IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 18, 2019, accepted October 2, 2019, date of current version October 16, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2945846

SVTN: Siamese Visual Tracking Networks With Spatially Constrained Correlation Filter and Saliency Prior Context Model

BO HUANG[1], TINGFA XU[1,2], SHENWANG JIANG[1], YU BAI[1], AND YIWEN CHEN[1]

[1]Image Engineering and Video Technology Lab, School of Optics and Photonics, Beijing Institute of Technology, Beijing 100081,China
[2]Key Laboratory of Photoelectronic Imaging Technology and System, Ministry of Education of China, Beijing 100081, China

Corresponding author: Tingfa Xu (ciom_xtf1@bit.edu.cn)

This work was supported in part by the Major Science Instrument Program of the National Natural Science Foundation of China under Grant 61527802, and in part by the General Program of National Natural Science Foundation of China under Grant 61371132 and Grant 61471043.

ABSTRACT Recently, Siamese network based trackers have been greatly developed and achieved state-of-the-art performance on multiple benchmarks. However, the decision-making mechanism needs to be studied more deeply in order to obtain higher accuracy. In this paper, we propose a novel Siamese network based visual tracking method, which enhances decision-making ability by Spatially Constrained Correlation Filter (SCCF) and Saliency Prior Context (SPC) model. We use the deep features extracted from Siamese networks to train the SCCF via the efficient Alternating Direction Method of Multipliers (ADMM), and our SCCF applies a penalizing matrix to suppress the boundary effect well. Meanwhile, we regard the end-to-end output of Siamese networks as a priori probability and utilize the spatio-temporal relationship to establish the SPC model. The SPC model can handle the various cases of feature distributions generated from different targets and their contexts. Further, we also take measures to solve some challenging problems in visual tracking, such as target scale change and target occlusion. We conduct extensive experiments to demonstrate the effectiveness of the proposed method, which obtains currently the best results on three large tracking benchmarks, including OTB-2013, OTB-2015, and VOT-2016.

INDEX TERMS Siamese network, spatially constrained correlation filter (SCCF), saliency prior context (SPC).

I. INTRODUCTION

Visual tracking has a large range of applications in multimedia processing, e.g. automatic driving, robotics, and human computer interaction. Although much progress has been made in the past decade, it has still been commonly recognized as a very challenging task as target objects often undergo significant appearance changes over time and may temporally leave the field of the view.

Recently, the deep learning based trackers have drawn much attention in the community. In the case of very limited training data of target tracking, Wang and Yeung [1] firstly use auxiliary non-tracking training data for pre-training to obtain a general representation of target features. In tracking, they fine-tune the pre-training model to enhance the classification performance for the current tracking target. Up to now, most deep trackers mainly consist of two parts, i.e. a deep feature extraction module and a decision-making mechanism. Some trackers put more emphasis on the deep feature extraction models pre-trained for the object recognition task. Wang *et al.* [2] make efficient use of Convolutional Neural Network (CNN) features offline pre-trained on ImageNet dataset, and deeply analyze the influence of Conv4-3 and Conv5-3 layers on tracking performance. Ma *et al.* [3], [4] further investigate Conv3-4, Conv4-4, and Conv5-4 layers of VGG-19 features, and give different weights to each layer to obtain the final response map. In DeepTrack [5], a candidate pool of multiple CNN is employed as a data-driven model of different instances of the target object, and the tracking task is finally regarded as a classification problem.

Some other trackers extensively explore various decision models, such as correlation filters [6], [7],

The associate editor coordinating the review of this manuscript and approving it for publication was Qilian Liang.

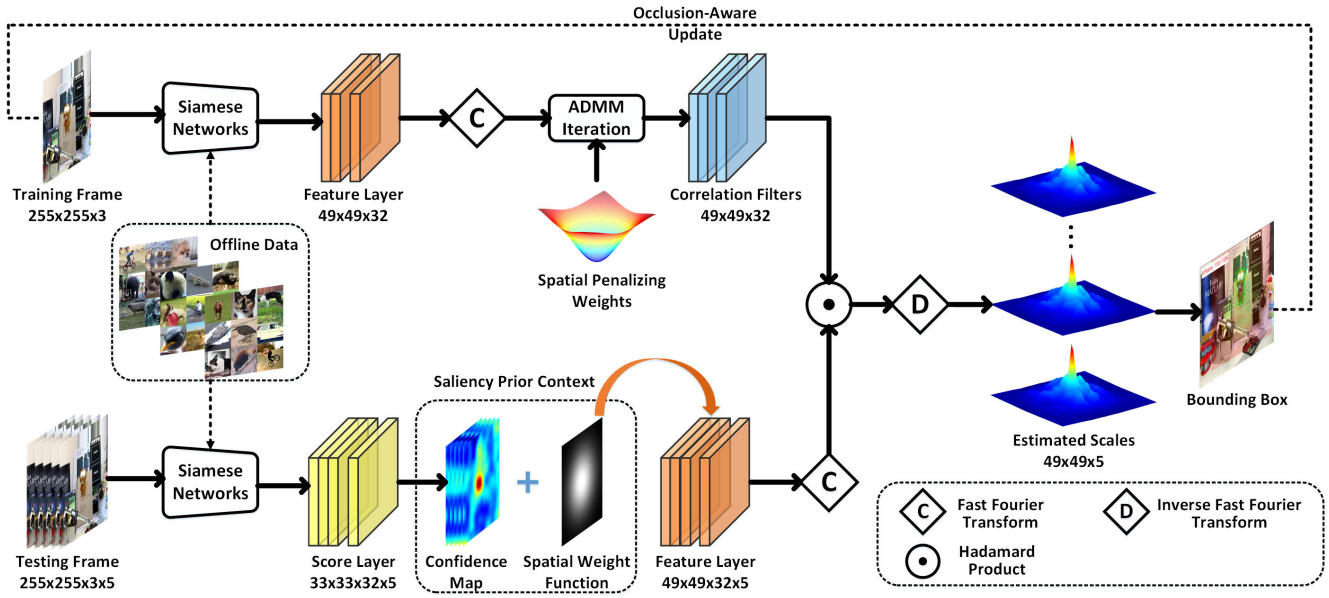


FIGURE 1. A scheme of the proposed SVTN for single object tracking.

regressors [8]–[11], and classifiers [12]–[14], considerably less attention is paid to learning more discriminative deep features. Although the deep feature is of great significance to improve the tracking performance, we note that the contributions of the decision-making mechanism for visual tracking are even more important as it is the most direct basis for the final output [7], [15]–[17]. For instance, Tao *et al.* [18] propose a fully-convolutional Siamese network (SiamFC) specially designed for target tracking, which computes the cross-correlation of target template and sliding-window on the search image. SiamFC2 (also named CFnet) [7] is obviously superior to the baseline SiamFC after equipping the correlation filter block in the decision-making system. SiamRPN [15] strengthens the decision-making ability of the model via a region proposal subnetwork including the classification branch and regression branch, which significantly improves tracking performance. Therefore, how to design an effective decision-making mechanism remains an open problem.

In this work, we propose a novel Siamese Visual Tracking Network (SVTN) with Spatially Constrained Correlation Filter (SCCF) and Saliency Prior Context (SPC) model. As shown in Figure 1, our SVTN consists of a training branch and a detection branch, which are both based on the Siamese structure. Inspired by the state-of-the-art method CFNet [7], the Siamese networks perform the Correlation Filters (CFs) as the decision-making mechanism and are pre-trained offline with large-scale image pairs in an end-to-end manner. Implementing CFs efficiently by Fast Fourier Transform (FFT) is, however, plagued by circular boundary effects, which dramatically hurt the tracking performance. And it is also difficult to update the filters in Siamese networks due to the end-to-end design. Different from standard CFnet, we extract the deep features from Siamese networks to train another correlation filter and apply a spatial constraint on the filter to alleviate the boundary effects. This spatial constrained function is a penalizing matrix that assigns higher bias weights on the background region. Since we implement a spatial constraint, we must solve the filter in the spatial domain. For the optimal solution of the model, we calculate the filter efficiently via the Alternating Direction Method of Multipliers (ADMM) [19]. An auxiliary variable is introduced to split the original solution into two subproblems, one in the spatial domain and the other in the frequency domain. Therefore, we can perform the spatial constraint in the spatial domain and solve the filter in the frequency domain at the same time. In the detection branch, we regard the end-to-end output of Siamese networks as a priori probability and utilize the spatio-temporal relationship to establish a Saliency Prior Context (SPC) model. The SPC model removes background interference and inhibits similar targets to a certain extent. To address the scale change of the target, we take advantage of convolution between the filter with fixed size and search regions with different resolutions to calculate the multi-scale response map. We estimate the scale variation of the target via the best matched scale in the scaling pool. Furthermore, we present an occlusion-aware template update scheme to handle appearance changes during tracking. We calculate the confidence score of the candidate target through the feedback of the final response map, and then decide whether to update the filter or not.

In summary, we have the following contributions in this paper:

- We successfully implement a Spatially Constrained Correlation Filter (SCCF) into Siamese Visual Tracking Networks. Our model not only owns a deep feature

extraction module with strong discriminating ability, but also has a better decision-making mechanism as we solve the boundary effects well.
- We encode the end-to-end output of Siamese networks as a priori probability and utilize the spatio-temporal relationship to establish a Saliency Prior Context (SPC) model. The SPC model removes background interference and inhibits similar targets to a certain extent.
- To handle appearance changes during tracking, we present a scale-aware strategy to address the target size variation and an occlusion-aware update scheme to avoid corruption in the model.

II. RELATED WORKS

Visual tracking has been extensively researched in recent decades. In the following we briefly review the development of tracking methods and list works closely related to our algorithm.

