Spatio-temporal Self-learning Object Tracking Model Based on Anti-occlusion Mechanism

Qiang Hu, Hongrun Wu, Jiahao Wu, Jinrong Shen, Haorong Hu, Yingpin Chen*, Lingzhi Wang*, and Hualin Zhang*

***Abstract*—The correlation filter tracking framework has attracted extensive attention from scholars in object tracking due to its efficient and accurate performance. However, this framework cannot adaptively perceive the target's effective spatio-temporal changes. Thus, the correlation filter is easily misled by the occlusion in the occluded scene, resulting in tracking failure. To solve the above limitation effectively, we propose a spatio-temporal self-learning object-tracking model based on an anti-occlusion mechanism. On the one hand, we design a spatially regularized factor based on the local response change vector to give the filter adaptive spatial adjustment ability. On the other hand, we construct a temporal regularized factor with adaptive adjustment capability based on a multimodal template pool to help the filter distinguish between effective appearance changes of the target and ineffective variations caused by occlusion. Then, the alternating direction multiplier method decomposes the proposed model into several simpler subproblems. A small number of iterative steps can obtain the optimal solution of the model. Finally, the OTB100 dataset is employed to compare the proposed method with several state-of-the-art tracking algorithms. Experiments show that the proposed method improves the tracking performance in occluded scenes.**

***Index Terms*—object tracking, spatio-temporal regularization, local response variation, multimodal template pool, anti-occlusion mechanism**

Manuscript received November 21, 2022; revised April 21, 2023.

This work was supported by National Natural Science Foundation of China (62106092); Natural Science Foundation of Fujian Province (2020J05169, 2022J01916); Innovation and Entrepreneurship Training Program for College Students (202210402009, S202210402038, S202210402025) and the Education Research Program of Minnan Normal University (202211).

Qiang Hu is a postgraduate student of Minnan Normal University, Zhangzhou, 363000 China. (e-mail: 137152437@qq.com).

Hongrun Wu is an associate professor of Minnan Normal University, Zhangzhou, 363000 China. (e-mail: wuhongrun@whu.edu.cn). (co-first author).

Jiahao Wu is an undergraduate student of Minnan Normal University, Zhangzhou, 363000 China. (e-mail: 1735024557@qq.com).

Jinrong Shen is an undergraduate student of Minnan Normal University, Zhangzhou, 363000 China. (e-mail: 1942613358@qq.com).

Haorong Hu is an undergraduate student of Minnan Normal University, Zhangzhou, 363000 China. (e-mail: 964983884@qq.com).

Yingpin Chen is an associate professor of Minnan Normal University, Zhangzhou, 363000 China. (e-mail: cyp1707@mnnu.edu.cn). (* correspondence author).

Lingzhi Wang is an associate professor of Xiamen City University, Xiamen, 361000 China. (e-mail: 64564254@qq.com). (* co-correspondence author).

Hualin Zhang is a professor of Minnan Normal University, Zhangzhou, 363000 China. (e-mail: 1072412086@qq.com). (* co-correspondence author).

I. Introduction

VISUAL object tracking is an important research direction of computer vision. The essence of its task is to simulate the observation of the human eye and the decision-making behavior of the human brain. In the object tracking task, it is necessary to provide the computer with the object's initial position and target size in the first frame. Then the computer is required to identify and track the object's position, size, and other states in the subsequent frames.

There are generative and discriminative-based models for object tracking. Generative models usually generate many random samples by employing the geometric state of the target in the previous frame as a reference[1, 2]. Then it evaluates the target state by reweighting the geometric states of the samples that are similar to the target. For example, zhou *et al*. [3] propose a novel online discriminative and low-rank dictionary learning technique, which can be seamlessly integrated into an online tracking framework for realizing more robust target localization. Unlike the generative model, the discriminant model adopts training classifiers to distinguish between targets and backgrounds and selects the sample with the highest classifier's score value as the tracking result [4].

Since Bolme *et al*. pioneered the minimum output sum of squared error (MOSSE) [5] filter-based tracking method, the discriminative correlation filter (DCF) has aroused extensive attention in academia and industry due to its excellent computational efficiency and tracking effect. Zhu *et al*. [6] proposed a complementary discriminative correlation filter technique based on collaborative characterization, which can respond to abrupt target appearance changes to achieve more robust target localization. Elayaperumal *et al*. [7] proposed the aberration suppression spatio-temporal correlation filter technique, which introduces temporal regularization and achieves a more robust appearance model. Lu *et al*. [8] proposed an adaptive region proposal scheme with feature channel regularization to facilitate robust object tracking. Henriques *et al*. proposed a target position estimation algorithm using the circulant structure with kernels (CSK) [9]. Based on the CSK and the histogram of oriented gradients (HOG), Henriques *et al*. proposed a kernel correlation filter (KCF) [10] high-speed tracking method, and the tracking effect is significantly improved. Although KCF achieves excellent tracking performance, it pre-sets a fixed target size and cannot adjust the target scale adaptively when the target size changes. In addition, scholars have fused the samples' hand-crafted features and deep features [11-14] to accurately

describe the appearance representation [15] of the target and background. However, there is a real-time problem in the employment of deep networks for object tracking. Therefore, further research is needed to reasonably simplify the deep networks and improve the tracking performance. Zhang *et al*. [16] proposed SCSTCF: Spatial-Channel Selection and Temporal Regularization Correlation Filter algorithm, which performs grouping feature selection from channel, spatial and temporal dimensions to establish relevance between multi-channel features and correlation filters.

Although the above methods have succeeded greatly, they also have limitations. (1) The traditional DCF methods adopt cyclic shifts to obtain negative samples, resulting in a lack of negative samples from real scenes [17]. As a result, traditional DCF methods greatly reduce the robustness of the tracker against cluttered backgrounds; (2) The traditional DCF methods are easy to fail in occlusion scenes. It is because the DCF framework identifies the target at the location with the maximum response and directly trains the filter with this sample. But ignoring the confidence level of this sample leads to a heavily contaminated training sample that misleads the tracker; (3) The traditional DCF methods have inferior adaptive learning ability with the change of spatio-temporal information. It fails to adaptively adjust the update speed of the filter according to the spatio-temporal variation of appearance. For example, Danelljan *et al*. proposed a spatially regularized discriminative correlation filter (SRDCF) [18] tracker to constrain the spatial distribution of the filter with a fixed spatial regularized term. Similarly, Li *et al*. proposed spatio-temporal regularized correlation filters (STRCF) [19], which cannot adaptively adjust the spatial regularization constraint parameters. Chen *et al*. proposed a correlation filter tracking algorithm based on visual attention learning and an anti-occlusion mechanism (VALACF) [20]. Therefore, this paper proposes a spatio-temporal self-learning (STSL) object tracking algorithm based on multimodal template pooling to solve the above limitations. On the one hand, this paper introduces background samples as real negative samples to participate in filter learning. Then we design the local response change matrix to perceive the variation of spatial appearance information according to the difference of adjacent frame response maps. In this way, an adaptive spatial regularized term with spatial perception capability is constructed. On the other hand, this paper constructs a multimodal template pool with hard positive samples in historical tracking results to evaluate the confidence of the best candidate samples. Thus, we avoid the contamination of the filter by the occlusion. If the best candidate sample is determined to be a hard positive sample, the learning of the filter is accelerated according to the local response change matrix. In summary, the main contributions of this paper are as follows:

(1) A spatial learning factor is introduced into the correlation filtering framework to limit the pixels with low confidence in the filter.

(2) A temporal learning factor is introduced in the correlation filtering framework to help the correlation filter distinguish effective appearance variations of the target and ineffective variations caused by occlusions.

(3) A historical multi-template target pool is established to avoid spatial filter contamination by unreliable samples, thus solving the problem of tracking algorithms failing due to occlusion during tracking.

The rest of this paper is organized as follows: Section II reviews the correlation filtering tracking method from the perspective of convolution. Section III elaborates on the technical details of the proposed method. Section IV provides multiple groups of experiments to verify the effectiveness of the proposed method. Finally, we summarize the paper and discuss the future research direction of the proposed model.

II. Correlation Filter Tracking in Convolutional View

The functional expression of correlation filter energy is as follows:

$$\varepsilon(\boldsymbol{f}_d^{(t)}) = \min_{\boldsymbol{f}_d^{(t)}} \frac{1}{2}\left\| \boldsymbol{y} - \sum_{d=1}^{D} \mathbf{vec}\left(\mathbf{mat}(\boldsymbol{x}_d^{(t)}) \star \mathbf{mat}(\boldsymbol{f}_d^{(t)})\right)\right\|_F^2 + \frac{\lambda}{2}\sum_{d=1}^{D}\left\|\boldsymbol{f}_d^{(t)}\right\|_F^2 \tag{1}$$

where $\boldsymbol{x}_d^{(t)} \in \mathbb{R}^{T\times 1}$ represents the feature of the d-th channel extracted from the training sample of the t-th frame (the matrix form of the feature is $\mathbf{mat}(\boldsymbol{x}_d^{(t)}) \in \mathbb{R}^{H\times W}$, and $T = H \times W$); $\boldsymbol{y} \in \mathbb{R}^{T\times 1}$ is the vectorized desired response with gaussian shape; $\boldsymbol{f}_d^{(t)} \in \mathbb{R}^{T\times 1}$ represents the d-th channel filter of the target; $\boldsymbol{y} \in \mathbb{R}^{T\times 1}$ represents the ideal response, which is generally set as a gaussian function; the symbol **vec** represents the matrix-to-column operator; the symbol **mat** represents the vector-to-matrix operator; the symbol $\star$ represents the two-dimensional correlation operator. If the matrix $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{M\times N}$, $\boldsymbol{Z} = \boldsymbol{X} \star \boldsymbol{Y}$, then $\boldsymbol{Z}(k,l) = \sum_{m=1}^{M}\sum_{n=1}^{N} \boldsymbol{X}(m,n)\boldsymbol{Y}(k+m-1, l+n-1)$. Because $\mathbf{mat}(\boldsymbol{x}_d^{(t)}) \star \mathbf{mat}(\boldsymbol{f}_d^{(t)}) = \mathbf{mat}(\boldsymbol{x}_d^{(t)}) * \overline{\mathbf{mat}(\boldsymbol{f}_d^{(t)})}$ ($\bar{\boldsymbol{X}}$ is generated by the following procedure: first, each row of $\boldsymbol{X}$ is arranged in reverse order and cyclically shifted by 1 bit to obtain an intermediate matrix. Then, each column of this intermediate matrix is arranged in reverse order and cyclically shifted by 1 bit to obtain $\bar{\boldsymbol{X}}$. For example, if $\boldsymbol{X} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}$, then $\bar{\boldsymbol{X}} = \begin{bmatrix} 1 & 4 & 3 & 2 \\ 13 & 16 & 15 & 14 \\ 9 & 12 & 11 & 10 \\ 5 & 8 & 7 & 6 \end{bmatrix}$; the symbol $*$ represents the two-dimensional convolution operator), Eq. (1) can be rewritten as Eq. (2).

