Selective Spatial Regularization by Reinforcement Learned Decision Making for Object Tracking

Qing Guo, Ruize Han, Wei Feng, *Member, IEEE*, Zhihao Chen, and Liang Wan

***Abstract*— Spatial regularization (SR) is known as an effective tool to alleviate the boundary effect of correlation filter (CF), a successful visual object tracking scheme, from which a number of state-of-the-art visual object trackers can be stemmed. Nevertheless, SR highly increases the optimization complexity of CF and its target-driven nature makes spatially-regularized CF trackers may easily lose the occluded targets or the targets surrounded by other similar objects. In this paper, we propose selective spatial regularization (SSR) for CF-tracking scheme. It can achieve not only higher accuracy and robustness, but also higher speed compared with spatially-regularized CF trackers. Specifically, rather than simply relying on foreground information, we extend the objective function of CF tracking scheme to learn the target-context-regularized filters using target-context-driven weight maps. We then formulate the online selection of these weight maps as a decision making problem by a Markov Decision Process (MDP), where the learning of weight map selection is equivalent to policy learning of the MDP that is solved by a reinforcement learning strategy. Moreover, by adding a special state, representing not-updating filters, in the MDP, we can learn when to skip unnecessary or erroneous filter updating, thus accelerating the online tracking. Finally, the proposed SSR is used to equip three popular spatially-regularized CF trackers to significantly boost their tracking accuracy, while achieving much faster online tracking speed. Besides, extensive experiments on five benchmarks validate the effectiveness of SSR.**

***Index Terms*— Visual object tracking, correlation filter, selective spatial regularization, MDP, reinforcement learning.**

I. Introduction

ONLINE object tracking is a fundamental problem of computer vision and has been widely employed in many tasks, such as smart surveillance, human-machine interaction and robotic perception [1]. Given the initial position of a target denoted by a bounding box at the first frame, a tracker aims to predict tight bounding boxes wrapping the target in subsequent video frames. Online learning an effective appearance model of the target is crucial for accurate and reliable visual object tracking [2]. With the rapid development of machine learning methods, such as support vector machine (SVM) [3], [4], subspace learning [5], online multi-instance boosting [6], sparse and compressive reconstruction [7], [8], correlation filter (CF) [9]–[12], and convolutional neural network (CNN) [13], [14], a number of powerful models representing the target appearance are proposed to construct successful visual object trackers.

In particular, CF is a notable tracking scheme that can learn a robust appearance model for the target online, based on which many successful real-time trackers have been constructed [9], [10], [15]–[17]. However, these CF-based trackers suffer from the boundary effect. Being a major inherent drawback of the CF scheme, boundary effect can easily fail the tracking under the condition of cluttered background and fast motion. Spatial regularization (SR) [18]–[22] is then proposed to alleviate this problem by using a spatially-variant weight map to penalize the filter coefficients in the background, which leads to a target-regularized CF model. Although spatially-regularized CF trackers can achieve much higher accuracy than the original CF counterparts, there still exist two limitations that hamper real-world feasibility of these successful trackers.

First, spatially-regularized CF trackers are prone to lose the target when it is severely occluded or surrounded by other objects with similar appearances, since the learned target-regularized model relies only on the foreground and ignores most of the context information. As shown in Fig. 1, when we track a girl who is fully occluded, a typical spatially-regularized CF tracker, i.e., SRDCF [18], produces a false high response at the background region and loses the girl, which further leads to erroneous filter updating and encumbers the re-detection of the girl when she re-appears in subsequent frames. Intuitively, such problem can be alleviated by using more discriminative features [16] and sophisticated learning methods [19]. However, it either requires much larger efforts to select suitable training data or significantly slows down the online tracking. Second, embedding a spatially-variant weight map into the objective function of CF scheme highly increases the complexity of online filter updating. As a result, spatially-regularized CF trackers are usually slow and not suitable for real-time visual tracking tasks.

Manuscript received January 26, 2019; revised September 30, 2019; accepted November 8, 2019. Date of publication November 28, 2019; date of current version January 28, 2020. This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 61671325, Grant 61572354, Grant 61672376, and Grant U1803264. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Guo-Jun Qi. *(Corresponding author: Wei Feng.)*

Q. Guo is with the School of Computer Science and Technology, College of Intelligence and Computing, Tianjin University, Tianjin 300350, China, with the Key Research Center for Surface Monitoring and Analysis of Cultural Relics (SMARC), State Administration of Cultural Heritage, Tianjin 300350, China, with the Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin University, Tianjin 300350, China, and also with the Cyber Security Research Centre, Nanyang Technological University, Singapore 639851.

R. Han and W. Feng are with the School of Computer Science and Technology, College of Intelligence and Computing, Tianjin University, Tianjin 300350, China, 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).

Z. Chen and L. Wan are with the School of Computer Software, College of Intelligence and Computing, Tianjin University, Tianjin 300350, China.

Digital Object Identifier 10.1109/TIP.2019.2955292

1057-7149 © 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

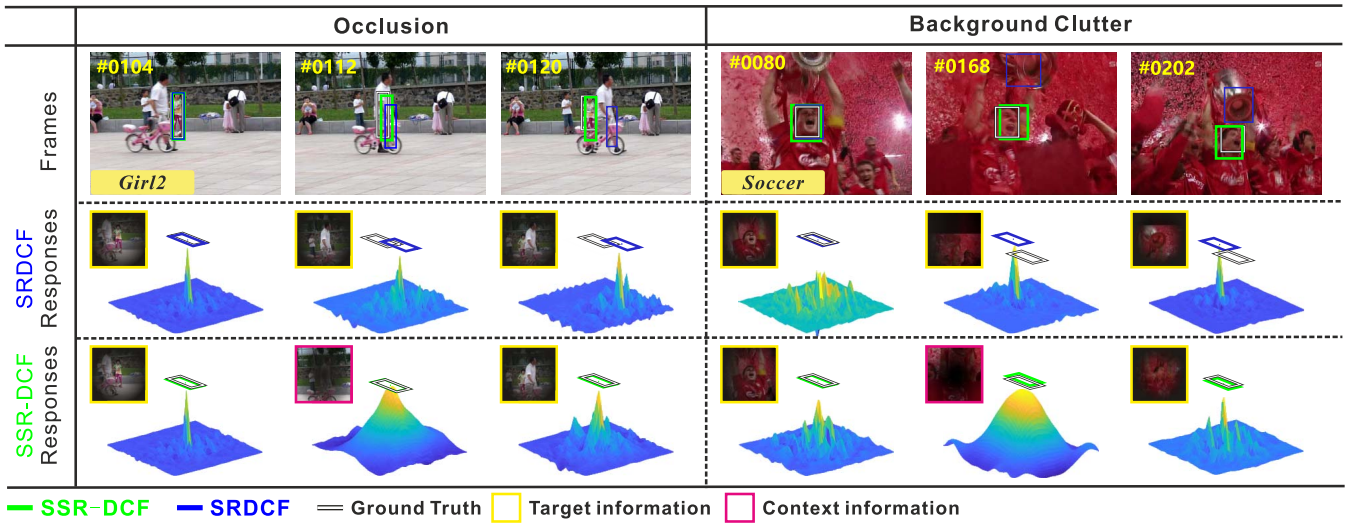


Fig. 1. Comparison between SRDCF [18] and the proposed selective spatial regularization based discriminative CF (SSR-DCF) on the cases of occlusion and background clutter. The bounding box results, their response maps, and the information they relied on are shown. When the targets are fully occluded or surrounded by similar objects, e.g., frame #112 in 'girl2' and frame #168 in 'soccer', SRDCF uses filters that only rely on the target information for localization and produces a false high response at the background, which further leads to erroneous filter updating and encumbers the re-detection of the target at subsequence frames. In contrast, SSR-DCF utilizes the context information when above severe situations happen and gets more discriminative response maps than SRDCF, which further avoids erroneous filter updating and enables to re-detect targets when interferences disappear. Please find intuitive explanations in the text.

To address the first problem, we consider tracking an interested target via its context region when the appearance of itself is unreliable due to the severe occlusion or background clutter. Specifically, when a video contains camera motion, object group motion, or the target moves slowly between neighboring frames, background around the target, i.e., the context, usually has the same or similar motion with the target itself temporarily and their relative positions between neighboring frames are almost the same. Hence, it is possible to track a target through its context when it is severely occluded and surrounded by similar objects. For example, when we try to track a girl, i.e., the first case in Fig. 1, who is fully occluded by a man at the 112 th frame, we can infer her position according to the unoccluded context, e.g., the boy at left side, the grass, or the ground, since their relative positions to the girl in the 112 th frame are almost the same with the ones in neighboring frames, e.g., the 104 th frame. Moreover, for the 'soccer' case where the camera keeps moving, although the target, i.e., the head of the center player, is surrounded by cluttered background with similar appearance, we can locate the player's head by his body or other players, since relative positions between them do not change among neighboring frames. To take the advantages of context, we extend the CF-tracking scheme and use filters learned from context for tracking when the target appearance is unreliable and we denote the filters as *context-regularized filters*. As shown in Fig. 1, compared the typical spatially-regularized CF tracker, i.e., SRDCF, our method using *context-regularized filters* can locate targets accurately under severe occlusion or background clutters. Moreover, *context-regularized filters* restrict the search region at subsequent frames in a close range from the position where the target is lost, and make the tracker able to re-detect the target when it re-appears.

Similar ideas have been introduced by [23]–[25] that use both object and context cues to estimate the target position. However, they construct models with intensities or local features that have not enough discriminative power for reliable visual tracking under complex scenes.

For the second problem, i.e., the complexity, of spatial regularization, we try to skip unnecessary or erroneous filter updating, which can not only speed up the online tracking significantly but also further improve the performance by avoiding the corruption of filters. How to learn such *context-regularized filters* effectively and deciding whether to skip unnecessary or erroneous filer updating are two keys to achieve two things mentioned above. In this paper, we propose *selective spatial regularization* (SSR) for the CF-tracking scheme, which can obtain higher tracking accuracy and robustness, and meanwhile is much faster during the online process. The major contributions of this work are:

- We propose an extended objective function for CF tracking scheme to generate *target-context-regularized filters* by selectively using *target-context-driven weight maps* to regularize the learning of correlation filters.
- We formulate the online selection of different weight maps as a decision making problem via a Markov Decision Process (MDP), where the learning of weight map selection is solved by policy learning of the MDP through a reinforcement learning strategy. Moreover, by adding a special state, representing not-updating filters, in the MDP, we effectively learn when to skip unnecessary or erroneous filter updating, thus to accelerate the online tracking without harming the accuracy.
- We use the proposed SSR to improve three popular spatially-regularized CF trackers, SRDCF [18], CCOT [19] and ECO [20], which validates the feasibility and generality of SSR. Extensive experiments on OTB-2013 [26], OTB-2015 [27], VOT-2016 [28],

TC-128 [29], and LaSOT [30] verify the superiority of our method over state-of-the-art competitors.

II. Related Work

A. Correlation Filter Based Tracking

An early work, i.e., [10], proposes a correlation filter (CF) tracker that can run at 669 fps with a single CPU. In recent years, numerous methods have been proposed to improve the CF tracker by equipping it with the kernel trick [9], multi-kernel learning [31], [32], multiple types of features [33], scale adaption strategies [15], [34], deep features [16], [35], [36], and optical flows [37]. Although these trackers have much higher accuracy than the one in [10], they still suffer from the inherent problem of CF scheme, i.e., the boundary effect [18], [38], which limits the further improvement of these methods.

Spatial regularization (SR) is proposed to learn a target-regularized CF model whose boundary effect is alleviated and leads to three effective trackers, i.e., Spatially-regularized Discriminative CF (SRDCF) [18], Continuous Convolution Operator Tracker (CCOT) [19], and Efficient Convolution Operator (ECO) [20] which significantly increase the tracking accuracy of the original CF tracker. However, the spatially-regularized CF trackers have two limitations. First, they are prone to lose the target when it is severely occluded or surrounded by similar objects, since the learned target-regularized CF model relies only on the foreground and ignores most of the context information. Second, the SR significantly increases the complexity of online filter updating. As a result, SRDCF and CCOT are slow and not suitable for real-time visual tracking tasks.