A. SIAMESE NETWORKS

Recently, the Siamese network based trackers have attracted increasing interest due to their well-balanced tracking accuracy and efficiency. The pioneering works are SINT [20] and SiamFC [18]. SINT employs a Siamese network to offline learn a matching function from a large set of sequences, then utilizes the fixed matching function to find the target in the tracking task. SiamFC introduces a fully convolutional Siamese network by training a similarity metric between the target object and candidate image patches. Inspired by them, a large amount of follow-up works [7], [8], [15]–[17] have been proposed. GOTURN [8] suggest a Siamese network based motion prediction model, which can track the generic object at 100 FPS. Valmadre *et al.* [7] propose a Siamese network to learn an end-to-end representation for correlation tracking filter. Li *et al.* [16] presents a residual attentional Siamese network, which introduces different kinds of the attention mechanisms to adapt the model without updating the model online. Notably, Bertinetto *et al.* [15] incorporate a region proposal subnetwork into Siamese network, and propose a one-stage Siamese-RPN tracker, achieving excellent performance. SiamMask [17] improves the offline training procedure of Siamese network for object tracking by augmenting its loss with a binary segmentation task.

B. CORRELATION FILTERS

The CF-based approaches for visual tracking have become increasingly popular due to its rapid computation speed. The standard formulation of CFs uses circular correlation, which allows to implement learning efficiently by Fast Fourier Transform (FFT). Bolme *et al.* [21] propose the pioneering MOSSE tracker, which utilizes the grey-scale image to extract single channel feature with high speed. Later, Henriques *et al.* [6] propose to use correlation filters in a kernel space with the KCF method. Danelljan *et al.* [22] propose an adaptive multi-scale correlation filter using HOG features to handle the scale change of target objects. SRDCF [23] introduces a spatial regularization term to penalize the CF coefficients according to their spatial locations and utilizes the Gauss-Seidel algorithm to solve the resulting normal equations. Galoogahi *et al.* [24] suggest a spatial cropping operation to address the boundary effects, and then solve the constrained optimization problem via ADMM. Li *et al.* [25] present a Spatial Temporal Regularized Correlation Filter (STRCF) by introducing temporal regularization to SRDCF.

C. TRACKING INCORPORATING SALIENCY

With the rapid development of saliency detection methods in recent years, incorporating saliency into tracking algorithms have been proposed in many publications to facilitate the robustness and accuracy for tracking. Mahadevan and Vasconcelos [26] propose a biologically inspired framework for visual tracking based on the top-down tuning of center-surround saliency mechanisms. Tran and Yaun [27] suggest to track the salient object by finding a spatio-temporal path of the highest saliency density in consecutive frames. Fan *et al.* [28] exploit discriminative spatial attention regions to handle the spatial distractions that exhibit similar visual appearances as the target. Zhang *et al.* [29] exploit a Bayesian framework to model the statistical correlation between the low-level features from the target and its surrounding regions. Hong *et al.* [12] propose an online visual tracking approach by learning discriminative saliency map using a Convolutional Neural Network (CNN). Zhu *et al.* [30] propose saliency proposals as prior information to handle the model drift problem caused by occlusion or distracter. Ma *et al.* [31] exploit low-level features for saliency and establish a new prior context model to deal with the various cases of feature distributions generated from different targets and their contexts.

III. THE PROPOSED SVTN METHOD

A. FULLY-CONVOLUTIONAL SIAMESE NETWORKS

In Siamese structure, a pair of fully-convolutional networks that use the target template x and the search region (larger) z as inputs, are employed to obtain a dense response map. x and z are, respectively, an image patch centered on the target object and a larger one centered on the last estimated position of the target. This pair of networks share all the parameters to ensure the same transformation applied to both x and z, which is crucial for the similarity metric learning. Both inputs are processed by a ConvNet χ with parameters θ. This yields two feature maps, which are cross-correlated as,

$$g_\theta(x, z) = \chi_\theta(x) * \chi_\theta(z) \tag{1}$$

where $*$ is the spatial correlation operator, and g_θ is the response map. The goal is to estimate the target location by finding the maximum value of the response map. To achieve this goal, the network is offline-trained with millions of random pairs taken from a collection of videos. We adopt the

logistic loss to train the network,

$$\arg\min_{\theta} \sum_{i} \ell(g_\theta(x_i, z_i), c_i) \tag{2}$$

Many previous works [6], [7] have proved that this similarity should be efficiently obtained in frequency domain by a correlation filter. CFnet [7] modifies the Siamese network by a correlation filter block, and this change can be formalized as,

$$g_\theta(x, z) = s(\chi_\theta(h)) * (\chi_\theta(z))) + b \tag{3}$$

where $\chi_\theta(h)$ is the correlation filter trained from the target feature map $\chi_\theta(x)$ by solving a ridge regression problem in the Fourier domain. s and b are the scalar parameters and bias respectively.

B. SALIENCY PRIOR CONTEXT MODEL

Researchers have pointed out that human attention fixations have a strong center bias [29], which indicates that more weight should be put on the locations near scene center in the saliency computing. Therefore, we use the Gaussian fashion function to label the context and distributes the center bias weight objectively to each pixel according to the location,

$$\varpi_\sigma(\rho) = \alpha \exp(-|\rho|^2/\sigma^2) \tag{4}$$

where ρ represents the spatial relative location distribution of all pixels in the searching region. σ is the shape parameter, and α is a constant used to control the ratio.

Denote x as a target location in the context and o as the real position of existing object in the scene. We achieve the goal of tracking by maximizing a confidence map $M(x)$, which denotes the posterior probability $P(o|x)$ and can be formulated as $P(x|o)P(o)$. Denote $c(z)$ as the feature at location $z \in \mathbb{Z}$ in the context, where $\mathbb{Z}$ represents the context location set. For each observation x, we use all the context features $c(z)$ to build a context model, and the likelihood function $P(x|o)$ can be computed as,

$$\begin{aligned} P(x|o) &= \sum_{z\in\mathbb{Z}} P(x, c(z)|o) \\ &= \sum_{z\in\mathbb{Z}} P(x|c(z), o)P(c(z)|o) \end{aligned} \tag{5}$$

Therefore we have,

$$\begin{aligned} M(x) &= P(x|o)P(o) \\ &= \sum_{z\in\mathbb{Z}} P(x|c(z), o)P(c(z)|o)P(o) \\ &= \sum_{z\in\mathbb{Z}} P(x|c(z), o)P(c(z), o) \end{aligned} \tag{6}$$

where the first item $P(x|c(z), o)$ indicates the conditional probability and the second item $P(c(z), o)$ represents the context prior probability.

Accordingly, the conditional probability $P(x|c(z), o)$ demonstrates the spatial relationship between target region and its context information, and is related to a spatial context model $H(x)$ using a function $\varpi_\sigma(x - z)$ describing relative distance and direction between two locations x and z, which can be learned and updated across frames. The prior probability $P(c(z), o)$ is modeled by $G(x)$, which is estimated by the CFnet [7]. CFnet designs end-to-end representation learning for correlation filter, so it can output the prior confidence map directly. In this paper, we encode the response score layer as the priori probability $G(x)$. Finally, our saliency prior context model is defined as following,

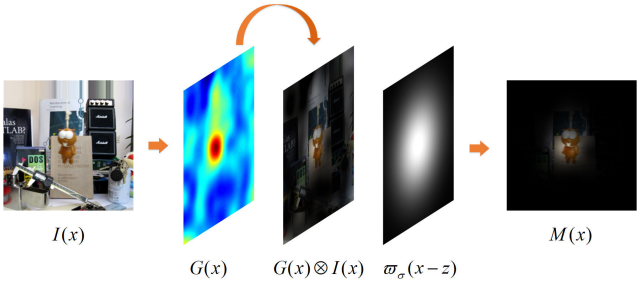


FIGURE 2. Illustration of the building of saliency prior context model.

$$M(x) = H(x) \otimes G(x) \otimes I(x) \tag{7}$$

where $\otimes$ indicates the Kronecker product. $I(x)$ is an intensity map for each observation x. An illustration of the modeling procedure is in Figure 2. In practice, the deep feature map $\chi_\theta(x)$ generated by Siamese networks are used to replace $I(x)$.

C. TRAINING CFS WITH SPATIAL CONSTRAINT

In order to make our model more robust, we utilize the deep features extracted from the Siamese networks to train a Spatially Constrained Correlation Filter (SCCF). Our SCCF can be expressed in the spatial domain by minimizing the following objective,

$$\varepsilon(\chi_\theta(h)) = \frac{1}{2}\|y - \chi_\theta(h) * \chi_\theta(x)\|^2 + \frac{1}{2}\|w \circ \chi_\theta(h)\|^2 \tag{8}$$

where $\chi_\theta(x) \in \mathbb{R}^L$ and $\chi_\theta(h) \in \mathbb{R}^L$ refer to the training deep feature and filter respectively, and L is the length of the signal $\chi_\theta(x)$. $y \in \mathbb{R}^L$ is the desired correlation response, and $\circ$ represents the element-wise multiplication. w is a spatial weight function used to penalize the magnitude of the filter coefficients in the learning. This spatial weights have been proved to solve the boundary effect well in SRDCF [23]. Similar to [24], (8) can be identically formulated as solving the following ridge regression problem,

$$\begin{aligned} \varepsilon(\chi_\theta(h)) = \frac{1}{2}\sum_{j=1}^{L} \|y[j] - \sum_{k=1}^{K} \chi_\theta^k(h)^{\mathsf{T}} \chi_\theta^k(x)[\Delta\tau_j]\|^2 \\ + \frac{1}{2}\sum_{k=1}^{K} \|w \circ \chi_\theta^k(h)\|^2 \end{aligned} \tag{9}$$

where K is the number of feature channels and T represents transpose operator. $[\Delta\tau_1, \ldots, \Delta\tau_L]$ generates all circular shifts, and $\chi_\theta^k(x)[\Delta\tau_j]$ represents applying a j-step discrete

circular shift to the signal $\chi_\theta^k(x)$. $y[j]$ represesnts the j-th element of y.

