Fast Learning of Spatially Regularized and Content Aware Correlation Filter for Visual Tracking

Ruize Han, Wei Feng, *Member, IEEE*, and Song Wang, *Senior Member, IEEE*

***Abstract*— With a good balance between accuracy and speed, correlation filter (CF) has become a popular and dominant visual object tracking scheme. It implicitly extends the training samples by circular shifts of a given target patch, which serve as negative samples for fast online learning of the filters. Since all these shifted patches are not real negative samples of the target, CF tracking scheme suffers from the annoying boundary effects that can greatly harm the tracking performance, especially under challenging situations, like occlusion and fast temporal variation. Spatial regularization is known as a potent way to alleviate such boundary effects, but with the cost of highly increased time complexity, caused by complex optimization imported by spatial regularization. In this paper, we propose a new fast learning approach to content-aware spatial regularization, namely weighted sample based CF tracking (WSCF). In WSCF, specifically, we present a simple yet effective energy function that implicitly weighs different training samples by spatial deviations. With the energy function, the learning of correlation filters is composed of two subproblems with closed-form solution and can be efficiently solved in an alternate way. We further develop a content-aware updating strategy to dynamically refine the weight distribution to well adapt to the temporal variations of the target and background. Finally, the proposed WSCF is used to enhance two state-of-the-art CF trackers to significantly boost their tracking accuracy, with little sacrifice on the tracking speed. Extensive experiments on five benchmarks validate the effectiveness of the proposed approach.**

***Index Terms*— Object tracking, correlation filter, boundary effects, fast spatial regularization, temporal variations.**

I. Introduction

VISUAL object tracking is a classical problem and plays an important role in computer vision, with many applications in practice [1]–[3]. In visual object tracking, fast learning an effective appearance model of the target is crucial for tracking accuracy and robustness [4], [5]. Many machine learning methods, e.g., support vector machines (SVM) [6], [7], subspace learning [8], online multi-instance boosting [9], sparse and compressive reconstruction [5], [10], correlation filter (CF) [11], [12], and convolutional neural network (CNN) [13], [14], have been developed to this end. Among all of them, correlation filter (CF) based trackers have shown great advantages due to the balance of accuracy and speed [15]. Specifically, the popularity of CF tracking scheme mainly comes from two aspects: 1) the circular convolution operation in CF effectively implicitly generates plenty of simulating negative samples that enables fast learning of more discriminative correlation filters; 2) the Fast Fourier Transform (FFT) significantly accelerates the computation and highly improves the efficiency of algorithm. As a result, some correlation filter (CF) based trackers [11] can run over 600 fps with a single CPU and generate quite promising tracking accuracy as well.

Despite the above advantages, the circularly shifted patches contain unwanted circular *boundary effects* [16] and are not real negative samples of the target. Such boundary effects can severely harm the performance of the CF tracking scheme, especially under challenging situations, such as occlusion and fast temporal variation. Recently, two categories of methods are proposed to alleviate the annoying boundary effects for CF tracking scheme. The first category includes SRDCF [17] and CSR-DCF [18]. They extend the region of training patch and introduce a spatial regularization (SR) map to adapt the filter to learn from the target region while suppress the biased background region. Theoretically, such methods are equivalent to assigning different weights to the training samples, i.e., the samples generated by strict circular shifting will get the lower weights. Spatial regularization [17], [18] has been shown to be very effective to improve both the accuracy and robustness of CF based trackers. However, it imports a spatial regularization term that leads to complex optimization to the learning of filters, thus harms the efficiency foundation of CF tracking scheme and highly increases the time complexity. The second category of CF based trackers tackling boundary effects includes CFLB [16] and BACF [19], which generate more realistic negative training samples extracted directly from the background. The limitation of such methods is that all the positive and negative training samples are given equal weights in the learning of filters, which is not good enough for

Manuscript received October 20, 2019; revised April 20, 2020; accepted May 16, 2020. Date of publication June 5, 2020; date of current version July 8, 2020. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant U1803264, Grant 61671325, and Grant 61672376. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Husrev T. Sencar. *(Corresponding author: Wei Feng.)*

Ruize Han and Wei Feng are with the School of Computer Science and Technology, College of Intelligence and Computing, Tianjin University, Tianjin 300350, China, also with the Key Research Center for Surface Monitoring and Analysis of Cultural Relics (SMARC), State Administration of Cultural Heritage, Tianjin 300350, China, and also with the Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin University, Tianjin 300350, China (e-mail: wfeng@ieee.org).

Song Wang is with the School of Computer Science and Technology, College of Intelligence and Computing, Tianjin University, Tianjin 300350, China, also with the Key Research Center for Surface Monitoring and Analysis of Cultural Relics (SMARC), State Administration of Cultural Heritage, Tianjin 300350, China, and also with the Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208 USA.

Digital Object Identifier 10.1109/TIP.2020.2998978

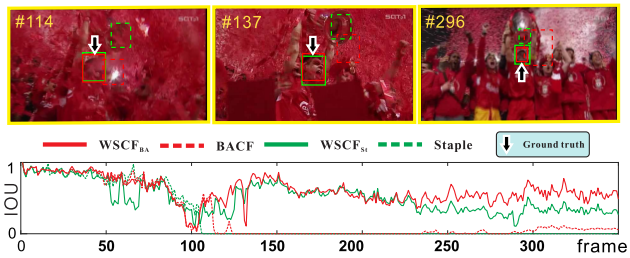


Fig. 1. Top: comparative tracking results on a video sequence using the baseline trackers Staple and BACF, and their spatially-regularized version WSCF$_{St}$ and WSCF$_{BA}$ enhanced by the proposed WSCF approach. Bottom: intersection-over-union (IoU) curve between the predicted and ground truth bounding boxes of the target tracked by the four trackers..

the tracking accuracy and cannot fully suppress the boundary effects either.

In this paper, we propose a new fast content-aware spatial-temporal regularization approach to the CF tracking scheme, namely weighted sample based CF tracking (WSCF). In WSCF, we assign different content-related weights to the implicit circular shifting generated training samples, while still preserving the efficiency of filter learning. Specifically, we propose a simple yet effective energy function, with the energy function, we can fast learn the correlation filters in each frame by a closed form solution. We further formulate the dynamic updating of such content-aware weight map as a constrained quadratic optimization problem. This enable our approach to well adapt to the temporal variations of the target and background. Since our WSCF provides a general way to alleviate boundary effects for the whole CF tracking scheme, it is applicable to many CF based trackers. In our experiments, we equip two state-of-the-art CF trackers with the capability of spatial regularization and significantly boost their tracking accuracy, without highly decreasing their tracking speed. Extensive experiments on five benchmarks validate the effectiveness of the proposed WSCF approach. As shown in Fig. 1, the enhanced WSCF$_{St}$ and WSCF$_{BA}$ trackers by our approach clearly outperforms their respective baselines, i.e., Staple and BACF. Note, compared to the spatial regularization based trackers, e.g., SRDCF [17] and CSR-DCF [18], WSCF$_{St}$ and WSCF$_{BA}$ also achieve better accuracy and higher speed. Compared to the real-sample based tracker CFLB [16], the proposed approach outperforms with a large margin. Moreover, the proposed WSCF approach can enhance BACF [19] to further improve its tracking accuracy.

Our major contributions are three-fold.

- A simple yet effective bound term is introduced into the CF formulation to alleviate the boundary effects and the new energy function can be optimized in closed form.
- A dynamic content-aware updating model is proposed to adjust the weight distribution of the samples to well adapt to the temporal variations, which can be solved by a fast ALM based algorithm.
- Experimental results on five benchmarks, i.e., OTB-2013, OTB-2015, VOT-2018, TC-128 and LaSOT validate that the proposed approach obviously improves the tracking accuracy of CF based trackers, while maintaining their real-time running speed.

The rest of this paper is organized as follows. Section II summarizes the related works. Our approach is elaborated in Section III. Section IV extensively evaluates and compares the proposed WSCF approach and state-of-the-art competitors. Finally, we conclude in Section V.

II. Related Work

A. Correlation Filter (CF) Based Tracking

Bolme *et al.* proposed the tracker MOSSE [11], which inaugurated the CF based tracking framework. CF framework shows two main advantage in handling tracking problem, i.e., the circular convolution operation extends the training samples sufficiently and the FFT operation decreases the computational complexity significantly. Due to the good trade off between accuracy and speed, CF tracking was developed quickly and has shown continuous performance improvement on benchmarks in recent years [12], [20]–[29]. Two typical strategies have been used to obtain better performance in CF tracking – integrating more effective features and using a more complete approach in filter learning. In the first strategy, multi-channel feature maps are integrated to CF tracking [12], [23]. Henriques *et al.* [12] proposed a CF tracker with multi-channel Histogram of Oriented Gradient (HOG) features, which can improve the tracking accuracy while maintaining a high running speed. Danelljan *et al.* applied multi-dimensional color attributes in DSST [23]. Li and Zhu further proposed SAMF tracker [24] using the feature combination for CF tracking. Recently, deep CNN based features have been applied to CF tracking [25], [27], which further improve the tracking performance but taking more computation time. In the second strategy, many conceptual improvements for filter learning in CF tracking are presented, e.g., non-linear kernelized correlation filter (KCF) used in [12], accurate scale estimation used in DSST [21], and color statistics integration proposed in Staple [20]. DRT [30] proposes a novel CF-based optimization problem to jointly model the discrimination and reliability information. Sun *et al.* [31] integrated ROI (region-of-interest) based pooling method into CF tracking and proposed a novel ROI pooled correlation filter (RPCF) algorithm for robust tracking. Recently, CFNet [32] models the CF tracking into an end-to-end framework by interpreting the CF learner as a differentiable layer in a deep neural network. Wang *et al.* [33] developed a new unsupervised learning method for CF-based deep tracking. Visual object tracking especially CF tracking has many application scenarios [34]. For example, Shao *et al.* employed the velocity feature [35], and two complementary features, i.e., the optical flow and the histogram of oriented gradient [36], to boost the CF tracking in satellite videos. Although these methods made much progress in CF tracking over the initial MOSSE, the boundary effect problem in CF tracking is still unsolved in these trackers.