$$\begin{aligned}\varepsilon(\boldsymbol{f}_d^{(t)}) = \min_{\boldsymbol{f}_d^{(t)}} \frac{1}{2}\left\| \boldsymbol{y} - \sum_{d=1}^{D} \mathbf{vec}\left(\mathbf{mat}(\boldsymbol{x}_d^{(t)}) * \overline{\mathbf{mat}(\boldsymbol{f}_d^{(t)})}\right)\right\|_F^2 \\ + \frac{\lambda}{2}\sum_{d=1}^{D}\left\|\mathbf{mat}(\boldsymbol{f}_d^{(t)})\right\|_F^2\end{aligned} \tag{2}$$

Let $\overline{\mathbf{mat}(\boldsymbol{f}_d^{(t)})}$ be marked as $\boldsymbol{G}_d^{(t)}$, Eq. (2) is rewritten as follows:

$$\varepsilon(\boldsymbol{g}_d^{(t)}) = \min_{\boldsymbol{g}_d^{(t)}} \frac{1}{2}\left\| \boldsymbol{y} - \sum_{d=1}^{D} \mathbf{vec}\left(\mathbf{mat}(\boldsymbol{x}_d^{(t)}) * \boldsymbol{G}_d^{(t)}\right)\right\|_F^2 + \frac{\lambda}{2}\sum_{d=1}^{D}\left\|\boldsymbol{g}_d^{(t)}\right\|_F^2 \tag{3}$$

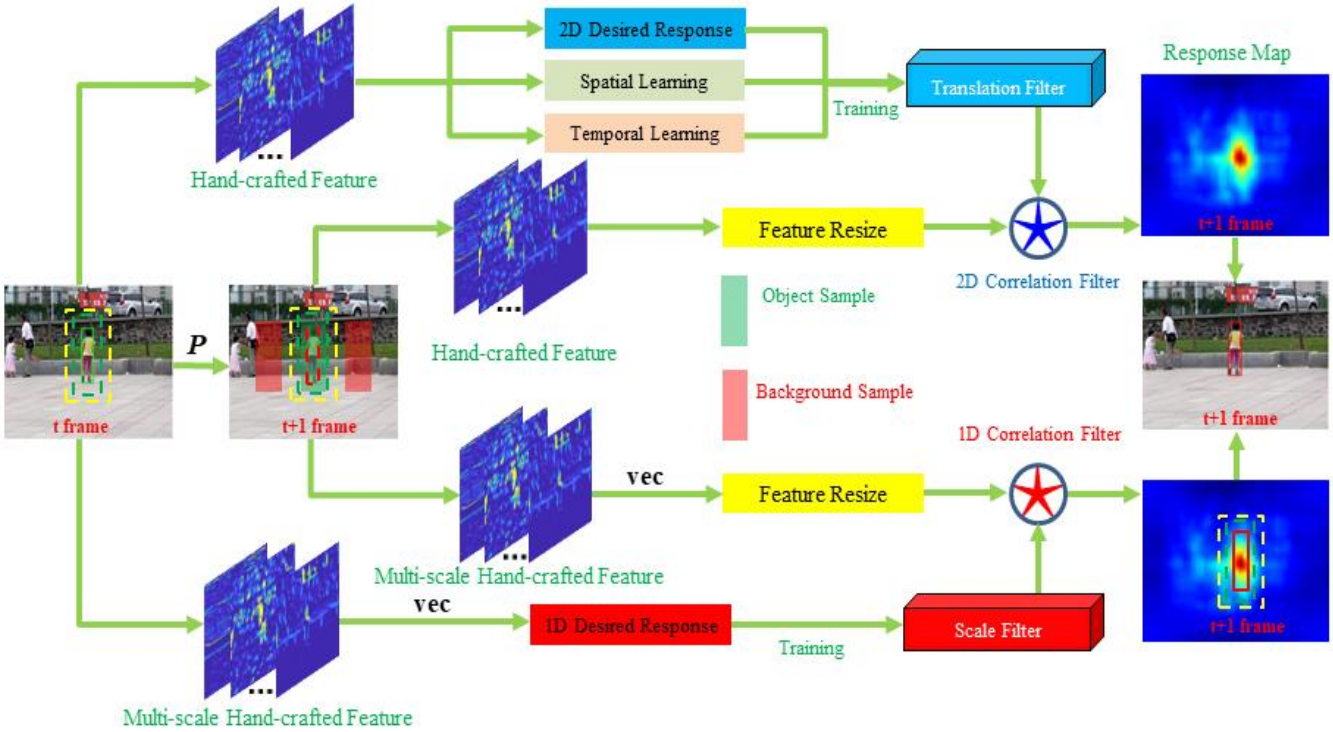

Fig. 1. General structure diagram of the proposed method.

where $\boldsymbol{G}_d^{(t)}=\mathbf{mat}(\boldsymbol{g}_d^{(t)})$.

According to the convolution theorem, we rewrite Eq. (3) into the frequency domain as follows:

$$\varepsilon(\hat{\boldsymbol{g}}_d^{(t)})=\min_{\hat{\boldsymbol{g}}_d}\frac{1}{2}\left\|\hat{\boldsymbol{y}}-\sum_{d=1}^{D}\hat{\boldsymbol{g}}_d^{(t)}\odot\hat{\boldsymbol{x}}_d^{(t)}\right\|_F^2+\frac{\lambda}{2}\sum_{d=1}^{D}\left\|\hat{\boldsymbol{g}}_d^{(t)}\right\|_F^2 \tag{4}$$

where $\odot$ is the element-wise multiplication operator, $\hat{\boldsymbol{x}}=\mathbf{vec}(\mathbf{fft2}(\mathbf{mat}(\boldsymbol{x})))$ is the vectorized column of two-dimensional Fourier transform for $\mathbf{mat}(\boldsymbol{x})$, $\mathbf{fft2}$ represents the two-dimensional Fourier transform.

By calculating the first-order derivative of Eq. (4) concerning $\hat{\boldsymbol{g}}_d^{(t)}$ and setting this derivative as zero, we have:

$$\hat{\boldsymbol{g}}_d^{(t)}=\frac{\hat{\boldsymbol{y}}\odot\left(\hat{\boldsymbol{x}}_d^{(t)}\right)^*}{\sum_{d=1}^{D}\left(\hat{\boldsymbol{x}}_d^{(t)}\right)^*\odot\hat{\boldsymbol{x}}_d^{(t)}+\lambda} \tag{5}$$

where the division in Eq. (5) represents the point-to-point division operator.

For a new sample $\boldsymbol{z}$, its multi-channel frequency feature $\hat{\boldsymbol{z}}_d^{(t)}$ is extracted first. Then the correlation filter response of this sample is calculated as follows:

$$\hat{\boldsymbol{r}}=\sum_{d=1}^{D}\hat{\boldsymbol{g}}_d^{(t)}\odot\hat{\boldsymbol{z}}_d^{(t)}=\frac{\sum_{d=1}^{D}\hat{\boldsymbol{y}}\odot\left(\hat{\boldsymbol{x}}_d^{(t)}\right)^*\odot\hat{\boldsymbol{z}}_d^{(t)}}{\sum_{d=1}^{D}\left(\hat{\boldsymbol{x}}_d^{(t)}\right)^*\odot\hat{\boldsymbol{x}}_d^{(t)}+\lambda} \tag{6}$$

The corresponding spatial response is:

$$\boldsymbol{r}=\mathbf{real}\left\{\mathcal{F}^{-1}(\hat{\boldsymbol{r}})\right\} \tag{7}$$

where $\mathcal{F}^{-1}$ represents the Fourier inverse transform operator; $\mathbf{real}$ represents the real part-taking operator.

III. Proposed Method

A. Proposed Model

The schematic diagram of the proposed method is shown in Fig. 1 As is shown in Fig. 1, on the one hand, we first extract the hand-crafted features (including HOG and color name feature) of the sample in the t-th frame and cropping matrix $\boldsymbol{P}$ in Fig. 1 is a binary matrix for cropping the sample elements. Then, these features are employed to design a two-dimensional position filter with reasonable spatial and temporal regularization factors. Using this filter makes it easy to detect the object's position in the $(t+1)$-th frame according to the highest value in the response. On the other hand, we extract and vectorize the multiscale hand-crafted features in the t-th frame to design a one-dimensional scale filter by DSST [21]. According to this filter, the optimal scale of the target in the $(t+1)$-th frame is finally determined.

