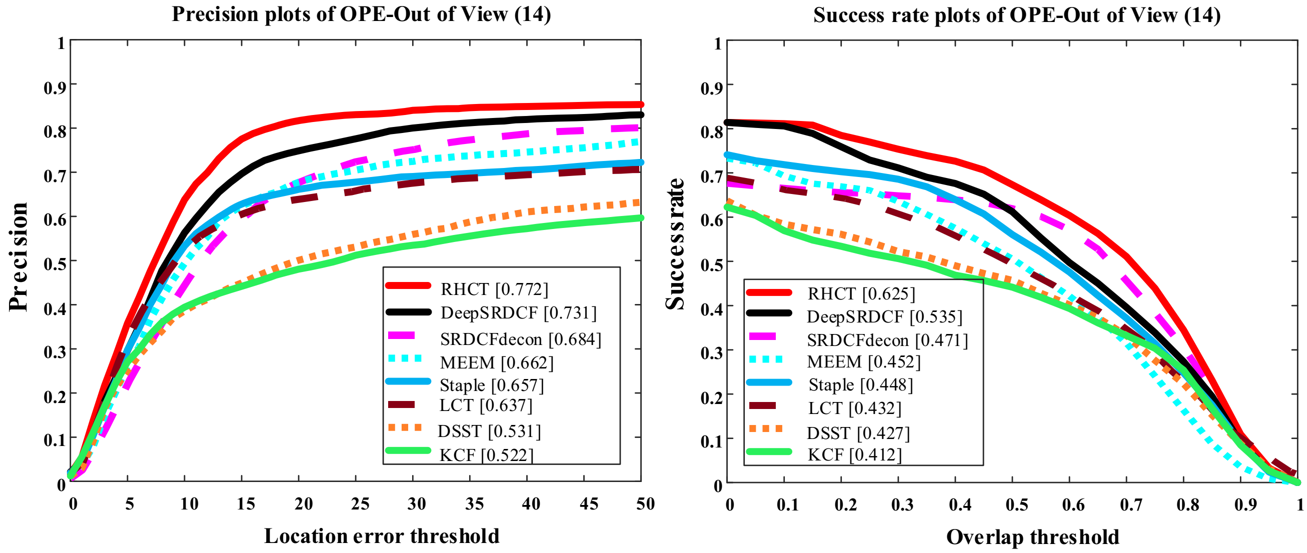


Fig. 9 The precision and success plots on out-of-view dataset

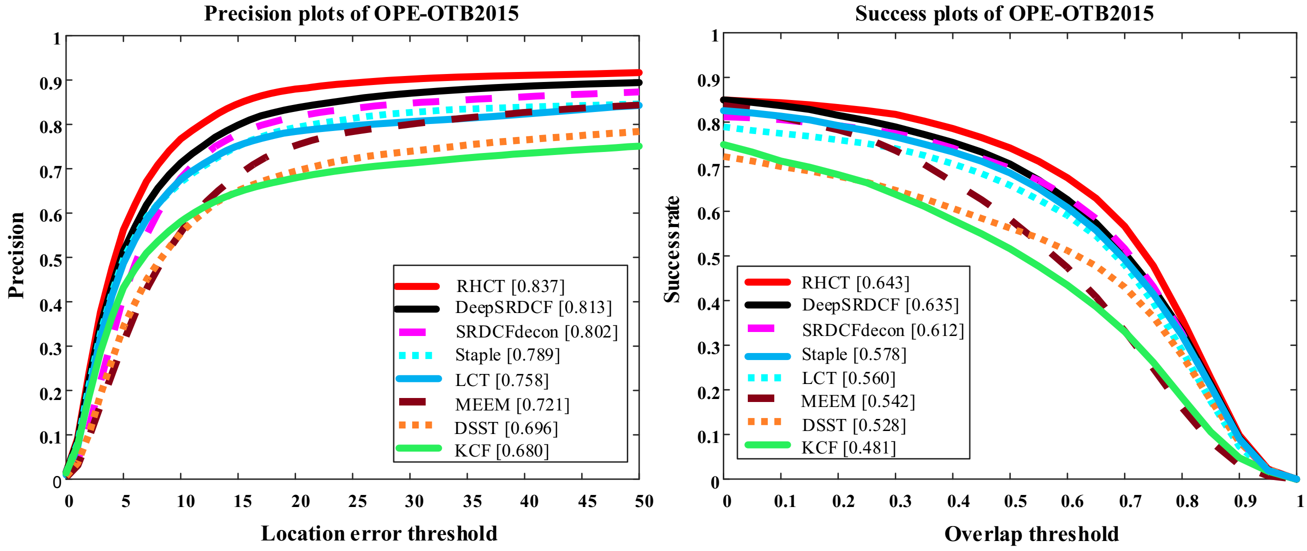


Fig. 10 The precision and success plots over 100 sequences on the OTB-2015 benchmark dataset

Table 3 Comparisons of the proposed method and state-of-the-art trackers on the OTB-2015 dataset

	RHCT	DeepSRDCF	SRDCFdecon	Staple	LCT	MEEM	DSST	KCF
Mean DP	**88.2**	86.5	84.3	82.8	80.4	75.3	71.2	70
Mean OP	**78.6**	77.4	76.7	70.2	64.5	63.2	60.6	54.8
Mean FPS	22.4	0.36	0.9	29.6	26.4	18.7	25.1	**84.2**

Bold values denote the best results

redetection module and the high-confidence updating module and can conquer the challenges of these sequences.