B. Spatial Regularization for CF Tracking

One issue in CF tracking is the unwanted boundary effects, which are resulted from the unreal training samples generated by circular convolution. Several popular methods have been proposed to address the boundary effects by considering

spatial constraints. In [16] and [19], CF is learned from more real training examples by enlarging the region of sample collection. Galoogahi *et al.* [16] proposed a CF tracker with limited boundary (CFLB) to reduce the boundary effects in CF tracking. In [19], the background-aware correlation filter (BACF) based tracking is developed to learn CF from real negative training samples extracted from the background. SRDCF [17] adopts a spatial regularization (SR) component to penalize CF values. The spatial regularization punishes the unreal training samples by assigning different weights to different samples. Based on SRDCF, Danelljan *et al.* [37] introduced a novel formulation for learning a convolution operator in the continuous spatial domain, which was further enhanced to tackle the problems of computational complexity and over-fitting simultaneously in ECO [38]. Moreover, based on [38], UPDT [39] further unveils the power of deep features for CF based tracking. Similarly, Lukezic *et al.* [18] proposed discriminative correlation filter with channel and spatial reliability (CSR-DCF) using the spatial regularization map to constrain the filter learned from the object region. Besides, CACF [40] allows the explicit incorporation of global context within CF trackers. BSCF [41] introduces spatial constraints in CF by suppressing the background region of the target in filter learning.

C. Spatial-Temporal Regularization for CF Tracking

An important factor in tracking task is to consider the temporal variation in tracking process, e.g., the size, appearance and shape change of the target over time. Several spatial-temporal regularization based methods were proposed to alleviate the incapability of CF tracking models in handling changeable scenes. Most of these methods are based on the spatial regularization proposed in SRDCF [17]. Specifically, a unified formulation for CF based tracking is proposed in [42] to dynamically manage the training set by estimating the quality of the samples. STRCF [43] introduces the temporal regularization to SRDCF to provide a more robust appearance model than the baseline in the case of large appearance variations. Similarly, STRSCF [44] integrates the spatial prior directly to the image and a temporal regularization term as in STRCF to improve the tracking performances. Zhang *et al.* [45] presented a part-based tracking framework by exploiting multiple SRDCFs. FOSR [46] incorporates an object-adaptive spatial regularization model for CF tracking. More recently, CRSRCF [47] integrates temporal content information of the target into the spatial constraint of [17] and DSAR-CF [48] introduces a dynamic saliency-aware spatial constraint into CF tracking. Similarly, ASRCF [49] incorporates an adaptive spatially regularized CF model to optimize the filter while adaptively tuning the spatial regularization weights. In addition, SSRCF [50] proposes a selective spatial regularization based on SRDCF. Besides above trackers extended from SRDCF, Hu *et al.* [51] proposed a spatial-aware temporal aggregation network to construct more efficient features for CF tracking. Guo *et al.* [52] derived a dynamic transformations to handle the variation of both the target and background.

The proposed WSCF is totally different from the above works in that we introduce the weighting constraint bound term into CF formulation to alleviate the boundary effects by assigning different weights to the training samples. Our formulation does not involve the elementwise multiplication operation on the correlation filter, and avoids the computational inefficiency of classical SR. This way, the filter can be solved by the closed-form solution, which almost does not increase the computational complexity.

III. The Proposed Method

A. Background and Motivation

1) CF Tracking and Boundary Effects: There are two primary stages in CF tracking framework, i.e., detection stage and filter learning stage. In the detection stage, an online updated correlation filter $\mathbf{f} \in \mathbb{R}^{N\times 1}$ is applied to detect the target location in a search region frame by frame ($\mathbf{z} \in \mathbb{R}^{N\times 1}$ represents the feature vector extracted from the search region), and response vector $\mathbf{c} \in \mathbb{R}^{N\times 1}$ can be computed by

$$\mathbf{c} = \mathbf{z} \circledast \mathbf{f}, \tag{1}$$

where $\circledast$ denotes circular convolution, and the target location is on the peak of $\mathbf{c}$. In filter learning stage, correlation filter $\mathbf{f}$ can be updated by minimizing

$$\mathrm{E}_{\mathrm{CF}}(\mathbf{f}) = \|\mathbf{x} \circledast \mathbf{f} - \mathbf{y}\|^2 + \lambda\|\mathbf{f}\|^2, \tag{2}$$

where $\mathbf{x}$ denotes the vectorized feature map extracted from the training samples, and $\mathbf{y} \in \mathbb{R}^{N\times 1}$ is the desired output (i.e., the Gaussian-shaped ground truth), $\lambda \geq 0$ is a control factor.[1] Optimizing Eq. (2) w.r.t. $\mathbf{f}$ can be efficiently solved in frequency domain where '$\circledast$' becomes elementwise multiplication, which leads to a super real-time algorithm [11]. As discussed in [16], the circular convolution in Eq. (2) assumes the cyclic shifts of $\mathbf{x}$, which causes the periodic repetitions on boundary positions. The cyclic shifts of base sample generate the unfaithful training samples as shown in Fig. 2 and harm the discriminative power of learned filters. This is called the *boundary effects* that inevitably degrades the CF tracking performance.

2) Spatial Regularization (SR) for Boundary Effects: Spatial regularization (SR) [17] is a classical method to alleviate the boundary effects in CF framework. We then revisit SR based tracker SRDCF [17], which introduces a spatial weight map to penalize filter values outside the object boundaries by the following energy function,

$$\mathrm{E}_{\mathrm{SR}}(\mathbf{f}) = \|\mathbf{x} \circledast \mathbf{f} - \mathbf{y}\|^2 + \lambda\|\tilde{\mathbf{w}} \odot \mathbf{f}\|^2, \tag{3}$$

where $\odot$ denotes the elementwise multiplication operation, and $\tilde{\mathbf{w}}$ is the vectorized spatial variant weight map. When we set $\mathbf{f}' = \tilde{\mathbf{w}} \odot \mathbf{f}$, Eq. (3) can be rewritten as

$$\mathrm{E}'_{\mathrm{SR}}(\mathbf{f}') = \left\| \frac{\mathbf{f}'}{\tilde{\mathbf{w}}} \circledast \mathbf{x} - \mathbf{y} \right\|^2 + \lambda\left\|\mathbf{f}'\right\|^2, \tag{4}$$

[1]For simplicity, what we describe above is for one-channel feature map. In practice, it can be extended to multiple channel feature maps by using multiple feature-extraction algorithms [53].

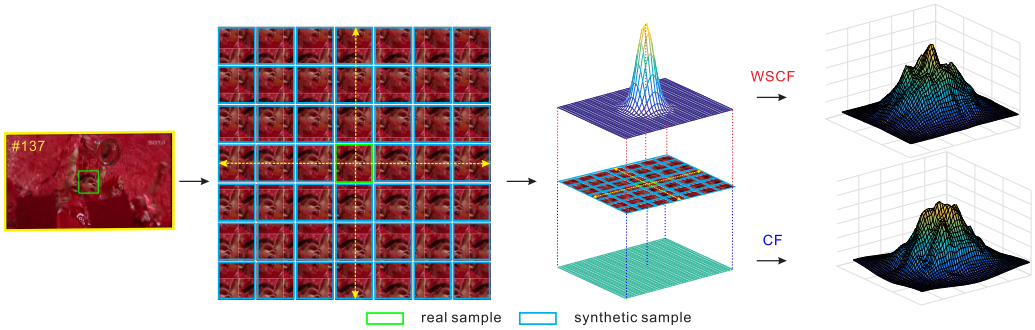


Fig. 2. An example frame from the video *soccer* (left) and the tracking results on this video has been shown in Fig. 1. An illustration of the training samples in CF framework and the weight distribution for different training samples using the proposed WSCF and original CF framework (middle). The response maps generated by these two types of methods (right)..

where $\frac{\mathbf{f}'}{\tilde{\mathbf{w}}} \circledast \mathbf{x}$ can be written as $\mathbf{C}\text{diag}(\frac{1}{\tilde{\mathbf{w}}})\mathbf{f}'$. Each row of $\mathbf{C}$ denotes a vector of circularly shifted $\mathbf{x}$, i.e., a training sample and diag$(\cdot)$ is the diagonalization function for a vector. Then $\mathbf{C}\text{diag}(\frac{1}{\tilde{\mathbf{w}}})\mathbf{f}'$ can be regarded as assigning the weights $\frac{1}{\tilde{\mathbf{w}}}$ to the training samples. As discussed in [48], in the original CF energy function of Eq. (2), the second term can be taken as $\lambda||\mathbf{1} \odot \mathbf{f}||$, which means both the real sample and synthetic samples are of equal importance for filter learning. However, those synthetic samples shifted to be far from the target center cannot represent the real scene. Hence, the synthetic samples may introduce disturbance in filter learning. SRDCF generates $\tilde{\mathbf{w}}$ according to the shifting distance of training samples, i.e., a synthetic sample with large shifting distance will be assigned a small weight. This effectively removes the influence of useless synthetic samples, thus learns much more discriminative filters. From this point, the spatial regularization can be regarded as *assigning different weights to the training samples to deal with the boundary effects*.