Correlation filters can be expressed as a Hadamard product in the frequency domain, for computational efficiency [32]. Unfortunately, since we are enforcing a spatial constraint, the filter must be solved in the spatial domain. Therefore, we express (9) as following,

$$\begin{aligned}\varepsilon(h,\hat{a}) &= \frac{1}{2}\|\hat{y}-\hat{X}\hat{a}\|^2+\frac{1}{2}\|(w\otimes I_K)h\|^2\\ s.t.\ \hat{a} &= \sqrt{L}(F\otimes I_K)h\end{aligned} \tag{10}$$

where $\hat{a}$ is an auxiliary variable, and the matrix $\hat{X}$ is defined as $\hat{X} = [diag(\chi_\theta^1(x))^\mathsf{T},\ldots,diag(\chi_\theta^K(x))^\mathsf{T}]$ of size $L\times KL$. A hat $\wedge$ is employed here as shorthand for the Discrete Fourier Transform (DFT) of a signal, such that $\hat{x}=\mathcal{F}(x)=\sqrt{L}Fx$, where F is the orthonormal $L\times L$ matrix of complex basis vectors for mapping to the Fourier domain for any L dimensional vectorized signal. $\hat{y}$ is the Fourier transform of y, and $h=[\chi_\theta^1(h)^\mathsf{T},\ldots,\chi_\theta^K(h)^\mathsf{T}]$ is the $KL\times L$ over-complete representations of the filters by concatenating their K vectorized channels. I_K is a $K\times K$ identity matrix.

The model in (10) is convex, and can be minimized to obtain the globally optimal solution via the Alternating Direction Method of Multipliers (ADMM). To this end, we form the augmented Lagrangian as,

$$\begin{aligned}\mathcal{L}(\hat{a},h,\hat{\varsigma}) &= \frac{1}{2}\|\hat{y}-\hat{X}\hat{a}\|^2+\frac{1}{2}\|(w\otimes I_K)h\|^2\\ &+\hat{\varsigma}^\mathsf{T}(\hat{a}-\sqrt{L}(F\otimes I_K)h)\\ &+\frac{\mu}{2}\|\hat{a}-\sqrt{L}(F\otimes I_K)h\|^2\end{aligned} \tag{11}$$

where μ is the penalty factor and $\hat{\varsigma}=[(\hat{\varsigma}^1)^T,\ldots,(\hat{\varsigma}^K)^T]^T$ is the $KL\times 1$ Lagrangian vector in the Fourier domain. The ADMM algorithm is then adopted by alternatingly solving the subproblems, $\hat{a}^*$ and h^*, each of the subproblems, has a closed form solution.

1) SUBPROBLEM h^*

$$\begin{aligned}h^* = \arg\min_h \Big\{&\frac{1}{2}\|(w\otimes I_K)h\|^2+\hat{\varsigma}^\mathsf{T}(\hat{a}-\sqrt{L}(F\otimes I_K)h)\\ &+\frac{\mu}{2}\|\hat{a}-\sqrt{L}(F\otimes I_K)h\|^2\Big\}\end{aligned} \tag{12}$$

By using $(A\otimes B)^\mathsf{T}=(A^\mathsf{T}\otimes B^\mathsf{T})$, we have,

$$\begin{aligned}\mathcal{H} &= \frac{1}{2}h^\mathsf{T}(w^\mathsf{T}\otimes I_K)(w\otimes I_K)h+\hat{\varsigma}^\mathsf{T}\hat{a}-\sqrt{L}\hat{\varsigma}^\mathsf{T}(F\otimes I_k)h\\ &+\frac{\mu}{2}(\hat{a}-\sqrt{L}(F\otimes I_K)h)^\mathsf{T}(\hat{a}-\sqrt{L}(F\otimes I_K)h)\\ &= \frac{1}{2}h^\mathsf{T}(w^\mathsf{T}\otimes I_K)(w\otimes I_K)h+\hat{\varsigma}^\mathsf{T}\hat{a}-\sqrt{L}\hat{\varsigma}^\mathsf{T}(F\otimes I_k)h\\ &+\frac{\mu}{2}(\hat{a}^\mathsf{T}\hat{a}-\sqrt{L}\hat{a}^\mathsf{T}(F\otimes I_K)h-\sqrt{L}h^\mathsf{T}(F^\mathsf{T}\otimes I_K)\hat{a}\\ &+Lh^\mathsf{T}(F^\mathsf{T}\otimes I_K)(F\otimes I_K)h)\end{aligned} \tag{13}$$

where $\mathcal{H}$ is the temporary name of the minimizing function. We find the minimum value by taking the derivative with respect to h,

$$\begin{aligned}\frac{\partial\mathcal{H}}{\partial h} &= \frac{1}{2}(w^\mathsf{T}\otimes I_K)(w\otimes I_K)h+\frac{1}{2}(w^\mathsf{T}\otimes I_K)(w\otimes I_K)h\\ &-\sqrt{L}(F^\mathsf{T}\otimes I_K)\hat{\varsigma}-\frac{\mu}{2}\sqrt{L}(F^\mathsf{T}\otimes I_K)\hat{a}\\ &-\frac{\mu}{2}\sqrt{L}(F^\mathsf{T}\otimes I_K)\hat{a}+\frac{\mu}{2}L(F^\mathsf{T}\otimes I_K)(F\otimes I_K)h\\ &+\frac{\mu}{2}L(F^\mathsf{T}\otimes I_K)(F\otimes I_K)h\end{aligned} \tag{14}$$

And because of $(A\otimes B)(C\otimes D)=(AC\otimes BD)$, we have,

$$\begin{aligned}\frac{\partial\mathcal{H}}{\partial h} &= (w^\mathsf{T}w\otimes I_K)h-\sqrt{L}(F^\mathsf{T}\otimes I_K)\hat{\varsigma}\\ &-\sqrt{L}(F^\mathsf{T}\otimes I_K)\hat{a}+\mu Lh\\ &= 0\end{aligned} \tag{15}$$

Therefore,

$$h^* = L\frac{\varsigma+\mu a}{w^\mathsf{T}w+\mu L} \tag{16}$$

where $a=\frac{1}{\sqrt{L}}(F^\mathsf{T}\otimes I_K)\hat{a}$ and $\varsigma=\frac{1}{\sqrt{L}}(F^\mathsf{T}\otimes I_K)\hat{\varsigma}$. We efficiently estimate a and ς by applying the Inverse Discrete Fourier transform (IDFT). The Kronecker product with the identity matrix can be obtained by K independent IDFT computations, $a^k=\frac{1}{\sqrt{L}}F^\mathsf{T}\hat{a}^k$ and $\varsigma^k=\frac{1}{\sqrt{L}}F^\mathsf{T}\hat{\varsigma}$. The over-complete vectors a and ς can be obtained by concatenating $\{a^k\}_{k=1}^K$ and $\{\varsigma^k\}_{k=1}^K$, respectively.

2) SUBPROBLEM $\hat{a}^*$

$$\begin{aligned}\hat{a}^* = \arg\min_{\hat{a}} \Big\{&\frac{1}{2}\|\hat{y}-\hat{X}\hat{a}\|^2+\hat{\varsigma}^\mathsf{T}(\hat{a}-\sqrt{L}(F\otimes I_k)h)\\ &+\frac{\mu}{2}\|\hat{a}-\sqrt{L}(F\otimes I_K)h\|^2\Big\}\end{aligned} \tag{17}$$

$\hat{X}$ is sparse banded, and thus, each element of $\hat{y}$ ($\hat{y}[l]$, $l=1,\ldots,L$) is dependent only on K values of the filter $\hat{a}[l]=[conj(\hat{a}^1[l]),\ldots,conj(\hat{a}^K[l])]^\mathsf{T}$ and sample $\widehat{\chi_\theta(x)}[l]=[\widehat{\chi_\theta^1(x)}[l],\ldots,\widehat{\chi_\theta^K(x)}[l]]^\mathsf{T}$. Therefore, solving (17) for $\hat{a}^*$ can be identically expressed as L smaller, independent objectives, solving for $\hat{a}[l]$, over $l=[1,\ldots,L]$,

$$\begin{aligned}\hat{a}[l]^* = \arg\min_{\hat{a}(l)} \Big\{&\frac{1}{2}\|\hat{y}[l]-(\widehat{\chi_\theta(x)}[l])^\mathsf{T}\hat{a}[l]\|^2\\ &+\hat{\varsigma}[l]^\mathsf{T}(\hat{a}[l]-\hat{h}[l])+\frac{\mu}{2}\|\hat{a}[l]-\hat{h}[l]\|^2\Big\}\end{aligned} \tag{18}$$

where $\hat{h}[l]=[\widehat{\chi_\theta^1(h)}[l],\ldots,\widehat{\chi_\theta^K(h)}[l]]$ and $\widehat{\chi_\theta^k(h)}=\sqrt{L}F\chi_\theta^k(h)$. In practice, we estimate each $\widehat{\chi_\theta^k(h)}$ by applying an efficient DFT of each $\chi_\theta^k(h)$. The solution for each $\hat{a}[l]$ is obtained by the following formula,

$$\hat{a}[l]^* = \frac{\hat{y}[l]\widehat{\chi_\theta(x)}[l]-\sqrt{L}\hat{\varsigma}[l]+\sqrt{L}\mu\hat{h}[l]}{(\widehat{\chi_\theta(x)}[l])(\widehat{\chi_\theta(x)}[l])^\mathsf{T}+\sqrt{L}\mu I_K} \tag{19}$$

Since $(\widehat{\chi_\theta(x)}[l])(\widehat{\chi_\theta(x)}[l])^\mathsf{T}$ is rank-1 matrix, (19) can be solved with the Sherman Morrsion formula [33], by following

formula,

$$\hat{a}[l]^* = \frac{1}{\sqrt{L}\mu}\left(I_K - \frac{(\widehat{\chi_\theta(x)}[l])(\widehat{\chi_\theta(x)}[l])^{\mathsf{T}}}{(\widehat{\chi_\theta(x)}[l])^{\mathsf{T}}(\widehat{\chi_\theta(x)}[l]) + \sqrt{L}\mu}\right) Q \quad (20)$$

where $Q = \hat{y}[l]\widehat{\chi_\theta(x)}[l] - \sqrt{L}\hat{\varsigma}[l] + \sqrt{L}\mu\hat{h}[l]$. Note that (20) only contains vector multiply-add operation and thus can be computed efficiently.