More recently, [20] proposes Efficient convolution operators for tracking (ECO) and further improves the performance and speed of CCOT by reducing the number of coefficients in filters, managing training samples and updating filters at a fixed interval. Nevertheless, improvements mentioned above cannot handle the first problem of spatially-regularized CF trackers directly. In addition, the skipping strategy for filter updating ignores the online detection results and may lead to erroneous updating. Reference [39] optimizes the SR-embedded CF objective function via alternating direction method of multipliers with a temporal regularization term and constructs a tracker that has much higher accuracy and runs even faster than SRDCF. Reference [40] online calculates spatial reliability map to select target parts suitable for tracking and achieves higher accuracy than SRDCF on VOT datasets [41]. Reference [42] proposes a background-aware CF tracker that learns a target-regularized CF model with real negative samples extracted from background. Although above methods do better than SRDCF and run even faster, they ignore the importance of context information and may easily lose the target when the target-regularized CF model becomes unreliable, which usually happens in cases of severe occlusion or cluttered background. In this paper, we propose selective spatial regularization (SSR)-based CF by using context-regularized filters to track a target when the target-regularized filters are unreliable, which helps track the target accurately under challenging situations.

Two recent works [33], [43] are related to our method. Reference [43] proposes the adaptive spatially-regularized CF (ASRCF) by simultaneously optimizing filters and the spatial regularization weight map. Intuitively, ASRCF generates an adaptive target-driven weight map and focuses on selecting effective target regions for filter learning. In contrast, our work mainly explores how to use effective context regions indicated by a context-driven weight map to get context-regularized filters for accurate tracking even under severe occlusion and background clutter. Reference [33] designs a multi-expert strategy to learn target-regularized CFs from various features and use the divergence of multiple experts to adaptively update filters. In this paper, we adopt a reinforcement learned MDP to guide the online updating of CFs. By adding a status representing not-updating filters, we can learn when to skip unnecessary or erroneous filter updating. More importantly, different from above two works, our method is a universal scheme for spatially-regularized CF trackers and improves three popular methods, i.e., SRDCF, CCOT, and ECO, with much higher performance on OTB, VOT-2016, TC128, and LaSOT benchmarks.

B. Context Assisted Tracking

The strong potential relationship between the target and its context has been studied by many works [23], [24], [44]–[49]. References [23] and [44] propose methods that simultaneously construct target and context appearance models base on color features to help locate the target even if it is occluded. References [24] and [46] use local feature points extracted from target and background regions to construct 'supporters' to decide the target position. However, local feature points from background are easily affected by occlusion, lighting and local geometry variation. Reference [45] then proposes to use region features and local feature points to build the context appearance model and target appearance model respectively to jointly track the target. Reference [47] models CF scheme as a context learning process that jointly uses target and context information for tracking. Although effective in some challenge scenes, above works focus on how to simultaneously use target and context information to track a target accurately. However, under a real-world scene, the context is usually dynamic and affected by complex interferences, e.g., occlusion, motion blur, and deformation. It has to take extra computing sources to find effective context regions for tracking [44], which may slow down the tracking process significantly. An alternative solution is to employ the context information when the target appearance is unreliable. In this paper, we propose a method selectively using target and context appearance models to track a target. The selection is formulated as a decision making problem by a Markov Decision Process (MDP) whose decision policies are learned via reinforcement learning.

C. Handling Occlusion and Background Clutter for Tracking

Occlusion and background clutter usually result in the erroneous updating of target appearance models. An available solution is to skip the updating process when we detect occlusion and background clutter [50]. A lot of works have

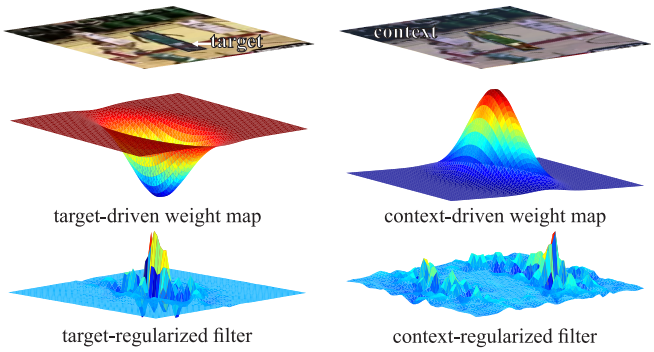


Fig. 2. Examples of target-context-driven weight maps and their corresponding target-context-regularized filters. By regularizing filters with target-context-driven weight maps, the learned filters only have values at target-context regions, respectively.

studied how to handle occlusion for object tracking [50]–[54]. References [50] and [51] explicitly detect pixel-level occluded regions through a background subtraction method. Such methods are only suitable for the videos with fixed cameras and easily fail the tracking in complex scenes. Reference [53] represents a target as a set of parts and determines if a part is occluded or not by comparing it with detected target and background regions. Instead of splitting the target into parts, [54] proposes to compare a detected target with its context and a set of target samples cropped from previous frames to determine if the target is occluded or not. Although effective, such method needs extra storage to save the target samples and is time-consuming due to several times of comparison. Instead of detecting occlusion and background clutter explicitly, we embed them into decision policies of a Markov Decision Process that determines when to skip updating or use context information to locate the target.

III. Background

In this paper, we focus on correlation filter (CF) [9], [10] based single object tracking. Given a set of samples cropped from historical frames according to tracking results of a target, we aim to learn filters to locate the target in a search region cropped from an incoming frame. The samples usually have larger sizes than the bounding boxes of the target and contain both target and context information, as shown in the first row of Fig. 2. To learn filters, we first extract features of those samples and obtain $\mathcal{X} = \{\mathbf{X}_k \in \Re^{M\times N\times D} | k = 1, \ldots, |\mathcal{X}|\}$, where $(M,\ N)$, D and $|\mathcal{X}|$ denote the spatial size, number of dimension and number of samples, respectively. Then, given a regression objective, i.e., $\mathbf{Y} \in \Re^{M\times N}$ being a 2D Gaussian map having high values at the target and low values at the background, we learn filters by minimizing

$$\mathrm{E}(\mathbf{F}, \mathcal{X}) = \frac{1}{2}\sum_{k=1}^{|\mathcal{X}|} \alpha_k \|\mathrm{S}(\mathbf{X}_k) - \mathbf{Y}\|^2 + \frac{\lambda}{2}\sum_{d=1}^{D} \|\mathbf{F}^d\|^2, \quad (1)$$

with

$$\mathrm{S}(\mathbf{X}_k) = \sum_{d=1}^{D} \mathbf{X}_k^d * \mathbf{F}^d \quad (2)$$

where $\mathbf{F} \in \Re^{M\times N\times D}$ is the desired filters, '$*$' denotes the circular convolution, and α_k is the exponentially decaying weight [18]. Minimizing Eq. (1) w.r.t. $\mathbf{F}$ can be efficiently solved in frequency domain where '$*$' becomes element-wise multiplication, which results in beyond real-time trackers [9], [10]. However, CF trackers mainly suffer from two drawbacks. First, they have to use samples and search regions with restricted size, thus struggle in the case of fast motion, since a naive expansion of training samples to include more context information significantly degrades the discriminative power of learned $\mathbf{F}$ [18]. Second, with the circular convolution, CF trackers actually regard the samples in $\mathcal{X}$ and their circularly shifted versions as positive and negative samples, respectively, which leads to the boundary effect and significantly reduces the tracking performance [9], [38]. Spatially regularized discriminative correlation filters (SRDCF) [18] is then proposed to alleviate the two problems by replacing the second term of Eq. (1) with a spatial regularization term

$$\mathrm{E}(\mathbf{F}, \mathcal{X}) = \frac{1}{2}\sum_{k=1}^{|\mathcal{X}|} \alpha_k \|\mathrm{S}(\mathbf{X}_k) - \mathbf{Y}\|^2 + \frac{1}{2}\sum_{d=1}^{D} \|\mathbf{W}_\mathrm{t} \odot \mathbf{F}^d\|^2, \quad (3)$$

where $\mathbf{W}_\mathrm{t} \in \Re^{M\times N}$ is a target-driven weight map that has high penalties at the context and low ones at the target, thus suppresses the coefficients of $\mathbf{F}$ in the context. We have

$$\mathbf{W}_\mathrm{t}(x, y) = \mu + \eta\left(\frac{x - x_0}{w}\right)^2 + \eta\left(\frac{y - y_0}{h}\right)^2, \quad (4)$$

where (x, y), (x_0, y_0) and (w, h) denote the coordinate, target position and target size on $\mathbf{W}_\mathrm{t}$, respectively. In practice, $\mathbf{W}_\mathrm{t}$ is further smoothed by preserving only 10 non-zero frequency coefficients. The spatial regularization term can not only alleviate the boundary effect of CF but also avoid inclusion of substantial amount of context information within the filters. The left subfigure of Fig. 2 shows a case of $\mathbf{W}_\mathrm{t}$ and the learned filters that only has valid values at the target region and can be regarded as target-regularized filters. Although much more accurate than the CF tracker, SRDCF still has two problems. First, it runs very slowly (about 4 fps with HOG features) since the spatial regularization term breaks the element-wise operations of CF in frequency domain when we minimize Eq. (3) w.r.t. $\mathbf{F}$. Second, SRDCF may lose the target when target-regularized filters become unreliable. For example, in Fig. 1, SRDCF misses the two targets when they are occluded or under background clutter. To overcome the two problems, we propose selective spatial regularization (SSR) that not only speeds up online process of spatially-regularized CF trackers but also keeps tracking under severe situations, e.g., occlusion and background clutter.

IV. Our Approach

In the following, we first introduce selective spatial regularization (SSR) for CF that regularizes filters by selectively using weight maps driven by the target and its context. We then formulate the online selection of weight maps as a decision making problem by a Markov Decision Process (MDP) and detail the way of training decision policies via

reinforcement learning. We finally present the implementation details of the proposed method.

A. Selective Spatial Regularization (SSR)

When we track an interested target, instead of using filters that mainly related to target appearance [18], we propose to use two kinds of filters to locate the target, i.e., *target-regularized filters* $\mathbf{F}_\mathrm{t} \in \Re^{M\times N\times D}$, and *context-regularized filters* $\mathbf{F}_\mathrm{c} \in \Re^{M\times N\times D}$ that take charge of the target and context appearance models, respectively. Specifically, given representative samples, i.e., $\mathcal{X} = \{\mathbf{X}_k \in \Re^{M\times N\times D} | k = 1, \ldots, |\mathcal{X}|\}$, that center at the target and are collected from historical frames, we separate each sample to two regions, i.e., target region and its context region, as shown in the first row of Fig. 2, and learn $\mathbf{F}_\mathrm{t}$ and $\mathbf{F}_\mathrm{c}$ relying on the two regions, respectively. To this end, we define an objective function to selectively learn $\mathbf{F}_\mathrm{t}$ and $\mathbf{F}_\mathrm{c}$

$$\mathrm{E}(\mathbf{F}, \mathcal{X}, s) = \frac{1}{2}\sum_{k=1}^{|\mathcal{X}|}\alpha_k \|\mathrm{S}(\mathbf{X}_k) - \mathbf{Y}\|^2 + \frac{1}{2}\sum_{d=1}^{D}\|\mathrm{W}(s)\odot\mathbf{F}^d\|^2, \tag{5}$$

with

$$\mathrm{W}(s) = \begin{cases} \mathbf{W}_\mathrm{t}, & \text{if } s = 1 \\ \mathbf{W}_\mathrm{c}, & \text{if } s = -1 \\ \mathbf{W}_\mathrm{n}, & \text{if } s = 0, \end{cases} \tag{6}$$

where $\mathbf{W}_\mathrm{c} = \max(\mathbf{W}_\mathrm{t}) - \mathbf{W}_\mathrm{t} + \min(\mathbf{W}_\mathrm{t})$ with $\mathbf{W}_\mathrm{t}$ defined in Eq. (4). We set $\mathbf{W}_\mathrm{n} = +\mathrm{Inf}$ that forces all coefficients of filters to be zero, which means that it is unnecessary to learn filters. $s \in \{0, 1, -1\}$ is a selector to determine which spatial weight map should be used to regularize the filters. Intuitively, when $\mathrm{W}(s) = \mathbf{W}_\mathrm{c}$, the target region is assigned larger penalties than the context region, as shown in the second row of Fig. 2, which makes the learned filters have zero values on the target region and only rely on the context information after optimizing Eq. (8), and we obtain the $\mathbf{F}_\mathrm{c}$. Similarly, we can get $\mathbf{F}_\mathrm{t}$ by setting $\mathrm{W}(s) = \mathbf{W}_\mathrm{t}$. We denote $\mathbf{W}_\mathrm{t}$ and $\mathbf{W}_\mathrm{c}$ as *target-driven and context-driven weight maps*, respectively, and have

$$\mathbf{F}_\mathrm{t} = \arg\min_{\mathbf{F}} \mathrm{E}(\mathbf{F}, \mathcal{X}_\mathrm{t}, 1), \tag{7}$$

$$\mathbf{F}_\mathrm{c} = \arg\min_{\mathbf{F}} \mathrm{E}(\mathbf{F}, \mathcal{X}_\mathrm{c}, -1). \tag{8}$$

Fig. 2 shows examples of learned $\mathbf{F}_\mathrm{t}$ and $\mathbf{F}_\mathrm{c}$ according to $\mathbf{W}_\mathrm{t}$ and $\mathbf{W}_\mathrm{c}$, respectively. The target-context-regularized filters, i.e., $\mathbf{F}_\mathrm{t}$ and $\mathbf{F}_\mathrm{c}$, only have valid values at the target and context regions, respectively.

