The Visual Computer (2020) 36:509–527
https://doi.org/10.1007/s00371-019-01634-5

ORIGINAL ARTICLE

Part-based visual tracking with spatially regularized correlation filters

Dejun Zhang[1] · Zhao Zhang[2] · Lu Zou[2] · Zhuyang Xie[2] · Fazhi He[3] · Yiqi Wu[4] · Zhigang Tu[5]

Published online: 13 February 2019

Abstract
Discriminative Correlation Filters (DCFs) have demonstrated excellent performance in visual object tracking. These methods utilize a periodic assumption of the training samples to efficiently learn a classifier on image patches; unfortunately, this also introduces unwanted boundary effects. Recently, Spatially Regularized Discriminative Correlation Filters (SRDCFs) were proposed to resolve this issue by introducing penalization weights to the filter coefficients, thereby efficiently reducing boundary effects by assigning higher weights to the background. However, due to the variable target scale, defining the penalization ratio is non trivial; thus, it is possible to penalize the image content while also penalizing the background. In this paper, we investigate SRDCFs and present a novel and efficient part-based tracking framework by exploiting multiple SRDCFs. Compared with existing trackers, the proposed method has several advantages. (1) We define multiple correlation filters to extract features within the range of the object, thereby alleviating the boundary effect problem and avoiding penalization of the target content. (2) Through the combination of cyclic object shifts with penalized filters to build part-based object trackers, there is no need to divide training samples into parts. (3) Comprehensive comparisons demonstrate that our approach achieves a performance equivalent to that of the baseline SRDCF tracker on a set of benchmark datasets, namely, OTB2013, OTB2015 and VOT2017. In addition, compared with other state-of-the-art trackers, our approach demonstrates superior performance.

Keywords Correlation filter tracking · Discriminative Correlation Filter · Part-based tracking · Spatially regularized filter

1 Introduction

Visual object tracking constitutes one of the most fundamental problems in the field of computer vision and has numerous applications, such as human–computer interaction, vehicle navigation and automatic surveillance. Generic tracking problems are considered online learning tasks in which the trajectory of a target is estimated within an image sequence specified by a bounding box in its first frame. Such problems are very challenging because the target can undergo numerous rapid variations, making it difficult to constrain. Typical examples of nuisances that must be overcome include scale variations, geometric deformations and occlusions.

Most state-of-the-art approaches [2,6,9,13,14,17,18,28,29,46,53] tackle tracking problems by learning the discriminative appearance model of the object target; this model is known as a classifier. In these scenarios, the tracker finds the target location that can differentiate the target from the environment. Recently, Discriminative Correlation Filter (DCF)-based approaches [2,3,6,13,14,18,22,27,45] have been shown to achieve a fairly rapid and robust tracking performance, and thus, they have attracted considerable attention. The main idea of a DCF is to learn a correlation filter that is used to localize the object within the next frame by identifying the location of the maximal correlation response (detection step); then, the location of the object is updated by computing a filter whose correlation with the training templates closely resembles a hand-crafted target response (training step), which is usually taken as a Gaussian centered on the current tracking result [2]. Nearly all DCF-based trackers utilize a periodic assumption of the training samples to

✉ Dejun Zhang
zhangdejun@cug.edu.cn; djz@sicau.edu.cn

1 Faculty of Information Engineering, China University of Geosciences, Wuhan 430074, China

2 College of Information and Engineering, Sichuan Agricultural University, Yaan 625014, China

3 School of Computer Science and Technology, Wuhan University, Wuhan 430072, China

4 College of Computer Science, China University of Geosciences, Wuhan 430074, China

5 School of Electrical and Electronic Engineering, Nanyang Technological University, 50 Nanyang Avenue 639798, Singapore

improve the efficiencies of both the correlation filter training and the target detection computation. However, this periodic assumption also introduces unwanted boundary effects; as a consequence, the learned filter usually contains substantial amounts of background information, thereby leading to problems with the growth of the search area [15].

Recently, Danelljan et al. [6] proposed the use of Spatially Regularized Discriminative Correlation Filters (SRDCFs) to alleviate the above-mentioned boundary problem. In the SRDCF, a spatial regularization component is introduced to penalize the correlation filter coefficients. Filter coefficients residing in the background region are penalized by assigning higher weights, thereby forcing the correlation filters to concentrate on the centers of the training patches and significantly mitigating the emphasis on background information in the learned classifier. However, there are two problems with the SRDCF tracker. First, the penalization of filter coefficients is a smoothly increasing function; therefore, the boundary edge between the object and background becomes confusing. However, the SRDCF simply regards the area beyond 1/2 of the target range as the background, thereby neglecting the boundary edge and inevitably penalizing the target content while also penalizing the background. Second, due to these penalized weights, the correlation filter cannot extract complete features from the target area.

In addition to DCF-based trackers, other researchers [18,23,28] have successfully applied part-based tracking strategies to correlation filter tracking. The main idea of part-based tracking is to construct an object appearance model based on multiple parts of an object. When an object is partially occluded, the remaining visible parts can still provide reliable cues for tracking. Therefore, part-based algorithms are helpful for improving the robustness of the tracking performance in the event of partial occlusions.

In this paper, we investigated the SRDCF framework and observed that the maximal target response of the correlation filter is actually mapped to the central location of the correlation filter learning area, indicating that the maximal target response can be used to determine the receptive field of the correlation filter. Therefore, when the penalization ratio is reasonably set, the correlation filter can extract features from different parts of an object by altering the location mapping to coincide with the maximal target response.

Based on the observations above, we present a novel and efficient Part-based Spatially Regularized Discriminative Correlation Filters (PSRDCF) (Sect. 4). In contrast to previous part-based methods, in the PSRDCF, it is not necessary to cut the object into parts to separately train the associated correlation filters. In this work, we introduce an enhanced spatial regularizer to penalize correlation filters corresponding to their spatial location, making the learning area of the correlation filters much smaller than the target area. Thus, the convolutional features of different parts of an object can

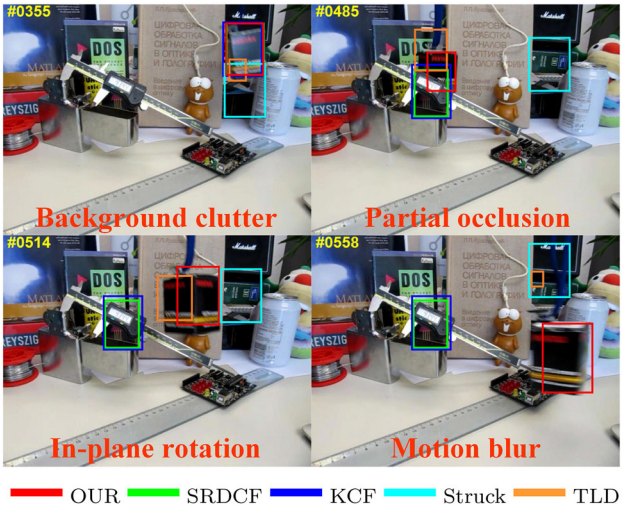


Fig. 1 Comparisons of our approach with the baseline SRDCF tracker and other state-of-the-art trackers in challenging situations of background clutter, partial occlusion, in-plane rotation, and motion blur on the *Box* sequence [41]. The proposed PSRDCF incorporates the part-based tracking strategy into regularized correlation filters to learn more robust convolution features, thereby demonstrating a better tracking performance

be learned by defining regression targets at different locations (Sects. 4.2 and 5.1). Additionally, the coefficients of our correlation filters are learned over the augmented training samples with a circulant structure using different regression targets. Moreover, we assume that the target response for each filter is a Gaussian function. During the tracking precess, the maximal convolution response of all correlation filters is recorded to estimate the state of the object in the current frame (Sects. 5.2 and 5.3). We further justify the SRDCF observations in Sect. 4.1 by altering the traditional target response in the training stage.

Figure 1 illustrates comparisons of our approach with the baseline SRDCF tracker and several other state-of-the-art trackers. In comparison with prior methods, our main contributions are as follows.

- We define multiple correlation filters to extract features within the range of the object; this approach not only alleviates boundary effects but also avoids problems in which the target content is penalized.
- To the best of our knowledge, we are the first to combine cyclic shifts of an object with penalized filters to build part-based trackers. Based on the circulant structure of the training samples, there is no need to divide the training samples into parts.
- Our method shows excellent experimental results on three large-scale benchmark datasets, namely, OTB-2013 with 50 sequences, OTB-2015 with 100 sequences and VOT2017 with 60 sequences.

2 Related work

Visual tracking has been studied extensively in the literature [2,12,27]. In this section, we provide a brief overview of the trackers that are the most relevant to our work.

2.1 Generative versus discriminative tracking

Based on the target appearance model, tracking algorithms are categorized as either generative [20,25,30,46,47,52] or discriminative [16,24,31,33,34,39,42,44,49]. Generative trackers search for a potential target location that is most similar in appearance to the generative model; therefore, it employs information from both the target and the background. Matthews et al. [30] developed a template update method that can reduce the drifting problem by aligning with the first template. Kwon and Lee [20] decomposed the observation model into multiple basic observation models to cover a wide range of pose and illumination variations. Zhang et al. [46] formulated an object tracking algorithm in a particle filter framework as a multitask sparse learning problem, and the algorithm was extended by exploiting the relationships between particles in their subsequent work [47]. Li et al. [25] proposed a comprehensive spatial feature similarity strategy to compute the confidence levels of target features that can be used to determine the current position of a target among candidates during the tracking process. Zhong et al. [52] performed visual tracking in a novel weakly supervised learning scenario where labels but no ground truth are provided by multiple imperfect oracles and proposed a probabilistic approach to simultaneously infer the most likely object position and the accuracy of each tracker.

Compared to generative trackers, discriminative approaches regard tracking as a classification problem in which the tracked targets are distinguished from the background. Wang et al. [39] combined multiple instance learning and Bayes' theorem to take full advantage of the information regarding the targets and their surrounding background. Mbelwa et al. [31] utilized objectness embedded in smoothing stochastic sampling and improved Tree coherency approximate nearest neighbor to address the problem of abrupt motions. Wu et al. [42] exploited scale invariant normalized rectangular features extracted from the adaptive compressive domain to improve the discriminative appearance model. Li et al. [24] adopted a self-adaptive multi-feature fusion strategy to adaptively adjust the joint weights of fused features. Zhao et al. [49] exploited structural sparse representation-based semi-supervised learning and edge detection to improve the performance of the discriminative tracker.

