Hindawi
Security and Communication Networks
Volume 2021, Article ID 9719745, 19 pages
https://doi.org/10.1155/2021/9719745

Research Article

An Enhanced Visual Attention Siamese Network That Updates Template Features Online

Wenqiu Zhu,[1,2] **Guang Zou**,[1,2] **Qiang Liu**,[1,2] **and Zhigao Zeng**[1,2]

[1]*College of Computer Science, Hunan University of Technology, Zhuzhou, Hunan 412007, China*
[2]*Intelligent Information Perception and Processing Technology, Hunan Province Key Laboratory, Zhuzhou, China*

Correspondence should be addressed to Guang Zou; 591891549@qq.com and Qiang Liu; liuqiang@hut.edu.cn

Received 1 May 2021; Revised 26 July 2021; Accepted 15 August 2021; Published 31 August 2021

Academic Editor: Yuan Yuan

Copyright © 2021 Wenqiu Zhu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recently, Siamese trackers have attracted extensive attention because of their simplicity and low computational cost. However, for most Siamese trackers, only a frame of the video sequence is used as the template, and the template is not updated in inference process, which makes the tracking success rate inferior to the trackers that can update the template online. In the current study, we introduce an enhanced visual attention Siamese network (ESA-Siam). The method is based on a deep convolutional neural network, which integrates channel attention and spatial self-attention to improve the discriminative ability of the tracker for positive and negative samples. Channel attention reflects different targets according to the response value of different channels to achieve better target representation. Spatial self-attention captures the correlation between two arbitrary positions to help locate the target. At the same time, a template search attention module is designed to implicitly update the template features online, which can effectively improve the success rate of the tracker when the target is interfered by the background. The proposed ESA-Siam tracker shows superior performance compared with 18 existing state-of-the-art trackers on five benchmark datasets including OTB50, OTB100, VOT2016, VOT2018, and LaSOT.

1. Introduction

Visual object tracking is a process of identifying the region of interest in the video, which can track the target in a given video. At present, visual object tracking is widely used in video surveillance [1], automatic driving [2], UAV tracking [3], and other fields. Although various researchers have done a lot of work on tracker to improve its performance, target tracking still faces such practical problems as fast motion, similar background interference, target scale transformation, low image resolution, and so on [4, 5].

The naive correlation filter tracker uses hand-crafted features, such as KCF [6], SRDCF [7], CACF [8], DSST [9], and SAMF [10]. Compared with the method of end-to-end learning using deep convolutional neural networks (CNNs), it is much inferior.

Recently, deep learning has been widely used in visual tracking. Trackers use CNNs to extract target features, and the tracking success rate and robustness are significantly improved. SiamFC [11] extracts template features and search features through AlexNet [12], uses the similarity measure method to perform cross-correlation operation on the extracted features to obtain the final response graph, and then predicts the target location according to the score of the response graph. Because the network model is simple and uses the offline pretraining network model, there is no online update and no complex calculation. Compared with the traditional online update method of correlation filter, SiamFC is faster and can meet the real-time requirements. Based on the Siamese network system, lots of trackers with state-of-the-art performance were proposed, such as SiamRPN [13], SiamMask [14], SiamRPN++ [15], and DaSiamRPN [16]. There are also some additional methods to build tracker based on different angles, such as thermal infrared [17], self-supervised [18], and focusing target regression model [19].

Siamese architecture has been applied in various fields of artificial intelligence, such as one-shot image

recognition [20], human reidentification [21], sentence similarity [22], and visual object tracking [23]. For visual object tracking, Siamese-based trackers train offline based on a large amount of data but do not update the target template online. Therefore, in the face of severe target deformation, scaling, occlusion, and other scenes, the target will be lost, resulting in the performance and robustness degradation of the tracker. In addition, the features extracted by CNN do not distinguish the weight in channel and space, and we know that different channel features correspond to different target information. The response channels that represent the tracking target should be given more weight, and not all of them should be given the same weight. The visual attention mechanism [24–26] can pay attention to the channel and location of interest and screen out the feature information that can represent the tracking target better. Based on this, we design a novel visual tracking method which can update the target online by enhancing the hybrid visual attention.

Inspired by the application of visual attention mechanism in RASNet [27] and EFCTA [28], we propose an enhanced visual attention Siamese network referred to as ESA-Siam. Considering that the information of search branch and template branch is mutually compensated, the context information of the search branch is also important. Combining the target information of the search branch can help the tracker identify positive and negative samples better. Therefore, we design a template search collaborative attention module, called T-SCAttn, which can update the template features online. It can improve the robustness and the positive and negative sample discrimination of the tracker and better deal with the problems of low image resolution and target occlusion. The main contributions of our work are as follows:

(1) We introduce a new twin network visual tracking algorithm based on the enhanced visual attention mechanism (including channel attention, spatial self-attention, and template search collaborative attention). Channel attention distinguishes background and targets according to different target response values, and spatial self-attention aggregates nonlocal context information to help target location better.