3) LAGRANGE MULTIPLIER UPDATE

The Lagrangians $\hat{\varsigma}$ is updated as follows,

$$\hat{\varsigma}^{(i+1)} \leftarrow \hat{\varsigma}^{(i)} + \mu(\hat{a}^{(i+1)} - \hat{h}^{(i+1)}) \quad (21)$$

where $\hat{a}^{(i+1)}$ and $\hat{h}^{(i+1)}$ indicate the current solutions to the above subproblems at iteration $i+1$ within the iterative ADMM, and $\hat{h}^{(i+1)} = \sqrt{L}(F \otimes I_K)h^{(i+1)}$.

4) UPDATING STEPSIZE PARAMETER

The scheme for selecting the stepsize parameter μ is the following,

$$\mu^{(i+1)} = \min(\mu_{\max}, \beta\mu^{(i)}) \quad (22)$$

where $\mu_{\max}$ represents the maximum value of μ and β denotes the scale factor.

D. ONLINE TRACKING

1) SCALE ESTIMATION

A good scale adaption mechanism could enhance the tracking performance significantly. Inspired by SAMF [34], we apply the Siamese networks on multiple resolutions of the search region, and build a scale pyramid to deal with the scale changes in videos. We fix the size of training region U_T and the scaling pool for the testing region is defined as $\Upsilon = \{\gamma_1, \gamma_2, \ldots, \gamma_S\}$, where S represents the number of scales. Intuitively, when a new frame comes out, we firstly crop the search region z with different scales $\{U_T\gamma_1, U_T\gamma_2, \ldots, U_T\gamma_S\}$ around the target center of the last frame. Then we employ bilinear-interpolation to resize z into the fixed template size U_T.

2) INCORPORATING SALIENCY

In order to compute our saliency prior context model, we input the multi-scale images into the Siamese networks. Then we extract the feature map from the convolution layer and the confidence map from the score layer, respectively. The confidence scores are encoded as the priori probability $G(z)$, the salient matrix $M(z)$ is then initialized by $G(z)$ and $\varpi_\sigma(z - z_c)$, where z_c is the center of the searching region z. Finally, we map the salient matrix to the feature space by Kronecker product to calculate the final response map. The localization of the target is estimated on the highest peak of the response map. The final response map to find the proper target is formulated as following,

$$g_{\theta,s}(z) = \underset{s=1,2,\ldots,S}{\arg\max}\,[\mathcal{F}^{-1}(\mathcal{F}^*(\chi_\theta(h)) \circ \mathcal{F}(M(z^{\gamma_s}) \otimes \chi_\theta(z^{\gamma_s})))] \quad (23)$$

where $M(z^{\gamma_s}) = G(z^{\gamma_s}) \otimes \varpi_\sigma(z^{\gamma_s} - z_c)$. z^{γ_s} refers to the search region with the size of $U_T\gamma_s$, which is resized to U_T. $\mathcal{F}$ and $\mathcal{F}^{-1}$ denotes the discrete Fourier transform (DFT) and inverse discrete Fourier transform (IDFT), respectively. Note that $\otimes$ here indicates that the dimension of saliency prior probability is inconsistent with the dimension of features.

E. OCCLUSION-AWARE UPDATE

Most existed trackers update tracking models in every frame or every several frames. It is efficient but it does not consider the occlusion occurred during tracking, which means that the model may be updated by a polluted new template and cause tracking failure. In our work, we use a nearest neighbor (NN) classifier to determine whether a new sample is occluded or not. If noise or occlusion exists in tracking results, the model will not be updated. Therefore, the occlusion in tracking results can be handled.

After getting the best candidate target of each frame by formula (23), we reconstruct the target by a Gaussian Mixture Model (GMM),

$$\mathbf{M} = \sum_{n=1}^{\mathsf{N}} \omega_n \mathcal{N}(D|\vartheta_n; \Sigma_n) \quad (24)$$

where $D = \{x_1, x_2, \ldots, x_{t-1}\}$ is the collection of target objects up to the frame $t-1$, and x_i is the i-th local patch extracted from the new sample. N is the number of Gaussian components $\omega_n\mathcal{N}(D|\vartheta_n; \Sigma_n)$, ω_n is the prior weight of the n-th component, ϑ_n is its mean, and Σ_n is the covariance matrix. In order to avoid introducing bad samples into the online template, we should check whether each new sample is reasonable before updating positive samples. The reconstruction errors between new samples and templates are calculated, and the sample with a large reconstruction error is regarded as occlusion or unreasonable. We compute the reconstruction error of a new sample as following,

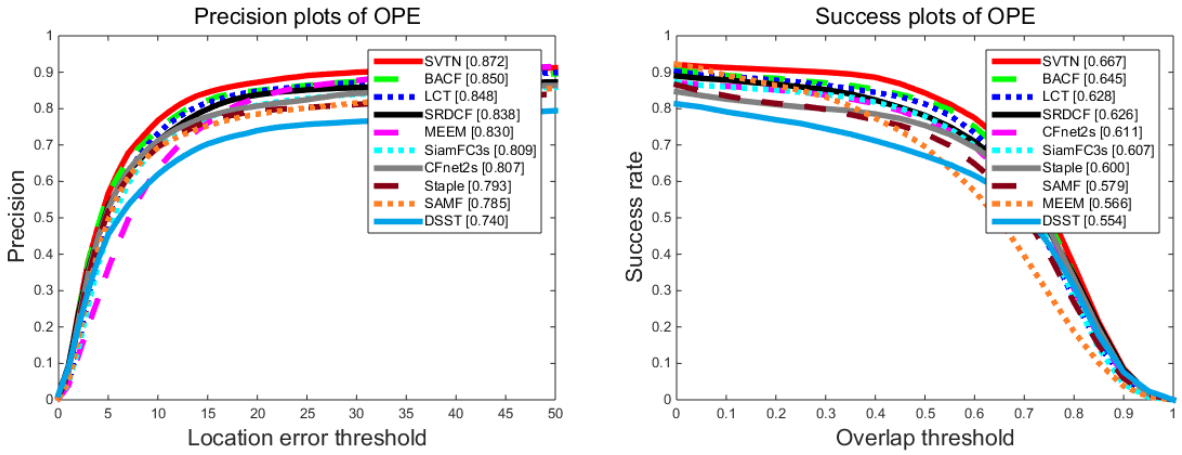
$$\delta = \sum_{n=1}^{\mathsf{N}} \|x_t - \omega_n\mathcal{N}(D|\vartheta_n; \Sigma_n)\|^2 \quad (25)$$

where x_t is the local patch extracted from the current frame t. If $\delta > \delta_0$, the new sample is regarded as occlusion, where δ_0 is a predefined threshold which distinguishes whether the detection is accurate or not. We discard these polluted target representations to avoid corruption in the model. The brief process of the proposed SVTN is shown in Algorithm 1.

IV. EXPERIMENT AND ANALYSIS

A. IMPLEMENTATION DETAILS

The approach proposed in this paper is implemented in MATLAB R2017b on the Ubuntu 16.04 x64 system with an Intel Core CPU (E5-2683 v3 2.00 GHz), a NVIDIA GPU (GTX1080TI) and 32 GB DDR4 RAM. The settings of the Siamese networks in this paper are the same as that in [7], except that we utilize boundary pixels to supplement areas beyond the image instead of the mean. In tracking, we crop the searching patch centered at the target, in which the side



(a) (b)

FIGURE 3. The precision plot (a) and success plot (b) of OPE (one pass evaluation) on OTB-2013 dataset for 10 trackers. The distance precision and overlap precision of each tracker are reported in brackets. Best viewed on the colored curves.

Algorithm 1 Our Proposed Tracking Method

Initial target bounding box $x_0 = (\mathrm{x}_0, \mathrm{y}_0, \mathrm{w}_0, \mathrm{h}_0)$ and other parameters;
Initial the target appearance model $\mathbf{M} = x_0$, and the filter $\widehat{\chi_\theta(h)}$;
for frame = 2, 3,..., until the last frame **do**
  Build the target pyramid around $(\mathrm{x}_{t-1}, \mathrm{y}_{t-1})$ and crop out the searching window z_t from the entire frame;
  Extract the feature map $\widehat{\chi_\theta(z_t)}$ and the confidence map $G(z_t)$ via Siamese networks;
  Build the saliency prior context model $M(z_t)$;
  Compute the correlation response map $g_\theta(z_t)$;
  Estimate the optimal scale s_t, the bounding box $x_t = (\mathrm{x}_t, \mathrm{y}_t, \mathrm{s}_t)$;
  Calculate the confidence of tracking results with formula (25);
  if $\delta < \delta_0$ **then**
    while ADMM iteration **do**
      Update subproblem $\hat{a}^{(i+1)}$ using $\hat{h}^{(i)}$ and $\hat{\varsigma}^{(i)}$, formula (20);
      Obtain $a^{(i)}$ and $\varsigma^{(i)}$ via inverse fast fourier transform;
      Calculate the subproblem $h^{(i+1)}$ in (16);
      Compute the Lagrangians $\hat{\varsigma}^{(i+1)}$, formula (21);
      Update the stepsize parameter $\mu^{(i+1)}$, formula (22);
    end while
    Update $\mathbf{M}$ and $\widehat{\chi_\theta(h)}$;
  end if
end for

length of the region is $\sqrt{5\mathrm{wh}}$ (w and h represent the width and height of the target, respectively), and the searching patch is resized to fixed size 255×255. We initialize the branch x of the Siamese networks in the first frame, and never change it again. In subsequent image frames, we extract the deep features from branch z, and these features are used to train the correlation filters. The online learning rate of feature representations η is set to 0.013 for all experiments. A 2D Gaussian function with bandwidth of $\sqrt{\mathrm{wh}}/16$ is used to define the correlation output for an object of size $[\mathrm{w}, \mathrm{h}]$. For the ADMM optimization, the initial stepsize parameter μ_0, the maximum value $\mu_{\max}$ and scale factor β are set to 2, 10 and 10^3, respectively. The number of scales is set to 5 with a scale-step of 1.01. The threshold for occlusion-aware update, δ_0, is set to 3.2.