2.2 Discriminative Correlation Filters

Recently, DCF-based approaches that exploit the properties of circular correlation have been shown to achieve a fairly rapid and robust tracking performance [1–3,6,13,14,18,22,27,45], and thus, they have attracted considerable attention with regard to visual tracking. Bolme et al. [3] initially proposed the Minimum Output Sum of Squared Error (MOSSE) tracker to encode the target appearance by learning an adaptive correlation filter that is restricted to using a single feature channel, typically a grayscale image. Henriques et al. exploited the circulant structure of adjacent image patches [13] and further improved the performance by using Histogram of Oriented Gradients (HOG) features [14] or color names [37]. To effectively handle variations in the scale and the drifting problem, Bai et al. [1] fused heterogeneous cues with different object scales to learn an adaptive set of filtering templates; this approach alleviates the drifting problem by carefully selecting object candidates in different situations to jointly capture the variations in the target appearance. Each of these DCF-based trackers uses the Fast Fourier Transform (FFT) to significantly reduce the computational effort required for training and detection based on a periodic assumption of the training samples. However, this periodic assumption also introduces unwanted boundary responses and severely degrades the trackers performance.

To investigate the problem of boundary effects encountered for single-channel DCFs, Galoogahi et al. [18] proposed an approach that solves a constrained optimization problem using the Alternating Direction Method of Multipliers (ADMM) to ensure a correct filter size; however, this method is restricted to the use of single-channel features and is therefore inapplicable for our purposes. Danelljan et al. [6] extended the findings of previous research [18] and presented the SRDCF; they introduced a spatial regularization component to penalize the correlation filter coefficients depending on their spatial location. Their work provided a notable improvement through the learning of multi-channel filters on multi-dimensional features (i.e., HOG features [5]). In another study [2], Bili et al. argued that the correlation operation employed in DCFs represents only an approximation of the actual sample translations; thus, the traditional use of a single centered Gaussian as the target response can lead to unrecoverable drift. To circumvent this problem, Bili et al. presented a generic framework that adaptively changes the target response from frame to frame; as a result, the tracker is less sensitive to cases in which circular shifts do not reliably approximate sample translations.

2.3 Part-based trackers

Many trackers divide an entire target into separate parts instead of learning an appearance model to increase the track-

ing robustness against partial occlusions [4,10,21,23,28]. Felzenszwalb et al. first demonstrated that the deformable parts model [8] can be employed to reliably detect objects even under heavy nongrid transformations and partial occlusions. Cehovin et al. [4] combined the global appearance with local appearance based on object parts using a novel coupled-layer visual model. Godec et al. [10] employed rough segmentation to describe the global appearance of a target. The proposed tracker includes an online feature selecting step, which enables a different part of the local appearance to be described by a different feature. Yang et al. [43] proposed a novel attentional tracking method that utilizes spatially attentional patches, which include salient and discriminative target regions; this method was proved to be robust on a large variety of real-world videos. Kwon et al. [21] represented a nongrid target object by a number of local patches with color histograms. Zhang et al. [48] tracked targets by matching parts among multiple frames. Latent structured learning was used in another study [51] that simultaneously addressed the tracker drift and occlusion problems and proposed a robust visual tracking algorithm via a patch-based adaptive appearance model driven by local background estimation. As demonstrated by the above-mentioned studies, part-based trackers can obtain more robust and accurate tracking results. However, the computational complexity of these methods is high; consequently, it is difficult for multiple part-based trackers to run in real time.

Our work is similar to other studies [23,27,28], insomuch that the part-based strategy is used in the correlation filter. However, in contrast to the direct use of an existing correlation filter to preserve the object structure for object appearance modeling, the above-mentioned approaches utilize correlation filter-based methods as base trackers, which take advantage of the model and perform tracking in the Fourier domain under the tracking-by-detection framework to significantly improve the tracking efficiency.

3 SRDCF

As discussed above, the way to handle the boundary effects in SRDCF [6] is to introduce a spatial regularization component as the penalty term of the correlation filter, enabling the learned filter to be less sensitive to the boundary of the sample. We provide some details of the training of the convolution filters, and the term "convolution" is used because the SRDCF is modeled with convolution instead of correlation [15].

Let f be the convolution filters that are learned from a set of training samples $\{(x_k, y_k)\}_{k=1}^{t}$ sampled at each frame $k = 1, 2, \ldots, t$. Here, t denotes the current frame number. Each training sample $x_k \in \mathbb{R}^{d\times M\times N}$ consists of a d-dimensional feature map (response map) with a spatial size of $M \times N$. y_k denotes the desired target response, which is usually assumed to be a Gaussian function with a peak value centered on the base patch [3]. We denote the feature channel l of x_k by superscript x_k^l, and y_k is the desired convolution response corresponding to training sample x_k. The objective of SRDCF is to learn a correlation filter f^l by minimizing the following loss:

$$\varepsilon_t(f) = \sum_{k=1}^{t} \alpha_k \left\| \sum_{l=1}^{d} x_k^l * f^l - y_k \right\|^2 + \sum_{l=1}^{d} \|\omega \cdot f^l\|^2. \quad (1)$$

Here, $*$ denotes circular convolution generalized to multi-channel signals via conventional means by computing inner products. $\cdot$ denotes element-wise multiplication. The weights $\alpha_k > 0$ determine the impact of each training sample. The regularization weights ω depend on their spatial locations for the importance of the filter coefficients. That is, coefficients in f residing outside the target region are suppressed by assigning higher weights in ω and vice versa. Hence, the emphasis on background information at the detection stage is reduced.

The loss function (Eq. (1)) can be efficiently minimized in the Fourier domain by exploiting the scarcity of the DFT coefficients $\hat{\omega}$. Instead of relying on approximate solutions, Danelljan et al. [6] proposed an iterative minimization scheme based on Gauss–Seidel, which converges to the global minimum of Eq. (1). We refer readers to Ref. [6] for a detailed description of the SRDCF training procedure.

4 Proposed approach

4.1 Motivation

DCFs, including SRDCFs [6], are always trained online with samples collected during tracking. Thus, the potentially large number of samples can become a computational burden that directly conflicts with real-time requirements. For this situation, Henriques et al. [13] introduced the concept of dense sampling in which the negative samples are generated by circular shifts of the actual samples; this approach not only obtains a sufficient number of training samples but also makes the kernel matrix highly structured. SRDCFs still utilize circular shifts to obtain a sufficient number of training samples, thereby enhancing the discriminative power of the learned model by emphasizing the appearance information within the target region. Figure 2 shows an example of circular shifts in a SRDCF.

As shown in Fig. 2, the SRDCF simply penalizes the background area, which is beyond 1/2 of the target size, to alleviate the boundary effects. The penalization is a smoothly increasing function; however, it inevitably affects the target content, leading to unrecoverable drift in many realistic tracking sce-

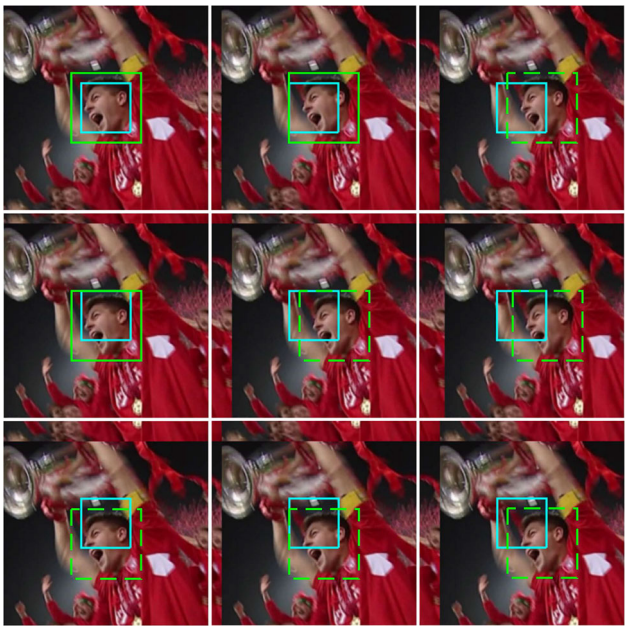


Fig. 2 Illustration of a circulant structure (with image sequence *Soccer* as the original vector) employed in the SRDCF. The green rectangle outlines the target, while the blue rectangle denotes the fixed learning area of the regularized correlation filter in the SRDCF. Due to the effect of penalization, the correlation filter concentrates only on the centers of the training patches, greatly alleviating the boundary effects. However, penalization also causes the correlation filter to extract only the local features of the target in certain patches (denoted as the green dashed rectangles)

narios, such as those involving fast motion and occlusions. Therefore, instead of simply setting the penalization ratio to 1/2, we conduct an in-depth investigation into the relationship between the learning area of the correlation filter and the target response. The results show that the maximal target response maps to the center of the learning area of the correlation filter in the SRDCF (training step). Therefore, the maximal target response can be used to determine the learning area of the correlation filter. When the spatial penalization is sufficiently large, the correlation filter can be employed to extract features from different parts of an object by altering the location mapping to the maximal target response; this approach coincides with the central tenet of training a part-based tracker.

Accordingly, this paper presents a novel part-based tracking framework by exploiting SRDCFs to extract global object features. In general, part-based trackers [23,28] use segmented samples to train structural correlation filters to extract the convolutional features of different target regions while sharing the same regression target. However, we define multiple correlation filters to separately extract the features from different parts of the object. In contrast to previous part-based methods, there is no need to cut the object into parts, and the training input is only an image sample with a circulant structure. Furthermore, a number of regression targets are defined as the label function. All filters are learned over the training sample with a circulant structure, and each filter corresponds to a different regression target to concentrate on different parts of the object. In this way, we can eliminate some redundant computations. Furthermore, we establish the regression response as a Gaussian function with a peak value placed at different positions.