4.5 Experiment 3: evaluation on OTB-2015 dataset

To validate the comprehensive performance of the proposed method, the proposed method is compared with seven state-of-the-art trackers including DeepSRDCF, SRDCFdecon [33], LCT [24], Staple [31], MEEM [34], DSST, KCF based on the OTB-2015 benchmark datasets [2].

The precision and success plots are shown in Fig. 10, which shows our tracker is superior comparing to state-of-the-art trackers on the OTB-2015 dataset. And the proposed method is compared quantitatively with seven state-of-the-

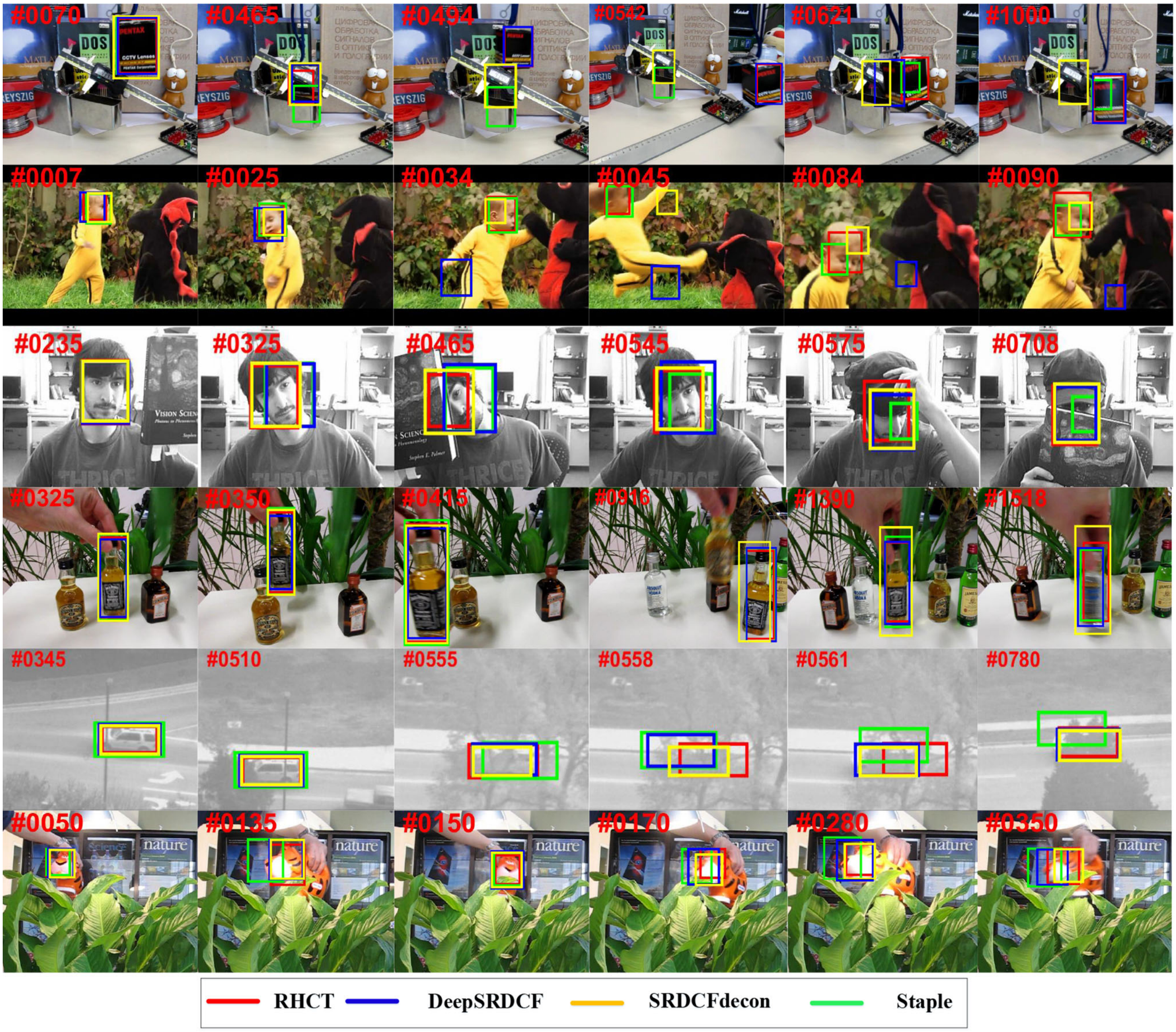


Fig. 11 Tracking results of three approaches (DeepSRDCF, SRDCFdecon and Staple) and our approach on six challenging sequences (from left to right and top down are Box, DragonBaby, FaceOcc2, Liquor, Suv, and Tiger2)