3) Limitation of SR: We transform Eq. (3) into Fourier domain and get

$$\mathrm{E}_{\mathrm{SR}}(\hat{\mathbf{f}}) = \left\| \hat{\mathbf{x}} \odot \hat{\mathbf{f}} - \hat{\mathbf{y}} \right\|^2 + \lambda \left\| \hat{\tilde{\mathbf{w}}} \circledast \hat{\mathbf{f}} \right\|^2, \tag{5}$$

where $\hat{\cdot}$ denotes the corresponding variable in Fourier domain. The second terms of Eq. (3) and Eq. (5) are the SR terms. We can see that the SR weight map $\tilde{\mathbf{w}}$ imports circular convolution operation into the second term of Eq. (5). This breaks the simplicity of elementwise multiplication in Fourier domain of the first term in Eq. (5), which is the efficiency foundation of CF framework. As a result, we have to use the Gauss-Seidel (GS) [17] or conjugate gradient (CG) [37], [38] algorithms to iteratively solve the corresponding linear system for learning $\mathbf{f}$, whose complexity is $O(N^3D^3)$ and $O(N^2D)$, respectively, where N and D denote the size and dimension of the feature map $\mathbf{x}$. Therefore, although the SR improves the accuracy of CF trackers, the use of SR term also significantly decelerates the algorithm speed, e.g., SRDCF runs at about 4 fps with HOG features. This way, we aim to construct a new spatial regularization alike method to alleviate the boundary effects without breaking the elementwise multiplication in Fourier domain for learning the filters.

B. Weighted Samples for Correlation Filter Learning

In this paper, we propose a new spatially variant weighted sample based CF model. First, the CF energy function of Eq. (2) can be rewritten as

$$\mathrm{E}_{\mathbf{w}}(\mathbf{f}, \mathbf{y}^*) = \left\| \mathbf{x} \circledast \mathbf{f} - \mathbf{y}^* \right\|^2 + \|\mathbf{f}\|^2 + \lambda_y \left\| \mathbf{w} \odot (\mathbf{y} - \mathbf{y}^*)^2 \right\|_1, \tag{6}$$

where $\mathbf{x}, \mathbf{f}$ and $\mathbf{y}$ have the same meanings as in Eq. (2), $\lambda_y \geq 0$ is a control factor and $\|\cdot\|_1$ denotes the vector L1-norm. The notation $\mathbf{w} \in \mathbb{R}^{N\times 1}$ denotes the vectorized weight distribution map, $\mathbf{y}^* \in \mathbb{R}^{N\times 1}$ represents the computed result of the filter applied to the training samples, i.e., $\mathbf{x} \circledast \mathbf{f}$. In original CF function as Eq. (2), it aims to minimize the difference between the desired output $\mathbf{y}$ and computed result $\mathbf{y}^*$, by treating all the training samples equally. As shown in Fig. 2, we introduce the spatially variant weight distribution $\mathbf{w}$ in Eq. (6) by assigning different weights to the training samples (the samples generated by seriously circular shift will get the lower weights) to alleviate the boundary effects problem of CF as in [17]. But differently we do not implement the elementwise operation of $\mathbf{w}$ with $\mathbf{f}$, which avoids the inefficient computation in [17].

The energy function Eq. (6) involves two variables $\mathbf{f}$ and $\mathbf{y}^*$. With respect to $\mathbf{y}^*$, the gradient of $\mathrm{E}_{\mathbf{w}}$ can be obtained by

$$\frac{\partial \mathrm{E}_{\mathbf{w}}(\mathbf{y}^*)}{\partial \mathbf{y}^*} = 2(\mathbf{f} \circledast \mathbf{x} - \mathbf{y}^*) + 2\lambda_y(\mathbf{w} \odot (\mathbf{y} - \mathbf{y}^*)). \tag{7}$$

By setting $\frac{\partial \mathrm{E}_{\mathbf{w}}(\mathbf{y}^*)}{\partial \mathbf{y}^*} = 0$, we get the closed-form solution

$$\mathbf{y}^* = (\mathbf{1} + \lambda_y \mathbf{w})^{-1} \cdot (\mathbf{x} \circledast \mathbf{f} + \lambda_y \mathbf{w} \odot \mathbf{y}). \tag{8}$$

With respect to $\mathbf{f}$, the form of Eq. (6) is the same as Eq. (2), while the only difference is the replacement of $\mathbf{y}$ with $\mathbf{y}^*$ and therefore, the filter $\mathbf{f}$ can be quickly solved as in original CF framework [11]. In the proposed spatially variant weights based CF tracking, we solve $\mathbf{y}^*$ and $\mathbf{f}$ alternately. Both $\mathbf{y}^*$ and $\mathbf{f}$ can be solved by the closed-form solution, which makes the algorithm more efficient. This way, the proposed method handles the boundary effects in CF by assigning different weights to the training samples and weakens the impact of unfaithful synthetic samples, which can achieve the similar effectiveness with spatial regularization (SR) [17] as discussed

in Section III-A. However, different from SR, the proposed formulation does not involve the elementwise multiplication operation on the filter $\mathbf{f}$ as in Eq. (3), which breaks the Fourier-domain efficient solution of CF tracking.

C. Dynamic Updating of Weight Distribution

We introduced the weight distribution $\mathbf{w}$ in Eq. (6). We can see that $\mathbf{w}$ is fixed in the whole tracking process after initialization. However, in real-world tracking tasks, object shape is usually irregular and may change frequently in the tracking process. From this perspective, it is unconscionable to define a constant weight distribution using only the spatial distance to the map center and fix it over time. Similar studies can be found in recent works [48], [49], which integrate the object-adaptive/temporal-dynamic constraint to improve the fixed SR map in [17]. In the following, we consider the object-related and temporal-varying information into the weight distribution to obtain more reliable filters and overcome the limitation of the fixed regularization weight distribution $\mathbf{w}$. Accordingly, we attempt to improve the weight distribution from fixed $\mathbf{w}$ into dynamic $\mathbf{d}$.

1) Problem Formulation: Based on Eq. (6), we introduce the dynamic weight distribution $\mathbf{d} \in \mathbb{R}^{N\times 1}$ and get

$$\begin{aligned} \mathrm{E_d}(\mathbf{f},\mathbf{y}^*,\mathbf{d}) = &\left\|\mathbf{x}\circledast\mathbf{f}-\mathbf{y}^*\right\|^2+\|\mathbf{f}\|^2 \\ &+\lambda_y\left\|\mathbf{d}\odot(\mathbf{y}-\mathbf{y}^*)^2\right\|_1+\lambda_w\left\|\frac{\mathbf{d}^2}{\mathbf{w}}\right\|_1 \\ &\text{s.t.}\ \begin{cases} d_k\geq 0 \\ \sum_k d_k=\mu \end{cases},\quad k=1,2\ldots N. \end{aligned} \tag{9}$$

where $\mathbf{w}=\{w_k|k=1\ldots N\}, \mathbf{d}=\{d_k|k=1\ldots N\}$. Different from the energy function in Eq. (6), the new formulation Eq. (9) is a function of three variables, i.e., the correlation filter $\mathbf{f}$, computed result $\mathbf{y}^*$, and the weight distribution $\mathbf{d}$. As a result, the weight map $\mathbf{d}$ is no longer pre-set constants as $\mathbf{w}$ in Eq. (6). The two constraints ensure that the weights in $\mathbf{d}$ are non-negative and summed up to μ. The last term in Eq. (9) is the regularization term on the sample weights in $\mathbf{d}$. This regularization is used to constrain the similarity between $\mathbf{d}$ and $\mathbf{w}$, which is controlled by the flexibility parameter $\lambda_w > 0$ and the prior sample weights $w_k > 0$, satisfying $\sum_k d_k = \sum_k w_k = \mu$. The parameter $\lambda_w > 0$ controls the adaptiveness of the sample weights d_k. Changing λ_w leads to a different degree of flexibility in the weights d_k, which will be discussed in the later part – '*Analysis of* λ_w'. Note that, we use '$\mathbf{d}^2$' rather than '$\mathbf{d}$' in Eq. (9). This is because d_k should be reserved in the differentiation of $\mathrm{E_d}$ w.r.t. d_k, which is shown in following Eq. (12) and Eq. (13).

2) Optimization: We extract the function $\mathrm{E_d}$ w.r.t. $\mathbf{d}$ and rewrite it in terms of scalar as

$$\begin{aligned} \mathrm{E_d}(\mathbf{d}) = &\lambda_y\sum_k|d_k(y_k-y_k^*)^2|+\lambda_w\sum_k|\frac{d_k^2}{w_k}| \\ &\text{s.t.}\ \begin{cases} d_k\geq 0 \\ \sum_k d_k=\mu \end{cases},\quad k=1,2\ldots N. \end{aligned} \tag{10}$$

Eq. (10) can be regarded as a constraint optimization problem. To optimize Eq. (10), we use Augmented Lagrange Method (ALM) [54] to solve for $\mathbf{d}$. We temporarily ignore the inequality constraint $d_k \geq 0$ and introduce Lagrange multipliers for the equality constraint,

$$\mathrm{E_d}(\mathbf{d})=\sum_k(\lambda_y|d_k(y_k-y_k^*)^2|+\lambda_w|\frac{d_k^2}{w_k}|)-\eta(\sum_k d_k-\mu), \tag{11}$$

where $\eta > 0$ denotes the Lagrange multiplier. Differentiation w.r.t. d_k gives,

$$\frac{\partial \mathrm{E_D}}{\partial d_k}=(L_k+2\lambda_w\frac{d_k}{w_k})-\eta, \tag{12}$$

where we denote $L_k = \lambda_y(y_k - y_k^*)^2$. The stationary point is computed by setting the partial derivatives to zero,

$$\frac{\partial \mathrm{E_D}}{\partial d_k}=0\Leftrightarrow d_k=\frac{\eta-L_k}{2\lambda_w}w_k. \tag{13}$$

The Lagrange multiplier η is computed by summing both sides of Eq. (13) over k and using $\sum d_k = \sum w_k = \mu$,

$$\begin{aligned} \sum_k d_k &= \sum_k\frac{\eta-L_k}{2\lambda_w}w_k \Leftrightarrow \\ \mu &= \sum_k\frac{\eta}{2\lambda_w}w_k-\sum_k\frac{L_k}{2\lambda_w}w_k \Leftrightarrow \\ \eta &= 2\lambda_w+\frac{1}{\mu}\sum_k L_k w_k. \end{aligned} \tag{14}$$

Combining (14) and (13) we have

$$d_k = w_k+\frac{w_k}{2\lambda_w\mu}(\sum_{l=1}^{N}L_l w_l-L_k), \tag{15}$$

where $L_l = \lambda_y(y_l - y_l^*)^2$.