4.2 PSRDCF model

With the aforementioned observations, we apply the part-based strategy to Eq. (1). Let the training sample x_k contains I reliable parts, f_i denotes the correlation filters of the ith part, and its corresponding regression target during training is predefined by y_i. Then, the resulting loss function is expressed as

$$\min_{\{f_i|i\in\{1,2,\ldots,I\}\}} \sum_{i=1}^{I} \varepsilon_t(f_i), \tag{2}$$

where

$$\varepsilon_t(f_i) = \sum_{k=1}^{t} \alpha_k \| \sum_{l=1}^{d} X_k^l * f_i^l - y_{ki} \|^2 + \sum_{l=1}^{d} \| \omega \cdot f_i^l \|^2. \tag{3}$$

Here, $X_k^l = C(x_k^l)$ denotes the circulant data matrix generated by x_k^l as the original vector. Other parameters have the same meaning as in Eq. (1). By applying Parseval's theorem to Eq. (3), the filter sets $\{f_i|i \in \{1, 2, \ldots, I\}\}$ can equivalently be obtained by minimizing the resulting loss function (Eq. (4)) over the Discrete Fourier Transformed (DFT) filters $\hat{f}$.

$$\varepsilon_t(\hat{f}_i) = \sum_{k=1}^{t} \alpha_k \left\| \sum_{l=1}^{d} D(\hat{x}_k^l) \hat{f}_i^l - \hat{y}_{ki} \right\|^2 + \sum_{l=1}^{d} \left\| \frac{C(\hat{\omega})}{MN} \hat{f}_i^l \right\|^2. \tag{4}$$

Here, $D(\hat{x})$ denotes the diagonal matrix with the elements of vector $\hat{x}$ in its diagonal and the hat denotes the DFT of a function. We reformulate Eq. (4) to an equivalent real-valued optimization problem via the method proposed in [6] to ensure faster convergence.

The DFT of a real-valued matrix constructs a matrix that conforms to the Hermitian symmetric. According to the property of the Hermitian symmetric, we construct a mapping to real value the Hermitian symmetric. Let $\tilde{f}_i^l = B\hat{f}_i^l$; $\tilde{f}_i^l$ is real valued by the Hermitian symmetry of $\hat{f}_i^l$. According to the property of the Hermitian symmetric, it can be concluded that B is an extremely sparse matrix that contains, at most, two nonzero entries in each row. Moreover, its value is fixed throughout the tracking procedure [6].

Now, we utilize the real-valued matrix demonstrated above to remapping Eq. (4) into real-valued space. To simplify the optimization function, we define $D_k^l = B\,D(\hat{x}_k^l)\,B^H$, $\tilde{f}_i^l = B\,\hat{f}_i^l$, $\tilde{y}_{ki} = B\,\hat{y}_{ki}$ and $C = \frac{BC(\hat{\omega})B^H}{MN}$

$$\varepsilon_t(\tilde{f}_i) = \sum_{k=1}^{t} \alpha_k \left\| \sum_{l=1}^{d} D_k^l \tilde{f}_i^l - \tilde{y}_{ki} \right\|^2 + \sum_{l=1}^{d} \|C\,\tilde{f}_i^l\|^2. \tag{5}$$

We formulate the multi-channel signals and the corresponding filter matrices into a unified form

$$\varepsilon_t(\tilde{f}_i) = \sum_{k=1}^{t} \alpha_k \|D_k\,\tilde{f}_i - \tilde{y}_{ki}\|^2 + \|W\,\tilde{f}_i\|^2. \tag{6}$$

Here, $D_k = (D_k^1, \ldots, D_k^d)$ and W are the $dMN \times dMN$ block diagonal matrix with each diagonal equal to C.

Now, Eq. (6) can be minimized by a closed form

$$\sum_{k=1}^{t} \alpha_k D_k^T D_k\,\tilde{f}_i - \alpha_k D_k^T\,\tilde{y}_{ki} + W^T W\,\tilde{f}_i = 0. \tag{7}$$

Finally, Eq. (7) can be solved by solving the normal equations $A_t\,\tilde{f}_i = \tilde{b}_{ti}$, where

$$A_t = \sum_{k=1}^{t} \alpha_k D_k^T D_k + W^T W, \tag{8}$$

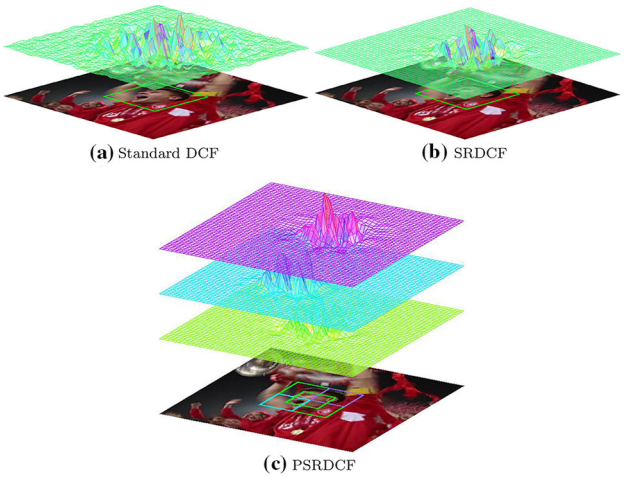
$$\tilde{b}_{ti} = \sum_{k=1}^{t} \alpha_k D_k^T\,\tilde{y}_{ki}. \tag{9}$$

A direct application of a sparse solver to the normal equations $A_t\,\tilde{f}_i = \tilde{b}_{ti}$ is computationally very demanding since the block structure is not attainable due to the structure of the regularization matrix $W^T W$ in Eq. (8). To solve the problem, Danelljan et al. [6] proposed an iterative approach based on the Gauss–Seidel method to efficiently compute the filter coefficients, and the construction of the weights ω ensures that both conditions are satisfied. In this paper, we use the same approach to optimize the problem.

Although the spatial regularizer is enhanced in this work, the properties of the regularization matrix remain the same. Thus, the Gauss–Seidel recursion will still converge to the solution of $A_t\,\tilde{f}_i = \tilde{b}_{ti}$.

According to the Gauss–Seidel method, the matrix A_t is decomposed into a lower triangular part L_t and a strictly upper triangular part U_t such that $A_t = L_t + U_t$. The algorithm then proceeds by solving the following triangular system for $\tilde{f}_i$ in each iteration $j = 1, 2, \ldots,$

$$L_t\,\tilde{f}_i^j = \tilde{b}_{ti} - U_t\,\tilde{f}_i^{j+1}. \tag{10}$$



(a) Standard DCF **(b)** SRDCF **(c)** PSRDCF

Fig. 3 Visualization of the filter coefficients trained using **a** the standard DCF [14], **b** the SRDCF [6] and **c** the proposed PSRDCF

The Gauss–Seidel recursion (Eq. (10)) converges to the solution of $A_t\,\tilde{f}_i = \tilde{b}_{ti}$ whenever the matrix A_t is symmetric and positive definite. Note that all the correlation filters are computed according to Eq. (10) based on A_t, and each correlation filter is computed independently.

Figure 3 illustrates the filters learned by optimizing the standard DCF loss [14], the SRDCF loss (Eq. (1)) and the proposed formulation (Eq. (2)) using the spatial regularization weights ω. The top layer plots the learned filters corresponding to the image region used for training. Here, the target area is outlined by the green rectangle. In standard DCF ((a), without spatial regularization), high convolution scores appear both in the target area and in the background area. In (b) SRDCF and (c) the proposed PSRDCF, regularization weights with a strong penalty on the background and high convolution values only appear around the central area, which increases the discriminative power of the learned model by emphasizing the appearance information within the target region. Furthermore, the proposed PSRDCF extracts features from different object parts, demonstrating robustness in partial occlusions. From the figure, the penalized area of the spatial regularization weights is larger than that of standard SRDCF. In other words, the filters learned by our approach contain less information within the background.

5 Proposed tracking framework

In this section, we introduce the overall tracking framework according to the PSRDCF proposed in Sect. 4.2. We first present an outline of the proposed tracker in Algorithm 1 and show the flowchart of our method in Fig. 4. More details are provided below.

Algorithm 1: PSRDCF algorithm

Input: Initial target state $\{(u, v)^{(0)}, scale^{(0)}\}$
Output: Estimated target state $\{(u, v)^{(t)}, scale^{(t)}\}$;
Correlation filter sets $\{f_i\}_{i\in\{1,2,\ldots,I\}}$
repeat
/*Model update*/
Crop out the sample $x_{(t-1)}$ according to $\{(u, v)^{(t-1)}, scale^{(t-1)}\}$ in frame $t-1$ and extract the features;
Update $A_{(t-1)}$ and $b_{(t-1)i}$ for f_i using Eqs. (11) and (12);
$j \leftarrow 0$;
repeat
Get the filters $\{f_i\}_{i\in\{1,2,\ldots,I\}}$ in frame t-1 using Eq.(10);
until $j \geq N_{GS}$;
/*Target estimation*/
Crop out the searching windows $\{J_s\}_{s\in\{a^c|c=\lfloor(1-C)/2\rfloor,\ldots,\lfloor(C-1)/2\rfloor\}}$ according to $\{(u, v)^{(t-1)}, scale^{(t-1)}\}$ in frame t;
Fast sub-grid detection by Eq. (13);
Get $\{(u, v)_i^{(t)}, score_i^{(t)}, scale_i^{(t)}\}_{i\in\{1,2,\ldots,I\}}$ for each image part using the peaks of all response maps;
Estimate the target state $\{(u, v)^{(t)}, scale^{(t)}\}$ using Eqs. (15), (16) and (17);
until *End of video sequences*;

5.1 Correlation filter learning

During the training stage, the correlation filter f_i is updated by extracting a new training sample x_t centered at the target location and combined with the target responses y_{ti} (*a.k.a* the label function [3]). Here, t denotes the current frame number and i denotes the index of the ith filter. We then update A_t and b_{ti} in Eqs. (8) and (9) with a learning rate $\gamma \geq 0$. In fact, we replace the weight coefficient α_k for each frame with a fixed update rate γ.

$$A_t = (1-\gamma)A_{t-1} + \gamma(D_t^T D_t + W^T W) \tag{11}$$

$$b_{ti} = (1-\gamma)b_{(t-1)i} + \gamma D_t^T \tilde{y}_{ti} \tag{12}$$

In the first frame, we set $A_1 = D_1^T D_1 + W^T W$ and $b_{1i} = D_1^T \tilde{y}_{1i}$. Note that $W^T W$ is a regularization matrix, which is previously computed once for the entire sequence, and all the regression targets are also previously computed. After the model update (Eqs. (8) and (9)), we perform a fixed number N_{GS} of Gauss–Seidel iterations (Eq. (10)) per frame to compute the new filter $\tilde{f}_i$.

5.2 Detection

At the detection stage, the location of the target in a new frame t is estimated by applying the filter $\hat{f}_{t-1}$ obtained in the $t-1$ frame. To handle the problem of scale variation, we construct a pyramid with different resolutions around the target. Let $P \times Q$ denote the target size in a test frame and C be the number of scales $S = \{a^c | c = \lfloor(1-C)/2\rfloor, \ldots, \lfloor(C-1)/2\rfloor\}$. Similar to [29], for each $s \in S$, we extracted a sample patch J_s of size $sP \times sQ$ centered at the previous target location. a^c is denoted as the scale incremental factor between feature layers.