B. COMPARISON ON OTB BENCHMARKS

In order to evaluate our STVN tracker, we employ all the sequences on OTB-2013 benchmark and OTB-2015 benchmark of Wu et al. [35], [36]. The organizers of the benchmarks maintain a large number of latest algorithm results on the public dataset, and recommend some metrics to evaluate all trackers. In this paper, all the tracking algorithms are evaluated based on the precision metric and the success metric with one pass evaluation (OPE). Precision metric indicates the percentage of frame locations within a certain threshold distance from those of the ground truth. The threshold distance is set as 20 for all trackers. Success measures the intersection over union (IoU) of ground truth and predicted bounding boxes. The success plot shows the rate of bounding boxes whose IoU score is larger than a given threshold. Area Under the Curve (AUC) of success plots is applied to rank the trackers. We provide a comparison of our tracker with several state-of-the-art methods, including: DSST [22], MEEM [37], SAMF [34], Staple [38], LCT [39], SRDCF [23], SiamFC3s [18], CFnet2s [7] and BACF [24].

1) OVERALL PERFORMANCE

Figure 3 and Figure 4 illustrates the overall performance of the 10 trackers in terms of the precision metric and success

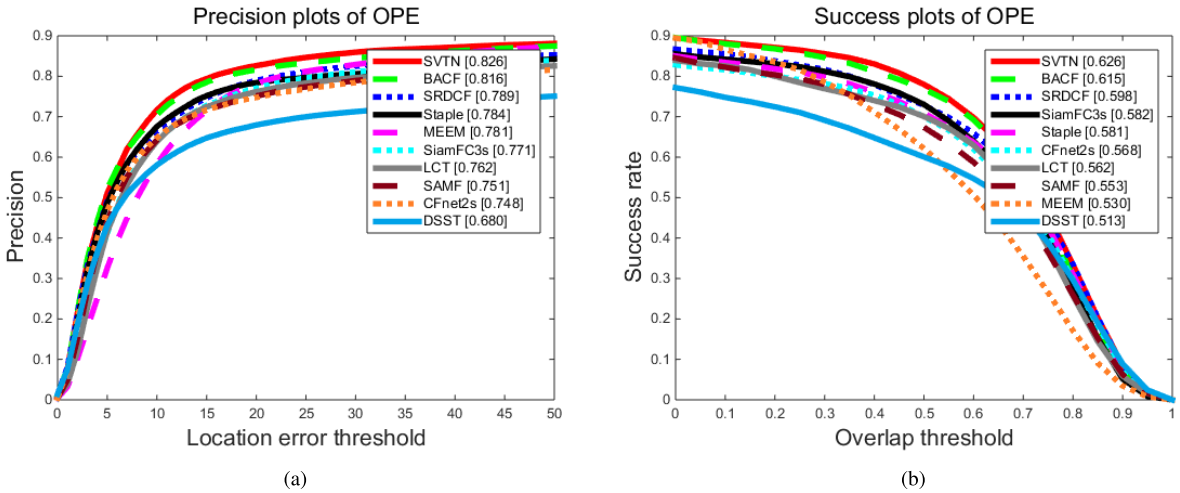


FIGURE 4. The precision plot (a) and success plot (b) of OPE (one pass evaluation) on OTB-2015 dataset for 10 trackers. The distance precision and overlap precision of each tracker are reported in brackets. Best viewed on the colored curves.

TABLE 1. VOT-2016 performance results. Red fonts indicate the best performance, the blue fonts indicate the second best ones and the green fonts indicate third ones.

Tracker	A-R rank		EAO	Speed	
	Accuracy	Robustness	EAO	FPS	EFO
SVTN	0.546	18.814	0.346	13.266	8.043
Staple	0.543	23.895	0.295	14.433	10.975
TGPR	0.452	41.012	0.181	0.207	0.312
DeepSRDCF	0.523	20.346	0.276	65.303	47.433
CCOT	0.533	16.582	0.331	82.182	50.998
DSST	0.527	44.814	0.181	13.900	9.714
SAMF	0.498	37.794	0.186	5.215	3.649

metric. One can see that the proposed method ranks first on both OTB-2013 and OTB-2015 dataset. On OTB-2013, the proposed algorithm achieves a score of 0.872 and 0.667 on the precision plot and the success plot, respectively, exceeding the baseline CFnet2s tracker more than 6.0%. Because the OTB-2015 dataset contains more complex video sequences, the results on OTB-2015 are slightly lower than that on OTB-2013, but our algorithm also leads to a significant gain of more than 5.0% compared to the baseline tracker.

2) ATTRIBUTE-BASED EVALUATION

We analyze the tracker performance using 11 annotated attributes on OTB-2015 dataset: Illumination Variation (IV), Out-of-Plane Rotation (OPR), Scale Variation (SV), Occlusion (OCC), Deformation (DEF), Motion Blur (MB), Fast Motion (FM), In-Plane Rotation (IPR), Out-of-View (OV), Background Clutter (BC), and Low Resolution (LR). Figure 5 and Figure 6 presents the results under one-pass evaluation regarding these challenging attributes for visual object tracking. It's worth noting that our STVN tracker achieves the best performance on 6 out of 11 attributes on both precision plots and success plots. Further, our method outperforms the baseline CFnet2s on 10 attributes on both metrics. For simplicity, we use the success metric to demonstrate the advantages of our algorithm. In case of scale variation, SAMF, a milestone tracker for handling target size changes, achieves a score of 49.2%. Our tracker provides a gain of 10% compared to SAMF, which is a significant improvement. Facing the challenges of OPR, our algorithm exceeds the second BACF by 2.0% in scores. Occlusion may pollute the target model and make it gradually corrupt. The proposed SVTN exploits an occlusion-aware model updating mechanism and obtains a good score of 58.4%. Although the proposed method can improve the tracking performance in most challenges, it should not be ignored that our tracker performs worse than the baseline in the LR sequences. Under the condition of low resolution, feature extraction of the model will be seriously affected, leading to tracking failure.

C. RESULTS ON VOT DATASET

We compare our tracker with 6 top participants on the VOT-2016 [40] dataset, including DSST [22], SAMF [34],