art trackers as shown in Table 3. From Table 3, it is obvious that the proposed method is superior to the existing methods in distance precision (DP) and overlap precision (OP). Among the tracking methods, DeepSRDCF achieves better results; the average DP and OP are 86.5% and 77.4%, respectively. However, the proposed method achieves the best test results. The average DP and OP are 88.2 and 78.6%, respectively, which are 1.7 and 1.2% higher than DeepSRDCF. The processing speed of the trackers in measured in mean FPS is also compared. The processing speed of the proposed tracking method is lower than that of DSST, but it is higher than most of the existing methods.

Figure 11 shows the tracking results of ours method, DeepSRDCF, SRDCFdecon, and Staple on challenging sequences. Figure 11 shows that the proposed method can track the object target accurately. The proposed method not only has high tracking accuracy, but also has good adaptability to the challenges of occlusion, out of view, and deformation.

The frame-by-frame comparisons of center location errors and overlap rate on the three challenging sequences are provided in Fig. 12. As shown in Fig. 12, the proposed method achieved the smallest center location error on most challenging sequences, which means the proposed method outperforms the other compared trackers remarkably.

In addition, in "Appendix" (as shown in Fig. 15), we give the average precision for each of the 11 challenging attributes. It is obvious that our tracker can achieve the best performance if there are occlusion, illumination variation, and out of view.

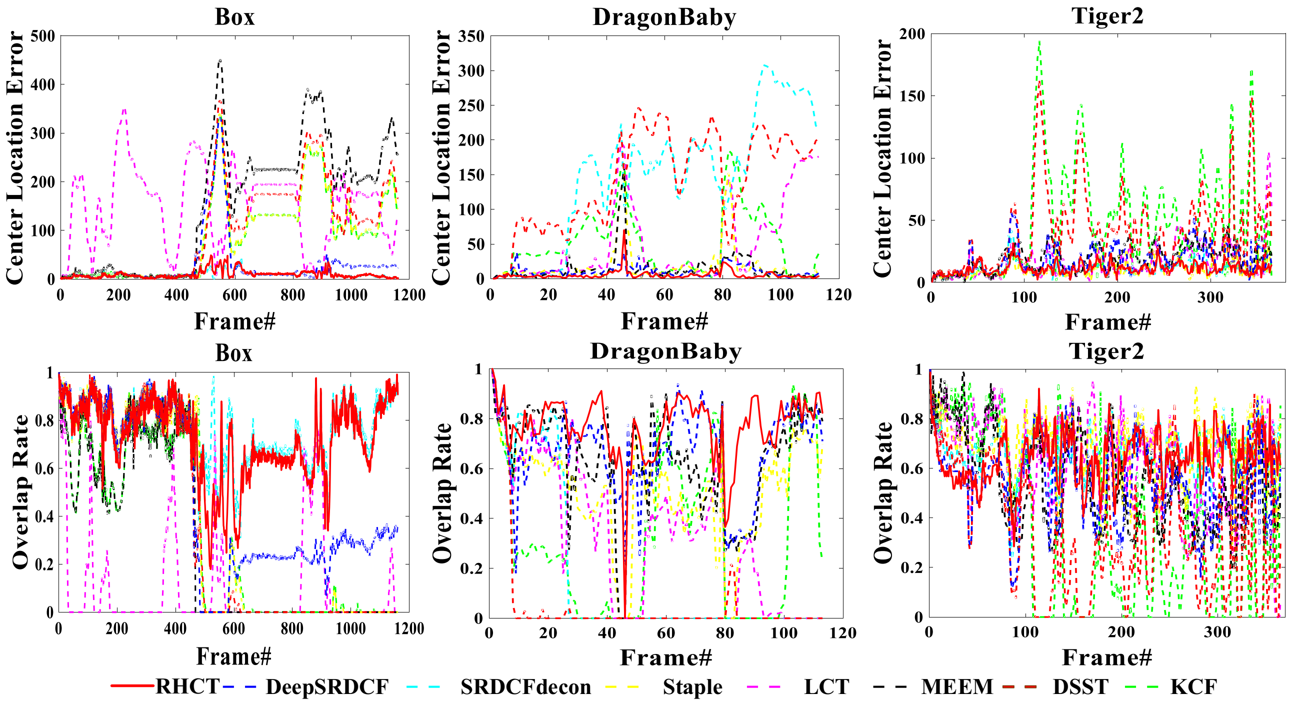


Fig. 12 Frame-by-frame comparison of center location errors and overlap rate on the 3 challenging sequences