Then we take into account the ignored constraint $d_k \geq 0$, and we define the constant[2]

$$\delta=\min_k 2\mu w_k\cdot|w_k(\sum_{l=1}^{N}L_l w_l-L_k)|^{-1}. \tag{16}$$

This choice ensures that $d_k > 0$, $\forall k$ if $0 < \frac{1}{\lambda_w} < \delta$. The inequality constraint is thus satisfied for $0 < \frac{1}{\lambda_w} < \delta$.

3) Analysis of λ_w*:* We analyze the effect of λ_w by considering the extreme cases of decreasing ($\lambda_w \to 0$) and increasing ($\lambda_w \to +\infty$) the flexibility parameter. **1)** $\lambda_w \to 0$: This corresponds to removing the last term in Eq. (9), implying no regularization on d_k. For the fixed Y and Y*, the energy function Eq. (9) is minimized by setting smaller values to d_k to those samples in which y_k is close to y_k^* and larger values to d_k when y_k is far from y_k^*. That is easy to make the weighting map lose the spatial constraint, if the last term in Eq. (9) is removed. Therefore, it is imperative to use a regularization on the weights d_k. **2)** $\lambda_w \to +\infty$: By introducing Lagrange multipliers, from Eq. (15) it can be shown that $d_k \to w_k$ when $\lambda_w \to +\infty$. Thus, increasing the parameter λ_w also reduces the flexibility of the weights d_k

[2]Please see the Appendix for more details.

about the prior weights w_k. The constant weight distribution in Eq. (6) is therefore obtained in the limit $\lambda_w \to +\infty$ by setting $d_k = w_k$. This way, the fixed spatial distribution $\mathbf{w}$ used in Section III-B is a special case of $\mathbf{d}$. The dynamic weight distribution can be seen as a generalization of Eq. (6) by introducing flexible sample weights d_k.

D. The New Tracking Scheme

To evaluate the influence of the proposed weight constrained training samples, we choose two famous state-of-the-art CF based trackers, i.e., Staple [20] and BACF [19] as the baseline methods. In the following, we denote the proposed tracker based on Staple [20] and BACF [19] as WSCF_{St} and WSCF_{BA} respectively. Similar to [17], we initialize the spatial weight map as a 2D Gaussian-shape map as shown in Fig. 2, then $\mathbf{w}$ is the vectorization of the weight map. For fair comparison, the proposed methods use the same features as our baseline methods. Specifically, following BACF, we employ 31-channel HOG features [55] using 4×4 cells multiplied by a Hann window [11] in WSCF_{BA}. For WSCF_{St}, we combine the HOG features and the global color histogram as in [20]. The parameters λ_y are set to 0.5 and 3 for WSCF_{St} and WSCF_{BA}, respectively. We set $\lambda_w = \frac{1}{\kappa\delta}(0 < \kappa < 1)$ to satisfy the constraint of $0 < \frac{1}{\lambda_w} < \delta$ and δ is calculated by Eq. (16) and κ is set as 0.9 and 0.5 in WSCF_{St} and WSCF_{BA}, respectively. Besides, we fix all the other parameters of the baseline trackers for fair comparison. We further discuss the influence of the parameters in Section IV-D.

The formal description of the proposed WSCF tracking scheme is shown in Algorithm 1. For scale estimation in the algorithm, following the most previous CF-based trackers [17], [19], we apply the filter on multi-scale searching areas for scale estimation [24]. Specifically, the learned filter $\mathbf{f}$ is applied on S searching areas with different covered regions, where the searching areas have been resized into the same size as the filter and S denotes the number of scales. This returns S correlation outputs.We employ the interpolation strategy in [17] to calculate detection scores of each output. The scale with the maximum score among all the outputs is taken as the optimal scale.

Algorithm 1 WSCF Tracking Scheme

Input: Frame $\{\mathbf{I}_t\}_1^T$, initial object bounding box $\mathbf{b}_1$
Output: Object bounding box of each frame $\{\mathbf{b}_t\}_2^T$
Initialization: initialize the correlation filter, initialize the spatially variant weight distribution $\mathbf{w}$.
Learn $\mathbf{f}$ by minimizing Eq. (6), $t = 2$.
while $t \leq T$ **do**
  Crop a search region $\mathbf{R}_t$ from $\mathbf{I}_t$ at the last bounding box $\mathbf{b}_{t-1}$ and extract its feature vector $\mathbf{z}$.
  Detect the object location $\mathbf{p}_t$ by calculating the response by Eq (1) via $\mathbf{z}$ and $\mathbf{f}$ and the estimate the scale of the target as in [24], thus get $\mathbf{b}_t$.
  Update $\mathbf{d}$ by Eq. (15) using the ALM algorithm.
  Update $\mathbf{y}^*$ by Eq. (9), where $\mathbf{y}^*$ can be calculated via Eq. (8) by setting $\mathbf{w} = \mathbf{d}$.
  Crop the image patch by the bounding box $\mathbf{b}_t$ and extract the feature map $\mathbf{x}$. Learn $\mathbf{f}$ by minimizing Eq. (9) via $\mathbf{x}$ and $\mathbf{y}^*$.
  $t = t + 1$
return $\{\mathbf{b}_t\}_2^T$.

IV. Experimental Results

A. Setup

1) Datasets and Metrics: The proposed method is implemented in MATLAB and runs on a desktop computer with an Intel Core i7 3.4 GHz CPU. We evaluate the proposed method on five standard benchmarks: OTB-2013, OTB-2015, VOT-2018, TC-128 and LaSOT. The first two OTB datasets contains 50 and 100 videos, respectively. We use two acknowledged metrics, i.e., precision and success rate under the OPE (one-pass evaluation) for quantitative evaluation. The precision metric measures the distance between the detected target locations and those of the ground truth. The success rate metric measures the intersection-over-union (IoU) between predicted and ground truth bounding boxes. For precision metric, we use a threshold, i.e., 20 pixels to judge whether a tracker is successful at each frame and calculate the percentage of successful frames out of all the frames in each sequence. For success rate metric, we calculate average success percentages w.r.t. different overlap ratio thresholds and obtain success plot and after that the area under curve (AUC) of each plot can be calculated as the success rate AUC score. The VOT-2018 [56] has 60 videos and re-initializes the object bounding box when the tracker fails to track the target. The expected average overlap (EAO) considers both bounding box overlap ratio – accuracy, and the failures times (re-initialization times) – robustness, which serves as the major evaluation metrics. The VOT-2018 also provides the EAO score with standard (baseline) and real-time experimental setup. The latter requires predicting bounding boxes faster or equal to the video frame-rate. TC-128 [57] contains 128 color sequences and it is used to explore the ability of trackers to encode color information. LaSOT [58] dataset is a recently proposed large-scale video dataset for visual object tracking which consists of 1,400 long-term sequences. Here, we report the results on its testing subset which contains 280 sequences with 690K frames. Following the metrics of OTB dataset, TC-128 and LaSOT also adopt the AUC of success plot and precision at 20 pixels as the evaluation metrics.

2) Comparison Methods: We compare the proposed method with 10 state-of-the-art trackers. Besides our baseline trackers i.e., Staple [20] and BACF [19], there are six hand-crafted features based trackers, i.e., KCF [12], DSST [21], SAMF [24], DLSSVM [7], SRDCF [17], CSR-DCF [18], and two deep feature based methods including HCF [25] and HDT [26]. Among them, KCF, DSST, SAMF, Staple, SRDCF, CSR-DCF and BACF are classical CF based trackers which obtain the performance improvement in filter learning. Besides, HCF and HDT are deep feature boosted CF trackers, DLSSVM is a SVM based tracker.