Danelljan et al. [6] employed the fast sub-grid detection strategy to efficiently compute pixel-dense convolution responses and demonstrated exceptional results. We follow the same strategy here.

Let $\hat{r}_i := \mathscr{F}\{R_{f_i}(J)\} = \sum_{l=1}^{d} \hat{J}^l \cdot \hat{f}_i^l$ be the DFT of the convolution responses $R_{f_i}(J)$ evaluated at sample patch J. Then, the convolution responses $r_i(u, v)$ at the continuous location $(u, v) \in [0, M) \times [0, N)$ in J are interpolated as

$$r_i(u, v) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \hat{r}_i(m, n) e^{j2\pi(\frac{m}{M}u + \frac{n}{N}v)}; \tag{13}$$

here, j denotes the imaginary unit.

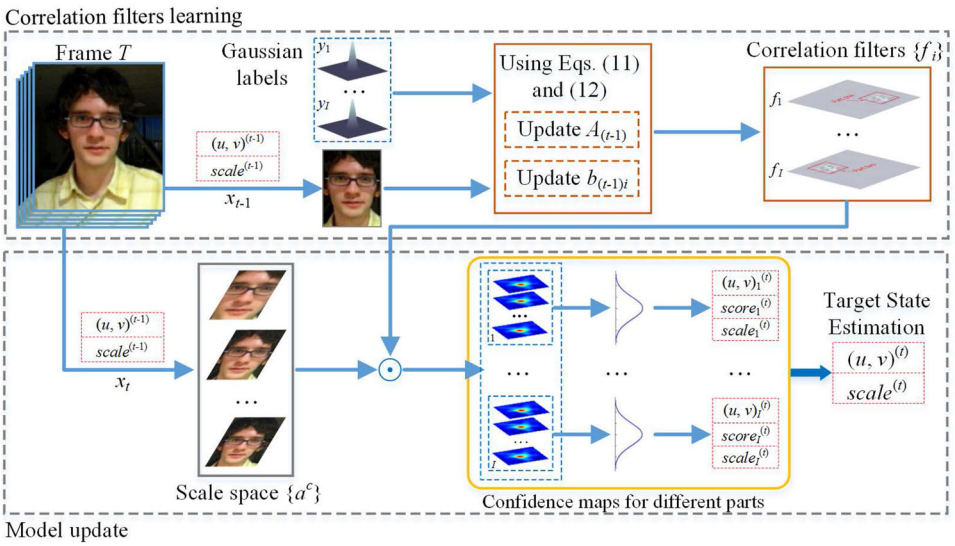


Fig. 4 Flowchart of the proposed tracking algorithm. Our tracking task is decomposed into training the correlation filter and updating the model. Each correlation filter independently tracks the corresponding part and outputs a confidence map, and we track the whole target by combining the confidence maps of individual parts using Eqs. (11) and (12)

To find the sub-grid location $(u^{(*)}, v^{(*)})_i \in \Omega$, which corresponds to the maximal convolution response. We first evaluated the sample patch J_s at all grid locations using $R_{f_i}(J_s) = \mathscr{F}^{-1}\{\sum_{l=1}^{d} \hat{J}_s^{\,l} \cdot \hat{f}_i^{l}\}$; here, $\cdot$ denotes point-wise multiplication, the $\hat{}$ denotes the DFT of a function and $\mathscr{F}^{-1}$ denotes the inverse DFT. Then, Eq. (13) is iteratively maximized using Newton's method by starting at the location $(u^{(0)}, v^{(0)})_i$. The gradient and Hessian in each iteration are computed by analytically differentiating Eq. (13). The sub-grid interpolation procedure is applied for each image patch independently. Finally, we can obtain the maximum convolution response of all the image patches as follows

$$score_i = \max_{s \in S} \max_{(u,v) \in \Omega} r_i^s(u, v). \tag{14}$$

The part level with the highest maximal convolution response is then used to update target location and scale. The target location $(u, v)_i$ and scale $scale_i$ of ith part can be obtained by $\arg\max_{(u,v)} score_i$ and $\arg\max_s score_i$, respectively.

5.3 Target state estimation

We use the fast target estimation method demonstrated in [29] to estimate the state of the target, including position prediction and scale estimation. Given I image parts of sample x_t in the frame t, as discussed above, we can obtain a set $\{score_i^{(t)}, (u, v)_i^{(t)}, scale_i^{(t)}\}_{i \in \{1,2,\ldots,I\}}$.

(a) *Position prediction.* Most of the shifted image patches should move in the same way between two consecutive samples. Therefore, we chose the translation distance as the criterion for object position estimation. First, n parts with large motions between adjacent images are excluded, and a new set $\{score_j^{(t)}, (u, v)_j^{(t)}\}_{j \in \{1,2,\ldots,(I-n)\}}$ is produced. Then, the translation of the target object could be calculated as follows:

$$trans^{(t)} = \frac{\sum_j^{I-n} score_j^{(t)} trans_j^{(t)}}{\sum_j^{I-n} score_j^{(t)}}, \tag{15}$$

where $trans_j^{(t)} = (u, v)_j^{(t)} - (u, v)_j^{(t-1)}$, which shows that more robust tracking parts with higher detection scores have a greater effect on the target position estimation. Finally, the position of the target in frame t is denoted as

$$(u, v)^{(t)} = (u, v)^{(t-1)} + trans^{(t)}. \tag{16}$$

(b) *Scale estimation.* At the detection stage, we determine the scale of each image part according to the maximal convolution response $score_i$. In general, the scale difference between two adjacent image patches is typically smaller than its translation. Therefore, n' parts with the largest scale difference are excluded, and a new set $\{scale_k^{(t)}\}_{k \in \{1,2,\ldots,(I-n')\}}$ is produced. The average of scales of all the remaining parts is taken as the scale of sample x_t, The scale of the target object could be calculated as follows:

$$scale_t = \frac{\sum_k^{I-n'} scale_k^{(t)}}{I - n'}. \tag{17}$$

6 Experiments

To demonstrate the performance of the proposed approach, we extensively evaluate our tracker on three challenging benchmark datasets, namely, OTB2013 [40], OTB2015 [41] and VOT2017 [19]. OTB2013 [40] contains 50 fully annotated sequences that are collected from commonly used tracking sequences, while OTB2015 [41] is an extension of OTB2013 and contains 100 video sequences. VOT2017 [19] is the last version of the visual object tracking toolkit which consists of 60 challenging videos that are automatically selected from a pool of 356 sequences.

For all three datasets, we follow the evaluation protocol established by the original authors and use the same parameter values for all the sequences and all sensitivity analyses to ensure a fair comparison. The experiments are implemented in MATLAB on an AMD Ryzen 5 1600X 3.6 GHz CPU with 16 GB RAM.

6.1 Experimental setup

6.1.1 Comparison scenarios

We conduct four experiments to validate the performance of the proposed approach.

- The first experiment is conducted to analyze the influence of the number of parts on the tracking performance, and we choose the best parameter setup for the following experiments.
- In the second experiment, we perform ablation analyses in accordance with different components of the proposed method and compare the results with the baseline approach.
- In the third experiment, we analyze the effectiveness of removing boundary effects with several challenging sequences; a comparison against the baseline approach is given.
- Finally, in the last experiment, comprehensive analyses are executed and we compare our tracker with state-of-the-art trackers.

Table 1 Exploration of different numbers of individual parts for the tracking performance

I	OTB2013			OTB2015			VOT2017			
	DP	OP	FPS	DP	OP	FPS	Acc.	Robust.	EAO	FPS
2	74.73	72.48	**2.65**	72.75	69.23	**2.59**	0.45	**1.23**	0.11	**3.56**
3	**84.54**	**81.20**	2.25	**80.60**	**75.61**	2.19	**0.47**	1.04	**0.14**	3.53
4	80.28	77.93	1.98	77.69	73.88	1.93	0.45	0.86	0.11	2.15

The best values are highlighted in bold

6.1.2 Performance evaluation methodology

For the two OTB datasets, we evaluate our tracking performance by three metrics, namely, the Distance Precision (DP), Overlap Precision (OP) and Center Location Error (CLE). The DP is computed as the relative number of frames in the sequence in which the CLE is smaller than a certain threshold, the OP is defined as the percentage of frames in which the bounding box overlap surpasses a threshold, and the CLE is defined as the average Euclidean distance between the center locations of the tracked targets and the manually labeled ground truth [40]. We present the results using the average OP, DP and CLE over all 50 and 100 sequences. In addition, following the standard evaluation strategy, we provide precision and success plots of the One-Pass Evaluation (OPE), and we use the Area Under the Curve (AUC) of each plot to rank the trackers.

The tracking performance for the VOT dataset is evaluated in terms of the accuracy (overlap with the ground truth), robustness (failure rate) and Expected Average Overlap (EAO). Furthermore, we provide our tracking speed on all three datasets.

6.1.3 Implementation details

(a) *Spatial regularization parameter.* In the proposed algorithm, the weight function ω is constructed by starting from a quadratic function $\omega(m, n) = \mu + \eta[(m/\zeta P)^2 + (n/\zeta Q)^2]$ with the minimum located at the sample center. $P \times Q$ denotes the target size (see Sect. 5.2), while μ and η are parameters. The minimum value of ω is set to μ=0.1, and the impact of the regularizer is set to η=3. In contrast to similar work [6], we introduce a stronger constraint ζ to penalize the filter coefficients, making the response maps smaller. Here, we set ζ=0.3. Furthermore, we simply remove all DFT coefficients smaller than a certain threshold (in practice, the threshold is set to 0.05) to ensure a sparse spectrum $\hat{\omega}$, and all parameters for the baseline trackers are set to the same values presented by the original authors.

(b) *Feature representation.* Similar to recent DCF-based trackers [14,22], we use HOG features for image representation with a cell size of 4×4 pixels. The samples are presented by a square $S \times \beta$ grid of cells, where S represents the target area and β (set as 4×4) represents the ratio of the search area to the target area. Finally, to alleviate boundary effects, the samples are multiplied by a Hann window [14].