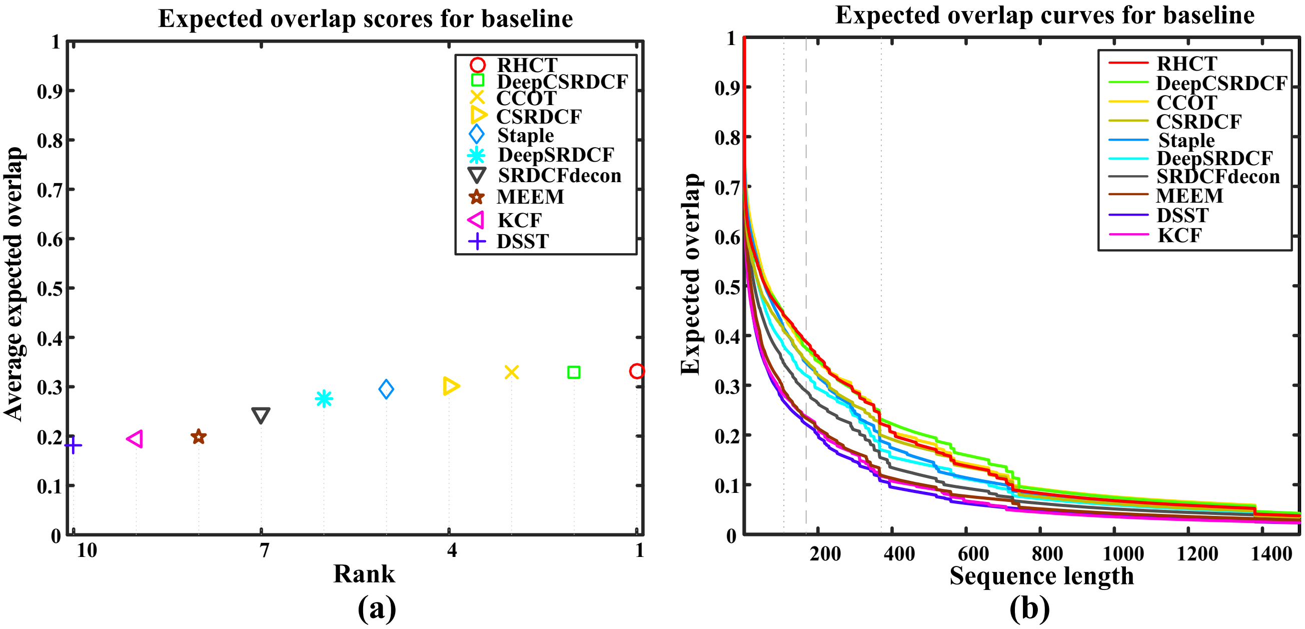


Fig. 13 Expected average overlap (EAO) curve on VOT-2016 dataset

Moreover, our tracker can perform well in the DEF, SV, OPR, and IPR. However, the performance of the proposed tracker is poor in FM, MB, LR, and BC. The main reason is that our method introduces a detector based on Keypoint matching, and the detector cannot be able to accurately locate the position of the target, resulting in low tracking precision if there are fast motion, motion blurring and so on.

4.6 Experiment 4: State-of-the-art comparison on VOT-2016 dataset

We also provide a comprehensive comparison of our trackers with nine state-of-the-art trackers on VOT-2016 benchmark datasets [26]: C-COT [23], DeepSRDCF [22], DeepC-

Table 4 The top-performing trackers on the VOT-2016 benchmark dataset

	Overlap	Accuracy	EAO	EFO
RHCT	**0.5593**	**0.5385**	**0.332**	8.2962
C-COT	0.5296	0.5125	0.3294	0.5023
DeepCSRDCF	0.528	0.5086	0.3294	0.3294
CSRDCF	0.5073	0.5014	0.302	4.6428
Staple	0.5104	0.5346	0.2942	10.9754
DeepSRDCF	0.5221	0.5245	0.2757	0.1286
SRDCFdecon	0.5259	0.5291	0.2459	0.3214
MEEM	0.4765	0.4722	0.1976	6.6786
KCF	0.4916	0.5084	0.1935	**40.7412**
DSST	0.5146	0.5037	0.1806	9.7141

Bold values denote the best results

SRDCF [21], SRDCFdecon [33], CSRDCF [21], Staple [31], DSST [15], KCF [13], and MEEM [34].

Figure 13 shows the EAO (Expected average overlap) plots with the proposed method and the nine state-of-the-art approaches. As shown in Fig. 13a, it shows the expected overlap scores rank of RHCT and other nine state-of-the-art method on VOT-2016 dataset. The proposed method outperforms all trackers and achieves the top rank. And Fig. 13b shows that with the increase in video frames, the expected overlap rate of each tracking algorithm will continue to decrease, but RHCT method has achieved the best results.

Table 4 shows the results reported by the VOT-2016. The first two columns contain the mean overlap score and accuracy over the VOT-2016 dataset. The remaining columns report the expected average overlap (EAO) and Equivalent Filter Operations (EFO) for each tracker. RHCT achieves the best final rank on this dataset. The RHCT outperforms the other nine state-of-the-art trackers with the EAO score equal to 0.332. Among the compared methods, RHCT achieves favorable results in terms of EAO, the mean overlap score and accuracy, at the cost of an EFO.