TABLE I

ATTRIBUTES BASED SUCCESS RATE AUC SCORES FOR $WSCF_{St}$, $WSCF_{BA}$ AND OTHER 12 STATE-OF-THE-ART TRACKERS ON OTB-2015. THE BEST THREE RESULTS ARE MARKED IN RED, GREEN AND BLUE RESPECTIVELY. ⇑ DENOTES THE IMPROVEMENT COMPARED TO BASELINE TRACKER

	OCC	BC	IV	FM	DEF	SV	OPR	IPR	OV	MB	LR
KCF	44.3	49.8	47.9	45.9	43.6	39.4	45.3	46.9	39.3	45.8	30.7
DSST	46.1	52.4	56.1	46.6	43.4	47.9	48.2	51.1	38.5	47.3	39.5
SAMF	53.4	53.4	53.4	52.6	49.5	49.3	52.7	52.7	50.0	52.3	43.1
DLSSVM	50.8	51.7	52.1	54.3	51.2	46.5	53.1	53.3	46.8	57.1	39.9
SRDCF	55.9	58.3	61.3	59.7	54.4	56.1	55.0	54.4	46.0	59.4	49.4
CSR-DCF	53.1	56.4	54.2	57.2	53.2	52.0	52.0	51.2	51.8	56.8	43.1
HCF	52.5	58.5	54.0	57.0	53.0	48.5	53.4	55.9	47.4	58.5	43.9
HDT	52.8	57.8	53.5	56.8	54.3	48.6	53.4	55.5	47.2	57.4	45.6
Staple	54.3	56.1	59.5	54.1	55.0	52.0	53.4	54.9	47.6	54.0	39.9
BACF	56.6	60.5	62.3	59.9	57.3	57.3	57.8	58.2	54.7	57.0	53.2
$WSCF_{St}$	55.4 ⇑	59.3 ⇑	62.2 ⇑	56.0 ⇑	55.2 ⇑	55.3 ⇑	56.4 ⇑	59.2 ⇑	52.7 ⇑	56.6 ⇑	38.7
$WSCF_{BA}$	58.2 ⇑	61.6 ⇑	63.6 ⇑	60.2 ⇑	59.3 ⇑	57.3	59.4 ⇑	58.3 ⇑	51.2	57.6 ⇑	50.7

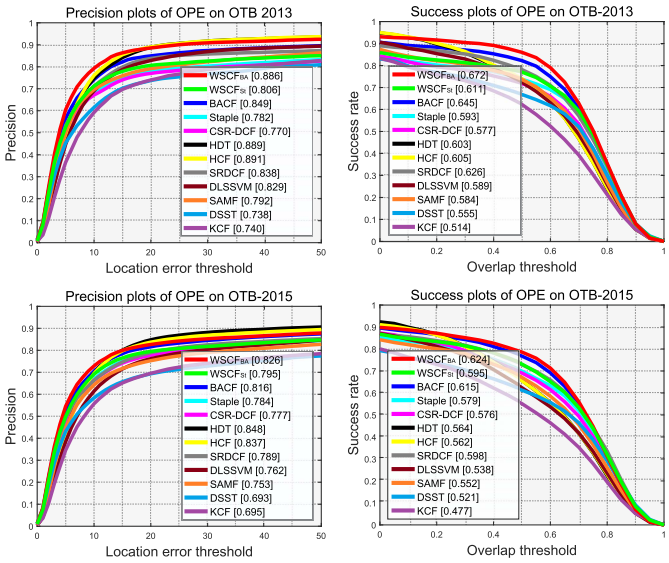


Fig. 3. Precision plots (left) and success plots (right) of both the proposed and comparison methods on OTB-2013 (first row) and OTB-2015 (second row) benchmark. The legend contains the average distance precision score at 20 pixels and the AUC of success plot of each method.

B. Comparative Results

1) Comparison Results on OTB: We evaluate the overall performance of the proposed method and compare with the baseline methods and other 10 state-of-the-art trackers on OTB-2013 and OTB-2015 benchmarks. As shown in the first row of Fig. 3, in terms of precision score, our methods $WSCF_{BA}$ and $WSCF_{St}$ outperform the baseline methods and achieve 3.7% and 2.4% improvement over BACF and Staple, respectively. In terms of the AUC of success plots, the proposed $WSCF_{BA}$ and $WSCF_{St}$ also get better performance than the baseline methods and achieve the state-of-the-art performance. We can see that both $WSCF_{BA}$ and $WSCF_{St}$ outperform the deep trackers HCF, HDT. We can see similar results on OTB-2015 in the second row of Fig. 3. The proposed $WSCF_{BA}$ and $WSCF_{St}$ outperform the baseline methods BACF and Staple in both precision score and AUC score of success rate. We can also see that several deep trackers, e.g., HDT, obtain higher precision scores than the proposed method. However, our $WSCF_{BA}$ obtains the highest performance in success plot AUC score, which is a more comprehensive metric to evaluate the object tracking performance.

2) Attribute Based Comparison: To more comprehensively evaluate the proposed tracker in various scenes, we present tracking performance in terms of different attributes on OTB-2015. The 100 videos of OTB-2015 are grouped into 11 subsets according to 11 attributes, i.e., occlusion (OCC), background clutter (BC), illumination variation (IV), fast motion (FM), deformation (DEF), scale variation (SV), out-of-plane rotation (OPR), in-plane rotation (IPR), out-of-view (OV), motion blur (MB), and low resolution (LR). Table I shows the success plot AUC of 10 comparison methods and the proposed $WSCF_{BA}$ and $WSCF_{St}$ on 11 subsets. In terms of results on 11 subsets, $WSCF_{BA}$ gets the best results on 7 subsets of OCC, BC, IV, FM, DEF, SV, and OPR and the second best performance on 2 subsets of IPR and LR. We can see that $WSCF_{St}$ outperforms the baseline tracker Staple on 10 subsets except for LR. $WSCF_{BA}$ also outperforms BACF on 8 subsets except for the comparable results on SV subset, and inferior results on OV and LR subsets. Note that, the subsets of OV and LR contain 14 and 9 sequences respectively, which are two attributes with the fewest sequences. Such small evaluation set can be easily biased.

3) Comparison Results on VOT: In addition to OTB benchmark, we also evaluate the proposed method on VOT-2018. We compare $WSCF_{St}$, $WSCF_{BA}$ with the baseline trackers, i.e., Staple [20] and BACF [19], and five methods that participate in the VOT challenge, i.e., KCF [12], DSST [21], SAMF [24], SRDCF [17] and CSR-DCF [18]. As shown in Table II, we can see that both $WSCF_{St}$ and $WSCF_{BA}$ outperform the baseline methods Staple and BACF in accuracy, robustness and expected average overlap (EAO) under baseline experiments, respectively. In terms of accuracy, $WSCF_{St}$ gets the best performance which improves the performance of Staple by 3.5%. Note that, CSR-DCF provides the best performance on robustness and baseline EAO score. However,

TABLE II

COMPARATIVE RESULTS ON VOT-2018 IN TERMS OF THE AVERAGE ACCURACY (ACCURACY), AVERAGE FAILURES (ROBUSTNESS), EAO UNDER BASELINE EXPERIMENTS AND EAO UNDER REALTIME EXPERIMENTS

Trackers	Baseline			Realtime
	Accuracy ↑	Robustness ↓	EAO ↑	EAO ↑
KCF	0.4336	2.55	0.1398	0.1385
DSST	0.3875	5.36	0.0829	0.0811
SAMF	0.4623	2.33	0.1515	0.0831
SRDCF	0.4765	3.77	0.1216	0.0614
CSR-DCF	0.4871	**1.28**	**0.2678**	0.1024
Staple	0.5222	2.49	0.1751	0.1753
BACF	0.5101	3.46	0.1408	0.1332
WSCF_{St}	**0.5404**	2.17	0.1841	**0.1783**
WSCF_{BA}	0.5210	3.34	0.1465	0.1400

as shown in the last column, it provides a very poor EAO score in real-time experimental setting, which is even lower than KCF. This is because CSR-DCF runs at $\sim$ 8 fps, far away from real-time speed. The proposed WSCF_{St} gets the best EAO score on real-time experiments, which can run at a real-time speed on VOT dataset.

4) Comparison Results on TC-128: We show the evaluation results on TC-128 in Fig. 4. Clearly, with our WS based method, the precision scores of BACF, and Staple get the relative improvements at 2.3% and 1.8%, respectively. For sucess rate AUC scores, WSCF_{BA} and WSCF_{St} get 2.9% and 2.2% relative improvements over their baseline methods, respectively. We can see that DLSSVM gets the best performance among the comparison methods. Note that, although DLSSVM gets the highest score in term of precision, for the compositive metric – success rate AUC score, our methods can obtain the similar (WSCF_{BA} at 49.9%) or better (WSCF_{St} at 51.1%) results compared to DLSSVM.

5) Comparison Results on LaSOT: We further evaluate the proposed method on LaSOT dataset. As shown in Fig. 5, we can see that the proposed method improves both BACF and Staple with 3.3% and 2.5% relative increments on precision scores, respectively. In terms of success rate AUC score, the relative increments are 2.3% and 3.7%, respectively. We can also see that, compared to original baseline tracker Staple with a success rate AUC score of 24.3%, the proposed method WSCF_{St} gets the AUC score of 25.2%, which outperforms several trackers that are superior to Staple, e.g., CSR-DCF, SRDCF, DLSSVM and HCF, and obtains the comparative performance with HDT. Moreover, the proposed method WSCF_{BA} gets the AUC score of 26.5% exceeding HDT. Above results demonstrate that the proposed WSCF algorithm not only improves the CF trackers on short-term sequences, e.g., OTB and TC-128 datasets, but also helps handle challenges from long-term sequences.

C. Ablation Study

To validate the effectiveness of our method, we show the results of ablation experiments on OTB-2013 and OTB-2015.

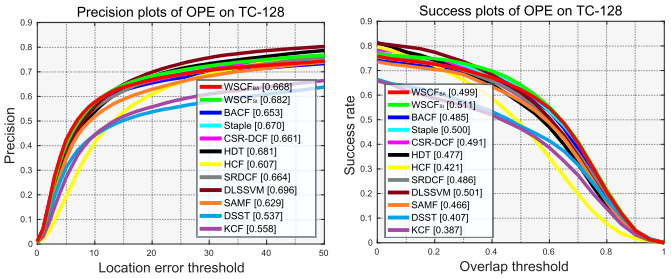


Fig. 4. Precision plots (left) and success plots (right) of both the proposed and comparison methods on TC-128 benchmark. The legend contains the average distance precision score at 20 pixels and the AUC of success plot of each method.