(c) *Label function.* At the training stage, each training sample is assigned a label. In the proposed approach, a standard regression response matrix ($a.k.a$ label function) is set to a Gaussian function, the peak value of which is placed at the target center location [3], and the matrix is cyclically shifted to construct different response maps for each shifted image patch. The step size of the cyclic shift is 1/4 the size of the sample area. Moreover, the number of response maps is set to 3, and the number of excluded samples n is set as 1.

(d) *Other parameters.* The learning rate γ and the number of Gauss–Seidel iterations N_{GS} are set to the same values as in the SRDCF [6] ($\gamma = 0.025$, $N_{\mathrm{GS}} = 4$).

6.2 Exploration study

In this section, we focus on an investigation of good practices for obtaining better tracking results. Because we propose a novel object partition approach in Sect. 4.2, the number of individual parts is the most critical parameter influencing the tracking performance. Therefore, we evaluate its effects when its value varies as $I = 2, 3, 4$. Table 1 reports the results on the three challenging datasets.

As shown in Table 1, the best tracking results are achieved when the number of individual parts is set to $I = 3$. Therefore, we choose $I = 3$ as good practice in the following experiments.

6.3 Ablation analyses

In this experiment, ablation analyses are performed to illustrate the effectiveness of the proposed components. To verify the contribution from each component in our algorithm, we implement and evaluate three variations in our approach. At first, the basic SRDCF [6] is implemented so that none of the proposed components are utilized. Then, to verify the superiority of the proposed object partition strategy, we apply it to the baseline and combine different target responses with two casually used merging algorithms, namely, max merging [16] and average merging [7]. Finally, our part-based track-

Table 2 Ablation analysis of the proposed method with the baseline tracker

	OTB2013		OTB2015		VOT2017			
	DP	OP	DP	OP	Accu.	Robust.	EAO	FPS
Basic SRDCF [6]	83.80	78.10	78.80	73.10	0.46	0.96	0.13	2.13
Part + Max [16]	83.80	78.70	79.60	74.20	0.45	1.02	0.13	3.50
Part + Average [7]	84.24	80.27	**80.71**	75.48	0.45	1.02	0.13	3.43
Part + TSE (OUR)	**84.54**	**81.20**	80.60	**75.61**	**0.47**	**1.04**	**0.14**	**3.53**

The best values are highlighted in bold

ing strategy is employed in combination with the adapted Target State Estimation (TSE) approach.

Table 2 shows the results of the analyses on OTB2013 [40], OTB2015 [41] and VOT2017 [19]. Evidently, all of the variations are helpful for improving the tracking performance. Specifically, the TSE algorithm gains the best performance among almost all evaluation criteria, thereby demonstrating the effectiveness of our approach.

6.4 Boundary effects analyses

The boundary effect problem is represented by the case in which the correlation filter learns an inaccurate representation of the image content because the training patches contain periodic repetitions. When either partial occlusion or plane rotation is present, the characteristics of the object will become unclear. As a result, the filter usually contains a large amount of background information about the nontarget area; then, once the target produces a response that is lower than the response generated by the nontarget area, the tracker will drift.

In this section, to validate whether our approach can effectively mitigate the boundary effect problem, we randomly select several challenging scenarios with small tracking targets for comparison purposes. As shown in Fig. 5, six video sequences with various challenges, such as occlusion and scale variations, are selected.

As shown in Fig. 5a, the tracking target first exhibits partial occlusion or slight rotation and then gradually appears in the field of view; it is obvious that the SRDCF tracker loses the target, indicating that the tracker is contaminated by the nontarget area resulting from the boundary effect problem. As shown in Fig. 5b, the tracking target first appears to be completely occluded. In this case, the object is tracked by the SRDCF tracker; however, when the target subsequently reappears, the filter fails to give the correct response and eventually loses the target. In addition, as shown in Fig. 5c, the target exhibits more complex variations, including plane rotation, nonrigid deformation and scale variations. In these cases, the SRDCF tracks only part of the target. Surprisingly, the tracking results in Fig. 5 demonstrate that the proposed approach can overcome all of the above-mentioned challenges and successfully track the target throughout the process. This shows that the tracker proposed in this paper can effectively alleviate the boundary effect problem during the tracking process.

6.5 Comparisons with state-of-the-arts

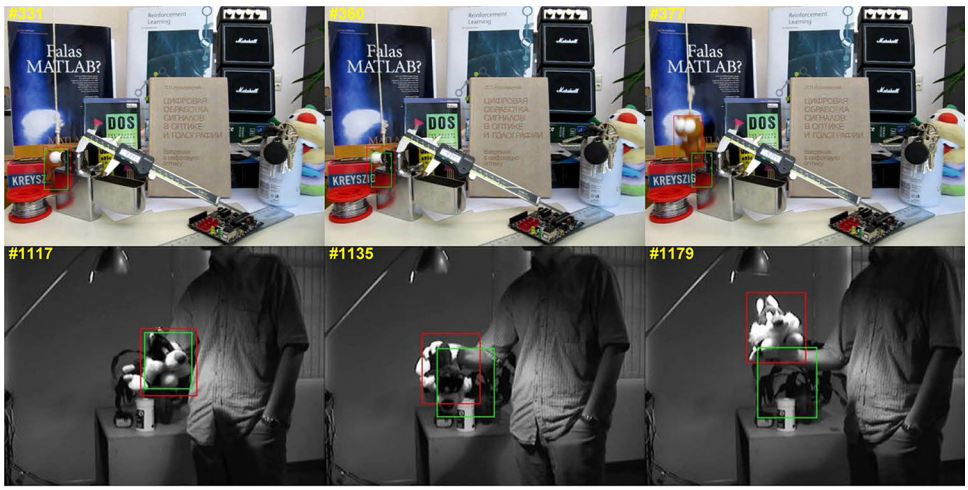
6.5.1 Quantitative results on OTB2013 and OTB2015

We quantitatively evaluate the proposed approach on the OTB datasets with comparisons against 32 state-of-the-art trackers, including the 29 trackers in [40], among which are SCM [53], TLD [17], CXT [32], STRUCK [12], KCF [14], TGPR [9], VTD [20], SRDCF [6] and LSK [26], among others. In addition, CNN-based trackers [36,38,50] have recently been proposed, and they have demonstrated great tracking performance; therefore, we also compare the proposed approach with the best performing CNN-based trackers, namely, CFNet [36] and DCFNet [38].

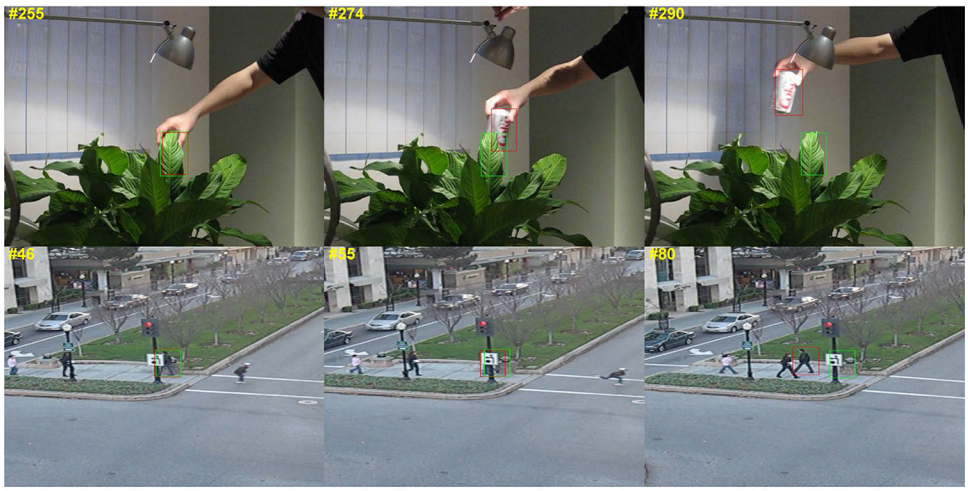
The OPE results over all 50 and 100 sequences for the top 10 ranked results are presented in Figs. 6 and 7 using the DP and overlap success rates, respectively, and Table 3 reports the results using the average OP, DP and CLE. For a clearer comparison, we provide an attribute-based evaluation of the 100 sequences in Fig. 8.

Table 3 shows that the proposed algorithm performs favorably against numerous state-of-the-art methods (including correlation filters [6,14,17], part-based trackers [53] and CNN-based trackers [36,38]). Note that over the entire 100 sequences, Benchmark II is more challenging, and all of the compared trackers perform worse on Benchmark I. The proposed method achieves the best results from among the state-of-the-art trackers with average OP values of 81.2% (I) and 75.6% (II) and average DP values of 84.5% (I) and 80.6% (II). Furthermore, the proposed algorithm performs with lower CLE values of 26.3 pixels (I) and 29.2 pixels (II) than the second best results of the SRDCF tracker of 8.9 pixels and 9.6 pixels, respectively.

As shown in Figs. 6 and 7, with regard to both the overlap plot and the precision plot, the proposed approach achieves the best performance, thereby outperforming the SRDCF, and it is far more effective than correlation filters, part-based trackers and CNN-based trackers. In summary, the proposed



(a) Partial occlusion or plane rotation (*Lemming* && *Sylvester*)



(b) Complete occlusion (*Coke* && *Human3*)

(c) Complex variations (*Freeman3* && *Crowds*)

OUR SRDCF

Fig. 5 Boundary effect analyses on 6 challenging sequences randomly selected from the 100-sequence benchmark [41] for the proposed PSRDCF tracker compared with the SRDCF tracker. The image sequences pose challenging situations such as **a** partial occlusion or plane rotation, **b** complete occlusion and **c** complex variations

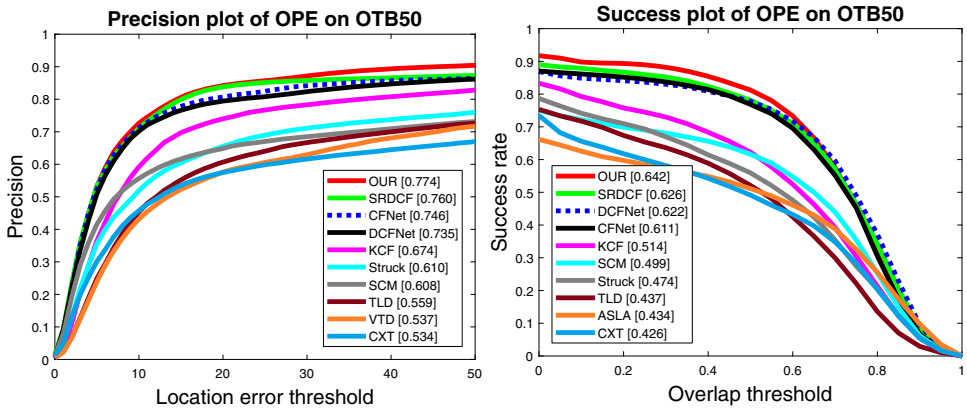


Fig. 6 Precision and success plots over all 50 sequences (OTB2013 [40]) using an OPE. The legend contains the AUC score for each tracker. The proposed method performs favorably against the state-of-the-art trackers