5 Conclusions

In this paper, a robust tracking algorithm was proposed. This tracking algorithm is based on a redetection module and high-confidence updating the proposed tracking algorithm is divided into four parts: translation and scale estimation, failure detection, redetection module, and high-confidence update module. The correlation filter is used to estimate the translation and scale of the target, and the maximal confidence score and PSR are used to detect the confidence degree of the tracking result. When the confidence of tracking result is below a set threshold, an online detector is used to redetect the target. In addition, in order to reduce the error model information introduced during the tracking process, an adaptive high-confidence update method is used to select a reliable tracking model to train classifier. The experimental results show that the proposed method can achieve promising performance on visual tracking especially there are occlusion and out of view in the video sequences. This will further improve the performance of our object tracking framework. Another research direction is to incorporate deep features into our framework.

Acknowledgements This work was partially supported by the Natural Science Foundation of China (No. 61603274), the Natural Science Foundation of Tianjin (No. 18JCYBJC87700), South African National Research Foundation Grants (Nos. 112108 and 112142), and South African National Research Foundation Incentive Grant (No. 114911), and Eskom Tertiary Education Support Programme Grant.

Compliance with ethical standards

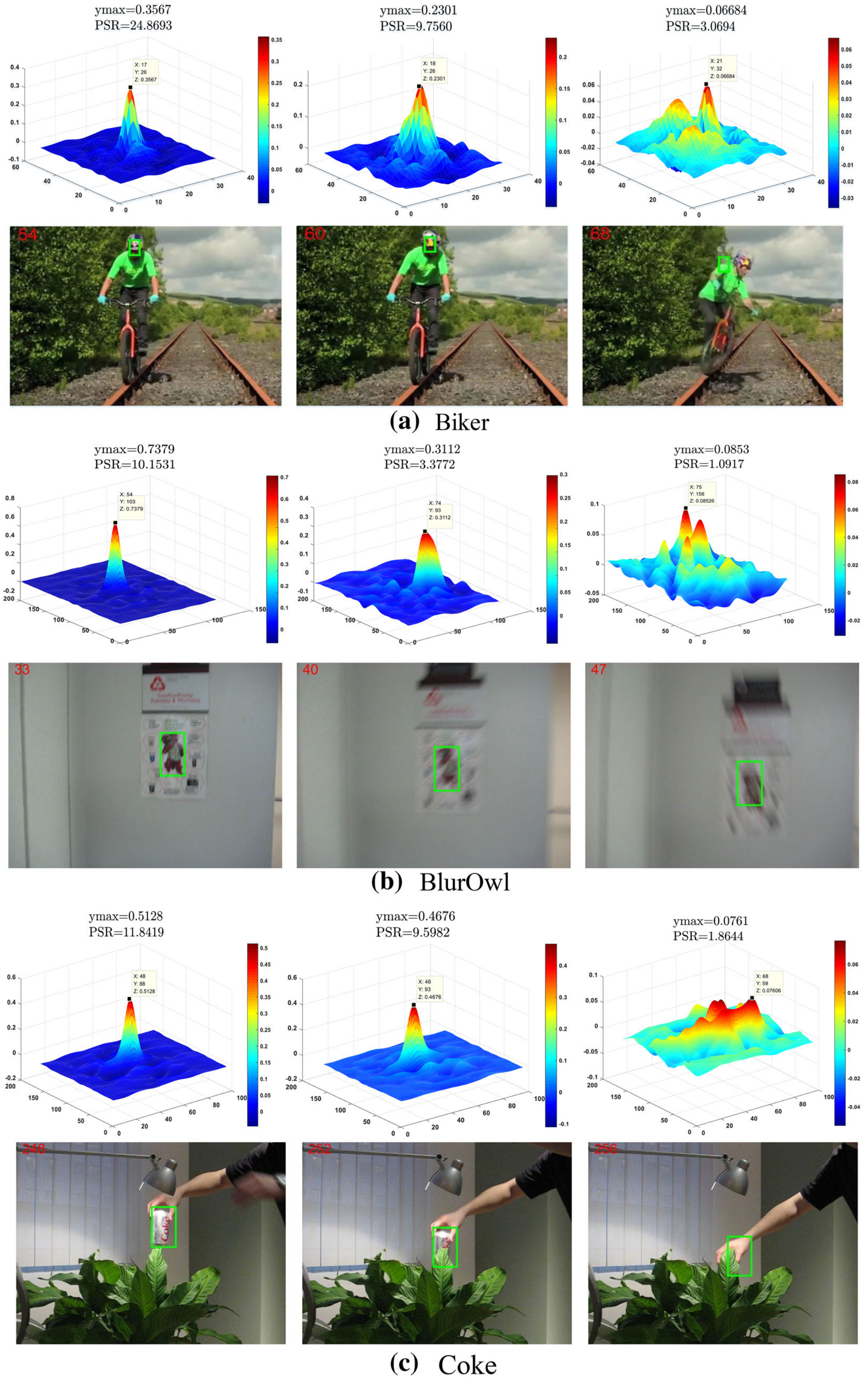
Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix

This appendix contains additional experimental results.