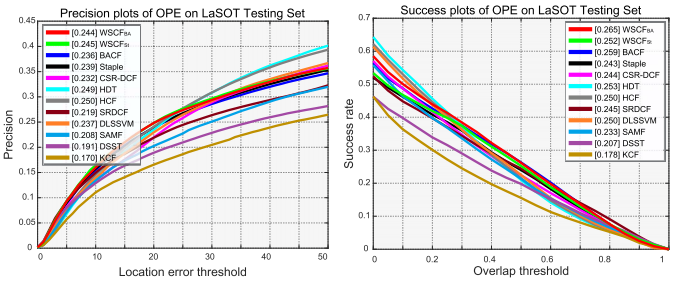


Fig. 5. Precision plots (left) and success plots (right) of both the proposed and comparison methods on LaSOT benchmark. The legend contains the average distance precision score at 20 pixels and the AUC of success plot of each method.

TABLE III

ABLATION STUDY ON OTB-2013 AND OTB-2015 VIA SUCCESS RATES (% AT IOU > 0.50 AND AUC SCORE)

Trackers	OTB-2013		OTB-2015	
	Succ. IOU	Succ. AUC	Succ. IOU	Succ. AUC
Staple	73.8	59.3	69.9	57.9
WSCF_{St}-no $\mathbf{d}$	75.6	59.9	71.7	58.4
WSCF_{St}	**77.0**	**61.1**	**73.1**	**59.5**
BACF	82.2	64.5	76.8	61.5
WSCF_{BA}-no $\mathbf{d}$	83.6	65.6	77.4	61.9
WSCF_{BA}	**86.0**	**67.3**	**78.6**	**62.4**

Specifically, as shown in Table III, Staple and BACF the two baseline trackers. WSCF_{St}-no $\mathbf{d}$ and WSCF_{BA}-no $\mathbf{d}$ use the fixed weight distribution i.e., $\mathbf{w}$ in Eq. (6), to assign the weights for training samples. The bottom row WSCF_{St} and WSCF_{BA} use the dynamically updated weight distribution i.e., $\mathbf{d}$ in Eq. (9), to assign the sample weights. For each baseline tracker, from the first and second rows we can see that the weighting for samples by the fixed weight distribution $\mathbf{w}$ can improve the performance of CF based trackers. Moreover, from the last two rows of each tracker we can also see that the dynamic updating of the weight distribution by $\mathbf{d}$ can further promote the improvement, which demonstrates the advantage of adding the updating of $\mathbf{w}$.

To further study the effectiveness of the dynamic weight distribution presented in Section III-C, we also evaluate the tracking performance in terms of different attributes on OTB-2013 and OTB-2015. We select six subsets with

TABLE IV

ABLATION STUDY ON SIX SUBSETS WITH DIFFERENT ATTRIBUTES IN OTB-2013 AND OTB-2015 VIA SUCCESS RATES (% AUC SCORE)

Trackers	OTB-2013						OTB-2015					
	SV	IPR	OPR	IV	DEF	OCC	SV	IPR	OPR	IV	DEF	OCC
Staple	54.5	57.6	56.9	56.1	58.5	55.7	52.0	54.9	53.4	59.5	55.0	54.3
WSCF_{St}-no $\mathbf{d}$	56.9	60.7	60.1	59.9	**62.4**	**58.4**	54.0	57.9	56.1	61.2	54.6	54.6
WSCF_{St}	**57.9**	**62.0**	**60.2**	**60.1**	60.0	58.2	**55.3**	**59.2**	**56.4**	**62.2**	**55.2**	**55.4**
BACF	61.0	63.0	63.0	59.0	62.2	62.4	57.3	58.2	57.8	62.3	57.3	56.6
WSCF_{BA}-no $\mathbf{d}$	61.6	63.4	64.1	61.0	64.3	64.4	57.2	57.9	58.4	63.5	58.2	57.2
WSCF_{BA}	**61.9**	**64.1**	**66.2**	**61.7**	**67.2**	**66.6**	**57.3**	**58.3**	**59.4**	**63.6**	**59.3**	**58.2**

TABLE V

COMPARATIVE STUDY OF DIFFERENT PARAMETER SETTINGS OF λ_y, κ ON OTB-2015 VIA SUCCESS RATES (% AT IOU > 0.50 AND AUC SCORE)

Parm.	WSCF_{St}		Parm.	WSCF_{BA}	
	Succ. IOU	Succ. AUC		Succ. IOU	Succ. AUC
$\lambda_y = 0.25$	71.7	58.7	$\lambda_y = 1.5$	77.6	61.7
$\lambda_y = 0.5$	**73.1**	**59.5**	$\lambda_y = 3$	**78.6**	**62.4**
$\lambda_y = 1$	71.7	58.5	$\lambda_y = 6$	77.0	61.5
$\kappa = 0.85$	72.0	58.9	$\kappa = 0.45$	78.5	62.4
$\kappa = 0.9$	**73.1**	**59.5**	$\kappa = 0.5$	**78.6**	**62.4**
$\kappa = 0.95$	71.9	58.7	$\kappa = 0.55$	78.1	62.1
Staple	69.9	57.9	BACF	76.8	61.5

corresponding attributes in which the targets or their backgrounds have frequent variations over time, i.e., scale variation (SV), in-plane rotation (IPR), out-of-plane rotation (OPR), illumination variation (IV), deformation (DEF), occlusion (OCC). As shown in Table IV, we can see that the proposed methods, including WSCF_{St} and WSCF_{BA}, using dynamic weights $\mathbf{d}$ perform better than both the baselines and those using fixed weights $\mathbf{w}$, on the selected attribute-aware subsets of OTB-2013 and OTB-2015. This way, the dynamic weight distribution $\mathbf{d}$ can better handle the cases where the targets/backgrounds have over-time changes, therefore it improves the final accuracy on overall dataset as shown in Table III.

D. Algorithm Analysis

1) Parameters Selection Analysis: We investigate the performance changes w.r.t. different setups of the two parameters, i.e., the control parameters λ_y and λ_w in Eq. (9). The parameter values in WSCF_{St} and WSCF_{BA} are slightly different due to the diversity of the baseline methods and we tune the setups of the parameters on two methods, respectively. As shown in Table V, we set λ_y as the original settings and enlarge/reduce it two times respectively. We can see that the tracking accuracy changes little with the huge displacement of λ_y. Hence, our method is not very sensitive to λ_y. The parameter λ_w in Eq. (9) is determined by two factors, i.e., κ and δ. The variable δ is calculated online in the tracking process, thus we only tune the hyperparameter κ. Table V compares the tracking results on OTB-2015 with different setups of κ. Clearly, the tracking accuracy changes little with different κ. In practice, we can see that the tracking results under different parameter settings all outperform the baseline trackers, which demonstrates the robustness of the proposed method.

2) Qualitative Analysis: Qualitative comparisons on several sequences are shown in Fig. 6. The two sequences *dragonbaby* and *shaking* are selected to show the robustness of trackers against object deformation. The targets in these two sequences are head of a baby and a singer, respectively, both have significant appearance variations when moving and turning. We can see that BACF fails to track the baby (e.g., #113) and Staple fails to track the singer (e.g., #31, #295, #335). The proposed trackers WSCF_{St} and WSCF_{BA} can track them continuously against the deformation. The second row in Fig. 6 shows the tracking results on two representative sequences *skater2* and *basketball*, where the targets are the whole body of a skater and a basketball player, respectively. The shape of the targets also vary drastically as the motion of athletes. We can see that Staple tracker fails to estimate the scale of the skater (e.g., #351, #339, #373) and both Staple and BACF lost the basketball player (e.g., #500, #666). Our trackers WSCF_{St} and WSCF_{BA} can track the target successfully over the whole sequence. The third row in Fig. 6 shows the target with severe or long-term occlusion. In the *girl* sequence, the girl can be tracked well until it is occluded by a man in a long term. Three trackers, i.e. Staple, BACF and CSR-DCF mistakenly track the man appearing in the foreground (e.g., #457), while the proposed trackers WSCF_{St} and WSCF_{BA} can unceasingly track the target. Similarly in the *jogging* sequence, both Staple and BACF lost the runner after leaving the obstruction (e.g., #070), while the proposed tracker WSCF_{BA} can keep tracking her continuously.

3) Failure Cases and Limitations Analysis: We have shown that proposed training sample weighting does help improve the tracking accuracy of CF based tracking method. However, the sample weighting strategy becomes less effective when the target moves very fast. Specifically, as shown in the last row of Fig. 6, when the target, e.g. the high jumper, moves fast, the center localization of the target may be far away between neighbor frames. As a result, the weight distribution may assign higher weights to the training samples that contain more background region instead of the target region, which makes the updated filter in frame $t-1$ less effective in detecting the target in frame t. Note that, although sample weighting strategy has such limitation, our method, i.e. WSCF, still outperforms the baseline trackers on the subset of fast motion