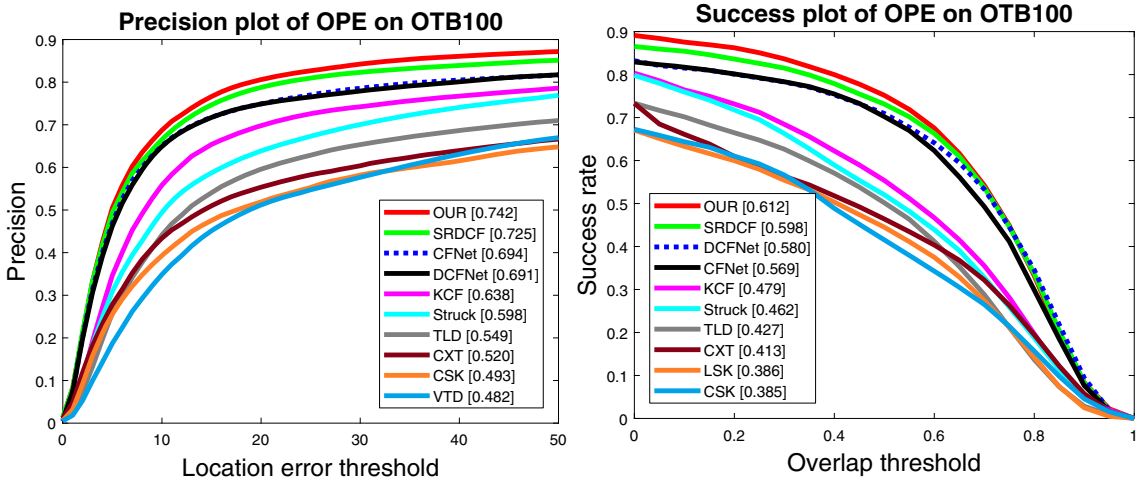


Fig. 7 Precision and success plots over all 100 sequences (OTB2015 [41]) using an OPE. The legend contains the AUC score for each tracker. The proposed method performs favorably against the state-of-the-art trackers

Table 3 Comparison with state-of-the-art trackers on the OTB2013 (I) [40] and OTB2015 (II) [41] benchmark sequences

		OUR	SRDCF [6]	CFNet [36]	DCFNet [38]	SCM [53]	Struck [12]	TLD [17]	CXT [32]	VTD [20]	KCF [14]
OP	I	**81.2**	**78.1**	76.8	**77.9**	61.6	55.9	52.1	49.2	49.3	62.3
	II	**75.6**	**73.1**	70.2	**71.0**	51.6	50.2	50.2	46.4	40.5	55.3
DP	I	**84.5**	**83.8**	**80.7**	79.5	64.9	65.6	60.8	57.5	57.6	74.0
	II	**80.6**	**78.8**	**74.9**	74.9	57.2	63.9	59.6	55.4	51.2	69.8
CLE	I	**26.3**	**35.2**	35.5	**31.3**	54.1	50.6	48.1	68.4	47.4	35.5
	II	**29.2**	**38.8**	**40.0**	40.5	61.6	47.2	60.3	67.7	62.0	44.8
FPS	I	2.0	7.8	-	41.2	0.37	10.0	21.7	-	-	-
	II	2.6	6.0	-	41.2	0.3	9.9	23.4	13.6	-	-

The proposed approach performs favorably against the existing methods with regard to the OP (%) at an overlap threshold of 0.5, DP (%) at a threshold of 20 pixels and CLE (in pixels). The top 3 ranked values are highlighted in bold and in the colors red, green and blue, respectively

algorithm retains the advantages of the SRDCF and can obtain improved results.

We further analyze the proposed approach under different video attributes (e.g., illumination variations, deformation and scale variations) annotated in the 100 sequences [41] (Benchmark I) that are expected to benefit the most from the PSRDCF framework. Figure 8 shows the success plot of OPE for nine main video attributes. Compared with the state-of-the-art trackers, our tracker achieves the best tracking performance except for occlusion, scale variation and both in-plane and out-of-plane rotations; in these cases, CNN-based trackers perform better than our proposed tracker. The reason for this is that CNN-based trackers utilize deep networks that are pre-trained on large-scale datasets as a feature extractor, which is more advantageous than training on hand-crafted features. Compared with the SRDCF tracker, the proposed tracker achieves a better performance on all attributes, including background clutter (by 6.2%), out-of-plane rotation (by

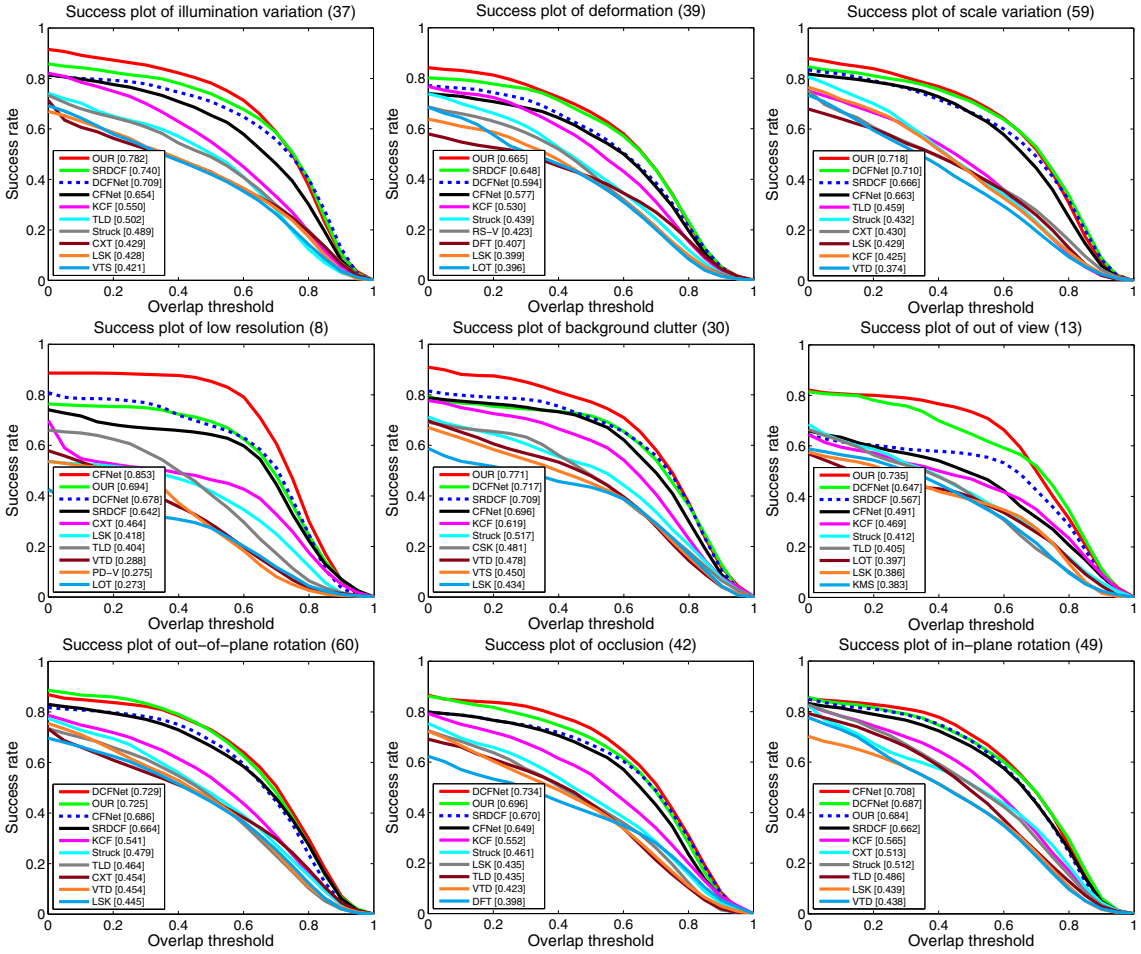


Fig. 8 Top 10 results of the attribute-based comparisons of the proposed tracker against various state-of-the-art methods over 100 sequences [41]. The number in each plot title indicates the number of sequences associated with the particular attribute

6.1%), scale variation (by 5.2%), out of view (16.8%) and low resolution (by 5.2%). In these cases, this algorithm achieves significant improvement over the baseline. In addition, in the other attributes, such as illumination variation (by 4.2%), in-plane rotation (by 2.2%) and occlusion (by 2.6%), our algorithm still performs slightly better. In conclusion, the proposed algorithm improves the visual tracking performance, especially in the case of out-of-view targets, background clutter, scale variations, out-of-plane rotation and low-resolution images.

6.5.2 Qualitative results on OTB

To visualize the superiority of our approach, we provide a qualitative comparison of the proposed method with four other state-of-the-art trackers, including SRDCF [6], KCF [14], Struck [12] and TLD [17], on 10 challenging sequences randomly selected from the OTB datasets [40,41]; these comparisons are shown in Fig. 9. Generally, these trackers perform well, but the existing trackers exhibit a number of issues. TLD and KCF drift in most of the scenes. Furthermore, the SRDCF and Struck trackers cannot handle partial occlusion (*Box*, *Jogging*-2), rotation (*Freeman*) or background clutter (*Skating1*). In addition, Struck does not perform well with partial occlusion (*Jogging*-2). Overall, the proposed PSRDCF tracker performs well at tracking objects on these challenging sequences. In addition, the CLEs are compared frame by frame on the 10 sequences in Fig. 10, which demonstrates that the proposed method performs well against the existing trackers.

6.5.3 Evaluation on VOT2017

The VOT Challenge[1] is a well-known competition among short-term, model-free VOT algorithms, and it has been held several times since 2013. In this section, we compare our tracker with 9 state-of-the-art trackers that participated in the VOT2017 challenge, including the baseline SRDCF tracker

[1] http://www.votchallenge.net/vot2017/.