Based on Eq. (7) and (8), we summarize SSR-based CF tracking as follows:

i. At the first frame, we learn target-regularized filters by Eq. (7) with $\mathcal{X}_\mathrm{t} = \{\mathbf{X}_1\}$ where $\mathbf{X}_1$ is the feature of a training sample cropped from the first frame. We then set the frame index as $\tau = 2$.
ii. Loading frame τ, we crop a search region centering at the target position of frame $\tau - 1$ and extract its feature, i.e., $\mathbf{Z}_\tau \in \Re^{M\times N\times D}$. We then obtain a response map with $\mathbf{C}_\tau = \sum_{d=1}^{D}\mathbf{Z}_\tau^d * \mathbf{F}_\mathrm{t}^d$.

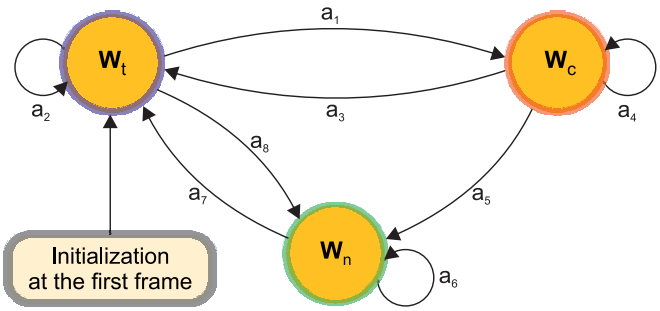


Fig. 3. State transition of MDP for online selecting weight maps. $\mathrm{a}_i \in \mathcal{A}$ is an action to transform one state to another. At the first frame, we initialize $s = 1$ to learn target-regularized filters.

iii. Selecting a spatial weight map from $\{\mathbf{W}_\mathrm{t}, \mathbf{W}_\mathrm{c}, \mathbf{W}_\mathrm{n}\}$ according to $\mathbf{C}_\tau$ and the MDP introduced in Section IV-B. If $\mathbf{W}_\mathrm{t}$ is selected, we locate the target according to position of the maximum of $\mathbf{C}_\tau$ and crop a sample centering at the position whose feature, i.e., $\mathbf{X}_\tau$, is added to $\mathcal{X}_\mathrm{t}$. We then update $\mathbf{F}_\mathrm{t}$ by Eq. (7). If $\mathbf{W}_\mathrm{c}$ is selected, $\mathbf{F}_\mathrm{t}$ is regarded unreliable due to interferences, e.g., occlusion and background clutter. We then set $\mathcal{X}_\mathrm{c} = \{\mathbf{X}_{\tau-1}\}$ and learn $\mathbf{F}_\mathrm{c}$ by Eq. (8). We can obtain a new response map by $\mathbf{C}'_\tau = \sum_{d=1}^{D}\mathbf{Z}_\tau^d * \mathbf{F}_\mathrm{c}^d$ whose maximum indicates the target position. If $\mathbf{W}_\mathrm{n}$ is selected, we skip the updating process at frame τ to avoid unnecessary or error updating of x$\mathbf{F}_\mathrm{t}$ and use the maximum of $\mathbf{C}_\tau$ to locate the target.
iv. Setting $\tau = \tau + 1$ and going to step ii to continue tracking.

By selectively using spatial weight maps to obtain target-context-regularized filters or skip updating, we can track a target with high online speed even if the target is under severe interference. As shown in Fig. 1, when a target is occluded or surrounded by similar objects, i.e., # 112 and # 168 in the cases of occlusion and background clutter, respectively, SRDCF only using $\mathbf{W}_\mathrm{t}$ to learn $\mathbf{F}_\mathrm{t}$, gets a high response on background and misses the target at subsequent frames. In contrast, by assuming the target and its context have similar motion between neighbor frames, $\mathbf{F}_\mathrm{c}$ learned by SSR-DCF with $\mathbf{W}_\mathrm{c}$ can still detect the target at # 112 and # 168 and helps re-detection at subsequent frames. However, it is important but difficult to determine which spatial weight map should be selected during online tracking. In Section IV-B, we regard the selection of weight maps as a Markov Decision Process (MDP) that makes a decision for selection with a learned policy.

B. Online Selection by Markov Decision Process (MDP)

We formulate the selection of weight maps in Eq. (6), as a Markov Decision Process (MDP) that consists of a tuple $(\mathcal{S}, \mathcal{A}, \mathrm{T}(\cdot), \mathrm{R}(\cdot))$ where $\mathcal{S}$, $\mathcal{A}$, $\mathrm{T}(\cdot)$ and $\mathrm{R}(\cdot)$ denote the set of states and actions, state transition function and real-valued reward function. We detail them in following:

1) MDP Formulation: We set the state space of weight map $\mathbf{W}$ as $\mathcal{S} = \{\mathbf{W}_\mathrm{t}, \mathbf{W}_\mathrm{c}, \mathbf{W}_\mathrm{n}\}$, and define eight transitions to change the state of $\mathbf{W}$, which corresponds to eight actions, i.e., $\mathcal{A} = \{\mathrm{a}_i | i \in [1, 8]\}$, in Fig. 3. Given a state of $\mathbf{W}$ and

an action, the transition function converts $\mathbf{W}$ to another state. For example, the conversion of $\mathbf{W}$ from $\mathbf{W}_\mathrm{t}$ to $\mathbf{W}_\mathrm{c}$ by a_1 can be represented as $\mathrm{T}(\mathbf{W}_\mathrm{t}, \mathrm{a}_1) = \mathbf{W}_\mathrm{c}$. We summary the available transitions in Fig. 3.

At the first frame, we initialize $\mathbf{W}$ as $\mathbf{W}_\mathrm{t}$ and learn target-regularized filters, i.e., $\mathbf{F}_\mathrm{t}$, by Eq. (7). During online tracking, if $\mathbf{F}_\mathrm{t}$ is unreliable, which is caused by some interferences, e.g., occlusion and background clutter, we do $\mathbf{W} = \mathrm{T}(\mathbf{W}_\mathrm{t}, \mathrm{a}_1)$ to learn context-regularized filters, i.e., $\mathbf{F}_\mathrm{c}$, to detect a target, if $\mathbf{F}_\mathrm{t}$ is discriminative enough to separate target from background, which implies the target does not change significantly, we set $\mathbf{W} = \mathrm{T}(\mathbf{W}_\mathrm{t}, \mathrm{a}_8)$ to skip unnecessary updating of $\mathbf{F}_\mathrm{t}$ to speed up tracking. Otherwise, we keep $\mathbf{W} = \mathrm{T}(\mathbf{W}_\mathrm{t}, \mathrm{a}_2)$. When $\mathbf{W} = \mathbf{W}_\mathrm{c}$, we convert it back to $\mathbf{W}_\mathrm{t}$ by $\mathrm{T}(\mathbf{W}_\mathrm{c}, \mathrm{a}_3)$ if $\mathbf{F}_\mathrm{t}$ is able to split the target from background again. However, if $\mathbf{W} = \mathrm{T}(\mathbf{W}_\mathrm{c}, \mathrm{a}_4)$ keeps for a long time (over 5 frames), we assume the target is lost and do $\mathbf{W} = \mathrm{T}(\mathbf{W}_\mathrm{c}, \mathrm{a}_5)$ to avoid error updating of filters. During this state, we re-detect the target with latest updated $\mathbf{F}_\mathrm{t}$. Once the target is re-detected, we set $\mathbf{W} = \mathrm{T}(\mathbf{W}_\mathrm{n}, \mathrm{a}_7)$ to update target-regularized filters.

To make the transitions metioned above work as expected during online tracking, a policy should be learned for each state of $\mathbf{W}$ to decide which action should be taken, which is equivalent to detecting if interferences, e.g., occlusion and background clutter, happen and make $\mathbf{F}_\mathrm{t}$ unreliable.

2) Policies of MDP: Given the state of $\mathbf{W}$, a policy determines which action to take, which is equivalent to selecting different weight maps according to the reliability of $\mathbf{F}_\mathrm{t}$.

For the policy at the states $\mathbf{W}_\mathrm{t}$ and $\mathbf{W}_\mathrm{c}$, MDP makes decision between $\mathbf{W}_\mathrm{t}$, $\mathbf{W}_\mathrm{c}$, and $\mathbf{W}_\mathrm{n}$ to regularize filters, which can be regarded as a classification problem. We thus train three binary Support Vector Machines (SVMs) for the three weight maps, respectively, by taking the response map $\mathbf{C}_\tau = \sum_{d=1}^{D} \mathbf{Z}_\tau^d * \mathbf{F}_\mathrm{t}^d$ as input and make a decision via Eq. (6) with

$$s = \arg \min_{l \in \{-1,0,1\}} \omega_l^\top \phi(\mathbf{C}_\tau) + \mathbf{b}_l, \tag{9}$$

where $l = -1, 0$ and 1 is to set weight map as $\mathbf{W}_\mathrm{c}$, $\mathbf{W}_\mathrm{n}$ and $\mathbf{W}_\mathrm{t}$, respectively, and $\phi(\cdot)$ calculates the features of $\mathbf{C}_\tau$ and will be introduced in Section IV-B.3. Similarly, for the policy at the state of $\mathbf{W}_\mathrm{n}$, we just need to select between $\mathbf{W}_\mathrm{t}$ and $\mathbf{W}_\mathrm{n}$ by solving $s = \arg\min_{l \in \{0,1\}} \omega_l^\top \phi(\mathbf{C}_\tau) + \mathbf{b}_l$.

We can learn ω_l and $\mathbf{b}_l$ by collecting $\mathbf{C}_\tau$ and the ground truth transitions of weight map as training samples, which is equivalent to maximizing following reward function

$$\mathrm{R}(\mathbf{W}, \mathrm{a}) = \sum_{l \in \{-1,0,1\}} \mathrm{y}_l(\mathrm{a})(\omega_l^\top \phi(\mathbf{C}_\tau) + \mathbf{b}_l), \tag{10}$$

where $\mathbf{W} \in \mathcal{S}$, $\mathrm{a} \in \mathcal{A}$, and $\mathrm{y}_l(\mathrm{a}) \in \{\text{false}, \text{true}\}$. We set $\mathrm{y}_l(\mathrm{a}) =$ true if an action a can do the same weight map transition with the one indicated by l. Otherwise, $\mathrm{y}_l(\mathrm{a}) =$ false. For example, when the activate weight map is $\mathbf{W}_\mathrm{t}$, the MDP takes action a_1 to transform it to $\mathbf{W}_\mathrm{c}$. Then, we should set $\mathrm{y}_{-1}(\mathrm{a}_1) =$ true, $\mathrm{y}_{-1}(\mathrm{a}_2) =$ false and $\mathrm{y}_{-1}(\mathrm{a}_8) =$ false. Given the ground truth actions during online tracking, we can maximize Eq. (10) to learn three SVMs. In practice, we realize the processs in a reinforcement learning fashion on a synthetic video, which will be introduced in Section IV-C.