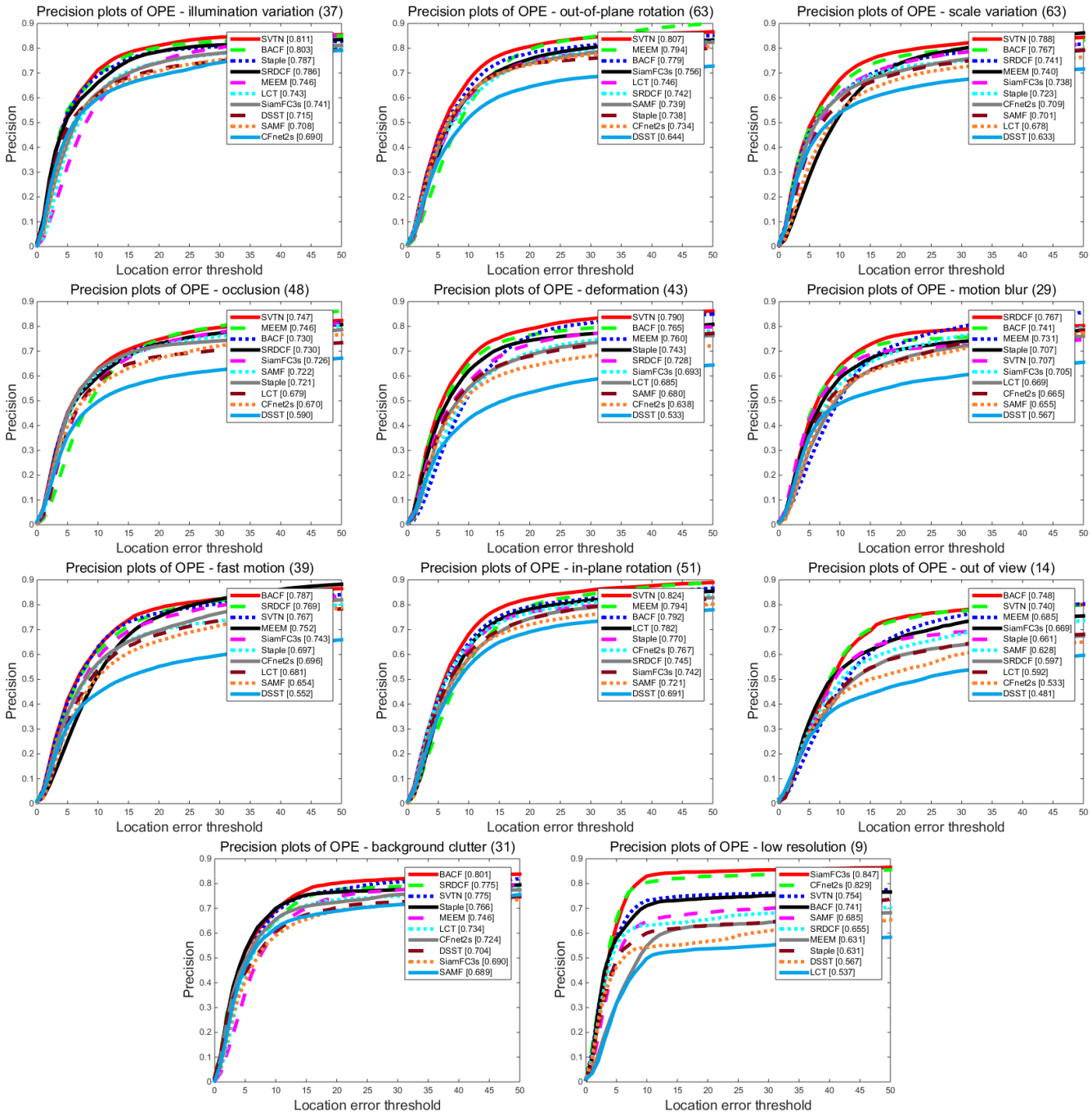


FIGURE 5. Attribute-based analysis of our approach on the OTB-2015 dataset with all videos. Precision plots are shown for 11 attributes: IV, OPR, SV, OCC, DEF, MB, FM, IPR, OV, BC, and LR. Number of videos for each attribute is appended to the end of each plot title.

CCOT [41], DeepSRDCF [42], TGPR [43], and Staple [38]. The results of these comparison algorithms are provided by the VOT website http://www.votchallenge.net/. In this paper, four primary measures are used to analyze tracking performance: Accuracy (A), Robustness (R), Expected Average Overlap (EAO) and Equivalent Filter Operation (EFO). In the baseline experiments, whenever a tracker predicts a bounding box with zero overlap with the ground truth, a failure is detected and the tracker will be re-initialized five frames after the failure [40].The Accuracy (A) is the average overlap between the predicted and ground truth bounding boxes during successful tracking periods. On the other hand, the Robustness (R) measures how many times the tracker loses the target (fails) during tracking. The Expected Average Overlap (EAO) is an estimator of the average overlap a tracker is expected to attain on a large collection of short-term sequences with the same visual properties as the given dataset. Finally, Equivalent Filter Operation (EFO)

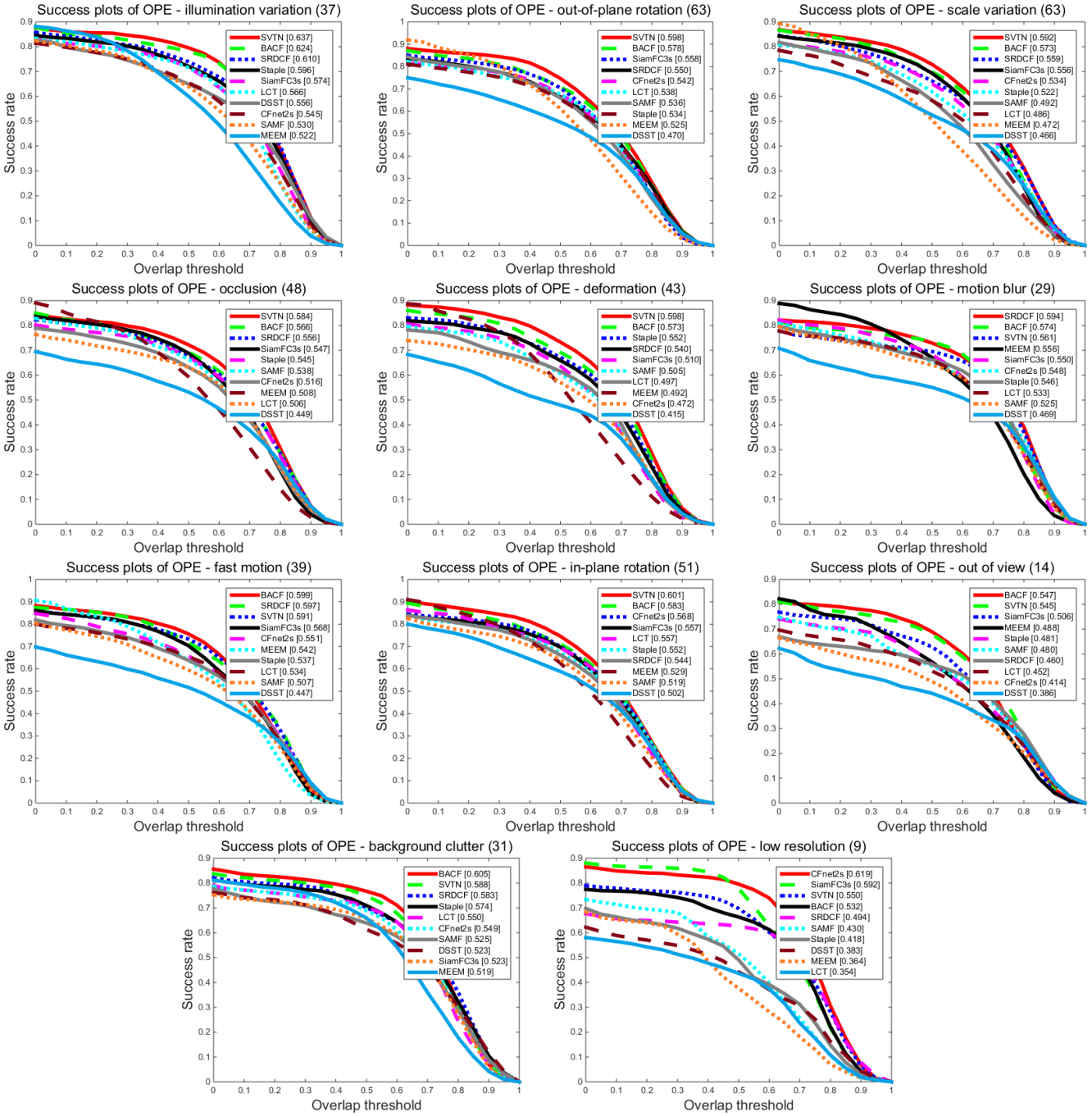


FIGURE 6. Attribute-based analysis of our approach on the OTB-2015 dataset with all videos. Success plots are shown for 11 attributes: IV, OPR, SV, OCC, DEF, MB, FM, IPR, OV, BC, and LR. Number of videos for each attribute is appended to the end of each plot title.

reports the tracker speed in terms of a predefined filtering operation that the toolkit automatically carries out prior to running the experiments. The results of the mentioned metrics are shown in Table 1. Our SVTN obtains the first place of Accuracy metric and EAO metric, and the second place of Robustness metric. The experimental results show that our tracker can not only achieve good accuracy but also appear very robust. On the other hand, our complex framework results in a certain amount of computational cost, so we do not have an advantage in the comparison of tracking speed.

D. COMPARISON TO SIAMESE BASELINE

For further analyses on the tracking performance, we also demonstrate the advantages of our algorithm through the baseline comparison on sequences of the OTB-2013 and

TABLE 2. Performance results produced by the OTB toolkit for the OTB-2013 and OTB-2015 datasets. Red fonts indicate the best performance, the blue fonts indicate the second best ones and the green fonts indicate third ones.

Method	Layer	OTB-2013		OTB-2015	
		Precision	Success	Precision	Success
CFnet1s	Conv1	0.776	0.578	0.713	0.536
CFnet2s	Conv2	0.807	0.611	0.748	0.568
CFnet5s	Conv5	0.785	0.589	0.777	0.586
SVTN1s (Ours)	Conv1	0.762	0.579	0.757	0.569
SVTN2s (Ours)	Conv2	0.826	0.630	0.819	0.620
SVTN5s (Ours)	Conv5	0.872	0.667	0.826	0.626