The proposed spatio-temporal self-learning tracking model is defined as:

$$\begin{aligned}\varepsilon\left(\boldsymbol{g}_d^{(t)},\lambda^{(t)}\right)=\min_{\boldsymbol{g}_d^{(t)},\lambda^{(t)}}\{&\frac{1}{2}\left\|\boldsymbol{y}-\sum_{d=1}^{D}\mathbf{vec}\left(\mathbf{mat}(\boldsymbol{x}_d^{(t)})*\mathbf{mat}(\boldsymbol{g}_d^{(t)})\right)\right\|_F^2\\&+\frac{1}{2}\sum_{d=1}^{D}\left\|\tilde{\boldsymbol{w}}\odot\boldsymbol{g}_d^{(t)}\right\|_F^2+\frac{\lambda^{(t)}}{2}\sum_{d=1}^{D}\left\|\boldsymbol{g}_d^{(t)}-\boldsymbol{g}_d^{(t-1)}\right\|_F^2+\frac{1}{2}\left\|\lambda^{(t)}-\tilde{\lambda}\right\|_F^2\}\end{aligned} \tag{8}$$

where $\lambda^{(t)}$ represents the temporal regularized factor; $\tilde{\lambda}$ denotes the reference of $\lambda^{(t)}$; $\tilde{\boldsymbol{w}}$ represents the spatial regularized factor.

B. Spatial Self-learning

The local response variation reveals the credibility of each pixel in the current frame search region. Therefore, the pixels with lower credibility in the filter should be restricted during the learning process. We introduce a local response change vector $\boldsymbol{m}$ in the spatial regularized term and design a spatial regularized factor $\tilde{\boldsymbol{w}}$ that can sense local response variations, as defined below:

$$\tilde{\boldsymbol{w}}=\boldsymbol{P}^T\boldsymbol{P}\delta\ln(\bar{\boldsymbol{m}}+\mathbf{1})+\boldsymbol{w}+\chi\left(\mathbf{1}-\boldsymbol{P}^T\boldsymbol{P}\right) \tag{9}$$

where $\boldsymbol{P}\in\mathbb{R}^{K\times T}$ is a binary matrix for cropping the K elements in the spatial sample; The elements of $\boldsymbol{P}$ are

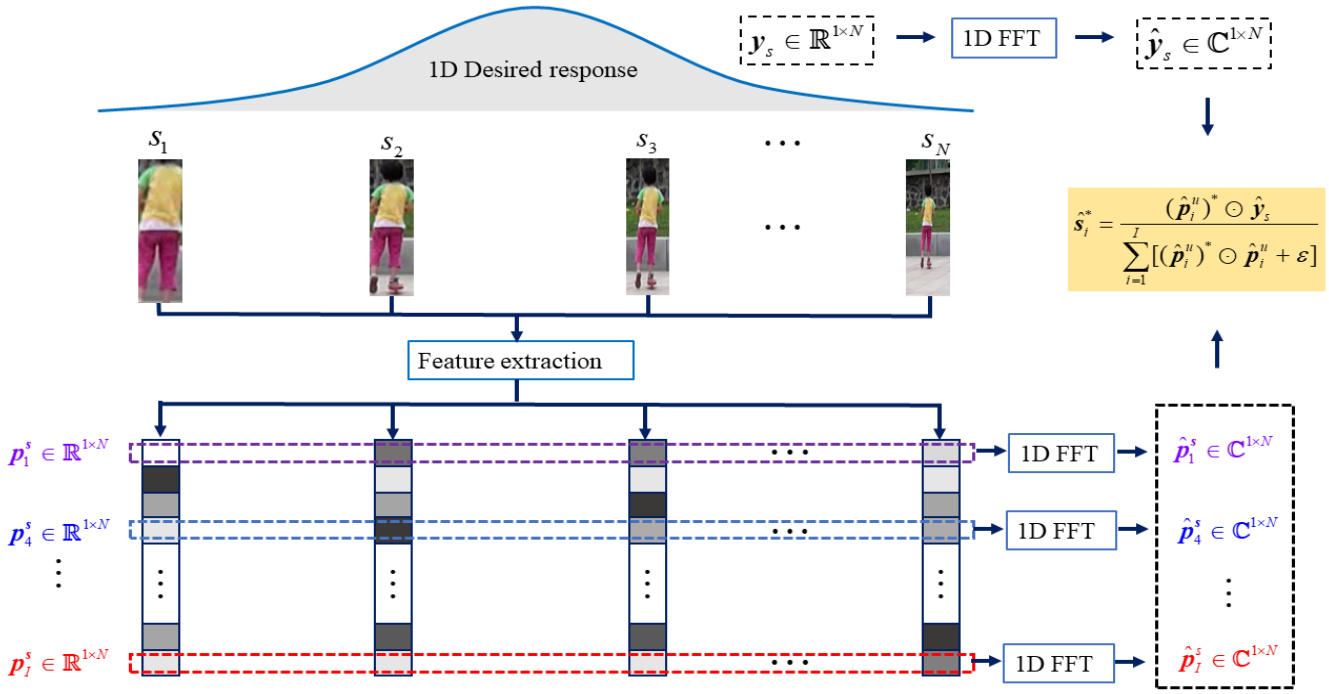

Fig. 2. Schematic diagram of scale adaptive filter.

defined as $\boldsymbol{P}(k,s)=\begin{cases}1, s\in S\\ 0, others\end{cases}$; S represents a set of clipped pixels determined by the size of the target template; $\boldsymbol{P}^T \in \mathbb{R}^{T\times K}$ is the transpose of $\boldsymbol{P}$; $\mathbf{1}$ denotes a matrix with all elements of 1. δ is a constant to adjust the weight of the local response variations, the parameter $\boldsymbol{w}$ is inherited from STRCF[19] to mitigate boundary effects. χ is a large number to punish the filter outside the size of the object. $\bar{\boldsymbol{m}}$=**vec**(**mat**($\boldsymbol{m}$)); The j-th element $\boldsymbol{m}_j$ of the local response change vector $\boldsymbol{m}$ is defined as:

$$\boldsymbol{m}_j = \left| \frac{\left\{\boldsymbol{r}\left[\psi_{\Delta}\right]_j\right\}^{(t)} - \boldsymbol{r}_j^{(t-1)}}{\boldsymbol{r}_j^{(t-1)}} \right| \tag{10}$$

where ψ_{Δ} represents the shift operator that coincides with the peaks of the two response maps $\boldsymbol{r}^{(t)}$ and $\boldsymbol{r}^{(t-1)}$ to remove motion influence; $\boldsymbol{r}_j^{(t-1)}$ represents the j-th element in the response map $\boldsymbol{r}^{(t-1)}$.

C. Temporal Self-learning

We automatically determine the value of the hyperparameters by jointly optimizing their values and filter.First, we construct the multimodal target pool. For the first frame, we have to fill the multimodal target pool with the face slices of the first frame due to the absence of historical data, i.e. $\boldsymbol{t}_n = \boldsymbol{p}^{(1)}(n=1,2,\cdots,N)$, where $\boldsymbol{p}^{(1)}$ represents the column vector drawn into the target face slices of the first frame; $\boldsymbol{t}_n$ denotes the n-th column vector of the multimodal target pool $\boldsymbol{T}$. Starting from the second frame, we assume that the face slice of the optimal sample obtained by the correlation response is $\boldsymbol{b}$, and extract the **HOG** features of $\boldsymbol{T}(:,n)$ and $\boldsymbol{b}$, as shown in the following equation:

$$\boldsymbol{e}_{t_n} = \frac{\mathbf{HOG}(\boldsymbol{t}_n)}{\max\left(\mathbf{HOG}(\boldsymbol{t}_n)\right)} \tag{11}$$

where **HOG** represents the directional gradient histogram extraction operator.

$$\boldsymbol{e}_b = \frac{\mathbf{HOG}(\boldsymbol{b})}{\max\left(\mathbf{HOG}(\boldsymbol{b})\right)} \tag{12}$$

We define a reference $\tilde{\lambda}$:

$$\tilde{\lambda} = \begin{cases} \dfrac{\zeta}{1+\ln\left(\upsilon\|\boldsymbol{m}\|_2+1\right)}, \text{if } \max\left(\cos\left(\boldsymbol{e}_{t_n},\boldsymbol{e}_b\right)\right) > \tau \\ \infty, \text{others} \end{cases} \tag{13}$$

where ζ and υ represent hyperparameters; τ is a threshold value in the range $[0,1]$. When the $\max\left(\cos\left(\boldsymbol{e}_{t_n},\boldsymbol{e}_b\right)\right)$ is higher than the set threshold, it indicates that the target is not occluded. Then $\boldsymbol{e}_b$ is updated to the target template pool and the face slice with the lowest similarity to $\boldsymbol{b}$ among the 2nd to N-th templates in the template pool is eliminated. At this point, the more dramatic the response map varies, the smaller the reference value will be. The temporal variation limit of the correlation filter can then be relaxed to speed up filter learning in the situation of large appearance variations. If the $\max\left(\cos\left(\boldsymbol{e}_{t_n},\boldsymbol{e}_b\right)\right)$ is lower than or equal to the set threshold, it indicates that the target is occluded and there are aberrances in the response map. Therefore, the correlation filter stops learning.