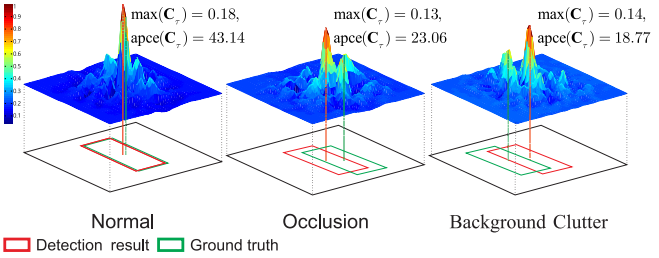


Fig. 4. Three examples of response maps, i.e., $\mathbf{C}_\tau$, under situations of normal, occlusion and background clutter, respectively. The red rectangles show the detect results according to maximum of $\mathbf{C}_\tau$. The green rectangles show the ground truth.

3) Features $\phi(\cdot)$: The reliability of target-regularized filters, i.e., $\mathbf{F}_\mathrm{t}$, is crucial to the state transition of MDP and can be represented through response map that is generated by using $\mathbf{F}_\mathrm{t}$ to detect the target, i.e., $\mathbf{C}_\tau = \sum_{d=1}^{D} \mathbf{Z}_k^d * \mathbf{F}^d$. Clearly, if $\mathbf{C}_\tau$ has very high values at the position of the target and low values at the background, $\mathbf{F}_\mathrm{t}$ is reliable and can separate target from the background very well. Otherwise, $\mathbf{F}_\mathrm{t}$ is unreliable. Hence, we extract information from $\mathbf{C}_\tau$ via ϕ to represent the reliability of $\mathbf{F}_\mathrm{t}$. We define $\phi(\mathbf{C}_\tau) = [\max(\mathbf{C}_\tau), \mathrm{apce}(\mathbf{C}_\tau)]$ where $\max(\mathbf{C}_\tau)$ outputs the maximum of $\mathbf{C}_\tau$. $\mathrm{apce}(\mathbf{C}_\tau)$ represents the average peak-to-correlation energy (APCE) [55] defined as $\mathrm{apce}(\mathbf{C}_\tau) = \frac{\|\max(\mathbf{C}_\tau) - \min(\mathbf{C}_\tau)\|^2}{\mathrm{avg}(\mathbf{C}_\tau - \min(\mathbf{C}_\tau))}$. Fig. 4 shows three examples of response maps and their maximum and APCE values in the situations of normal, occlusion and background clutter. When a target is occluded or surrounded by similar objects, the response maps lead to erroneous detection results with the $\max(\mathbf{C}_\tau)$ and $\mathrm{apce}(\mathbf{C}_\tau)$ reducing significantly. Hence, we can use the maximum and APEC of the response map to determine if $\mathbf{F}_\mathrm{t}$ is reliabile.

C. Reinforcement Learning for MDP

Since the value range of response map differ from each video, it is difficult to learn unified weight ω_l and bias $\mathbf{b}_l$ in Eq. (10) for all videos offline. Thus, we learn ω_l and $\mathbf{b}_l$ through reinforcement learning on a synthetic video generated from the first frame of a test video using data augmentation.

1) Synthetic Video Generation: Given the first frame $\mathbf{I}_1$ of a testing video $\mathcal{V} = \{\mathbf{I}_\tau\}_1^{|\mathcal{V}|}$, we generate a synthetic video $\tilde{\mathcal{V}} = \{\tilde{\mathbf{I}}_\tau\}_1^{N_v}$ by circularly shifting $\mathbf{I}_1$, adding artificial occlusion and motion blur to simulate target translation, occlusion and appearance variation, respectively. N_v is the specified video length. To make the transition between states of MDP happen frequently, we implement image shifting and artificial occlusion alternately, and add motion blur with random kernel size and direction at each frame. Specifically, when generating frame τ, i.e., $\tilde{\mathbf{I}}_\tau$, we just circularly shift $\mathbf{I}_1$ with random values on horizontal and vertical coordinates, respectively. Then, a randomized motion blur is added to $\mathbf{I}_1$. To generate $\tilde{\mathbf{I}}_{\tau+1}$, we add occlusion to $\tilde{\mathbf{I}}_\tau$ by replacing part of the target in $\tilde{\mathbf{I}}_\tau$ with a region randomly cropped from the background of the first image. An example of a synthetic video is shown in Fig. 5.

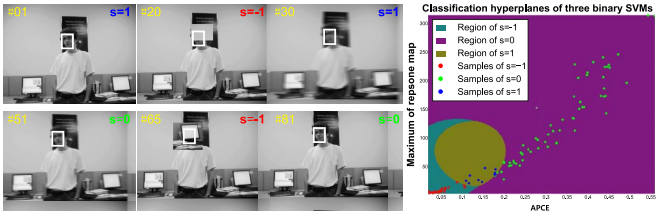


Fig. 5. An example of generated synthetic video and learned binary SVMs. The left subfigure shows the tracking results via white bounding boxes and selectors s at the top right corner. The right subfigure shows the sample distribution of the synthetic video and the classification hyperplanes of three SVMs which separate the sample space into three regions. Each sample corresponds to $\phi(\mathbf{C}_\tau) = [\max(\mathbf{C}_\tau), \text{apce}(\mathbf{C}_\tau)]$ calculated from frame τ and is colored by red, blue and green, respectively, according to selector.

2) Reinforcement Learning of ω_l and $\mathbf{b}_l$: With $\tilde{\mathcal{V}} = \{\tilde{\mathbf{I}}_\tau\}_1^{N_v}$, we aim to learn MDP policy to track the target on all frames of $\tilde{\mathcal{V}}$ accurately. We initialize $\omega_l = 0.15 + \Delta\omega$ and $\mathbf{b}_l = 0.05 + \Delta\mathbf{b}$ where $\Delta\omega$ and $\Delta\mathbf{b}$ are randomly selected from [-0.1,0.1] and [-0.01, 0.01], respectively. The 0.15 and 0.05 are empirical values and help train better policies for the MDP. The training set $\mathcal{T}_l$ is initialized as empty. With such initial policy, we track the target through the process introduced in Section IV-A. At frame τ, we first obtain the ground truth action by comparing the localization precisions of using target-regularized filters, i.e., $\mathbf{F}_\text{t}$, context-regularized filters, i.e., $\mathbf{F}_\text{c}$, and updated $\mathbf{F}_\text{t}$ to detect target, respectively, where the updated $\mathbf{F}_\text{t}$ is to update $\mathbf{F}_\text{t}$ with sample cropped from frame τ. Specifically, if $\mathbf{F}_\text{t}$ misses the target while $\mathbf{F}_\text{c}$ detects it accurately, the action that transfers $\mathbf{W}$ to $\mathbf{W}_\text{c}$ should be taken. If $\mathbf{F}_\text{t}$ could obtain similar or better precision than its updated version or both $\mathbf{F}_\text{t}$ and $\mathbf{F}_\text{c}$ miss the target, the action that transfers $\mathbf{W}$ to $\mathbf{W}_\text{n}$ should be taken to avoid unnecessary or error updating of $\mathbf{F}_\text{t}$. Otherwise, we keep $\mathbf{W} = \mathbf{W}_\text{t}$. If Eq. (9) cannot generate the same decision with the ground truth action, we add the corresponding response map $\phi(\mathbf{C}_\tau)$ and ground truth action a_τ^gt into $\mathcal{T}_l$ and learn the ω_l and $\mathbf{b}_l$ by solving a soft-margin optimization problem

$$\min_{\omega_l, \mathbf{b}_l, \xi_i} \frac{1}{2}\|\omega_l\|^2 + \alpha \sum_{\tau=1}^{|\mathcal{T}_l|} \xi_i$$
$$\text{s.t. } \text{y}_l(\text{a}_\tau^\text{gt})(\omega_l^\top \phi(\mathbf{C}_\tau) + \mathbf{b}_l) \geq 1 - \xi_\tau, \quad \xi_\tau \geq 0, \ \forall\tau \qquad (11)$$

where ξ_i with $i = 1, \ldots, |\mathcal{T}_l|$ are the slack variables. We do this for all frames and keep iterating until all targets are successfully tracked. Please find the detailed process of learning method in Algorithm 1. In Fig. 5, we show an example of learned SVMs which separate the sample space into three regions corresponds to $s = -1, 0, 1$, respectively. Clearly, the region of $s = 1$ takes the largest proportion, which means that a large part of learning $\mathbf{F}_\text{t}$ or $\mathbf{F}_\text{c}$ can be avoided to speed up tracking significantly.

D. Implementation Details

Selective spatial regularization (SSR) is a universal scheme which helps improve various spatially-regularized CF trackers. In this paper, we validate SSR by equipping it to three popular spatially-regularized CF trackers, i.e., SRDCF [18], CCOT [19], and ECO [20]. We inherit their parameter setup. Specifically, for SRDCF, we use HoG, gray and color name as features and solve Eq. (7) and (8) via Gaussian-Seidel method with 4 iterations. We calculate $\mathbf{W}_\text{t}$ with $\mu = 0.1$ and $\eta = 12$. To realize scale adaption, we perform correlation on 7 scales with scale step being 1.01. α_k is set as exponentially decaying weights and corresponds to a fixed learning rate 0.025. For CCOT, we use VGG-M as features and solve Eq. (7) and (8) via the Conjugate Gradient method with 5 iterations. We generate $\mathbf{W}_\text{t}$ with $\mu = 10^{-4}$ and $\eta = 10^{-2}$ and detect target on 5 scales with scale step being 1.02. The learning rate is set as 0.0075. For ECO, we adopt the features of VGG-M and HoG and learn online via the Conjugate Gradient method with 5 iterations and the learning rate is 0.025. The parameters for $\mathbf{W}_\text{t}$, i.e., μ and η, are set as 10^{-4} and 10^{-2}. Note, ECO uses a sparse updating strategy and updates filters every 5 frames, which is not suitable for the SSR-based scheme. We thus remove this strategy and update filters according to the status changes of the MDP. We denote improved SRDCF, CCOT, and ECO as SSR-DCF, SSR-CCOT, and SSR-ECO, respectively.

For synthetic video generation, we set $N_v = 20$ and add motion blur by randomly selecting kernel size from 5 to 11 and direction from 0 to 5 degrees.