Analyze the effectiveness of PSR

Here, we added some experiment to analyze the effectiveness of PSR in Sect. 3.2. As shown in Fig. 14, the experimental results show that the maximal confidence score and the PSR can reflect the confidence degree about the tracking performance to some extent. Therefore, the maximal confidence score and PSR are used as reference for judging whether the tracking result is reliable.

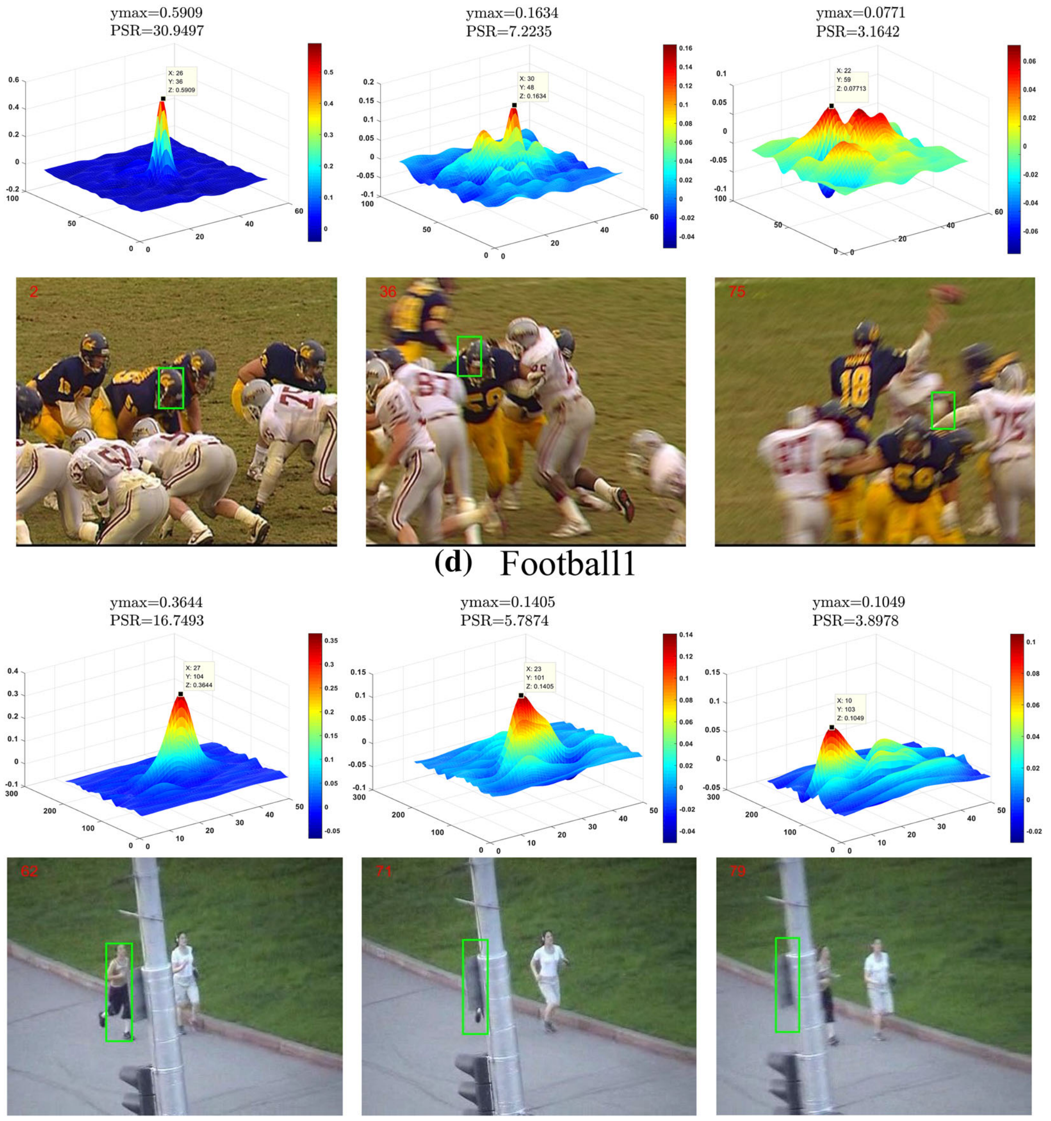


(a) Biker

(b) BlurOwl

(c) Coke

Fig. 14 The changes in the confidence map of the 5 image sequences



(d) Football1

(e) Jogging

Fig. 14 continued

Additional results on out-of-view dataset

Here, we provide further experimental evaluation on 14 out-of-view videos in the OTB-2015 dataset. As shown in Table 5, it shows the average overlap accuracy of the proposed method for each sequence and is compared to seven state-of-the-art trackers.