D. Solver of the Proposed Model

To optimize the objective function, we introduce an auxiliary variable $\hat{\boldsymbol{h}}_d^{(t)}=\hat{\boldsymbol{g}}_d^{(t)}=\boldsymbol{F}\boldsymbol{g}_d^{(t)}$=**vec**(**fft2**(**mat**($\boldsymbol{g}_d^{(t)}$))); $\boldsymbol{F}=\boldsymbol{F}_{\sqrt{T}}\otimes\boldsymbol{F}_{\sqrt{T}}\in\mathbb{C}^{T\times T}$ ($\boldsymbol{F}_{\sqrt{T}}\in\mathbb{C}^{\sqrt{T}\times\sqrt{T}}$ is the discrete Fourier transform matrix; $\otimes$ represents the Kronecker product operator). Then Eq. (8) is converted into the frequency domain, that is:

$$\varepsilon\left(\boldsymbol{g}_d^{(t)},\lambda^{(t)},\hat{\boldsymbol{H}}^{(t)}\right)=\min_{\boldsymbol{g}_d^{(t)},\lambda^{(t)},\hat{\boldsymbol{H}}^{(t)}}\{\frac{1}{2}\left\|\hat{\boldsymbol{y}}-\sum_{d=1}^{D}\hat{\boldsymbol{x}}_d^{(t)}\odot\hat{\boldsymbol{h}}_d^{(t)}\right\|_F^2+\frac{1}{2}\sum_{d=1}^{D}\left\|\tilde{\boldsymbol{w}}\odot\boldsymbol{g}_d^{(t)}\right\|_F^2$$
$$+\frac{\lambda^{(t)}}{2}\sum_{d=1}^{D}\left\|\hat{\boldsymbol{h}}_d^{(t)}-\hat{\boldsymbol{h}}_d^{(t-1)}\right\|_F^2+\frac{1}{2}\left\|\lambda^{(t)}-\tilde{\lambda}\right\|_F^2\}$$
$$\text{s.t. }\hat{\boldsymbol{h}}_d^{(t)}=\boldsymbol{F}\boldsymbol{g}_d^{(t)} \tag{14}$$

where $(\hat{\boldsymbol{H}}^{(t)}=\left[\hat{\boldsymbol{h}}_1^{(t)},\hat{\boldsymbol{h}}_2^{(t)},\hat{\boldsymbol{h}}_3^{(t)},...,\hat{\boldsymbol{h}}_D^{(t)}\right])$.

The optimal solution of Eq. (14) can be obtained through the alternating direction method of multipliers (ADMM). The augmented Lagrangian form of Eq. (14) can be formulated as:

$$\mathcal{L}\left(\boldsymbol{g}_d^{(t)},\lambda^{(t)},\hat{\boldsymbol{H}}^{(t)},\hat{\boldsymbol{L}}^{(t)}\right)=\varepsilon\left(\boldsymbol{g}_d^{(t)},\lambda^{(t)},\hat{\boldsymbol{H}}^{(t)}\right)$$
$$+\frac{\gamma}{2}\sum_{d=1}^{D}\left\|\hat{\boldsymbol{h}}_d^{(t)}-\hat{\boldsymbol{h}}_d^{(t-1)}\right\|_F^2+\sum_{d=1}^{D}\left(\hat{\boldsymbol{l}}_d^{(t)}\right)^H\left(\hat{\boldsymbol{h}}_d^{(t)}-\boldsymbol{F}\boldsymbol{g}_d^{(t)}\right) \tag{15}$$

where $\hat{\boldsymbol{x}}=\boldsymbol{F}\boldsymbol{x}=\mathbf{vec}(\mathbf{fft2}(\mathbf{mat}(\boldsymbol{x})))$ is the signal spectrum; $\hat{\boldsymbol{L}}^{(t)}=\left[\hat{\boldsymbol{l}}_1^{(t)},\hat{\boldsymbol{l}}_2^{(t)},...,\hat{\boldsymbol{l}}_D^{(t)}\right]$ is the frequency domain variable of the Lagrange multiplier; γ represents the quadratic penalty parameter. Let $\boldsymbol{v}_d^{(t)}=\frac{\boldsymbol{l}_d^{(t)}}{\gamma},\boldsymbol{V}_d^{(t)}=\left[\boldsymbol{v}_1^{(t)},\boldsymbol{v}_2^{(t)},...,\boldsymbol{v}_D^{(t)}\right]$, we organize Eq. (15) using the matching method as:

$$\mathcal{L}(\boldsymbol{g}_d^{(t)},\lambda^{(t)},\hat{\boldsymbol{H}}^{(t)},\hat{\boldsymbol{V}}^{(t)})=\varepsilon(\boldsymbol{g}_d^{(t)},\lambda^{(t)},\hat{\boldsymbol{H}}^{(t)})+\frac{\gamma}{2}\sum_{d=1}^{D}\left\|\hat{\boldsymbol{h}}_d^{(t)}-\boldsymbol{F}\boldsymbol{g}_d^{(t)}+\hat{\boldsymbol{v}}_d^{(t)}\right\|_F^2 \tag{16}$$

Then we iteratively solve for each variable by ADMM. First fixing $\boldsymbol{g}_d^{(t)}$, $\lambda^{(t)}$, $\boldsymbol{V}_d^{(t)}$, then $\hat{\boldsymbol{H}}^{(t)}$ can be calculated by the following equation:

$$\left\{\hat{\boldsymbol{H}}^{(t)}\right\}^{(i+1)}=\arg\min_{\hat{\boldsymbol{H}}^{(t)}}\left\{\frac{1}{2}\left\|\hat{\boldsymbol{y}}-\sum_{d=1}^{D}\hat{\boldsymbol{x}}_d^{(t)}\odot\hat{\boldsymbol{h}}_d^{(t)}\right\|_2^2+\frac{\left\{\lambda^{(t)}\right\}^{(i)}}{2}\sum_{d=1}^{D}\left\|\hat{\boldsymbol{h}}_d^{(t)}-\hat{\boldsymbol{h}}_d^{(t-1)}\right\|_F^2\right.$$
$$\left.+\frac{\gamma}{2}\sum_{d=1}^{D}\left\|\hat{\boldsymbol{h}}_d^{(t)}-\boldsymbol{F}\left\{\boldsymbol{g}_d^{(t)}\right\}^{(i)}+\left\{\hat{\boldsymbol{v}}_d^{(t)}\right\}^{(i)}\right\|_F^2\right\} \tag{17}$$

where i (the initial value of i is 0) represents the number of iterations; $\left\{\boldsymbol{g}_d^{(t)}\right\}^{(0)}=\boldsymbol{g}_d^{(t-1)}$, $\left\{\lambda^{(t)}\right\}^{(0)}=\lambda^{(t-1)}$, $\left\{\hat{\boldsymbol{v}}_d^{(t)}\right\}^{(0)}=\mathbf{0}$.

Due to the complexity of Eq. (17), we simplify the above formulation by the following equation:

$$\Gamma_j\left(\left\{\hat{\boldsymbol{H}}^{(t)}\right\}^{(i+1)}\right)=\arg\min_{\Gamma_j\left(\hat{\boldsymbol{H}}^{(t)}\right)}\left\{\left\|\hat{\boldsymbol{y}}_j-\Gamma_j\left(\hat{\boldsymbol{X}}^{(t)}\right)^T\Gamma_j\left(\hat{\boldsymbol{H}}^{(t)}\right)\right\|_F^2\right.$$
$$+\gamma\left\|\Gamma_j\left(\hat{\boldsymbol{H}}^{(t)}\right)+\Gamma_j\left(\left\{\hat{\boldsymbol{V}}^{(t)}\right\}^{(i)}\right)-\Gamma_j\left(\boldsymbol{F}\left\{\boldsymbol{G}^{(t)}\right\}^{(i)}\right)\right\|_F^2 \tag{18}$$
$$\left.+\left\{\lambda^{(t)}\right\}^{(i)}\left\|\Gamma_j\left(\hat{\boldsymbol{H}}^{(t)}\right)-\Gamma_j\left(\hat{\boldsymbol{H}}^{(t-1)}\right)\right\|_F^2\right\}$$

where $\Gamma_j\left(\hat{\boldsymbol{X}}^{(t)}\right)$ means that the elements in the j-th row of $\hat{\boldsymbol{X}}^{(t)}=\left[\hat{\boldsymbol{x}}_1^{(t)},\hat{\boldsymbol{x}}_2^{(t)},...,\hat{\boldsymbol{x}}_D^{(t)}\right]\in\mathbb{C}^{T\times D}$.

By using the derivation of the Sherman-Morrison formula, we can obtain its solution.

$$\Gamma_j\left(\left\{\hat{\boldsymbol{H}}^{(t)}\right\}^{(i+1)}\right)$$
$$=\frac{1}{\gamma+\left\{\lambda^{(t)}\right\}^{(i)}}\left(\mathbf{I}-\frac{\Gamma_j\left(\hat{\boldsymbol{X}}^{(t)}\right)\Gamma_j\left(\hat{\boldsymbol{X}}^{(t)}\right)^T}{\left\{\lambda^{(t)}\right\}^{(i)}+\gamma+\Gamma_j\left(\hat{\boldsymbol{X}}^{(t)}\right)^T\Gamma_j\left(\hat{\boldsymbol{X}}^{(t)}\right)}\right)\boldsymbol{\rho} \tag{19}$$

where $\mathbf{I}\in\mathbb{R}^{D\times D}$ is the unit matrix,

$$\boldsymbol{\rho}=\Gamma_j\left(\hat{\boldsymbol{X}}^{(t)}\right)\hat{\boldsymbol{y}}_j+\left\{\lambda^{(t)}\right\}^{(i)}\Gamma_j\left(\hat{\boldsymbol{H}}^{(t-1)}\right)-\gamma\Gamma_j\left(\left\{\hat{\boldsymbol{V}}^{(t)}\right\}^{(i)}\right)+\gamma\Gamma_j\left(\boldsymbol{F}\left\{\boldsymbol{G}^{(t)}\right\}^{(i)}\right)$$
, $\left\{\hat{\boldsymbol{H}}^{(t)}\right\}^{(i+1)}=\left[\left\{\hat{\boldsymbol{h}}_1^{(t)}\right\}^{(i+1)},\left\{\hat{\boldsymbol{h}}_2^{(t)}\right\}^{(i+1)},\left\{\hat{\boldsymbol{h}}_3^{(t)}\right\}^{(i+1)},...,\left\{\hat{\boldsymbol{h}}_D^{(t)}\right\}^{(i+1)}\right]$.