Algorithm 1: Reinforcement Learning for SSRCF

Input: A synthetic video: $\tilde{\mathcal{V}} = \{\tilde{\mathbf{I}}_\tau\}_1^{N_v}$; Ground truth target location: $\mathcal{P}^\text{gt} = \{\mathbf{p}_\tau^\text{gt}\}_{\tau=1}^{N_v}$
Output: Weight ω_l and bias $\mathbf{b}_l$ in Eq. (10)
Random initialize ω_l and $\mathbf{b}_l$;
Initialize $\mathbf{F}_\text{t}$ by Eq. (7);
for $1 < \tau \leq N_v$ **do**
  Crop a search region and extract its feature $\mathbf{Z}_\tau^d$ from frame τ;
  Get target location $\mathbf{p}_\tau$ via $\mathbf{C}_\tau = \sum_{d=1}^{D} \mathbf{Z}_\tau^d * \mathbf{F}_\text{t}^d$;
  Calcuate center location error (CLE) via $\mathbf{p}_\tau$ and $\mathbf{p}_\tau^\text{gt}$;
  Select an action a by our MDP;
  Calculate ground truth action a_τ^gt according to CLE;
  if $\text{a} \neq \text{a}_\tau^\text{gt}$ **then**
    Add $(\phi(\mathbf{C}_\tau), \text{y}_l(\text{a}_\tau^\text{gt}))$ to training set $\mathcal{T}_l$;
    Get ω_l and $\mathbf{b}_l$ by solving Eq. (11);
    Break;
  else
    According to the action a, we select a weight map to learn filters by Eq. (7) or (8);
  end
end

V. Experimental Results

A. Setup

1) Datasets and Metrics: We evaluate our method and baseline trackers on OTB-2013 [26], OTB-2015 [27], TC-128 [29], LaSOT [30], and VOT-2016 dataset [28]. OTB-2015 contains 98 sequences with 100 targets. OTB-2013 is a subset of OTB-2015 and contains 50 sequences with 51 targets. The intersection-over-union (IoU) and center location error (CLE) between tracking results and ground truth bounding boxes are used to evaluate a tracker quantitatively. By setting

thresholds for IoU and CLE, we get average success rate and precision over all frames, respectively. With a range of thresholds, we finally obtain the success plots of IoU and precision plots for CLE. The area under curve (AUC) of success plots and the precision of CLE at 20 pixels are regarded as the final metrics for each tracker. Besides, OTB dataset contains 11 subsets according to 11 interference attributes, i.e., illumination variation (IV), scale variation (SV), in-plane rotation (IPR), out-plane rotation (OPR), deformation (DEF), occlusion (OCC), motion blur (MB), fast motion (FM), background clutter (BC), out-of-view (OV) and low resolution (LR) [27].

TC-128 [29] has 128 color sequences and is used to explore the ability of trackers to encode color information. LaSOT [30] is a recently proposed large-scale video dataset for single object tracking and consists of 1400 lone-term sequences. Here, we report the results on its testing subset that contains 280 sequences with 690K frames. Following the setup of OTB datasets, TC-128 and LaSOT also adopt the AUC of success plot and CLE at 20 pixels as final evaluation metrics.

We also take the VOT-2016 [28] as an evaluation dataset that contains 60 videos. The expected average overlap (EAO) is regarded as the metric and considers both the accuracy and robustness of trackers [56].

2) Baseline Trackers: For OTB, TC-128, and LaSOT datasets, we choose two groups of baseline trackers. The first one is CF trackers including HCF [16], SRDCF [18], Staple [17], HDT [57], CCOT [19], ECO [20], CSRDCF [40], BACF [42] and CFNet [36]. The second group includes trackers base on other frameworks, such as DLSSVM [58], SiamFC [59], LMCF [55] and DSiam [14]. Among these trackers, HCF, CFNet, SiamFC, CCOT, ECO, HDT and DSiam use deep features. Note, since the source code of LMCF is not available, we only show its results on the OTB datasets. For VOT-2016 dataset, we compare our trackers with the five best trackers, i.e., CCOT, TCNN [60], SSAT [28], MLDF [28] and Staple, and other five CF trackers, i.e., ECO, HCF, CSRDCF, BACF and SRDCF.

B. Comparison of Benchmark Datasets

1) OTB Dataset Evaluation Results: We show the evaluation results on OTB-2013 and OTB-2015 in Fig. 6 and Fig. 7. With our selective spatial regularization (SSR), the improved spatially-regularized CF trackers, i.e., SSR-DCF, SSR-CCOT, and SSR-ECO, achieve the highest accuracy among all compared trackers. Specifically, for OTB-2013 results, SSR-DCF obtains 8.9% relative improvement over SRDCF on both AUC score and CLE precision at 20 pixels. The relative improvements of SSR-CCOT and SSR-ECO over CCOT and ECO are smaller than those of SSR-DCF over SRDCF, i.e., 1.9% and 2.5%, according to the AUC scores, since CCOT and ECO use deep features that help address interference and limit the ability of SSR in improving performance. Compared with other state-of-the-art CF trackers, e.g., BACF and CSRDCF, SSR-ECO gets 9.0% and 21.8% relative improvements according to their AUC scores. In terms of other tracking frameworks, e.g., DSiam, LMCF, DLSSVM and SiamFC,

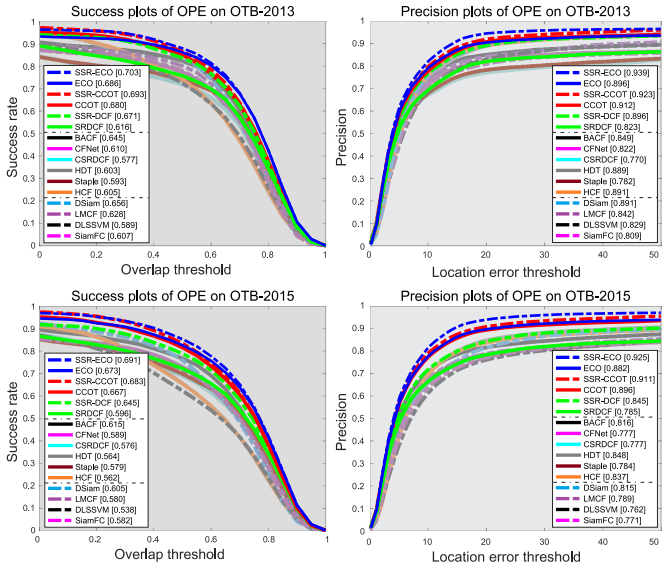


Fig. 6. Comparison results on OTB-2013 [26] and OTB-2015 [27]. The legend of each tracker shows the AUC score of success plots and precision at 20 pixels of precision plots.

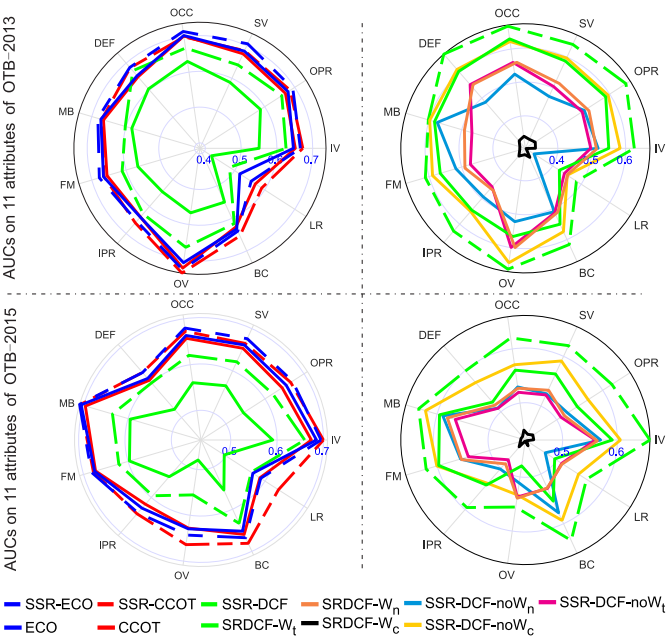


Fig. 7. Average AUC scores on 11 subsets of OTB-2013 [26] and OTB-2015 [27].

SSR-DCF using hand-crafted features can still achieve 2.3%, 6.8%, 13.9%, and 14.2% relative improvements on AUC scores. We also compare SSR-DCF, SRDCF, SSR-CCOT, CCOT, SSR-ECO, and ECO on the 11 subsets of OTB-2013. As shown in the left column of Fig. 7, SSR-based methods outperform their original versions on all 11 subsets and achieve high accuracy gain on OV, IV, IPR, OPR, DEF, BC and OCC. SSR-ECO and SSR-CCOT also get larger AUC scores than ECO and CCOT on all subsets. However, the accuracy gains are much smaller than SSR-DCF over SRDCF, since deep features they used help overcome various interference.

According to the results on OTB-2015 that contains more challenge sequences than OTB-2013, we observe that accuracy gains of SSR-based methods are usually larger than that on OTB-2013. Specifically, SSR-DCF, SSR-CCOT, and SSR-ECO obtain 14.6%, 2.4%, and 2.7% relative improvements over SRDCF according to their AUC scores, which are larger than the gains on OTB-2013. It shows that SSR helps SRDCF, CCOT, and ECO handle more challenging situations. Meanwhile, SSR-DCF obtains better relative improvements, i.e., 6.6%, 11.2% and 19.9%, over DSiam, LMCF, and DLSSVM on OTB-2015 than those on OTB-2013. Considering the results on 11 subsets of OTB-2015, we also see that SSR-DCF, SSR-CCOT, and SSR-ECO outperform SRDCF, CCOT, and ECO on all subsets.

Besides quantitative analysis, we compare the visualization results of ten trackers in Fig. 9. In sequences of 'Box' and 'Lemming', SSR-ECO, SSR-CCOT, and SSR-DCF can handle the severe occlusion properly and locate targets accurately. However, state-of-the-art CF trackers, e.g., CSRDCF, BACF, HDT, CFNet and SRDCF, easily fail to track when the targets are severely occluded by background. We observe similar results in 'Girl2', 'Human3' and 'Kitesurf'. Particularly, in sequence 'Human3', except SSR-ECO, SSR-DCF and ECO, all trackers miss the target. Although SSR-CCOT fails at frame #1617, the target is still within the search region of SSR-CCOT and can be re-detect at following frames. In sequence 'Singer2', the singer is within background clutter introduced by audios and illumination variation caused by the screen. Only SSR-ECO, SSR-CCOT, SSR-DCF and BACF can track the singer accurately and adapt its scale variation.

2) TC-128 Dataset Evaluation Results: We show the evaluation results on TC-128 in the first row of Fig. 8. Clearly, with our SSR, the AUC scores of SRDCF, CCOT, and ECO achieve 5.8%, 1.6%, and 1.7% relative improvements. For CLE precision results, SSR-DCF, SSR-CCOT, and SSR-ECO achieve 5.0%, 2.1%, and 2.7% relative improvements over their original versions, respectively. More importantly, our three trackers outperform all compared methods according to the AUC scores and it demonstrates the effectiveness of our SSR for handling color sequences.

3) LaSOT Dataset Evaluation Results: We further validate the proposed method on the LaSOT dataset. Note, since LaSOT is large-scale and CCOT is too slow to be evaluated in a limited time, we only report results of SSR-DCF and SSR-ECO. As shown in the second row of Fig. 8, the SSR improves both SRDCF and ECO with 11.0% and 4.0% relative improvements on AUC scores, respectively. In terms of the CLE precision, the relative improvements are 6.6% and 13.7%, which demonstrates the proposed SSR not only improves spatially-regularized CF trackers on short-term sequences, i.e., OTB and TC-128 videos but also helps handle challenges, e.g., long-term occlusion, etc., from long-term sequences. Compared with state-of-the-art trackers, SSR-ECO achieves the largest AUC score and SSR-DCF also gets better results than recent CF trackers, e.g., BACF, CSRDCF, HDT, Staple, and HCF, while being slightly worse than CFNet. SSR-ECO gets the second best result according to the CLE precision and outperforms CF trackers using deep features,

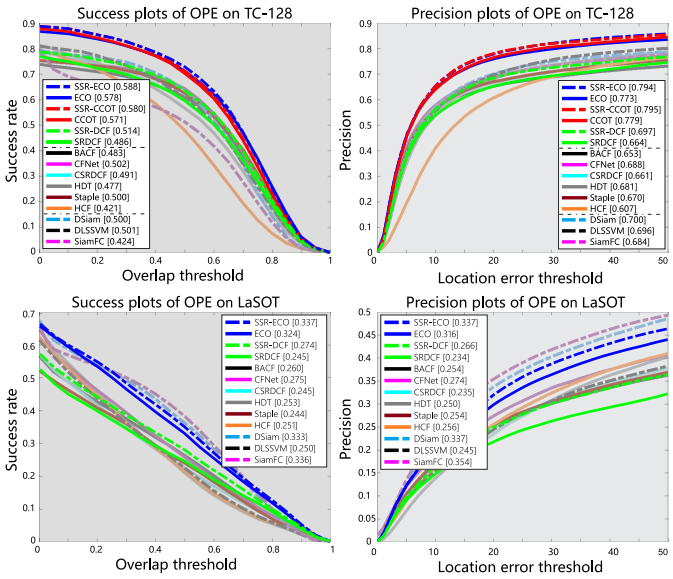


Fig. 8. Comparison results on TC-128 [29] and LaSOT-2015 [30]. The legend of each tracker shows the AUC score of success plot and precision at 20 pixels of precision plot.

i.e., CFNet, HDT, and HCF, with 23.0%, 34.8%, and 31.6% relative improvement, respectively.