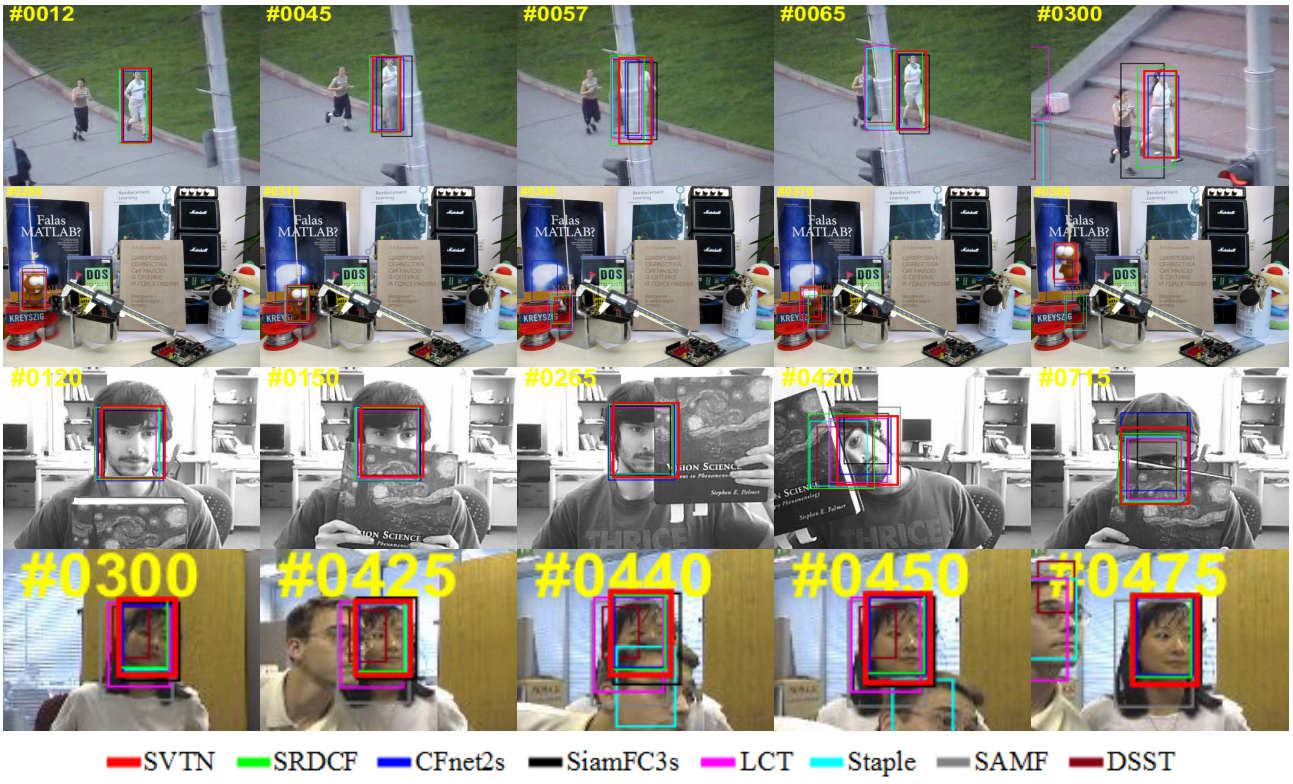


FIGURE 7. Visual comparison of our approach with 7 state-of-the-art trackers on the "Jogging-2", "Lemming", "FaceOcc2" and "Girl" video sequences. Occlusion is the main challenge illustrated in the figure.

OTB-2015 dataset. We compare our algorithm with CFnet using different AlexNet layers, including Conv1, Conv2 and Conv5. The complete comparisons of Precision and Success scores with 3 different layers are illustrated in Table 2. It's worth noting that our SVTN5s and SVTN2s achieve the best and the second performance on both OTB-2013 and OTB-2015 dataset. The CFnet2s tracker achieves the third performance on OTB-2013, while the CFnet5s tracker obtains the third place on OTB-2015. Although the performance of our SVTN1s is not quite good, our SCCF and SPC model also play a role in improving performance. It is worth noting that the STVN algorithm mentioned above refers to SVTN5s.

E. VISUAL COMPARISONS

For visual comparisons, we evaluate our SVTN with 7 state-of-the-art trackers, including SiamFC3s [18], CFnet2s [7], DSST [22], SAMF [34], LCT [39], SRDCF [23], and Staple [38]. SiamFC3s and CFnet2s are the baseline tracking methods. DSST and SAMF are designed to handle scale variations. LCT algorithm performs strong robustness in case of long term tracking, which can deal with the challenge of heavy occlusion. SRDCF can address boundary effects in the standard DCF trackers. Staple tracker applies more comprehensive features, which combines HOG with color. To better analyze the effectiveness and robustness of the proposed tracker, we mainly discuss the performance over

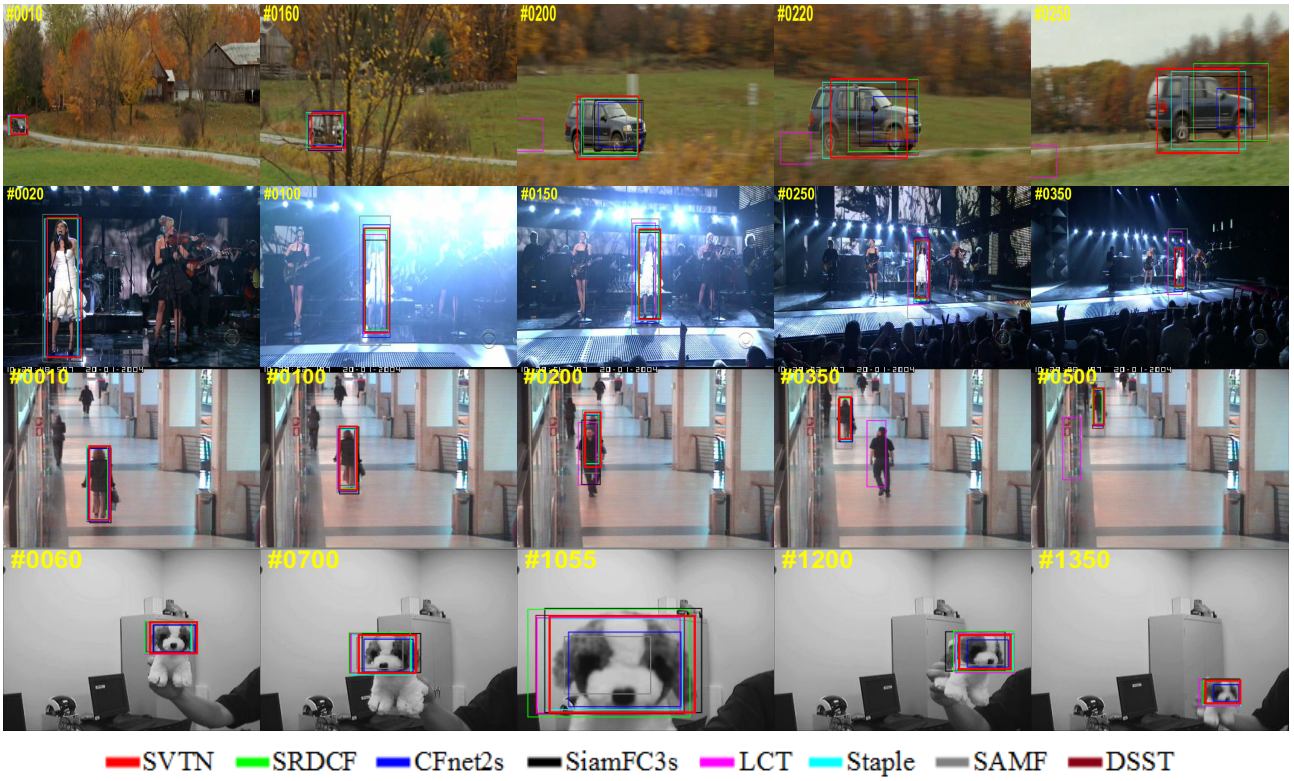


FIGURE 8. Visual comparison of our approach with 7 state-of-the-art trackers on the "CarScale", "Singer1", "Walking2" and "Dog1" video sequences. Scale Variation is the main challenge illustrated in the figure.

most common challenging factors, namely Occlusion, Scale Variation, Rotation and Background Clutter. We choose some example sequences from the OTB-2015 dataset to illustrate our experimental results.

1) EVALUATION OF OCCLUSION HANDLING

The visual evaluation results are shown in Figure 7. In the "Jogging-2" scene, the target is occluded for a very short time, and then the four trackers lose the target, but our tracker can follow the target tightly. In "Lemming" sequences, the target is occluded for a period of time, and our tracker is able to cope with this situation too. The performance of the partial occlusion is shown in the "FaceOcc2" and "Girl" video sequences, our algorithm is also outstanding. The reason why our algorithm can outperform than other trackers is that we exploit an occlusion-aware update strategy.

2) EVALUATION OF SCALE HANDLING

Scale handling tests are shown in Figure 8. The target in "CarScale" sequences changes from small to large, and many trackers are unable to adapt to this change, resulting in smaller and smaller overlap. Similarly, the target appearance in the "Singer1" and "Walking2" sequences become smaller and smaller, which makes it difficult to obtain the bounding box with high overlap with the ground truth. In "Dog1" video sequences, the target size changes repeatedly, which leads to the scale drift of many trackers. As we utilize a scaling pool for the searching region, our tracker can capture the target with different scales for further selection. Therefore, Our tracker handles these situations well.

3) EVALUATION OF ROTATION HANDLING

The challenge of rotation is caused by the movement of the target or the change of the viewpoint, and this challenge makes it difficult to model the appearance of the target. The scene in the "Basketball" video sequences shows the situation in which the targets suffer Out-of-Plane Rotation (OPR), while the target in "Diving" video sequences rotates in the image plane. In "Sylvester" and "DragonBaby" sequences, the trackers face the challenges of OPR and IPR at the same time. As shown in Figure 9, our tracker can cope with the problem of rotation well. This benefits from the use of deep Siamese features, which are robust when the target rotates.