Table 5 Average overlap accuracy (OP) for each sequence on the out-of-view dataset (%)

	RHCT	DeepSRDCF	SRDCFdecon	LCT	Staple	MEEM	KCF	DSST
Biker	33.4	**52.4**	31.0	33.8	23.7	24.5	22.4	26.8
Bird1	17.2	19.2	4.6	**23.2**	18.8	4.9	5.4	6.6
Board	**91.4**	82.1	83.0	67.3	55.7	59.4	63.3	84.1
Box	**84.6**	46.3	71.4	9.9	35.6	52.3	28.9	39.6
ClifBar	**92.4**	51.4	59.0	53.0	43.5	37.1	24.6	88.6
DragonBaby	52.3	**64.4**	17.9	27.4	45.9	53.1	30.4	6.3
Dudek	**100.0**	82.5	79.8	85.7	70.9	67.6	97.4	98.6
Human6	58.4	**64.4**	36.7	23.3	82.1	19.2	20.4	45.6
Ironman	**21.7**	19.5	6.7	9.7	14.3	40.1	15.1	13.3
Lemming	**85.4**	70.3	75.1	70.1	23.2	66.0	44.2	27.2
Liquor	**92.8**	84.3	8.7	57.4	68.2	75.1	83.9	41.0
Panda	**65.2**	15.4	11.0	25.2	31.2	50.3	16.8	13.3
Suv	**100.0**	54.9	71.8	76.1	80.9	63.9	87.7	98.4
Tiger2	55.3	53.9	62.4	61.1	**68.7**	53.7	36.4	29.6
Mean OP	**67.9**	54.4	49.7	44.5	47.3	47.7	41.2	44.2

Bold values denote the best results

Additional results on OTB-2015 dataset

Here, as shown in Fig. 15, we give the average precision for each of the 11 challenging attributes.