4) VOT-2016 Dataset Evaluation Results: As shown in Table I, we see that SSR-ECO and SSR-CCOT get much higher EAO than their original versions and are the best two trackers among all compared methods. Although having much lower EAO than SSR-ECO, SSR-DCF also has larger EAO and is more robust than SRDCF. In terms of robustness, SSR-ECO, SSR-CCOT, and SSR-DCF obtain better results than ECO, CCOT and SRDCF, respectively. Moreover, SSR-ECO and SSR-CCOT have the best robustness among all methods.

C. Detailed Analysis

1) Ablation Study: To analyze the contribution of different weight maps for accurate tracking, we remove $\mathbf{W}_\mathrm{t}$, $\mathbf{W}_\mathrm{c}$ and $\mathbf{W}_\mathrm{n}$ from the status set $\mathcal{S}$ of our MDP, respectively, and get three variants of SSR-DCF, i.e., SSR-DCF-no$\mathbf{W}_\mathrm{t}$, SSR-DCF-no$\mathbf{W}_\mathrm{c}$ and SSR-DCF-no$\mathbf{W}_\mathrm{n}$. For example, SSR-DCF-no$\mathbf{W}_\mathrm{c}$ only converts between $\mathbf{W}_\mathrm{t}$ and $\mathbf{W}_\mathrm{n}$ during online tracking with the same SVMs used by SSR-DCF. For a comprehensive comparison, we construct three baseline trackers base on SRDCF with the three weight maps, i.e., $\mathbf{W}_\mathrm{t}$, $\mathbf{W}_\mathrm{c}$, and $\mathbf{W}_\mathrm{n}$, respectively, and denote them as SRDCF-$\mathbf{W}_\mathrm{t}$, SRDCF-$\mathbf{W}_\mathrm{c}$, and SRDCF-$\mathbf{W}_\mathrm{n}$. Specifically, SRDCF-$\mathbf{W}_\mathrm{t}$ is the SRDCF in [18]. SRDCF-$\mathbf{W}_\mathrm{n}$ uses target-regularized filters learned at the first frame to track targets without online filter updating. SRDCF-$\mathbf{W}_\mathrm{c}$ locates the target through context-regularized filters calculated by Eq. (8) with the sample cropped from the previous frame.

We evaluate SSR-DCF, its three variants, and the three baseline trackers on OTB-2013 and OTB-2015, respectively. As shown in Fig. 10, removing any status of $\mathcal{S}$ leads to significant performance reduction of SSR-DCF. Specifically, SSR-DCF shows 27.6%, 8.1% and 23.3% relative improvement over SSR-DCF-no$\mathbf{W}_\mathrm{n}$, SSR-DCF-no$\mathbf{W}_\mathrm{c}$ and

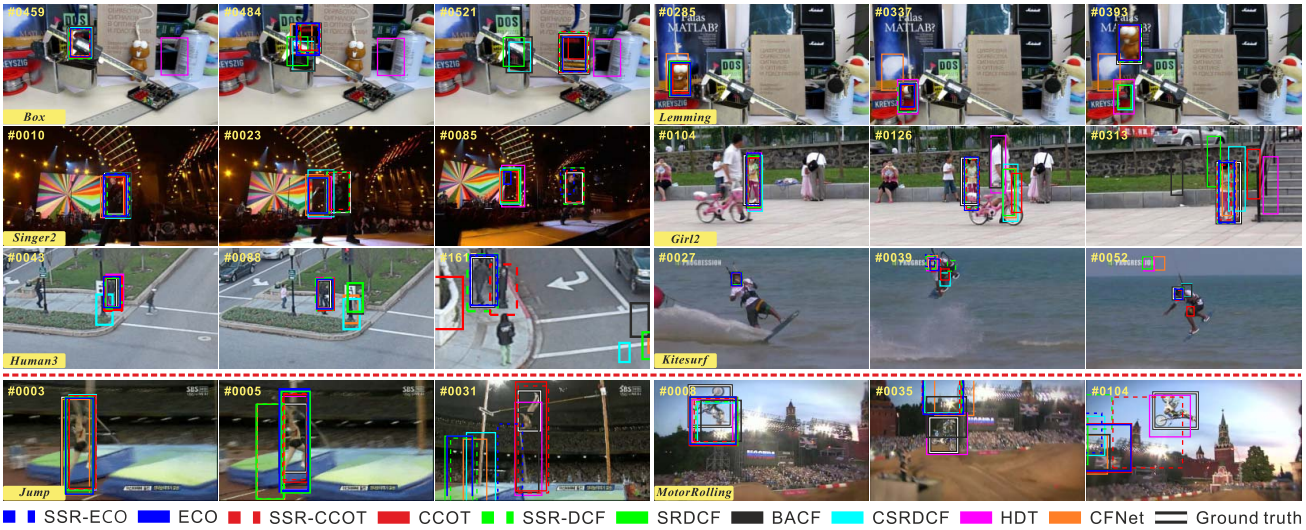


Fig. 9. Visualization results of SSR-ECO, ECO, SSR-CCOT, CCOT, SSR-DCF, SRDCF, and other four CF trackers. The first six sequences show the advantages of SSR-based CF trackers in handling occlusion and background clutter. The last two cases show the limitation of SSR-based CF trackers in handling fast motion with huge deformation.

TABLE I

PERFORMANCE EVALUATION ON VOT-2016 DATASET. IN THIS DATASET, WE COMPARE OUR SSR-ECO, SSR-CCOT, AND SSR-DCF WITH THE ECO, CCOT, SRDCF AND OTHER 7 STATE-OF-THE-ART TRACKERS. THE BEST THREE RESULTS ARE MARKED IN RED, GREEN AND BLUE BOLD FONTS, RESPECTIVELY

	SSR-ECO	ECO	SSR-CCOT	CCOT	SSR-DCF	SRDCF	TCNN	SSAT	MLDF	Staple	HCF	CSRDCF	BACF
EAO	**0.396**	**0.363**	**0.356**	0.339	0.248	0.244	0.325	0.321	0.311	0.295	0.237	0.338	0.223
Accuracy	**0.55**	**0.55**	0.52	0.51	0.53	0.53	0.54	**0.57**	0.48	0.54	0.47	0.51	**0.56**
Robustness	**0.71**	**0.82**	**0.78**	0.87	1.43	1.50	0.96	1.04	0.83	1.35	1.38	0.85	1.88

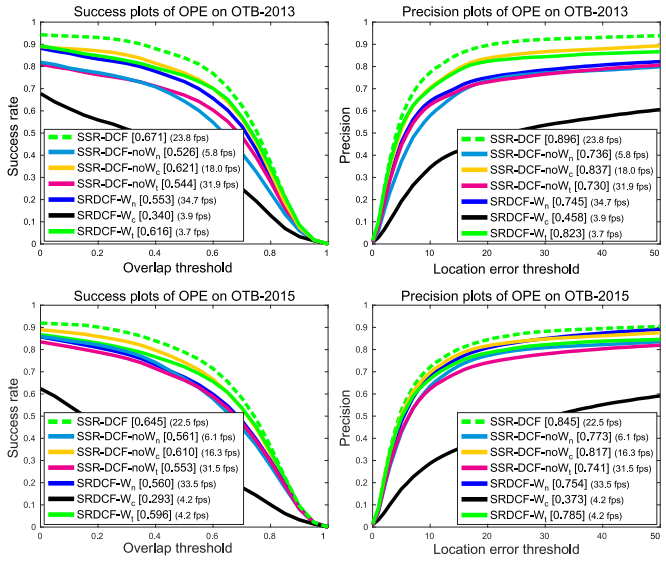


Fig. 10. Ablation study by comparing SSR-DCF and its three variants with SRDCF on OTB-2013 and OTB-2015, respectively. The speed, AUC scores and CLE precision are shown at legends.

SSR-DCF-no$\mathbf{W}_t$, respectively, according to the AUC scores on OTB-2013. We observe similar results on OTB-2015. Hence, all three weight maps help get much better tracking accuracy. Particularly, SSR-DCF-no$\mathbf{W}_n$ obtains the worst results and runs very slowly on OTB-2013, since the status $\mathbf{W}_c$ is frequently activated even if the target-regularized filters are discriminative enough to get good results. SSR-DCF-no$\mathbf{W}_c$ that only converts between $\mathbf{W}_t$ and $\mathbf{W}_n$ achieves higher accuracy and run 4.9 times faster than SRDCF-$\mathbf{W}_t$, which demonstrates that skipping unnecessary or erroneous updating helps improve tracking accuracy and shorten running time.

According to the results of three baseline trackers, we see that: 1) Without any target appearance information, SRDCF-$\mathbf{W}_c$ still gets 37.3% CLE precision, which shows that context-regularized filters may locate a target without using the target appearance information. That is why the SSR-based trackers selectively utilizing context information can address severe occlusion or background clutter. We further discuss the advantage of SRDCF-$\mathbf{W}_c$ in Section V-C.2. 2) Without online updating, SRDCF-$\mathbf{W}_n$ only reduce relative AUC by 6% w.r.t. SRDCF-$\mathbf{W}_t$ while running near 8 times faster. It shows the possibility to accelerate the SRDCF-$\mathbf{W}_t$ without harming its accuracy by doing sparse filter updating.

We further compare the 7 trackers on 11 subsets of OTB-2013 and OTB-2015, respectively. As shown in the second column of Fig. 7, SSR-DCF outperforms its three variants on all subsets, which demonstrates that all three weight maps help track target accurately under various interferences.

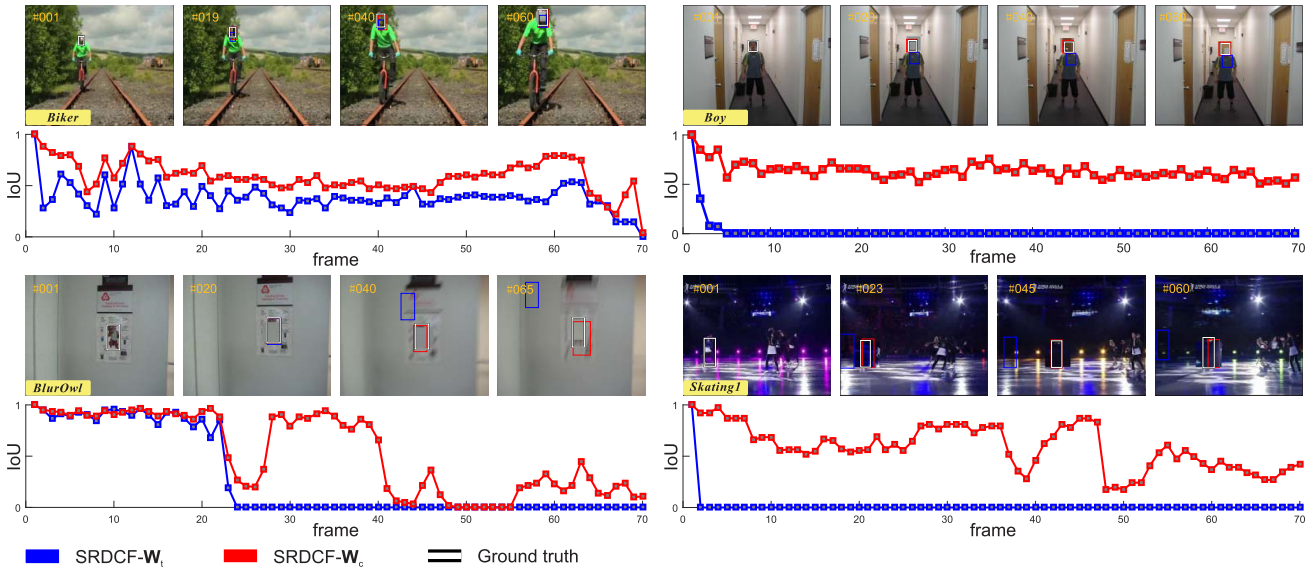


Fig. 11. Validation of context-driven CF tracking that only uses $\mathbf{W}_c$ to learn filters during online tracking and is denoted as SRDCF-$\mathbf{W}_c$. We compare it with target-driven CF tracking, i.e., SRDCF-$\mathbf{W}_t$, that only uses $\mathbf{W}_t$ to learn filters to track a target. Except the first frames, all frames of four testing sequences are added random occlusion. The intersection-over-union (IoU) of each frame is also shown. In sequences of 'Biker' and 'Boy', their cameras are fixed while targets keep moving. In sequence 'Blurowl', its target holds its position while the camera is moving. In sequence 'Skating1', both target and camera are moving.