4) EVALUATION OF BACKGROUND CLUTTER HANDLING

The target undergoes the Background Clutter (BC) challenge in Figure 10, the bounding box may drift onto the background, as it is difficult to distinguish between the target object and the background by a fairly simple model. The changing lights in "Shaking" and "Singer2" scenes make it even more difficult to distinguish the target from the background, and the bright stadium in "Football1" and "Bolt" videos also makes the feature of the target less obvious. To handle this problem, one must enhance the robustness

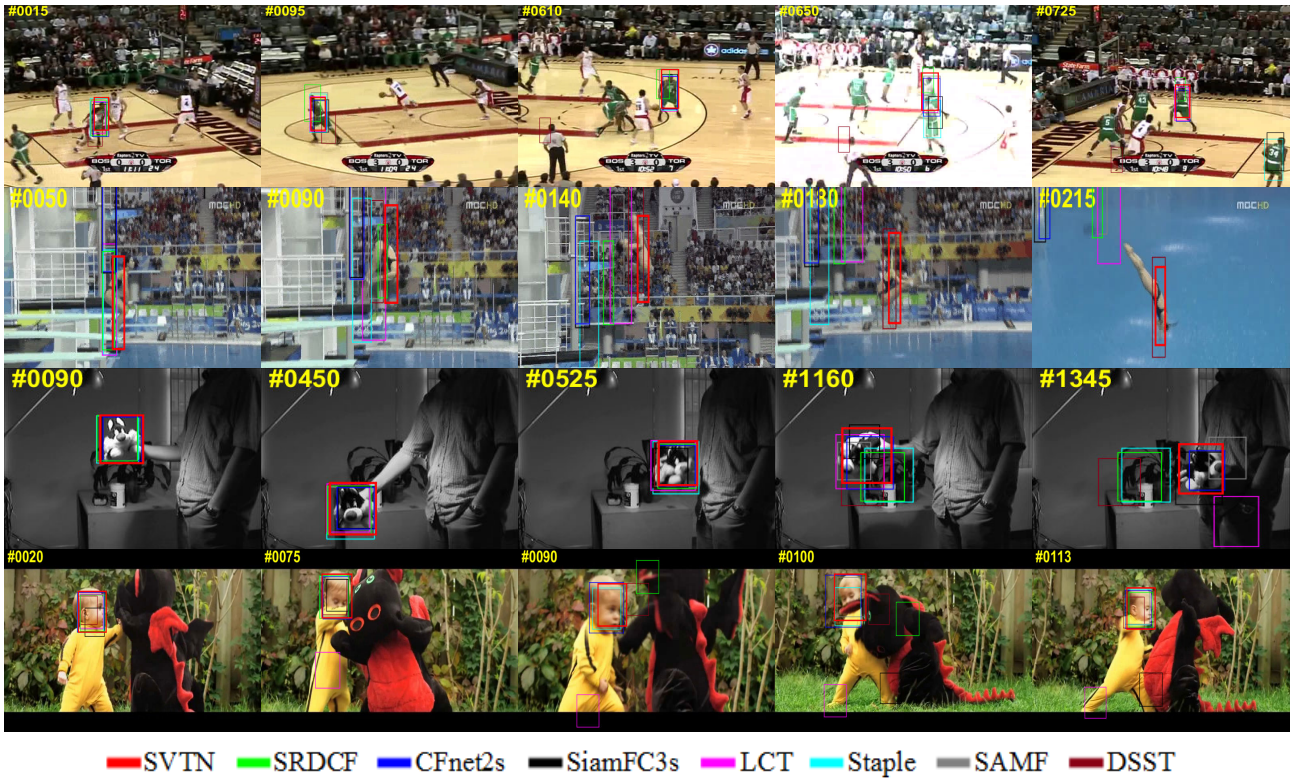


FIGURE 9. Visual comparison of our approach with 7 state-of-the-art trackers on the "Basketball", "Diving", "Sylvester" and "DragonBaby" video sequences. Rotation is the main challenge illustrated in the figure.

FIGURE 10. Visual comparison of our approach with 7 state-of-the-art trackers on the "Shaking", "Singer2", "Football1" and "Bolt" video sequences. Background Clutter is the main challenge illustrated in the figure.

of target appearance modeling. Our tracker tracks the target tightly when BC challenge occurs, which means that our SPC model can suppress the position drift caused by background interference through the spatio-temporal relationship in video.

V. CONCLUSION

In this paper, we employ the Spatially Constrained Correlation Filter (SCCF) and the Saliency Prior Context (SPC) model for Siamese visual tracking. Our tracker not only enhances the ability of decision-making module, but also solves the boundary effect well. The solution of our SCCF is optimized via the Alternating Direction Method of Multipliers (ADMM) so that we can implement spatial penalties in the spatial domain and compute correlation filters in the frequency domain at the same time. The SPC model makes full use of the spatio-temporal prior information, so that the position drift caused by similar objects can be well suppressed. Furthermore, we suggest an occlusion-aware update strategy to avoid the model corruption problem. The experimental results demonstrate that the proposed tracker performs superiorly against several state-of-the-art algorithms on OTB-2013, OTB-2015 and VOT-2016 dataset.

ACKNOWLEDGMENT

The authors sincerely acknowledge the editor and the anonymous reviewers for their insightful comments on the manuscript.

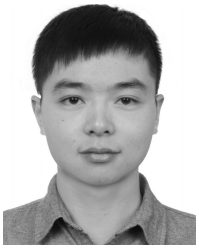
REFERENCES

[1] N. Wang and D.-Y. Yeung, "Learning a deep compact image representation for visual tracking," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 809–817.
[2] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3119–3127.
[3] 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.
[4] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Robust visual tracking via hierarchical convolutional features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2709–2723, Nov. 2019.
[5] H. Li, Y. Li, and F. Porikli, "DeepTrack: Learning discriminative feature representations online for robust visual tracking," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1834–1848, Apr. 2016.
[6] 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.
[7] 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.
[8] D. Held, S. Thrun, and S. Savarese, "Learning to track at 100 FPS with deep regression networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 749–765.
[9] Y. Song, C. Ma, L. Gong, J. Zhang, R. W. H. Lau, and M.-H. Yang, "CREST: Convolutional residual learning for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2555–2564.
[10] C. Sun, D. Wang, H. Lu, and M.-H. Yang, "Correlation tracking via joint discrimination and reliability learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 489–497.
[11] C. Sun, D. Wang, H. Lu, and M.-H. Yang, "Learning spatial-aware regressions for visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8962–8970.
[12] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 597–606.
[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] Y. Qi, S. Zhang, L. Qin, H. Yao, Q. Huang, J. Lim, and M.-H. Yang, "Hedged deep tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4303–4311.
[15] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8971–8980.
[16] Q. Wang, Z. Teng, J. Xing, J. Gao, W. Hu, and S. Maybank, "Learning attentions: Residual attentional siamese network for high performance online visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4854–4863.
[17] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. S. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 1328–1338.
[18] 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.* Cham, Switzerland: Springer, 2016, pp. 850–865.
[19] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.
[20] R. Tao, E. Gavves, and A. W. Smeulders, "Siamese instance search for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1420–1429.
[21] D. S. 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.
[22] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. Brit. Mach. Vis. Conf.*, Nottingham, U.K., Sep. 2014, pp. 1–11.
[23] 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.*, Dec. 2015, pp. 4310–4318.
[24] 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. 1135–1143.
[25] 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.
[26] V. Mahadevan and N. Vasconcelos, "Saliency-based discriminant tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1007–1013.
[27] D. Tran and J. Yuan, "Optimal spatio-temporal path discovery for video event detection," in *Proc. CVPR*, Jun. 2011, pp. 3321–3328.
[28] J. Fan, Y. Wu, and S. Dai, "Discriminative spatial attention for robust tracking," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2010, pp. 480–493.
[29] K. Zhang, Z. Lei, M. H. Yang, and D. Zhang, "Fast tracking via spatio-temporal context learning," 2013, *arXiv:1311.1939*. [Online]. Available: https://arxiv.org/abs/1311.1939
[30] G. Zhu, J. Wang, Y. Wu, X. Zhang, and H. Lu, "MC-HOG correlation tracking with saliency proposal," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 1–7.
[31] C. Ma, Z. Miao, X.-P. Zhang, and M. Li, "A saliency prior context model for real-time object tracking," *IEEE Trans. Multimedia*, vol. 19, no. 11, pp. 2415–2424, Nov. 2017.
[32] B. V. Kumar, A. Mahalanobis, and R. D. Juday, *Correlation Pattern Recognition*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
[33] J. Sherman and W. J. Morrison, "Adjustment of an inverse matrix corresponding to a change in one element of a given matrix," *Ann. Math. Statist.*, vol. 21, no. 1, pp. 124–127, 1950.
[34] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 254–265.
[35] 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.
[36] 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.

[37] J. Zhang, S. Ma, and S. Sclaroff, "MEEM: Robust tracking via multiple experts using entropy minimization," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 188–203.
[38] 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.
[39] C. Ma, X. Yang, C. Zhang, and M.-H. Yang, "Long-term correlation tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5388–5396.
[40] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Č. Zajc, T. Vojír, G. Häger, A. Lukežič, G. Fernández, A. Gupta, A. Petrosino, A. Memarmoghadam, A. Garcia-Martin, and A. S. Montero, "The visual object tracking VOT2016 challenge results," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Oct. 2016, pp. 777–823. [Online]. Available: http://www.springer.com/gp/book/9783319488806
[41] 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.* Cham, Switzerland: Springer, 2016, pp. 472–488.
[42] 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. Workshops*, Dec. 2015, pp. 58–66.
[43] J. Gao, H. Ling, W. Hu, and J. Xing, "Transfer learning based visual tracking with Gaussian processes regression," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 188–203.

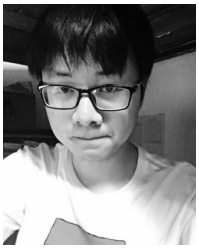
BO HUANG is currently pursuing the Ph.D. degree with the School of Optoelectronics, Beijing Institute of Technology, Beijing, China. His research interests include computer vision and real-time image/video processing.

TINGFA XU received the Ph.D. degree from the Changchun Institute of Optics, Fine Mechanics and Physics, Changchun, China, in 2004. He is currently a Professor with the School of Optoelectronics, Beijing Institute of Technology, Beijing, China. His research interests include optoelectronic imaging and detection and hyper-spectral remote sensing image processing.



SHENWANG JIANG received the B.E. degree from the School of Optoelectronics, Beijing Institute of Technology, Beijing, China, in 2014, where he is currently pursuing the Ph.D. degree in optical engineering. His research interests include image classification, object detection, and image processing.

YU BAI is currently pursuing the M.E. degree with the School of Optoelectronics, Beijing Institute of Technology, Beijing, China. Her research interests include computer vision and real-time image/video processing.



YIWEN CHEN is currently pursuing the M.E. degree with the School of Optoelectronics, Beijing Institute of Technology, Beijing, China. His research interests include computer vision and real-time image/video processing.