Fig. 9 Qualitative results on 10 challenging sequences randomly selected from the 100-sequence benchmark [41] for the proposed PSRDCF tracker compared with the top 4 trackers (denoted in different colors and lines). These image sequences pose challenging situations such as **a** partial occlusion, **b** scale variation, **c** illumination variation, **d** rotation and **e** background clutter. The proposed model outperforms the other methods

[6]. Figure 11 illustrates that our PSRDCF tracker ranks first among all 10 trackers according to the EAO criterion. In addition, Table 4 shows that our tracker outperforms all other trackers in the VOT2017 challenge with an EAO of 14.0%, an accuracy of 45% and a failure rate of 86%, thereby achieving relative gains of 0.6%, 1.0% and − 6.0% over the SRDCF [6].

7 Conclusion and future work

In this paper, we propose a novel part-based tracking method that accounts for parts of an object in multiple constrained correlation filters. The utilization of circular shifts of the training samples allows penalized filters to automatically concentrate on target regions with different locations. Thus, the proposed model preserves the ability to address boundary problems existing in traditional DCF-based trackers and also avoids damage to the image content due to this penalization. Experimental results on three large-scale benchmark datasets, namely, OTB2013, OTB2015 and VOT2017, demonstrate that the proposed approach performs better than state-of-the-art methods.

Several limitations exist in our work. First, the proposed PSRDCF model integrates a part-based tracking strategy with

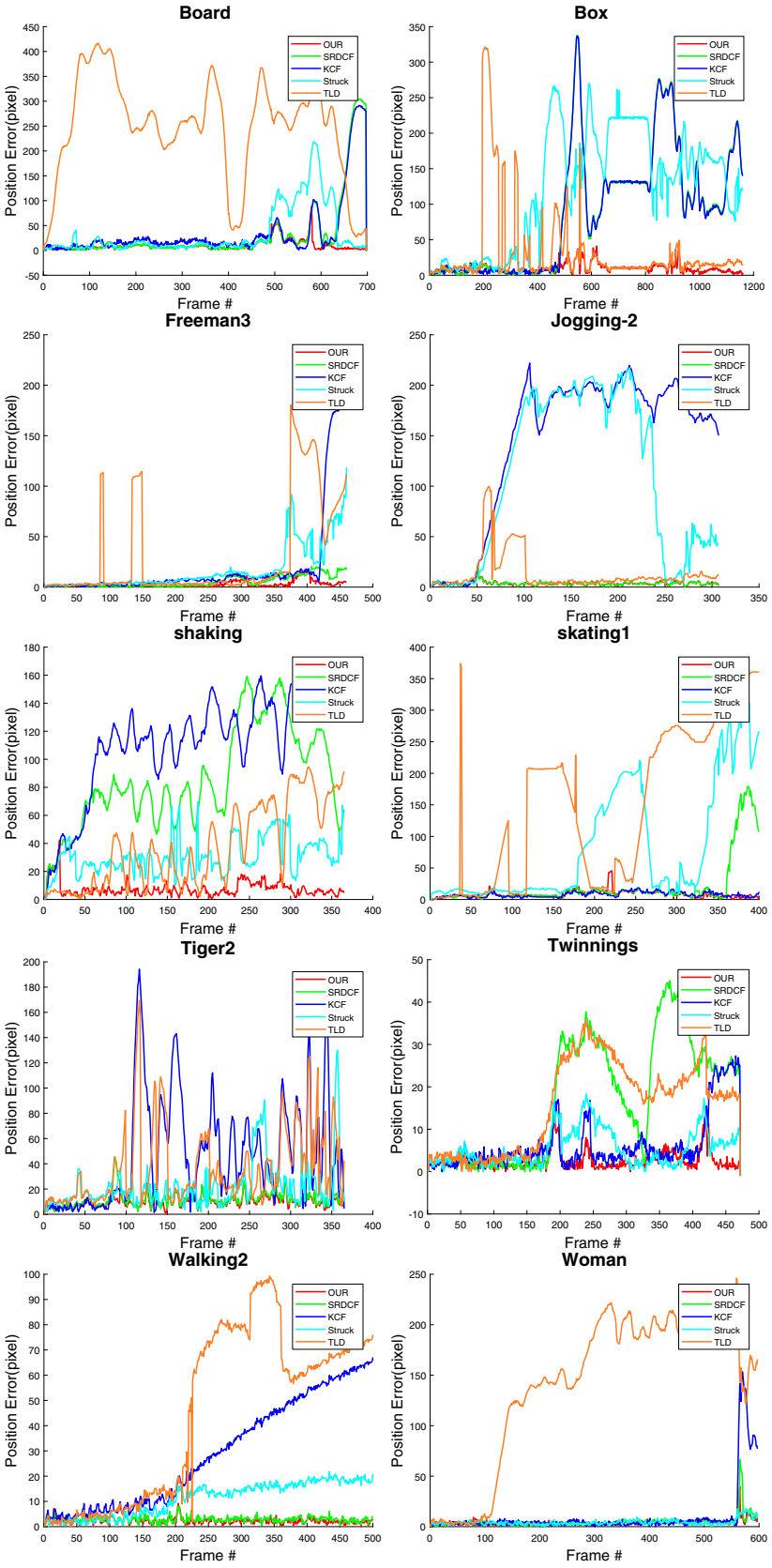


Fig. 10 A frame-by-frame comparison of the CLEs (in pixels) on 10 challenging sequences: Box, Jogging-2, Human5, Walking2, shaking, Tiger2, Freeman3, Twinnings, Board and Skating1. The proposed approach provides promising results compared with the top 4 existing trackers (we encourage the reader to zoom in to the individual panels for better viewing)

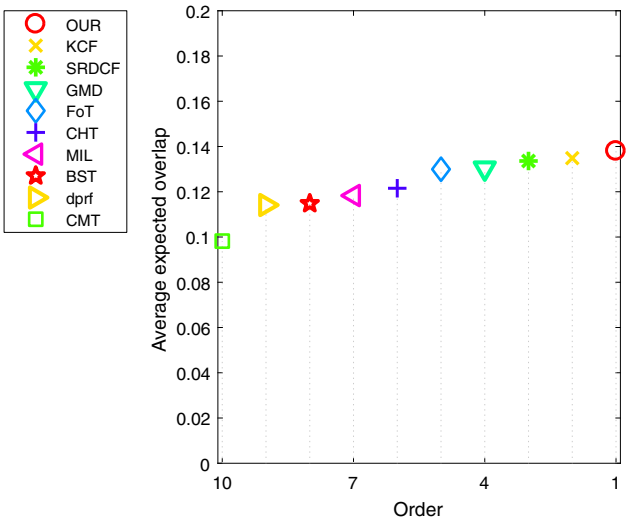


Fig. 11 An illustration of the EAO plot on the VOT2017 challenge

Table 4 Comparisons with the top trackers on VOT2017 [19]

	EAO	Accuracy	Failures	FPS
OUR	**0.140**	**0.45**	**0.86**	3.53
KCF	**0.135**	0.44	**0.80**	**60.01**
SRDCF	**0.134**	**0.46**	0.96	2.13
GMD	0.130	0.45	0.93	4.11
FoT	0.130	0.39	1.14	**163.10**
CHT	0.121	0.41	1.03	**112.97**
MIL	0.118	0.38	1.13	5.99
BST	0.115	0.29	0.90	1.71
dprf	0.115	**0.47**	1.09	0
CMT	0.098	0.32	**0.55**	16.67

The top 3 ranked values are highlighted in bold and the colors are red, green and blue, respectively

correlation filters that partially employs the intrinsic relationship among local parts via spatial constraints to improve the object detection accuracy; however, this strategy neglects to apply structural information during the filter learning stage. In future work, we will concentrate on how to better utilize structural information. Second, we exploited the Gauss–Seidel algorithm to optimize the objective function during the training stage which is far from being real time. In future work, we will employ a better optimization algorithm to remedy this issue. Third, CNNs are commonly used to address computer vision problems and recently demonstrated state-of-the-art performance [11,35]. Future research will consider how to best apply CNN-based features to improve the accuracy of our tracker.

Acknowledgements This study was funded by the National Natural Science Foundation of China (Grant nos. 61702350 and 61472289) and the Open Project Program of State Key Laboratory of Digital Manufacturing Equipment and Technology at HUST (Grant no. DMETKF2017016).

References

1. Bai, B., Zhong, B., Ouyang, G., Wang, P., Liu, X., Chen, Z., Wang, C.: Kernel correlation filters for visual tracking with adaptive fusion of heterogeneous cues. Neurocomputing **286**, 109–120 (2018)
2. Bibi, A., Mueller, M., Ghanem, B.: Target response adaptation for correlation filter tracking. In: European Conference on Computer Vision, pp. 419–433. Springer (2016)
3. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2544–2550. IEEE (2010)
4. Cehovin, L., Kristan, M., Leonardis, A.: Robust visual tracking using an adaptive coupled-layer visual model. IEEE Trans. Pattern Anal. Mach. Intell. **35**(4), 941–953 (2013)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR 2005. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, vol. 1, pp. 886–893. IEEE (2005)
6. Danelljan, M., Hager, G., Shahbaz Khan, F., Felsberg, M,: Learning spatially regularized correlation filters for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4310–4318 (2015)
7. Fan, H., Xiang, J., Xu, J., Liao, H.: Part-based visual tracking via online weighted p–n learning. Sci. World J. **2014**, 402185 (2014)
8. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. **32**(9), 1627–1645 (2010)
9. Gao, J., Ling, H., Hu, W., Xing, J.: Transfer learning based visual tracking with Gaussian processes regression. In: European Conference on Computer Vision, pp. 188–203. Springer (2014)
10. Godec, M., Roth, P.M., Bischof, H.: Hough-based tracking of non-rigid objects. Comput. Vis. Image Underst. **117**(10), 1245–1256 (2013)
11. Guan, H., Cheng, B.: How do deep convolutional features affect tracking performance: an experimental study. Visual Comput. **34**(12), 1701–1711 (2018)
12. 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 (2016)
13. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: Exploiting the circulant structure of tracking-by-detection with kernels. In: European Conference on Computer Vision, pp. 702–715. Springer (2012)
14. 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)
15. Hu, X., Yang, Y.: Faster spatially regularized correlation filters for visual tracking. arXiv preprint. arXiv:1706.00140 (2017)
16. Hwang, J.P., Baek, J., Choi, B., Kim, E.: A novel part-based approach to mean-shift algorithm for visual tracking. Int. J. Control Autom. Syst. **13**(2), 443–453 (2015)
17. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking–learning–detection. IEEE Trans. Pattern Anal. Mach. Intell. **34**(7), 1409–1422 (2012)
18. Kiani Galoogahi, H., Sim, T., Lucey, S.: Correlation filters with limited boundaries. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4630–4638 (2015)
19. Kristan, M., Eldesokey, A., Xing, Y., Fan, Y., Zhu, Z., Zhang, Z., He, Z., Fernandez, G., Garciamartin, A., Muhic, A.: The visual object tracking VOT2017 challenge results. In: IEEE International Conference on Computer Vision Workshop, pp. 1949–1972 (2017)
20. Kwon, J., Lee, K.M.: Visual tracking decomposition. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1269–1276. IEEE (2010)