2) Validation of Context-Driven CF Tracking: In this subsection, we validate and discuss the effectiveness and advantages of using context-regularized filters to track the target. We construct two trackers, i.e., SRDCF-$\mathbf{W}_c$ and SRDCF-$\mathbf{W}_t$ in Section V-C.1, by using context-driven and target-driven weight maps to learn filters and track the target at each frame, respectively. To evaluate their ability of addressing severe occlusion, we add random occlusion to a target in all frames except the first one of a testing sequence and use the two trackers to track the target, respectively. As shown in Fig. 11, SRDCF-$\mathbf{W}_c$ outperforms SRDCF-$\mathbf{W}_t$ on all four sequences. Due to the severe occlusion, SRDCF-$\mathbf{W}_t$ relying on the target appearance easily fails tracking and regards backgrounds as targets. However, SRDCF-$\mathbf{W}_c$ utilizing the context information can still locate the target or even detect the scale variation. Specifically, in sequences 'Biker' and 'Boy', the cameras are fixed while the two men keep moving. Since other parts of the two men become the context of SRDCF-$\mathbf{W}_c$ and have the same motion with targets, i.e., the heads, SRDCF-$\mathbf{W}_c$ can keep tracking the target while estimating the scale variation. Similarly, in the sequence 'BlurOwl', since the camera is moving while all objects in the scene are fixed, the context has the same motion with the owl and helps SRDCF-$\mathbf{W}_c$ track target accurately. In 'Skating1', although both camera and the target keep moving, bounding boxes generated by SRDCF-$\mathbf{W}_c$ are always not far away from the ground truth position, which guarantees the target is included in search region during tracking, thus would help target-regularized filters re-detect the target when occlusion is removed.

3) Validation of Selective Spatial Regularization: In Fig. 12, we show two cases of s transition during online tracking, where s can be -1, 0 or 1, which corresponds to use $\mathbf{W}_c$, $\mathbf{W}_n$ or $\mathbf{W}_t$ to regularize filters. In both sequences of Fig. 12, the weight map keeps $\mathbf{W}_n$, i.e., $s = 0$, at the most of time, which means that target-regularized filters are rarely updated during tracking. Hence, SSR-DCF runs about 5 times faster than SRDCF and obtains much higher accuracy by avoiding erroneous updating of target-regularized filters, as discussed in Section V-C.1. When target appearance changes during tracking, the weight map is set as $\mathbf{W}_t$ with $s = 1$, which means to update target-regularized filters by the detection result to adapt target appearance variation. For example, at the frame '#498' of the first sequence and frame '#876' of the second sequence, the two targets slightly change due to view changing and motion blur, respectively. Our MDP converts the weight map to $\mathbf{W}_t$ immediately to update the target-regularized filters. When a target is occluded or surrounded by similar objects, the MDP sets weight map as $\mathbf{W}_c$ to learn context-regularized filters to track the target. For example, at frame '#532' and '#680' in the first sequence and frame '#728' in the second sequence, the two targets are still accurately located even though they are fully occluded by trees and a glass bottle, respectively. Note, before or after the period of $\mathbf{W}_c$, the weight map is converted to $\mathbf{W}_t$, since partial occlusion usually happens before and after severe occlusion and is regarded as target appearance variation by the MDP. This may lead to error updating of target-regularized filters. However, the problem has little effect on the overall tracking accuracy due to the low frequency of such situation.

4) Analysis of Handling Long-Term Occlusion and Out-of-View: In this subsection, we analyze the ability of our SSR to handle long-term occlusion or out-of-view by comparing SRDCF and SSR-DCF on the VOT-2018LT dataset [41], [61]. VOT-2018LT contains 35 sequences with 14687 frames and each sequence has average 12 long-term target disappearances each of which lasting on average 40 frames. Moreover,

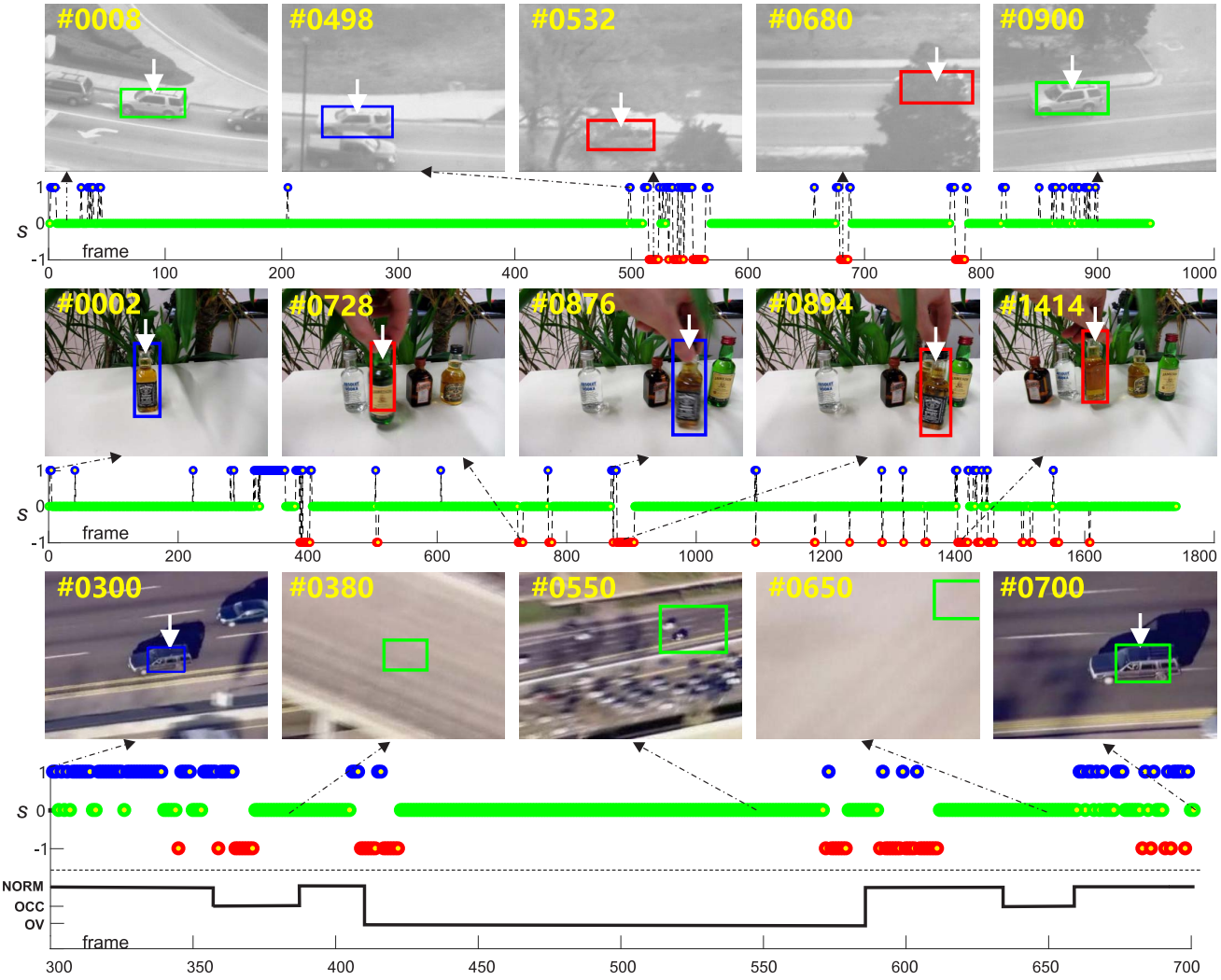


Fig. 12. Validation of selective spatial regularization for short-term and long-term occlusion and out-of-view. With the proposed MDP, the weight map $\mathbf{W}$ converts between three statuses w.r.t. the value of selector s. The white arrows show the ground truth of target locations. Red, Green and Blue bounding boxes correspond to detection results generated by the filters regularized by $\mathbf{W}_c$, $\mathbf{W}_n$ and $\mathbf{W}_t$, respectively. At the bottom of the third case, we visualize the status of the target with 'NORM', 'OCC', and 'OV' representing that the target is within the visual field, occluded, and out-of-view, respectively.

TABLE II

COMPARISON BETWEEN SRDCF AND SSR-DCF ON VOT-2018LT AND ITS THREE SUBSETS, I.E., FULL OCCLUSION (FULL OCC.), PARTIAL OCCLUSION (PARTIAL OCC.), AND OUT-OF-VIEW (OV). WE USE F-SCORE DEFINED IN [61] AS THE EVALUATION METRIC

Tracker	Full Occ.	Partial Occ.	OV	All
SRDCF	0.1204	0.2393	0.1364	0.2313
SSR-DCF	0.1576	0.2721	0.1581	0.2623

the sequences are assigned with ten visual attributes corresponding to ten subsets and we focus on the full occlusion, partial occlusion, and out-of-view subsets. We use F-score that considers both tracking precision and recall and is defined in [61] as the evaluation metric. As shown in Table II, according to the F-score, SSR-DCF outperforms SRDCF with 13.4%, 31.0%, 13.7%, and 15.9% relative improvements on the whole dataset and the three subsets, i.e., full occlusion, partial occlusion, and out-of-view, respectively, and it demonstrates SSR not only helps SRDCF address long-term full or partial occlusion but also out-of-view. This is because the SSR can convert the regularization weight map to $\mathbf{W}_c$ or $\mathbf{W}_n$ when occlusion or out-of-view happens, and avoids the corruption of target-regularized filters, which enables the tracker to re-detect the target when it appears again. We show an intuitive result from VOT-2018LT in the third row of Fig. 12. Long term occlusion and out-of-view happen between frame 360 and frame 660. During this period, the weight map mainly keeps $\mathbf{W}_c$ and $\mathbf{W}_n$ and the target is re-detected at the 700th frame.

5) Disadvantage of Using Context-Regularized Filters: Using context to address occlusion and background clutter is base on the assumption that the target and its context have similar motion temporarily. However, this assumption tends to fail when the target has huge deformation while running fast. Under this situation, the context-regularized filters would be triggered and fail to track, since the context between two neighboring frames would be different due to the fast motion. We show two examples, i.e., 'Jump' and 'MotorRolling' whose targets move very fast with huge shape

TABLE III
TIME COST COMPARISON AMONG SRDCF, SSR-DCF, CCOT, SSR-CCOT, ECO AND SSR-ECO ON OTB-2015

	FPS	Online Avg. cost per frame (s)	Avg. cost at 1^{st} frame (s)		
			Syn. Gen.	RL Train.	Init.
SRDCF	4.2	0.122	-	-	0.533
SSR-DCF	22.5	0.086	1.59	1.66	-
CCOT	0.48	3.35	-	-	4.33
SSR-CCOT	1.74	1.75	1.70	8.85	-
ECO	22.2	0.046	-	-	1.70
SSR-ECO	24.1	0.042	1.65	6.05	-

deformation, in the last row of Fig. 9. Particularly, in the case of 'MotorRolling', all SSR-based trackers miss the motorcycle. One possible solution for this problem is to equip the SSR-based trackers with an effective motion model that helps avoid error triggering of context-regularized filters.