Fixing $\left\{\hat{\boldsymbol{H}}^{(t)}\right\}^{(i+1)}$ and $\left\{\hat{\boldsymbol{V}}^{(t)}\right\}^{(i)}$, we can optimize $\boldsymbol{g}_d^{(t)}$ by:

$$\left(\boldsymbol{g}_d^{(t)}\right)^{(i+1)}=\arg\min_{\boldsymbol{g}_d^{(t)}}\{J_{\boldsymbol{g}_d^{(t)}}\}$$
$$=\arg\min_{\boldsymbol{g}_d^{(t)}}\left\{\frac{1}{2}\left\|\tilde{\boldsymbol{w}}\odot\boldsymbol{g}_d^{(t)}\right\|_F^2+\frac{\gamma}{2}\left\|\left\{\hat{\boldsymbol{h}}_d^{(t)}\right\}^{(i+1)}-\boldsymbol{F}\boldsymbol{g}_d^{(t)}+\left\{\hat{\boldsymbol{v}}_d^{(t)}\right\}^{(i)}\right\|_F^2\right\} \tag{20}$$

Let $\frac{\partial J_{\boldsymbol{g}_d^{(t)}}}{\partial\boldsymbol{g}_d^{(t)}}=\mathbf{0}$, then we have:

$$\tilde{\boldsymbol{w}}\odot\tilde{\boldsymbol{w}}\odot\boldsymbol{g}_d^{(t)}+\gamma\boldsymbol{F}^H\left(\boldsymbol{F}\boldsymbol{g}_d^{(t)}-\left\{\hat{\boldsymbol{v}}_d^{(t)}\right\}^{(i)}-\left\{\hat{\boldsymbol{h}}_d^{(t)}\right\}^{(i+1)}\right)=\mathbf{0} \tag{21}$$

Then, the optimal solution of $\boldsymbol{g}_d^{(t)}$ is:

$$\left(\boldsymbol{g}_d^{(t)}\right)^{(i+1)}=\frac{\gamma\boldsymbol{F}^H\left(\left\{\boldsymbol{v}_d^{(t)}\right\}^{(i)}+\left\{\hat{\boldsymbol{h}}_d^{(t)}\right\}^{(i+1)}\right)}{\tilde{\boldsymbol{w}}\odot\tilde{\boldsymbol{w}}+\gamma T} \tag{22}$$

The division in Eq. (22) denotes the element-wise division operation.

Given other variables in Eq. (16), the optimal solution of $\lambda^{(t)}$ can be calculated as follows:

$$\left\{\lambda^{(t)}\right\}^{(i+1)}=\arg\min_{\lambda^{(t)}}\left\{\frac{\lambda^{(t)}}{2}\sum_{d=1}^{D}\left\|\hat{\boldsymbol{h}}_d^{(t)}-\hat{\boldsymbol{h}}_d^{(t-1)}\right\|_F^2+\frac{1}{2}\left\|\lambda^{(t)}-\tilde{\lambda}\right\|_F^2\right\}$$
$$=\tilde{\lambda}-\frac{\left\|\left\{\hat{\boldsymbol{H}}^{(t)}\right\}^{(i+1)}-\hat{\boldsymbol{H}}^{(t-1)}\right\|_F^2}{2} \tag{23}$$

The following equation can update the Lagrange multiplier:

$$\hat{\boldsymbol{V}}^{(i+1)}=\hat{\boldsymbol{V}}^{(i)}+\gamma\left(\left\{\hat{\boldsymbol{H}}^{(t)}\right\}^{(i+1)}-\left\{\hat{\boldsymbol{G}}^{(t)}\right\}^{(i+1)}\right) \tag{24}$$

where $\left\{\hat{\boldsymbol{G}}^{(t)}\right\}^{(i+1)}=\left[\left(\hat{\boldsymbol{g}}_1^{(t)}\right)^{(i+1)},\left(\hat{\boldsymbol{g}}_2^{(t)}\right)^{(i+1)},\left(\hat{\boldsymbol{g}}_3^{(t)}\right)^{(i+1)},...,\left(\hat{\boldsymbol{g}}_D^{(t)}\right)^{(i+1)}\right]$; γ (the initial value of γ is 1) is iterated as $\gamma^{(i+1)}=\min\left(\gamma_{\max},\beta\gamma^{(i)}\right)$ (β = 10, $\gamma_{\max}$ = 10000).

After calculating the optimal filter and the temporal regularized parameter, they are adopted to calculate the filter response for the next frame.

E. Object Localization and Scale Estimation

The tracking object is located by searching the maximum value of the response. The response is calculated as follows:

$$\boldsymbol{r}^{(t)}=\mathbf{real}\left\{\mathcal{F}^{-1}\sum_{d=1}^{D}\left(\hat{\boldsymbol{z}}_d^{(t)}\odot\hat{\boldsymbol{h}}_d^{(t-1)}\right)\right\} \tag{25}$$

where $\boldsymbol{r}^{(t)}$ is the response in the frame t, $\hat{\boldsymbol{z}}_d^{(t)}$ represents the spectrum of the extracted features concerning the sample in the frame t. This paper employs the same scale estimation method in DSST [21].

The DSST scale estimation is as follows. We suppose that the target size of the previous frame is $H\times W$. The target samples are first sampled with different scales; the scale range is $\boldsymbol{S}=\left\{a^n\mid n=\left\lfloor-\frac{N-1}{2},\ldots,\frac{N-1}{2}\right\rfloor\right\}$. The target scale size is $\left\{sH\times sW\mid s\in\boldsymbol{S}\right\}$ at each scale, and the image features are obtained at different scales. Then the image features at the n-th scale are pulled into column vectors and weighted with the scale primary function

$\boldsymbol{w}_s(n)=\frac{1}{\sqrt{2\pi}\sigma_1}\exp[-\frac{(n-\frac{N}{2})^2}{2\sigma_1^2}]$ (the scale primary function is a gaussian function and σ_1 denotes the standard deviation of the scale primary function) to form a multiscale feature matrix $\boldsymbol{P}_x \in \mathbb{R}^{D\times N}$. The matrix is defined as $\boldsymbol{P}_x=\begin{pmatrix}\boldsymbol{p}_1^x\\ \boldsymbol{p}_2^x\\ \vdots\\ \boldsymbol{p}_I^x\end{pmatrix}$, where $\boldsymbol{p}_i^x \in \mathbb{R}^{1\times N}(i=1,2,\cdots,I)$, as shown in Fig. 2.

The objective function of the scale-adaptive filter is designed as follows:

$$J=\min_{\bar{\boldsymbol{s}}_i}\left\|\sum_{i=1}^{I}\bar{\boldsymbol{s}}_i * \boldsymbol{p}_i^x-\boldsymbol{y}_s\right\|_F^2+\varepsilon\sum_{i=1}^{I}\left\|\bar{\boldsymbol{s}}_i\right\|_F^2 \tag{26}$$

where $\boldsymbol{y}_s \in \mathbb{R}^{1\times N}$ denotes the scale training label with elements defined as $\boldsymbol{y}_s(n)=\frac{1}{\sqrt{2\pi}\sigma_2}\exp[-\frac{(n-\frac{N}{2})^2}{2\sigma_2^2}]$ and σ_2 denotes the standard deviation of the desired response gaussian function. $\bar{\boldsymbol{s}}_i$ is arranged in reverse order of $\boldsymbol{s}_i$ and then cyclically shifted by 1 bit. ε denotes the parameter of ridge regression balance.

We rewrite Eq. (26) into the frequency domain as follows:

$$J=\min_{\hat{\boldsymbol{s}}_i^*}\left\|\sum_{i=1}^{I}\hat{\boldsymbol{s}}_i^*\odot\hat{\boldsymbol{p}}_i^x-\hat{\boldsymbol{y}}_s\right\|_F^2+\varepsilon\sum_{i=1}^{I}\left\|\hat{\boldsymbol{s}}_i^*\right\|_F^2 \tag{27}$$

By calculating the first-order derivative of the filter $\hat{\boldsymbol{s}}_i^*$ and setting this derivative as zero, the scaled filter is as follows:

$$\hat{\boldsymbol{s}}_i^*=\frac{(\hat{\boldsymbol{p}}_i^x)^*\odot\hat{\boldsymbol{y}}_s}{\sum_{i=1}^{I}[(\hat{\boldsymbol{p}}_i^x)^*\odot\hat{\boldsymbol{p}}_i^x+\varepsilon]}\in\mathbb{C}^{1\times N} \tag{28}$$

For the new sample $\boldsymbol{z}$, we perform pyramidal sampling to obtain N scale face slices and the multiscale feature matrix $\boldsymbol{P}_z \in \mathbb{R}^{D\times N}$. Similarly, this matrix is binned by rows to obtain $\boldsymbol{P}_z=\begin{pmatrix}\boldsymbol{p}_1^z\\ \boldsymbol{p}_2^z\\ \vdots\\ \boldsymbol{p}_I^z\end{pmatrix}$. The scale response is:

$$\hat{\boldsymbol{r}}_s=\sum_{i=1}^{I}\hat{\boldsymbol{s}}_i^*\odot\hat{\boldsymbol{p}}_i^z\in\mathbb{C}^{1\times N} \tag{29}$$

The inverse Fourier transform of Eq. (29) is performed to obtain the scale-dependent response as follows:

$$\boldsymbol{r}_s=\boldsymbol{real}\{\mathcal{F}^{-1}(\hat{\boldsymbol{r}}_s)\}\in\mathbb{R}^{1\times N} \tag{30}$$

Then we select the scale corresponding to the maximum scale response as the final target scale.

IV. Experiments

The computer configuration for experimentation was: Intel (R) Core (TM) i5-3210M CPU @ 2.50GHz 2.50 GHz, 16GBRAM, Windows 10 64-bit operating system. The OTB100 dataset [22] is adopted to evaluate the proposed method.