21. Kwon, J., Lee, K.M.: Highly nonrigid object tracking via patch-based dynamic appearance modeling. IEEE Trans. Pattern Anal. Mach. Intell. **35**(10), 2427–2441 (2013)
22. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration. In: ECCV Workshops, no. 2, pp. 254–265 (2014)
23. Li, Y., Zhu, J., Hoi, S.C.H.: Reliable patch trackers: robust visual tracking by exploiting reliable patches. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 353–361 (2015)
24. Li, Z., He, S., Hashem, M.: Robust object tracking via multi-feature adaptive fusion based on stability: contrast analysis. Visual Comput. **31**(10), 1319–1337 (2015)
25. Li, Z., Xiaoping, Y., Li, P., Hashem, M.: Moving object tracking based on multi-independent features distribution fields with comprehensive spatial feature similarity. Visual Comput. **31**(12), 1633–1651 (2015)
26. Liu, B., Huang, J., Kulikowski, C., Yang, L.: Robust visual tracking using local sparse appearance model and k-selection. IEEE Trans. Pattern Anal. Mach. Intell. **35**(12), 2968–2981 (2013)
27. Liu, S., Zhang, T., Cao, X., Xu, C.: Structural correlation filter for robust visual tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4312–4320 (2016)
28. Liu, T., Wang, G., Yang, Q.: Real-time part-based visual tracking via adaptive correlation filters. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4902–4912 (2015)
29. Ma, C., Yang, X., Zhang, C., Yang, M.-H.: Long-term correlation tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5388–5396 (2015)
30. Matthews, L., Ishikawa, T., Baker, S.: The template update problem. IEEE Trans. Pattern Anal. Mach. Intell. **26**(6), 810–815 (2004)
31. Mbelwa, J.T., Zhao, Q., Lu, Y., Liu, H., Wang, F., Mbise, M.: Objectness-based smoothing stochastic sampling and coherence approximate nearest neighbor for visual tracking. Visual Comput. 1–14 (2018). https://doi.org/10.1007/s00371-018-1470-5
32. Medioni, G.: Context tracker: exploring supporters and distracters in unconstrained environments. In: Computer Vision and Pattern Recognition, pp. 1177–1184 (2011)
33. Quan, W., Chen, J.X., Yu, N.: Robust object tracking using enhanced random ferns. Visual Comput. **30**(4), 351–358 (2014)
34. Quan, W., Jiang, Y., Zhang, J., Chen, J.X.: Robust object tracking with active context learning. Visual Comput. **31**(10), 1307–1318 (2015)
35. Zhigang, T., Xie, W., Qin, Q., Poppe, R., Veltkamp, R.C., Li, B., Yuan, J.: Multi-stream CNN: learning representations based on human-related regions for action recognition. Pattern Recognit. **79**, 32–43 (2018)
36. Valmadre, J., Bertinetto, L., Henriques, J., Vedaldi, A., Torr, P.H.S.: End-to-end representation learning for correlation filter based tracking. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5000–5008. IEEE (2017)
37. Van De Weijer, J., Schmid, C., Verbeek, J., Larlus, D.: Learning color names for real-world applications. IEEE Trans. Image Process. **18**(7), 1512–1523 (2009)
38. Wang, Q., Gao, J., Xing, J., Zhang, M., Hu, W.: Dcfnet: discriminant correlation filters network for visual tracking. *arXiv preprint*arXiv:1704.04057 (2017)
39. Wang, Z., Yoon, S., Xie, S.J., Lu, Y., Park, D.S.: Visual tracking with semi-supervised online weighted multiple instance learning. Visual Comput. **32**(3), 307–320 (2016)
40. Wu, Y., Lim, J., Yang, M.-H.: Online object tracking: a benchmark. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2411–2418 (2013)
41. Yi, W., Lim, J., Yang, M.-H.: Object tracking benchmark. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1834–1848 (2015)
42. Yunxia, W., Jia, N., Sun, J.: Real-time multi-scale tracking based on compressive sensing. Visual Comput. **31**(4), 471–484 (2015)
43. Yang, M., Yuan, J., Wu, Y.: Spatial selection for attentional visual tracking. In: CVPR'07. IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8. IEEE (2007)
44. Zhan, J., Zhuo, S., Hefeng, W., Luo, X.: Robust tracking via discriminative sparse feature selection. Visual Comput. **31**(5), 575–588 (2015)
45. Zhang, H., Liu, G.: Coupled-layer based visual tracking via adaptive kernelized correlation filters. Visual Comput. **34**(1), 41–54 (2018)
46. Zhang, T., Ghanem, B., Liu, S., Ahuja, N.: Robust visual tracking via multi-task sparse learning. In: Computer Vision and Pattern Recognition, pp. 2042–2049 (2012)
47. Zhang, T., Ghanem, B., Liu, S., Ahuja, N.: Low-Rank Sparse Learning for Robust Visual Tracking. Springer, Berlin (2012)
48. Zhang, T., Jia, K., Xu, C., Ma, Y., Ahuja, N.: Partial occlusion handling for visual tracking via robust part matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1258–1265 (2014)
49. Zhao, L., Zhao, Q., Liu, H., Lv, P., Dongbing, G.: Structural sparse representation-based semi-supervised learning and edge detection proposal for visual tracking. Visual Comput. **33**(9), 1169–1184 (2017)
50. Zhong, B., Zhang, J., Wang, P., Du, J., Chen, D.: Jointly feature learning and selection for robust tracking via a gating mechanism. PLOS ONE **11**(8), e0161808 (2016)
51. Zhong, B., Chen, Y., Shen, Y., Chen, Y., Cui, Z., Ji, R., Yuan, X., Chen, D., Chen, W.: Robust tracking via patch-based appearance model and local background estimation. Neurocomputing **123**, 344–353 (2014)
52. Zhong, B., Yao, H., Chen, S., Ji, R., Chin, T.J., Wang, H.: Visual tracking via weakly supervised learning from multiple imperfect oracles. Pattern Recognit. **47**(3), 1395–1410 (2014)
53. Zhong, W., Lu, H., Yang, M.-H.: Robust object tracking via sparsity-based collaborative model. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1838–1845. IEEE (2012)

Dejun Zhang received the Ph.D. degree from the department of computer school, Wuhan University, China, in 2015. He is currently a lecturer with the Faculty of Information Engineering, China University of Geosciences, Wuhan, China. Since 2015, he has been serving as a senior member of the China Society for Industrial and Applied Mathematics (CSIAM) and a committee member of the geometric design & computing of CSIAM. His research areas include computer graphics, computer vision, image and video processing. He has conducted a considerable amount of research in digital geometric processing and computational photography, and he has published more than 30 papers in journals and conferences.

Zhao Zhang was born in Chongqing, China, 1998. He received the BS degree from the Sichuan Agricultural University in 2018. He is currently a graduate student in the College of Computer Science at Sichuan University, China. He is a member of the China Computer Federation (CCF). He has been engaged in scientific research for 3 years and has authored two articles in referred journals and proceedings in the areas of computer vision.

Lu Zou was born in Chengdu, China, 1996. She received the BS degree from the Sichuan Agricultural University in 2018. She is currently a graduate student in the School of Data Science at University of Science and Technology of China, China. She is a member of the China Computer Federation (CCF). She is a member of the China Society for Industrial and Applied Mathematics (CSIAM). She has been engaged in scientific research for 3 years and has authored nine articles in referred journals and proceedings in the areas of computer graphics, computer vision and bioinformatics.

Zhuyang Xie was born in Mianyang, China, 1995. He received the BS degree from the Sichuan Agricultural University in 2018. He is currently a graduate student in the College of Computer Science at Southwest Jiaotong University, China. He is a member of the China Society for Industrial and Applied Mathematics (CSIAM). He has been engaged in scientific research for 3 years and has authored two articles in referred journals and proceedings in the areas of computer vision.

Fazhi He received Ph.D. degree from Wuhan University of Technology. He was post-doctor researcher in The State Key Laboratory of CAD & CG at Zhejiang University, a visiting researcher in Korea Advanced Institute of Science & Technology and a visiting faculty member in the University of North Carolina at Chapel Hill. He became an assistant professor at Wuhan University in 2001 and became a professor in 2006. Now, he is a professor in State Key Laboratory of Software Engineering, School of Computer, Wuhan University. He was a program committee for the Asian Conference on Design and Digital Engineering 2017 (ACDDE 2017) and a program committee for the 2018 IEEE 22st International Conference on Computer Supported Cooperative Work in Design (CSCWD 2018). He has been serving as a senior member of the China Society for Industrial and Applied Mathematics (CSIAM) and a committee member of the geometric design & computing of CSIAM. Currently, he is a member of the editorial board for the Journal of Computer-Aided Design & Computer Graphics. His research interests are computer graphics, computer-aided design, natural language processing and computer-supported cooperative work. He has published more than 100 refereed articles in journals and conference proceedings and has received several best paper awards to date.

Yiqi Wu received the BS degree from the Huazhong University of Science and Technology in 2007 and the MS degree from China University of Geosciences in 2011. He received the Ph.D. degree from the department of computer school, Wuhan University, China, in 2017. He is currently a lecturer in the College of Computer Science, China University of Geosciences, China. His research interests include computer-aided design, computer-supported cooperative work, cloud-based design and manufacturing and computer graphics.

Zhigang Tu started his Master Degree in image processing at the School of Electronic Information, Wuhan University, China, 2008. In 2015, he received the Ph.D. degree in Computer Science from Utrecht University, Netherlands. From 2015 to 2016, he was a postdoctoral researcher at Arizona State University, USA. Then from 2016 to 2018, he was a research fellow at the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He is currently a professor at the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote sensing, Wuhan University. His research interests include computer vision, image processing, video analytics and machine learning, especially for motion estimation, object segmentation, object tracking, action recognition and localization, and anomaly detection.