6) Time-Consuming Analysis: In Table III, we report the average speed, average online time cost per frame, and average time cost at the first frame of SRDCF, SSR-DCF, CCOT, SSR-CCOT, ECO and SSR-ECO on OTB-2015. In terms of the FPS, SSR-DCF and SSR-CCOT run 5.4 and 3.6 times faster than SRDCF and CCOT, respectively. SSR-ECO is slightly faster than ECO, since ECO updates filters every 5 frames, which significantly speeds up itself. As a result, our SSR being a universal component not only helps spatially-regularized CF trackers improve tracking accuracy but also increase their online speed significantly. In terms of the cost at the first frame, we consider three values, the average cost of synthetic video generation (Syn. Gen.), reinforcement training for MDP (RL Train.), and initialization of target-regularized filters (Init.). For SSR-based trackers, the Init. process is included in the RL Train. stage while spatially-regularized CF trackers do not contain the Syn. Gen. and RL Train. processes. According to the results in Table III, SSR-DCF, SSR-CCOT, and SSR-ECO take 6.1, 2.5, and 4.5 times more cost than SRDCF, CCOT, and ECO at the first frame due to the extra time for Syn. Gen. and RL Train. Although SSR-based trackers are inefficient at the first frame, the total cost will be reduced when we handle long-term videos, since SSR-based trackers always have a lower time cost during the online process than the original spatially-regularized CF trackers. More importantly, SSR-based trackers significantly outperform the original trackers on long-term object tracking according to the results on LaSOT and VOT2018LT as shown in Fig. 8 and Table. II. Note, SRDCF, SSR-DCF, CCOT, and SSR-CCOT are evaluated on the same platform with the CPU Intel i7-3770 and RAM 16 GB while ECO and SSRECO run on the NVIDIA RTX2080.

VI. CONCLUSION AND DISCUSSION

In this paper, we proposed the selective spatial regularization (SSR) for correlation filter (CF)-tracking scheme that selectively learns the *target-context-regularized filters* and can reliably track a target even if it is severely occluded or within cluttered background. Specifically, we proposed an extended objective function for the CF-tracking scheme to generate *target-context-regularized filters* by selectively using *target-context-driven weight maps* in the online CF optimization. We then constructed a Markov Decision Process (MDP) whose state policy set decides which weight map should be selected during the online tracking process. We effectively learned the state policy set of the MDP on a synthetic video sequence that is generated with the ground truth target in the first frame via reinforcement learning. Besides, by adding a special state in the MDP representing not updating filters, we also learned when to skip unnecessary or erroneous filter updating, thus to accelerate the online tracking speed without harming the accuracy. We have shown that SSR is a universal component for the CF-tracking scheme and improved three popular spatially-regularized CF trackers, i.e., SRDCF [18], CCOT [19], and ECO [20], with much faster online speed. We validated the effectiveness and superiority of our trackers over various state-of-the-art competitors on five benchmark datasets, OTB-2013, OTB-2015, LaSOT, TC-128, and VOT-2016.

For the MDP of our SSR, the desired implementation is to adopt the deep reinforcement learning by offline training a Q-net as done in early-stopping tracker (EAST) [62]. Nevertheless, compared with EAST, the SSR has two challenges with deep reinforcement learning: 1) An effective large-scale training dataset for SSR is not easily obtained. The objective of EAST is to adopt an MDP that makes decisions across feature layers to predict a tight bounding box wrapping the target with few feature layers at each frame. Any annotated image pairs can be added to the training set to offline train the Q-net. In contrast, our objective is to use the MDP to make decisions across frames and decide when to skip filter updating or use context-regularized filters to address severe occlusion or background clutter. However, severe situation, e.g., occlusion, rarely happens in a real-world video dataset, which limits the effectiveness of training sequences. 2) To handle all test videos with one universal MDP, deep reinforcement learning for our SSR needs a novel Q-net that can generate robust decisions and tolerate spatial-temporal variations of different targets and backgrounds. The Q-net used by EAST takes the response map of one frame as inputs and does not consider the spatial-temporal variation across different videos. Hence, it cannot be used to realize a universal MDP for SSR directly and should be carefully designed to let the MDP make robust decisions across videos. In summary, this work opens a door to improve the CF trackers and more recent advanced deep learning techniques could be used to make the SSR more powerful in the future by overcoming its specific challenges.

REFERENCES

[1] F.-P. Tian *et al.*, "Active camera relocalization from a single reference image without hand-eye calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 12, pp. 2791–2806, Dec. 2019.
[2] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. van den Hengel, "A survey of appearance models in visual object tracking," *ACM Trans. Intell. Syst. Technol.*, vol. 4, no. 4, 2013, Art. no. 58.
[3] S. Avidan, "Support vector tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 1064–1072, Aug. 2004.

[4] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: Structured output tracking with kernels," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 263–270.

[5] 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, 2008.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

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

[11] 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.

[12] 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.

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

[14] 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.*, Oct. 2017, pp. 1781–1789.

[15] 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. 1–11.

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

[17] 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.*, Jun. 2016, pp. 1401–1409.

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

[19] 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.

[20] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6931–6939.

[21] 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.

[22] 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.

[23] L. Cerman, J. Matas, and V. Hlaváč, *Sputnik Tracker: Having a Companion Improves Robustness of the Tracker*. Berlin, Germany: Springer, 2009, pp. 291–300.

[24] H. Grabner, J. Matas, L. Van Gool, and P. Cattin, "Tracking the invisible: Learning where the object might be," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 1285–1292.

[25] C. Zhou, Q. Guo, L. Wan, and W. Feng, "Selective object and context tracking," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2017, pp. 1947–1951.

[26] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2411–2418.

[27] 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.

[28] M. Kristan *et al.*, "The visual object tracking VOT2016 challenge results," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, Nov. 2016, pp. 777–823.

[29] 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.

[30] H. Fan *et al.*, "LaSOT: A high-quality benchmark for large-scale single object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 5374–5383.

[31] M. Tang and J. Feng, "Multi-kernel correlation filter for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3038–3046.

[32] M. Tang, B. Yu, F. Zhang, and J. Wang, "High-speed tracking with multi-kernel correlation filters," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4874–4883.

[33] N. Wang, W. Zhou, Q. Tian, R. Hong, M. Wang, and H. Li, "Multi-cue correlation filters for robust visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4844–4853.

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

[35] Q. Wang, J. Gao, J. Xing, M. Zhang, and W. Hu, "DCFNet: Discriminant correlation filters network for visual tracking," 2017, *arXiv:1704.04057*. [Online]. Available: https://arxiv.org/abs/1704.04057

[36] 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.*, Jul. 2017, pp. 2805–2813.

[37] W. Zou, Z. Zhu, W. Wu, and J. Yan, "End-to-end flow correlation tracking with spatial-temporal attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 548–557.

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

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

[40] A. Lukezic, T. Vojir, L. C. Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6309–6318.

[41] M. Kristan *et al.*, "The sixth visual object tracking VOT2018 challenge results," in *Proc. Eur. Conf. Comput. Vis. Workshop*, 2018, pp. 3–53.

[42] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 1144–1152.

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

[44] M. Yang, Y. Wu, and G. Hua, "Context-aware visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 7, pp. 1195–1209, Jul. 2009.

[45] T. B. Dinh, N. Vo, and G. Medioni, "Context tracker: Exploring supporters and distracters in unconstrained environments," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 1177–1184.

[46] F. Xiong, O. I. Camps, and M. Sznaier, *Dynamic Context for Tracking behind Occlusions*. Berlin, Germany: Springer, 2012, pp. 580–593.

[47] K. Zhang, L. Zhang, Q. Liu, M.-H. Yang, and D. Zhang, "Fast visual tracking via dense spatio-temporal context learning," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 127–141.

[48] A. Li and S. Yan, "Object tracking with only background cues," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 11, pp. 1911–1919, Nov. 2014.

[49] B.-J. Chen and G. Medioni, "Exploring local context for multi-target tracking in wide area aerial surveillance," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 787–796.

[50] H. T. Nguyen and A. W. M. Smeulders, "Fast occluded object tracking by a robust appearance filter," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 1099–1104, Aug. 2004.

[51] A. Senior, A. Hampapur, Y.-L. Tian, L. Brown, S. Pankanti, and R. Bolle, "Appearance models for occlusion handling," *Image Vis. Comput.*, vol. 24, no. 11, pp. 1233–1243, 2006.

[52] J. Pan and B. Hu, "Robust occlusion handling in object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

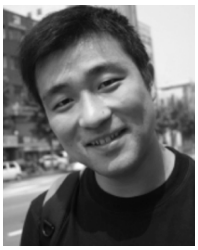
[53] D. Chen, Z. Yuan, Y. Wu, G. Zhang, and N. Zheng, "Constructing adaptive complex cells for robust visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1113–1120.

[54] X. Dong, J. Shen, D. Yu, W. Wang, J. Liu, and H. Huang, "Occlusion-aware real-time object tracking," *IEEE Trans. Multimedia*, vol. 19, no. 4, pp. 763–771, Apr. 2016.

[55] M. Wang, Y. Liu, and Z. Huang, "Large margin object tracking with circulant feature maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4800–4808.
[56] M. Kristan *et al.*, "The visual object tracking VOT2015 challenge results," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, Dec. 2015, pp. 564–586.
[57] Y. Qi *et al.*, "Hedged deep tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4303–4311.
[58] 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.*, Jun. 2016, pp. 4266–4274.
[59] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional siamese networks for object tracking," 2016, *arXiv:1606.09549*. [Online]. Available: https://arxiv.org/abs/1606.09549
[60] H. Nam, M. Baek, and B. Han, "Modeling and propagating CNNs in a tree structure for visual tracking," 2016, *arXiv:1608.07242*. [Online]. Available: https://arxiv.org/abs/1608.07242
[61] A. Lukežič, L. C. Zajc, T. Vojíř, J. Matas, and M. Kristan, "Now you see me: Evaluating performance in long-term visual tracking," 2018, *arXiv:1804.07056*. [Online]. Available: https://arxiv.org/abs/1804.07056
[62] C. Huang, S. Lucey, and D. Ramanan, "Learning policies for adaptive tracking with deep feature cascades," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 105–114.

Qing Guo received the B.S. degree in electronic and information engineering from the North China Institute of Aerospace Engineering in 2011, the M.E. degree in computer application technology from the College of Computer and Information Technology, China Three Gorges University in 2014, and the Ph.D. degree in computer application technology from the School of Computer Science and Technology, Tianjin University, Tianjin, China. He is currently with the School of Computer Science and Technology, College of Intelligence and Computing, Tianjin University. He is also a Research Fellow with the Cyber Security Research Centre, Nanyang Technological University, Singapore. His research interests include visual object tracking, image and video object segmentation, image denoising, and other related vision problems.

Ruize Han received the B.S. degree in mathematics and applied mathematics from the Hebei University of Technology, China, in 2016, and the M.Eng. degree in computer technology from Tianjin University, China, in 2019, where he is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, College of Intelligence and Computing. 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 (M'06) received the B.S. and M.Phil. degrees in computer science from Northwestern Polytechnical University, China, in 2000 and 2003, respectively, and 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 the City University of Hong Kong. He is currently a Full Professor with the School of Computer Science and Technology, College of Intelligence and Computing, Tianjin University, Tianjin, 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. He focuses on solving preventive conservation problems of cultural heritages via computer vision and machine learning.

Zhihao Chen received the B.E. degree from the School of Computer Software, Tianjin University, China, in 2017, where he is currently pursuing the Ph.D. degree with the College of Intelligence and Computing. His major research interest is visual intelligence, specifically including single object tracking, semantic segmentation, and medical image processing.

Liang Wan received the B.Eng. and M.Eng. degrees in computer science and engineering from Northwestern Polytechnical University, China, in 2000 and 2003, respectively, and the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong in 2007. She is currently a Full Professor with the College of Intelligence and Computing, Tianjin University, China. Her research interests are mainly on image processing and computer vision, including panoramic image processing, visual object tracking, and medical image analysis.