For the hyperparameters of STSL, we set $\delta=0.2$, $\nu=2\times10^{-5}$, $\zeta=13$, and the ADMM iteration is set as 4.

A. Ablation Experiments

In this section, we carry out some ablation experiments on the STSL algorithm to show the importance of the key components of the proposed method.

Ablation Experiment on Spatial Regularization

In this section, we compare the STSL and STSL without spatial regularization (referred to as STSL without SR, whose objective function is $\varepsilon\left(\lambda^{(t)},\hat{\boldsymbol{H}}^{(t)}\right)=\min_{\lambda^{(t)},\hat{\boldsymbol{H}}^{(t)}}\{\frac{1}{2}\left\|\hat{\boldsymbol{y}}-\sum_{d=1}^{D}\hat{\boldsymbol{x}}_d^{(t)}\odot\hat{\boldsymbol{h}}_d^{(t)}\right\|_F^2+\frac{1}{2}\sum_{d=1}^{D}\left\|\boldsymbol{h}_d^{(t)}\right\|_F^2+\frac{\lambda^{(t)}}{2}\sum_{d=1}^{D}\left\|\hat{\boldsymbol{h}}_d^{(t)}-\hat{\boldsymbol{h}}_d^{(t-1)}\right\|_F^2+\frac{1}{2}\left\|\lambda^{(t)}-\tilde{\lambda}\right\|_F^2\}$).

As shown in Fig. 3 (a), the red tracking frame representing the STSL with spatial regularization and the green tracking frame representing the STSL without spatial regularization can achieve better tracking of the target when the target is running normally at frame 10. As shown in Fig. 3 (b) (c), the target enters the shadow after frame 27, then after resembling the background, the green tracking frame cannot distinguish the background from the target resulting in tracking failure. But the red tracking frame can still track the target accurately. As shown in Fig. 3 (d), at frame 102, the green tracking frame is influenced by the previous occlusion and consistently fails to re-track the correct target. In contrast, the red tracking frame is undisturbed and continues to track the target accurately and steadily.

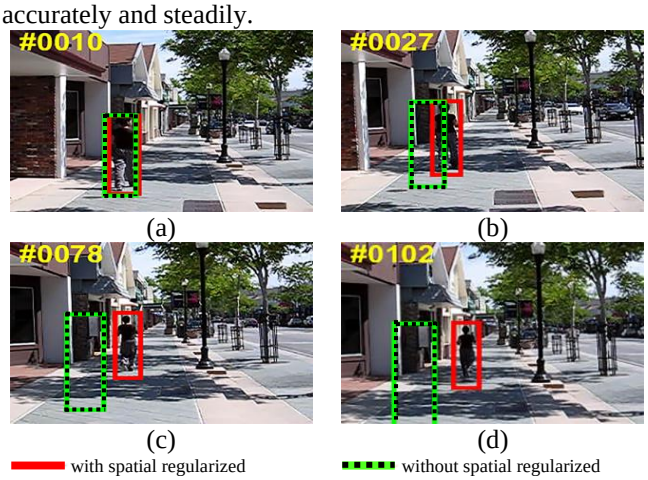

Fig. 3. Ablation experiment on spatial regularization. (a) Frame 10; (b) Frame 27; (c) Frame 78; (d) Frame 102.

Experiments show that the STSL with spatial regularized can stop the correlation filter learning when the samples are contaminated. Thus the DCF can focus on the target object and avoid introducing a large number of contaminated samples that lead to tracking failure.

Ablation Experiment on Temporal Regularization

In this section, we compare the STSL and STSL without temporal regularization (referred to as STSL without TR, whose objective function is $\varepsilon\left(\boldsymbol{g}_d^{(t)},\hat{\boldsymbol{H}}^{(t)}\right)=\min_{\boldsymbol{g}_d^{(t)},\hat{\boldsymbol{H}}^{(t)}}\{\frac{1}{2}\left\|\hat{\boldsymbol{y}}-\sum_{d=1}^{D}\hat{\boldsymbol{x}}_d^{(t)}\odot\hat{\boldsymbol{h}}_d^{(t)}\right\|_F^2+\frac{1}{2}\sum_{d=1}^{D}\left\|\tilde{\boldsymbol{w}}\odot\boldsymbol{h}_d^{(t)}\right\|_F^2$).

As shown in Fig. 4 (a), the target does not move substantially and quickly in frame 15. Thus, the red tracking frame representing the STSL with temporal regularization and the green tracking frame representing the STSL without temporal regularization can achieve good target tracking. As shown in Fig. 4 (b) (c) (d), after the 26th frame, the target makes a rapid jumping motion. Then the green tracking frame produces a large tracking drift, while the red tracking frame can always track the target accurately.

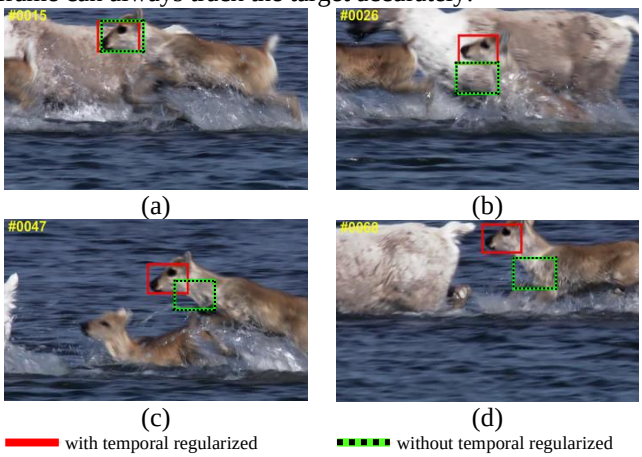

Fig. 4. Ablation experiment on temporal regularization. (a) Frame 15; (b) Frame 26; (c) Frame 47; (d) Frame 68.

Experiments show that in the situation of rapid motion, the STSL with temporal regularization can adjust the filter's learning limitation, update the filter quickly, and avoid tracking failure due to the rapid movement of the target.

Ablation Experiment on Anti-occlusion Module

We compare the STSL and the STSL without the anti-occlusion module to test the influence of adding an anti-occlusion module.

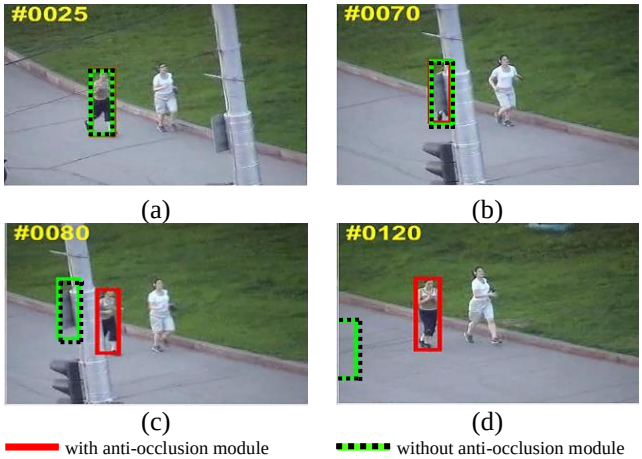

Fig. 5. Ablation experiment on anti-occlusion module. (a) Frame 25; (b) Frame 70; (c) Frame 80; (d) Frame 120.

As shown in Fig. 5 (a), when the target is not occluded at frame 25, both the red tracking frame representing the STSL with the anti-occlusion module and the green tracking frame representing the STSL without the anti-occlusion module can achieve better tracking of the target. As shown in Fig. 5 (b) (c), the target is strongly occluded at frames 70 and 80. Thus the green tracking frame could not capture the target, while the red tracking frame could still track the target accurately. As shown in Fig. 5 (d), at frame 120, the green tracking frame is influenced by the previous occlusion, never retraces the correct target, and fails to track. In contrast, the occlusion does not interfere with the red tracking frame and continues to track the target accurately and stably.

Experiments show that in the situation of the strong occlusion, the STSL with the anti-occlusion module can effectively decrease the influence of the occlusion on the target tracking and avoid tracking failure due to the strong occlusion of the target.

B. Qualitative Analysis

In this section, we compare the proposed STSL method with some state-of-the-art tracking methods (including AutoTrack [23], BACF [17], ARCF_H [24], DeepSRDCF [25], SRDCF [18], Struck [26], STAPLE_CA [27], DSST [21], KCF [10]) in OTB100 dataset. Fig. 6 intercepts some frames of different visuals to show the tracking performance of our algorithm and the nine algorithms compared under different tracking challenges.

The following is a qualitative analysis of several typical tracking challenges:

(1) Rapid motion. In Fig. 6 (a), a group of basketball players plays a game. The target is moving fast and has some motion blur accompanied by rotation. Except for DSST and STSL, which always track the target stably, the tracking of other algorithms has some deviations. The ARCF_H, BACF, SRDCF, and Struck finally fail to track. In Fig. 6 (e), the target is performing skating. The DSST algorithm has a large deviation, while other algorithms track the target more stably. Although the target changes rapidly, the STSL still tracks the target due to the simultaneous use of local response variations.

(2) Occlusion. In Fig. 6 (b), the girl is fully or partially occluded by passersby several times. The girl is completely obscured between video frames 110 and 120. STSL can always track the target accurately, while other algorithms fail to track it. After video frame 390, the DeepSRDCF can reidentify the tracking target. In Fig. 6 (d), the Jogging2 target is strongly occluded by the pole. Then BACF, ARCF_H, Struck, STAPLE_CA, and KCF cannot track the target effectively. After 114 frames, the ARCF_H recognizes the tracking target again. The STSL is always able to track the target stably. The adopted multimodal template pooling strategy can effectively handle partial or complete occlusion.