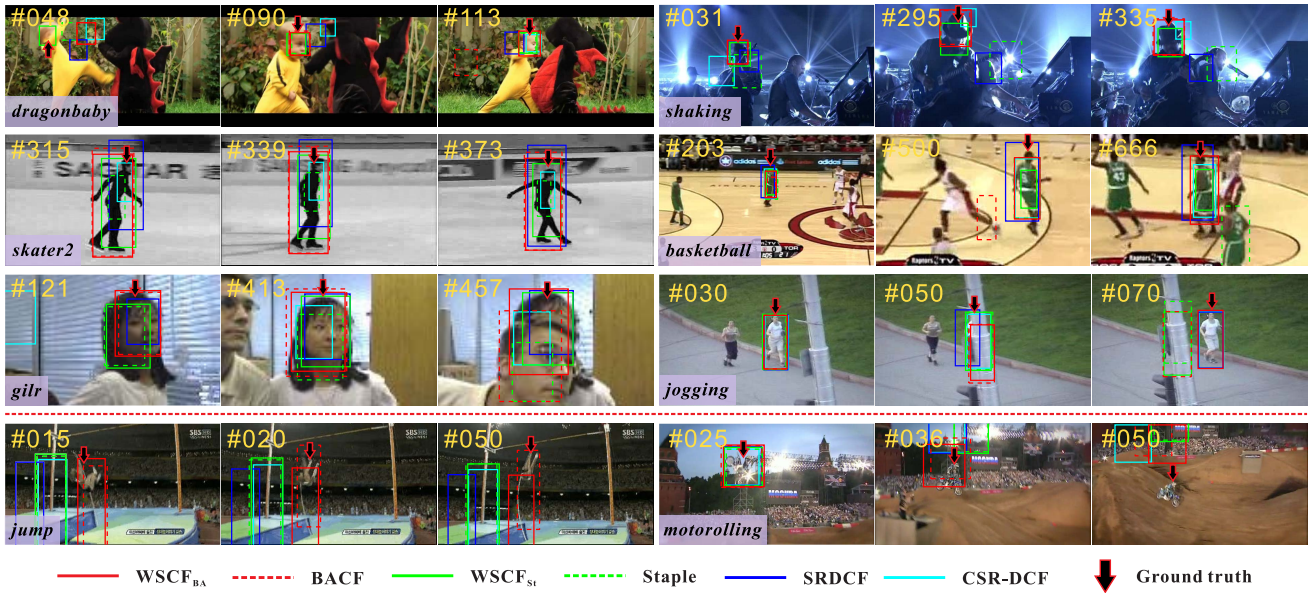


Fig. 6. Tracking results of WSCF$_{BA}$, WSCF$_{St}$ and other four other CF trackers on eight challenging videos in OTB-2015. The last two cases show the limitation of WSCF based trackers in handing object fast motion.

TABLE VI

COMPARATIVE STUDY OF DIFFERENT EXPERIMENTAL SETTINGS FOR SR AND WS BASED TRACKERS ON OTB-2015 VIA SUCCESS RATES (% AUC SCORE) AND AVERAGE RUNNING SPEED (FPS)

Type	KCF		Type	SR for CF		Type	WS for CF	
	Succ. AUC	Avg. FPS		Succ. AUC	Avg. FPS		Succ. AUC	Avg. FPS
KCF	47.7	153.5	KCF + SR	49.3	7.8	KCF + WS	49.7	84.8
KCF + SE	50.7	37.6	KCF + SE + SR	57.5	4.5	KCF + SE + WS	56.7	33.0
-	-	-	SRDCF	59.8	3.5	WSCF	58.6	27.3

in OTB-2015, as shown in Table I. This is because the above sudden motion does not happen frequently in a sequence and the proposed sample weighting strategy could help improve tracking accuracy for most of the time.

4) Comparison of WS and SR Model: To compare the proposed weighted samples (WS) based method for CF with the spatial regularization (SR) in SRDCF, we conduct some tracking algorithms with different settings. Specifically, we select KCF as the baseline and integrate the SR and WS methods to boost it, respectively. As shown in the the first row of Table VI, both SR and WS can improve the tracking performance of KCF. Although both the SR and WS reduce the running speed of the baseline, the WS based method can still runs at real-time speed. However, the SR severely pulls down the speed into 7.8 fps. Besides, we also integrate the scale estimation (SE) method [17], [24] into the baseline for further comparison as shown in the second row. We can see that the WS based method can get comparative accuracy with SR and maintain a higher running speed. As shown in the last row, we compare the complete SRDCF [17] algorithm with our WS based tracker using the same scale of search region, training data and detection strategy, i.e., WSCF. We can see that although the tracking performance of WSCF is slightly lower than SRDCF, WSCF has an approximate seven-time speedup compared to SRDCF and achieves real-time running speed. Moreover, as shown in Fig. 3, the proposed techniques can be applied to existing trackers, e.g., Staple and BACF, to obtain further performance gain.

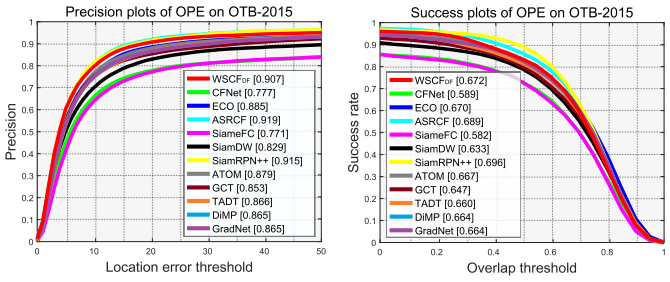


Fig. 7. Precision plots (left) and success plots (right) of both the proposed method with deep features and state-of-the-art deep learning based methods on OTB-2015. The legend contains the average distance precision score at 20 pixels and the AUC of success plot of each method.

5) Deep Feature Based Tracker: To further assess WSCF, we exploit the deep features (Norm1 from VGG-M, Conv4-3 from VGG-16) for object representation to enhance our method WSCF$_{BA}$. In the following, we name it as WSCF$_{DF}$ for simplicity. We compare it with several most state-of-the-art deep learning based methods, including the latest CF trackers: ECO [38], CFNet [32], and ASRCF [49]; a series of Siamese

TABLE VII
SUCCESS RATES (% AT IOU>0.50 AND AUC SCORE) OF WSCF_{St},WSCF_{BA} VERSUS RELATED TRACKERS, AND THE CORRESPONDING WEIGHTED AVERAGE SPEEDS ON OTB-2015 DATASET

	KCF (PAMI2015)	CFLB (CVPR2015)	SRDCF (ICCV2015)	CSR-DCF (CVPR2017)	Staple (CVCPR2016)	BACF (ICCV2017)	WSCF_{St} (Ours)	WSCF_{BA} (Ours)
Succ.rate (IoU)	55.1	44.7	72.8	69.1	69.9	76.8	73.1	78.6
Succ.rate (AUC)	47.7	41.5	59.8	57.6	57.9	61.5	59.5	62.4
Avg.FPS	153.5	87.1	3.5	7.5	43.9	18.4	32.9	16.3

Network based trackers: SiameseFC [14], SiamDW [59], and SiamRPN++ [60]; and other methods proposed in 2019: ATOM [61], GCT [62], TADT [63], DiMP [64], GradNet [65]. The comparison results on OTB-2015 are shown in Fig. 7, we can see that WSCF_{DF} achieves comparative or better performance compared to these state-of-the-art deep learning based methods in term of both the precision and success rate AUC scores.

6) Speed Analysis: Algorithm efficiency is also an important factor for the tracking task. Table VII compares several related well-known CF trackers, where the speed is measured on a desktop computer with an Intel Core i7 3.4GHz CPU. We can see that the proposed WSCF_{St} and WSCF_{BA} can improve the baseline Staple and BACF in terms of success rate while with little sacrifice on tracking speed. Our method WSCF_{St} achieves the real-time running speed over 30 fps and WSCF_{BA} runs near real time at an average of 16 fps, both of which are more efficient than the spatial regularization based CF tracking methods, e.g., SRDCF, CSR-DCF. The performance improvement of WSCF_{BA} is relatively small compared with WSCF_{St}, because the real samples used in BACF can alleviate the boundary effects to some extent.

Note that, the proposed methods do not rely on the offline trained models and pre-trained deep features, which can effectively save training time and storage space. Moreover, our method is implemented on the Matlab platform without any optimization strategies and can be implemented for real time applications by further optimizing the code with GPU acceleration or parallel computing.

V. CONCLUSION

In this paper, we have proposed a fast content-aware spatial regularization approach for correlated filter (CF), one of the most popular visual object tracking schemes. Our approach implicitly weighs circular shifted training samples and solve the spatially regularized learning of correlation filters in closed form. This provides a new efficient spatial regularization way for the CF tracking scheme. Furthermore, to adapt to temporal variations, we present a content-aware updating strategy to dynamically optimize the weight distribution by solving a constrained quadratic optimization problem. Since the proposed fast content-aware spatial regularization approach is general for the CF tracking scheme, it is able to help many CF based trackers to alleviate the boundary effects without harming their original tracking speed. Particularly, our approach is used to improve two state-of-the-art CF trackers resulting in very promising performance improvement. On five benchmark datasets, OTB-2013, OTB-2015, VOT-2018, TC-128 and LaSOT, we have validated the effectiveness and superiority of our approach over various state-of-the-art competitors. In the future, we plan to further explore the potentials of our approach to more recent sophisticated CF based tracking methods, and study its possible application within other popular tracking scheme, such as Siamese network.

APPENDIX

We discuss the derivation of constraint condition defined in Eq. (16), which is to guarantee $d_k > 0$. From Eq. (15),

$$d_k = w_k + \frac{1}{\lambda_w} \cdot \frac{w_k}{2\mu} \Delta L, \tag{17}$$

where we denote $\Delta L = \sum_{l=1}^{N} L_l w_l - L_k$. Given the preset parameters $w_k \geq 0$ and $\mu \geq 0$. 1) If $\Delta L \geq 0$, d_k will monotonically increase with $\frac{1}{\lambda_w}$ and it is easy to get that $d_k > 0$ when $\frac{1}{\lambda_w} \geq 0$. 2) If $\Delta L < 0$, d_k will monotonically decrease with $\frac{1}{\lambda_w}$ and d_k will get the minimum value when $\frac{1}{\lambda_w}$ take the maximum. To calculate the upper bound of $\frac{1}{\lambda_w}$, we solve

$$w_k + \frac{1}{\lambda_w} \cdot \frac{w_k}{2\mu} \Delta L = 0, \tag{18}$$

and then we get

$$\frac{1}{\lambda_w} = 2\mu w_k \cdot |w_k \Delta L|^{-1}. \tag{19}$$

Thus we take $\delta = \min\ 2\mu w_k \cdot |w_k \Delta L|^{-1}$. To sum up, the constraint condition $0 < \frac{1}{\lambda_w} < \delta$ will guarantee $d_k > 0$.