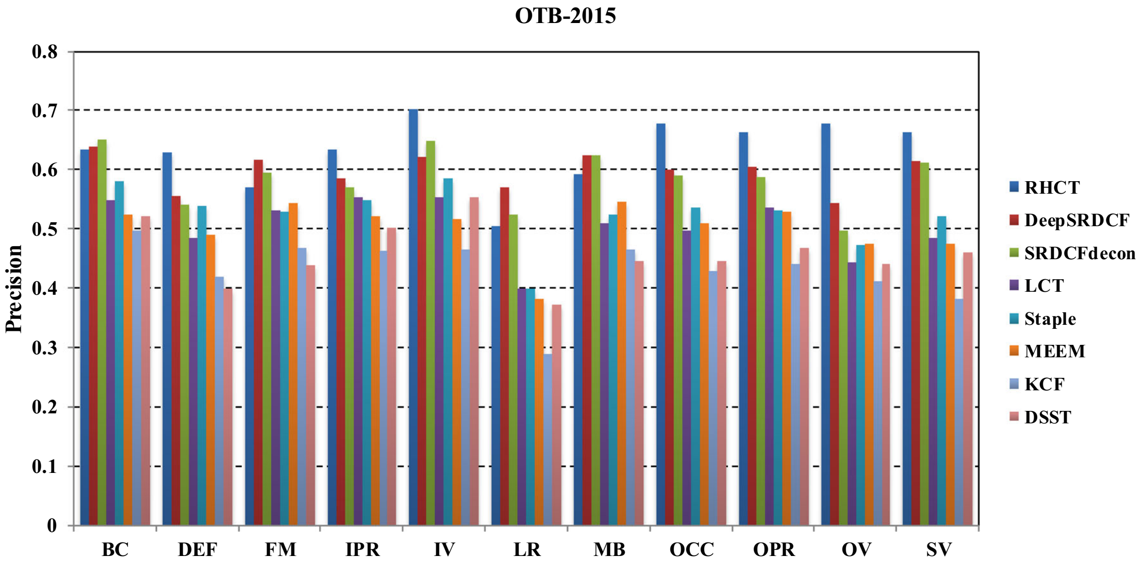


Fig. 15 The average precision over the eleven challenges on the OTB-2015 dataset

References

1. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: a benchmark. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2411-2418 (2013)
2. Wu, Y., Lim, J., Yang, M.H.: Object tracking benchmark. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1834–1848 (2015)
3. Lu, H., Jia, X., Yang, M.H.: Visual tracking via adaptive structural local sparse appearance model. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1822-1829 (2012)
4. He, S.F., Yang, Q., Lau, R., Wang, J.: Visual tracking via locality sensitive histograms. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2427–2434 (2013)
5. Vojir, T., Noskova, J., Matas, J.: Robust scale-adaptive mean-shift for tracking. Pattern Recogn. Lett. **49**, 250–258 (2014)
6. Danelljan, Y.M., Khan, F.S., Felsberg, M.: Adaptive color attributes for real-time visual tracking. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1090-1097 (2014)
7. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. IEEE Trans. Softw. Eng. **34**(7), 1409–1422 (2011)
8. Hare, S., Saffari, A., Torr, P.H.S.: Struck: structured output tracking with Kernels. IEEE Trans. Pattern Anal. Mach. Intell. **38**(10), 2096–2109 (2016)
9. Lu, D., Li, L.S., Yan, Q.S.: A survey: target tracking algorithm based on sparse representation. In: Seventh International Symposium on Computational Intelligence and Design (2015)
10. Zhang, K.H., Zhang, L., Yang, M.H.: Real-time compressive tracking. In: European Conference on Computer Vision, pp. 864–877 (2012)
11. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2544-2550 (2010)
12. João, F., Henriques., Caseiro, R., Martins, P.: Exploiting the circulant structure of tracking-by-detection with Kernels. In: 2012 European Conference on Computer Vision, pp. 702-715 (2012)
13. Henriques, J.F., Caseiro, R., Martins, P.: High-speed tracking with Kernelized correlation filters. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 583–596 (2015)
14. Danelljan, M., Khan, F.S., Felsberg, M.: Adaptive color attributes for real-time visual tracking. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1090–1097 (2014)
15. Danelljan, M., Häger, G., Khan, F.S.: Accurate scale estimation for robust visual tracking. In: Proceedings of British Machine Vision Conference (2014)
16. Wang, M.M., Liu, Y., Huang, Z.: Large margin object tracking with circulant feature maps. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, pp. 4800–4808 (2017)
17. Danelljan, M., Häger, G., Gustav., Khan, FS.: Learning spatially regularized correlation filters for visual tracking. In: 2015 IEEE International Conference on Computer Vision, pp. 4310–4318 (2015)
18. Zhang, D., Zhang, Z., Zou, L.: Part-based visual tracking with spatially regularized correlation filters. Vis. Comput. **36**, 509–527 (2020). https://doi.org/10.1007/s00371-019-01634-5
19. Zhang, H., Liu, G.: Coupled-layer based visual tracking via adaptive kernelized correlation filters. Visual Comput. **34**(1), 41–54 (2018)
20. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration, pp. **254–265** (2014)
21. Lukežič, A., Tomáš, V., Luka, Č.: Discriminative correlation filter with channel and spatial reliability. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, pp. 4847–4856 (2017)
22. Danelljan, M., Häger, G., Khan, F.S.: Convolutional features for correlation filter based visual tracking. In: 2015 IEEE International Conference on Computer Vision Workshop, pp. 621–629 (2015)
23. Danelljan, M., Robinson, A., Khan, F.S.: Beyond correlation filters: learning continuous convolution operators for visual tracking. In: European Conference on Computer Vision, pp. 472–488 (2016)
24. Ma, C., Yang, X., Zhang, N.C.: Long-term correlation tracking. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition, pp. 5388–5396 (2015)
25. Li, C.L., Lin, L., Zuo, W.M.: Visual tracking via dynamic graph learning. IEEE Trans. Pattern Anal. Mach. Intell. **41**(11), 2770–2782 (2019)
26. Kristan, M., Leonardis, A., Matas, J.: The visual object tracking VOT2016 challenge results. In: IEEE International Conference on Computer Vision Workshops, pp. 191–217 (2016)
27. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
28. Nebehay, G., Pflugfelder, R.: Clustering of static-adaptive correspondences for deformable object tracking. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2784–2791 (2015)
29. Xu, R., Wunsch, D.: Survey of clustering algorithms. IEEE Trans. Neural Netw. **16**(3), 645–678 (2005)
30. Nebehay, G., Pflugfelder, R.: Consensus-based matching and tracking of keypoints for object tracking. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2784–2791 (2015)
31. Bertinetto, L., Valmadre, J., Golodetz, S.: Staple: complementary learners for real-time tracking. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1401–1409 (2016)
32. Leutenegger, S., Chli, M., Siegwart, R.Y.: BRISK: binary robust invariant scalable keypoints. In: 2011 International Conference on Computer Vision, pp. 2548–2555 (2011)
33. Danelljan, M., Häger, Gustav., Khan, F.S.: Adaptive decontamination of the training set: a unified formulation for discriminative visual tracking. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (2016)
34. Zhang, J.M., Ma, S.G., Sclaroff, S.: MEEM: Robust tracking via multiple experts using entropy minimization. In: European Conference on Computer Vision, pp. 188–203 (2014)

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

Enzeng Dong graduated with Doctoral degree in Operational Research and Cybernetics, Nankai University, China, in 2006. From 2016, he is a professor in School of Electrical and Electronic Engineering, Tianjin University of Technology. He is mainly working on machine vision and pattern recognition

Mengtao Deng received his Bachelor degree in Electrical and Electronic Engineering from Tianjin University of Technology, China, in 2017. He is a graduate student at Tianjin University of Technology in control science and Engineering. Her research interests include image processing, object tracking and so on.

Zenghui Wang graduated with Doctoral degree in Control Theory and Control Engineering, Nankai University, China, in 2007. He is currently a Professor in the Department of Electrical and Mining Engineering at University of South Africa. His research interest is in the fields of evolutionary optimization, image and video processing, artificial intelligence, and so on.