(3) Background clutter. In Fig. 6 (c), the target's clothes and the background color are very similar, so the target is indistinguishable from the background. Struck produces large tracking drift, AutoTrack, SRDCF, STAPLE_CA, and KCF tracking failure. The proposed STSL can still track the target accurately and stably. The local response spatial vector lets DCF focus on the learning object, decreasing background clutter's influence.

(4) Rotation. In Fig. 6 (a), the DSST, ARCF_H, and KCF experienced varying levels of tracking drift after 35 frames, while the other algorithms were largely unaffected. In Fig. 6 (b), the BACF, SRDCF, STAPLE-CA, ARCF_H, AutoTrack, KCF, and SRDCF are lost before the target starts rotating at frame 430. The DSST has some tracking drift when the target undergoes 90 degrees rotation. The SRDCF reidentifies the target. The STSL always tracks the target accurately. In Fig. 6 (e), after the target rotated 360 degrees, DSST lost the target, STAPLE_CA and SRDCF drifted slightly, and the rest of the

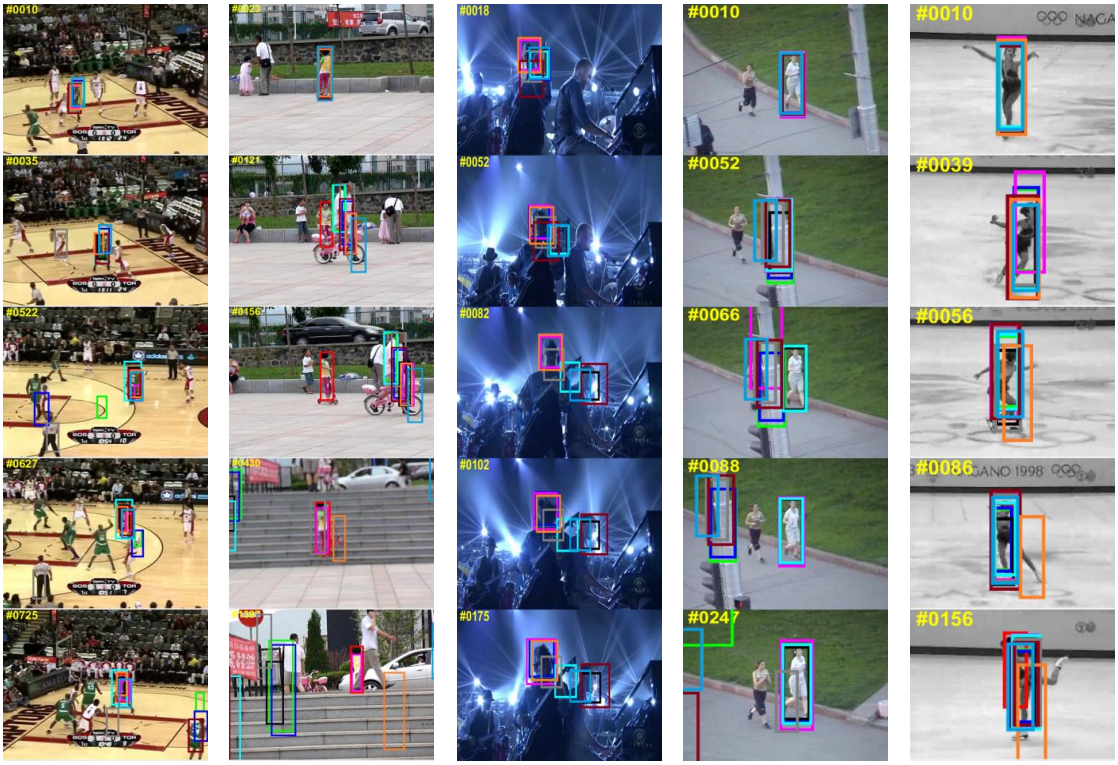

Fig. 6. Visualize the position error and overlap rate on different video sequences.

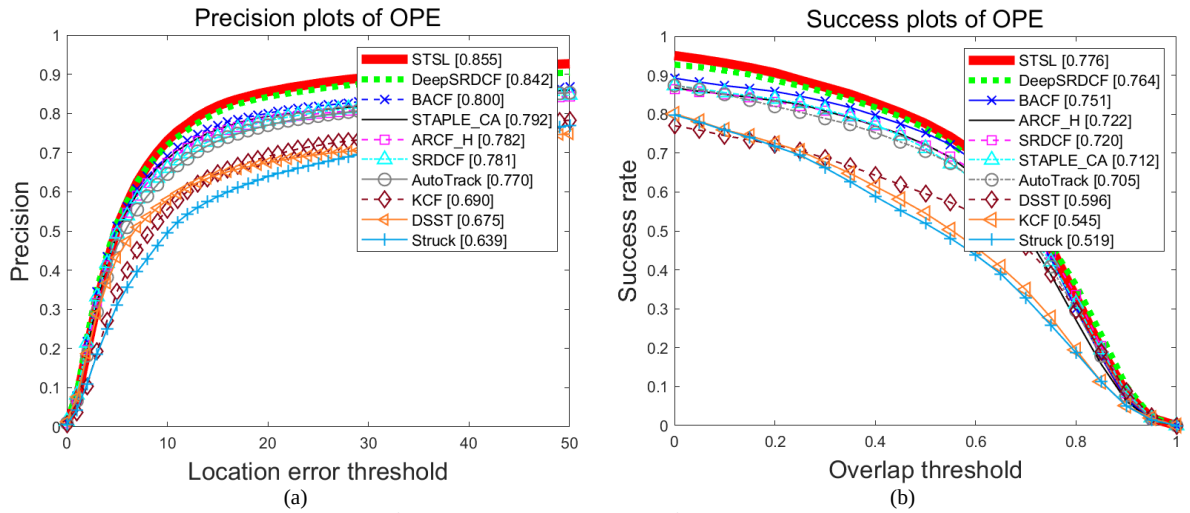

Fig. 7. Comprehensive comparison of the success rate and accuracy of the algorithm. (a) Precision plots; (b) Success plots.

TABLE I
SCORE RATES OF 10 ALGORITHMS FOR EACH CHALLENGE ATTRIBUTE

Challenge Attributes	SRDCF	KCF	ARCF_H	DSST	STAPLE_CA	Struck	DeepSRDCF	AutoTrack	BACF	STSL
Illumination Variation	0.764	0.704	0.746	0.703	0.782	0.554	0.763	0.730	0.760	0.805
Out-of-plane Rotation	0.729	0.665	0.724	0.637	0.737	0.590	0.821	0.725	0.754	0.834
Scale Variation	0.741	0.638	0.736	0.633	0.746	0.607	0.817	0.717	0.755	0.82
Occlusion	0.712	0.621	0.658	0.579	0.701	0.541	0.804	0.679	0.697	0.792
Deformation	0.709	0.602	0.720	0.522	0.735	0.532	0.759	0.712	0.727	0.810
Motion Blur	0.737	0.582	0.688	0.551	0.690	0.590	0.792	0.680	0.686	0.784
Fast Motion	0.747	0.608	0.736	0.540	0.733	0.622	0.791	0.737	0.765	0.787
In-plane Rotation	0.728	0.691	0.734	0.681	0.763	0.635	0.801	0.729	0.760	0.798
Out Of View	0.596	0.515	0.674	0.481	0.661	0.503	0.779	0.726	0.747	0.791
Background Clutter	0.774	0.719	0.802	0.703	0.797	0.566	0.841	0.757	0.801	0.848
Low Resolution	0.655	0.560	0.694	0.567	0.628	0.671	0.708	0.763	0.741	0.802

TABLE II
SCORE RATES OF 10 ALGORITHMS FOR EACH CHALLENGE ATTRIBUTE

Challenge Attributes	SRDCF	KCF	ARCF_H	DSST	STAPLE_CA	Struck	DeepSRDCF	AutoTrack	BACF	STSL
Illumination Variation	0.717	0.53	0.722	0.636	0.687	0.486	0.721	0.707	0.734	0.768
Out-of-plane Rotation	0.651	0.515	0.637	0.544	0.647	0.464	0.724	0.642	0.681	0.732
Scale Variation	0.663	0.417	0.642	0.525	0.628	0.424	0.732	0.633	0.685	0.713
Occlusion	0.661	0.500	0.606	0.521	0.644	0.436	0.722	0.632	0.658	0.734
Deformation	0.639	0.486	0.644	0.468	0.651	0.418	0.672	0.654	0.652	0.729
Motion Blur	0.701	0.529	0.673	0.534	0.655	0.544	0.751	0.654	0.677	0.743
Fast Motion	0.696	0.511	0.708	0.505	0.684	0.534	0.733	0.683	0.735	0.729
In-plane Rotation	0.646	0.541	0.638	0.579	0.659	0.503	0.700	0.627	0.678	0.687
Out Of View	0.561	0.466	0.662	0.442	0.589	0.421	0.653	0.669	0.692	0.729
Background Clutter	0.702	0.613	0.761	0.613	0.729	0.518	0.749	0.717	0.768	0.783
Low Resolution	0.626	0.295	0.572	0.442	0.494	0.318	0.587	0.669	0.663	0.686

TABLE III
AVERAGE TRACKING OVERLAP RATES FOR EACH TRACKING ALGORITHM IN SOME VIDEOS

Video	SRDCF	KCF	ARCF_H	DSST	STAPLE_CA	Struck	DeepSRDCF	AutoTrack	BACF	STSL
BlurCar2	0.88	0.75	0.76	**0.90**	0.89	0.74	0.89	**0.90**	0.75	**0.90**
David3	0.76	**0.77**	0.72	0.48	**0.77**	0.29	**0.77**	**0.77**	0.76	**0.77**
Deer	**0.81**	0.62	0.80	0.64	0.80	0.74	0.77	0.79	0.73	0.80
Girl2	0.07	0.06	0.07	0.09	0.07	0.22	0.67	0.06	0.06	**0.72**
Jogging-1	**0.80**	0.19	0.18	0.19	0.18	0.17	0.79	0.19	0.77	**0.80**
Jogging-2	0.71	0.12	0.61	0.13	0.14	0.20	0.60	**0.76**	0.13	**0.76**
Woman	0.67	0.71	**0.80**	0.71	0.72	0.73	0.72	0.70	**0.80**	0.70
Faceocc2	0.74	0.22	0.73	**0.79**	0.78	**0.79**	0.61	0.51	0.76	0.74
Average	0.68	0.43	0.58	0.49	0.54	0.49	0.72	0.59	0.60	**0.77**