ACKNOWLEDGMENT

The authors thank all reviewers and the associate editor for their valuable comments. They thank Q. Guo, P. Zhang, and Z. Chen for the mutual academic discussion. They especially thank A. Waseem for helping them to remotely access the devices in lab for implementing experiments during the school holiday.

REFERENCES

[1] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1442–1468, Jul. 2014.

[2] C. A. Perez *et al.*, "Automatic safety video surveillance-tracking system to avoid vehicle-workers interaction for mining applications," in *Proc. Int. Symp. Optomechatronic Technol.*, Nov. 2014, pp. 23–27.

[3] R. Han *et al.*, "Complementary-view multiple human tracking," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 1–8.

[4] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel, "A survey of appearance models in visual object tracking," *ACM Trans. Intell. Syst. Technol.*, vol. 4, no. 4, pp. 1–48, Sep. 2013.

[5] Q. Guo, W. Feng, C. Zhou, C.-M. Pun, and B. Wu, "Structure-regularized compressive tracking with online data-driven sampling," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 5692–5705, Dec. 2017.

[6] S. Avidan, "Support vector tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 1064–1072, Aug. 2004.

[7] J. Ning, J. Yang, S. Jiang, L. Zhang, and M.-H. Yang, "Object tracking via dual linear structured SVM and explicit feature map," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4266–4274.

[8] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *Int. J. Comput. Vis.*, vol. 77, nos. 1–3, pp. 125–141, May 2008.

[9] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 983–990.

[10] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2042–2049.

[11] D. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2544–2550.

[12] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.

[13] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4293–4302.

[14] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional Siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 850–865.

[15] Y. Wu, J. Lim, and M. H. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1834–1848, Sep. 2015.

[16] H. K. Galoogahi, T. Sim, and S. Lucey, "Correlation filters with limited boundaries," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4630–4638.

[17] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4310–4318.

[18] A. Lukežič, T. Vojiř, L. Č. Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6309–6318.

[19] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1135–1143.

[20] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1401–1409.

[21] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 1–5.

[22] T. Zhang, S. Liu, C. Xu, B. Liu, and M.-H. Yang, "Correlation particle filter for visual tracking," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2676–2687, Jun. 2018.

[23] M. Danelljan, F. S. Khan, M. Felsberg, and J. V. D. Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1090–1097.

[24] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 254–265.

[25] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3074–3082.

[26] Y. Qi *et al.*, "Hedged deep tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4303–4311.

[27] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop (ICCVW)*, Dec. 2015, pp. 58–66.

[28] T. Zhang, C. Xu, and M.-H. Yang, "Multi-task correlation particle filter for robust object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4335–4343.

[29] N. Wang, W. Zhou, Y. Song, C. Ma, and H. Li, "Real-time correlation tracking via joint model compression and transfer," *IEEE Trans. Image Process.*, vol. 29, pp. 6123–6135, 2020.

[30] C. Sun, D. Wang, H. Lu, and M.-H. Yang, "Correlation tracking via joint discrimination and reliability learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 489–497.

[31] Y. Sun, C. Sun, D. Wang, Y. He, and H. Lu, "ROI pooled correlation filters for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5783–5791.

[32] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2805–2813.

[33] N. Wang, Y. Song, C. Ma, W. Zhou, W. Liu, and H. Li, "Unsupervised deep tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1308–1317.

[34] B. Du, S. Cai, and C. Wu, "Object tracking in satellite videos based on a multiframe optical flow tracker," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 8, pp. 3043–3055, Aug. 2019.

[35] J. Shao, B. Du, C. Wu, and L. Zhang, "Tracking objects from satellite videos: A velocity feature based correlation filter," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7860–7871, Oct. 2019.

[36] J. Shao, B. Du, C. Wu, and L. Zhang, "Can we track targets from space? A hybrid kernel correlation filter tracker for satellite video," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 8719–8731, Nov. 2019.

[37] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 472–488.

[38] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6638–6646.

[39] G. Bhat, J. Johnander, M. Danelljan, F. S. Khan, and M. Felsberg, "Unveiling the power of deep tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 483–498.

[40] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1396–1404.

[41] Z. Chen, Q. Guo, L. Wan, and W. Feng, "Background-suppressed correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2018, pp. 1–6.

[42] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1430–1438.

[43] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4904–4913.

[44] B. Li, C. Liu, J. Liu, H. Gao, X. Song, and W. Liu, "Visual tracking using spatial-temporal regularized support correlation filters," in *Proc. Int. Conf. Commun., Image Signal Process.*, 2018, pp. 1–7.

[45] D. Zhang *et al.*, "Part-based visual tracking with spatially regularized correlation filters," *Vis. Comput.*, vol. 36, no. 3, pp. 509–527, Mar. 2020.

[46] P. Zhang, Q. Guo, and W. Feng, "Fast and object-adaptive spatial regularization for correlation filters based tracking," *Neurocomputing*, vol. 337, pp. 129–143, Apr. 2019.

[47] R. Han, Q. Guo, and W. Feng, "Content-related spatial regularization for visual object tracking," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2018, pp. 1–6.

[48] W. Feng, R. Han, Q. Guo, J. Zhu, and S. Wang, "Dynamic saliency-aware regularization for correlation filter-based object tracking," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3232–3245, Jul. 2019.

[49] K. Dai, D. Wang, H. Lu, C. Sun, and J. Li, "Visual tracking via adaptive spatially-regularized correlation filters," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4670–4679.

[50] Q. Guo, R. Han, W. Feng, Z. Chen, and L. Wan, "Selective spatial regularization by reinforcement learned decision making for object tracking," *IEEE Trans. Image Process.*, vol. 29, pp. 2999–3013, 2020.

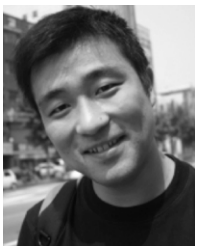
[51] T. Hu, L. Huang, X. Liu, and H. Shen, "Real time visual tracking using spatial-aware temporal aggregation network," 2019, *arXiv:1908.00692*. [Online]. Available: http://arxiv.org/abs/1908.00692

[52] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, "Learning dynamic siamese network for visual object tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1763–1771.

[53] H. K. Galoogahi, T. Sim, and S. Lucey, "Multi-channel correlation filters," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3072–3079.

[54] S. Boyd, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010.
[55] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 886–893.
[56] M. Kristan *et al.*, "The sixth visual object tracking VOT2018 challenge results," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–53.
[57] P. Liang, E. Blasch, and H. Ling, "Encoding color information for visual tracking: Algorithms and benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5630–5644, Dec. 2015.
[58] H. Fan *et al.*, "LaSOT: A high-quality benchmark for large-scale single object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5374–5383.
[59] Z. Zhang and H. Peng, "Deeper and wider siamese networks for real-time visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4591–4600.
[60] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "SiamRPN++: Evolution of siamese visual tracking with very deep networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4282–4291.
[61] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ATOM: Accurate tracking by overlap maximization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4660–4669.
[62] J. Gao, T. Zhang, and C. Xu, "Graph convolutional tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4649–4659.
[63] X. Li, C. Ma, B. Wu, Z. He, and M.-H. Yang, "Target-aware deep tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1369–1378.
[64] G. Bhat, M. Danelljan, L. Van Gool, and R. Timofte, "Learning discriminative model prediction for tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6182–6191.
[65] P. Li, B. Chen, W. Ouyang, D. Wang, X. Yang, and H. Lu, "GradNet: Gradient-guided network for visual object tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6162–6171.

Ruize Han received the B.S. degree in mathematics and applied mathematics from the Hebei University of Technology, China, in 2016, and the M.E. degree in computer technology from Tianjin University, China, in 2019. He is currently pursuing the Ph.D. degree with the College of Intelligence and Computing, Tianjin University. His major research interest is visual intelligence, specifically including multi-camera video collaborative analysis and visual object tracking. He is also interested in solving preventive conservation problems of cultural heritages via artificial intelligence.



Wei Feng (Member, IEEE) received the Ph.D. degree in computer science from the City University of Hong Kong in 2008. From 2008 to 2010, he was a Research Fellow with The Chinese University of Hong Kong and City University of Hong Kong. He is currently a Professor with the School of Computer Science and Technology, College of Computing and Intelligence, Tianjin University, China. His major research interests are active robotic vision and visual intelligence, specifically including active camera relocalization and lighting recurrence, general Markov random fields modeling, energy minimization, active 3D scene perception, SLAM, and generic pattern recognition. Recently, he focuses on solving preventive conservation problems of cultural heritages via computer vision and machine learning. He is an Associate Editor of *Neurocomputing* and *Journal of Ambient Intelligence and Humanized Computing*.

Song Wang (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana Champaign (UIUC), Champaign, IL, USA, in 2002. He was a Research Assistant with Image Formation and Processing Group, Beckman Institute, UIUC, from 1998 to 2002. In 2002, he joined the Department of Computer Science and Engineering, University of South Carolina, Columbia, SC, USA, where he is currently a Professor. His current research interests include computer vision, image processing, and machine learning. He is a member of the IEEE Computer Society. He is currently serving as the Publicity/Web Portal Chair of the Technical Committee of Pattern Analysis and Machine Intelligence of the IEEE Computer Society, an Associate Editor of the IEEE TRANSACTION ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, *Pattern Recognition Letters*, and *Electronics Letters*.