TABLE IV
AVERAGE CENTER POINT ERROR OF EACH TRACKING ALGORITHM IN SOME VIDEOS

Video	SRDCF	KCF	ARCF_H	DSST	STAPLE_CA	Struck	DeepSRDCF	AutoTrack	BACF	STSL
BlurCar2	3.61	6.81	4.07	**2.86**	3.37	10.42	2.98	3.29	4.50	3.29
David3	**3.62**	4.30	4.39	88.38	4.17	106.5	4.32	4.73	4.43	4.70
Deer	**3.97**	21.16	4.22	16.69	**3.97**	5.27	4.64	4.03	12.40	4.03
Girl2	322.97	264.58	254.29	128.42	370.77	144.35	36.43	290.43	253.39	**7.39**
Jogging-1	4.70	88.27	95.58	87.90	98.91	62.06	**4.39**	89.88	4.77	4.70
Jogging-2	**3.65**	144.47	24.19	144.03	151.61	107.69	5.72	4.12	189.31	4.12
Woman	4.81	10.06	3.01	9.74	2.55	4.17	3.68	2.70	2.6	2.70
Faceocc2	**5.55**	7.67	9.64	6.73	7.27	5.96	13.51	32.62	8.03	10.14
Average	44.11	68.42	49.92	60.59	80.33	55.80	9.45	53.98	59.93	**5.14**

algorithms were able to track consistently and stably. In summary, the spatio-temporal adaptive learning framework adopted by the proposed STSL is reliable.

C. Attribute-based Comparison on OTB

We perform a specific analysis of the 11 challenging attributes. TABLE I shows that STSL performs best on all precision mapping attributes except motion blur, occlusion, and fast motion. TABLE II shows that STSL performs best on all success mapping attributes except scale variation, motion blur, and fast motion.

D. Evaluation on OTB100

Fig. 7 shows the accuracy and success rate plots of the ten algorithms in the OTB100 dataset. The top right corner of the figure shows the ranking of the algorithms. As shown in Fig. 7 (a), STSL ranked first in accuracy with 85.5%. As shown in Fig. 7 (b), the success rate of STSL ranks first and reaches 77.6%. It indicates that the proposed method has good tracking performance.

Table III shows each algorithm's average tracking overlap rate in different video sequences. Larger values indicate better algorithm tracking performance. The bold, underlined, and wavy lines indicate the top 3 algorithms. The average tracking overlap rates of the STSL in BlurCar2, David3, Deer, Girl2, Jogging-1, Jogging-2, Woman, and Faceocc2 sequences are 0.90, 0.77, 0.80, 0.72, 0.80, 0.76, 0.70 and 0.74, and the total average value reached 0.77. This algorithm ranked first among the 10 algorithms, and all these nine video sequences could achieve better tracking.

Table IV shows each tracking algorithm's average center point error in different video sequences. The smaller value indicates a smaller error relative to the target's true position. Then bolded, underlined, and wavy lines indicate the three algorithms with the highest tracking accuracy. The center point errors of the STSL for BlurCar2, David3, Deer, Girl2, Jogging-1, Jogging-2, Woman, and Faceocc2 video

sequences are 3.29, 4.70, 4.03, 7.39, 4.71, 4.12, 2.71, and 10.14 pixels, and the total mean pixel error is 5.14 pixels. This algorithm ranks first among the 10 algorithms. Then it also has the best tracking effect among the comparison algorithms. The STSL can still achieve excellent target tracking under multiple factors, such as scale change, occlusion, deformation, motion blur, and out-of-plane rotation in the Girl2 sequence.

V. Conclusions

Traditional DCF-based trackers improve filter learning by introducing predefined regularized. However, predefining these parameters requires a lot of energy and cannot be adapted to various situations. In this work, an anti-occlusion mechanism based on a multimodal template pool helps the correlation filter exploit the local and global information hidden in the response map to automatically adjust the hyperparameters in real time. The experiments show that the proposed algorithm has the following advantages:

(1) A spatial regularized factor is introduced to sense the local response variation, thus limiting the pixels with low confidence in the filter.

(2) A temporal regularized factor is introduced to help the correlation filter distinguish between effective appearance changes of the target and ineffective changes caused by occlusions, thus adaptively adjusting the filter learning.

(3) A pool of historical multimodal templates is introduced. Historical multimodal templates are employed to filter samples by setting thresholds, thus avoiding sample contamination and improving the reliability of tracking samples.

In the proposed tracking model, we only employ hand-craft features like HOG features and intensity features, which may limit the improvement of the tracing performance. Therefore, in the future, we will introduce deep features in the proposed model to improve the robust performance of STSL further.

References

[1] Q. Yu, T. B. Dinh, and G. G. Medioni, "Online tracking and reacquisition using co-trained generative and discriminative trackers," in *European Conference on Computer Vision*, Marseille, 2008, pp. 678-691.

[2] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, USA, 2010, pp. 2691-2698.

[3] T. Zhou, F. Liu, H. Bhaskar, and J. Yang, "Robust visual tracking via online discriminative and low-rank dictionary learning," *IEEE transactions on cybernetics*, vol. 48, pp. 2643-2655, 2017.

[4] M. Danelljan, F. S. Khan, M. Felsberg, and J. Van De Weijer, "Adaptive color attributes for real-time visual tracking," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, 2014, pp. 1090-1097.

[5] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *2010 IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, 2010, pp. 2544-2550.

[6] X.-F. Zhu, X.-J. Wu, T. Xu, Z.-H. Feng, and J. Kittler, "Complementary discriminative correlation filters based on collaborative representation for visual object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, pp. 557-568, 2020.

[7] D. Elayaperumal and Y. H. Joo, "Aberrance suppressed spatio-temporal correlation filters for visual object tracking," *Pattern Recognition*, vol. 115, pp. 79-89, 2021.

[8] X. Lu, C. Ma, B. Ni, and X. Yang, "Adaptive region proposal with channel regularization for robust object tracking," *IEEE transactions on circuits and systems for video technology*, vol. 31, pp. 1268-1282, 2019.

[9] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the Circulant Structure of Tracking-by-detection with Kernels," in *European Conference on Computer Vision*, Singapore, 2012, pp. 702-715.

[10] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed Tracking with Kernelized Correlation Filters," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 37, pp. 583-596, 2014.

[11] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, 2018, pp. 8971-8980.

[12] Q. Wang, Z. Teng, J. Xing, J. Gao, and S. Maybank, "Learning attentions: residual attentional siamese network for high performance online visual tracking," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, 2018, pp. 4854-4863.

[13] B. Yu, M. Tang, L. Zheng, G. Zhu, J. Wang, H. Feng, *et al.*, "High-performance discriminative tracking with transformers," in *2021 IEEE/CVF International Conference on Computer Vision*, Montreal, QC, Canada, 2021, pp. 9836-9845.

[14] D. Guo, Y. Shao, Y. Cui, Z. Wang, L. Zhang, and C. Shen, "Graph attention tracking," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Online, 2021, pp. 9538-9547.

[15] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *2017 IEEE Conference on Computer Vision and Pattern Recognition* Hawaii, 2017, pp. 6931-6939.

[16] J. Zhang, W. Feng, T. Yuan, J. Wang, and A. K. Sangaiah, "SCSTCF: spatial-channel selection and temporal regularized correlation filters for visual tracking," *Applied Soft Computing*, vol. 118, pp. 118-121, 2022.

[17] H. K. Galoogahi and A. L. S. Fagg, "Learning Background-aware Correlation Filters for Visual Tracking," in *2017 IEEE International Conference on Computer Vision(ICCV)*, New York, 2017, pp. 1144-1152.

[18] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Learning Spatially Regularized Correlation Filters for Visual Tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, Santiago, 2015, pp. 4310-4318.

[19] F. Li, C. Tian, W. Zuo, L. Zhang, and M. H. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 4904-4913.

[20] Y. Huang, Y. Chen, C. Lin, Q. Hu, and J. Song, "Visual attention learning and antiocclusion-based correlation filter for visual object tracking," *Journal of Electronic Imaging*, vol. 32, pp. 13-23, 2023.

[21] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference*, Nottingham, 2014, pp. 1-11.

[22] W. Yi, L. Jongwoo, and Y. Ming-Hsuan, "Object Tracking Benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, pp. 1834-1848, 2015.

[23] Y. Li, C. Fu, F. Ding, Z. Huang, and G. Lu, "AutoTrack: Towards High-Performance Visual Tracking for UAV with Automatic Spatio-Temporal Regularization," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2567-2587.

[24] Z. Huang, C. Fu, Y. Li, F. Lin, and P. Lu, "Learning Aberrance Repressed Correlation Filters for Real-time UAV Tracking," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Seoul, 2019, pp. 2891-2900.

[25] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Convolutional Features for Correlation Filter Based Visual Tracking," in *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, New York, 2015, pp. 621-629.

[26] S. Hare, A. Saffari, and P. Torr, "Struck: Structured Output Tracking with Kernels," in *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain*, 2011, pp. 2566-2578.

[27] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. Torr, "Staple: Complementary Learners for Real-Time Tracking," in *Computer Vision & Pattern Recognition*, New York, 2016, pp. 1401-1409.