(2) We design a template search collaborative attention module which can update the template features online by recalculating the template images and search images.

(3) We change the traditional pooling layer. Based on this, we propose golden threshold stochastic pooling to activate the target features with a higher probability and ignore other background features.

(4) Our approach in the benchmark datasets OTB50 [29], OTB100 [30], VOT2016 [31], VOT2018 [32], and LaSOT [33] has excellent tracking performance, the tracking of which can reach speeds of up to 60 fps.

2. Related Work

Since MOSSE [34], trackers based on correlation filtering have been widely used due to the convenience and simplicity of the hand-crafted features. Such methods can update targets online and have high accuracy. However, due to its simple feature, the robustness is poor when the target is blocked and the appearance is deformed. The depth features based on CNNs can more fully express the target features. As a result, a number of tracking methods that combine related filtering and depth features have emerged, such as C-COT [35], CFNet [36], MDNet [37], DeepSRDCF [38], and ECO [39], to achieve better tracking performance. CFNet combines correlation filtering with SiamFC to win the VOT2017 real-time challenge and introduces a cyclic displacement matrix in SiamFC to improve performance. MDNet proposes a multidomain learning model based on CNN to distinguish multiple different independent targets.

In recent years, the current branch of building trackers is based on the Siamese network system. Since the SiamFC was proposed, more tracking methods based on this Siamese network have been proposed. Li et al. introduced candidate regions for target detection and proposed SiamRPN to treat the tracking task as a two-stage task: one is target classification and the other is target regression. Wang et al. combined target tracking with image segmentation; SiamMask segmented the target through a mask and completed the image segmentation while completing the target extraction. Zhu et al. proposed DaSiamRPN to effectively control the sampling strategy on the basis of SiamRPN, balanced the distribution of positive and negative samples, and improved the tracking performance. SA-Siam [40] uses two Siamese networks: one is to extract the semantic branch of high-level features of the target and the other is to extract the appearance branch of low-level features of the target; the network branches are trained separately and feature fusion is performed to improve the robustness of the tracker. Recently, Li et al. proposed SiamRPN++, using the deeper network ResNet50 [41] as the backbone network, analyzed the reason the Siamese network system cannot use the deep network, and further improved the tracking performance. However, since there is no online update, Siamese-based trackers are easily interfered by target occlusion and complex background.

Recently, with the widespread application of the visual attention mechanism in the field of computer vision, Hu et al. proposed SENet [42], which gives weights to different channels by squeeze and excitation channels and statistically the global information of the image at the characteristic channel level, selecting features in a targeted manner. Shaw et al. proposed novel self-attention, which pays more attention to the correlation between internal feature elements and obtains the global dependence of any two positions in the feature map. Wang et al. proposed a generalized and simple nonlocal block [43] that can be directly embedded in the network, which can capture time and space information for integration. Particularly, Wang et al. combined the visual attention mechanism to propose RASNet, which separated feature learning and discriminant analysis and used

cross-correlation to update the target to enhance the ability to distinguish between target and background. However, only using the feature information of the target, the distinction between the target and the background is not enough, and it is impossible to face complex scenes.

3. Proposed Method

The overall framework of the enhanced visual attention Siamese network is shown in Figure 1. Compared with other Siamese-based trackers, ESA-Siam uses a template and search area to coordinate the attention block to update the target online implicitly to adapt to changes in the target's appearance. Second, our network uses a golden threshold stochastic pooling layer to activate target features with greater probability. Finally, we use the channel attention mechanism and spatial self-attention to filter feature maps and combine the correlation between global context information and local features to help locate targets and estimate target contours. The following sections describe in detail the various components of the proposed tracker.

3.1. Siamese-Based Trackers. The key point of the SiamFC tracking algorithm is the use of offline training and online fine-tuning of the network, which can effectively improve the speed of the algorithm. Its network structure is composed of a template branch and a search branch, and the two branches extract features through the same shared network (AlexNet). We perform cross-correlation of the extracted two branch features to calculate the feature similarity and locate the target according to the similarity value. The position with high similarity is the target position. When a full convolutional network is used, the size of the search image does not need to be consistent with the template image, which can provide a larger search area for the network and calculate the similarity of more subwindows. The cross-correlation function is shown in the following formula:

$$f(z, x) = \varphi(z) * \varphi(x) + b_1, \tag{1}$$

where x is the input search image, z is the input template image, $\varphi(\cdot)$ is the feature extraction network, $*$ represents the convolution operation, b represents the offset of each position in the score map, and $f(z, x)$ represents the similarity score map between the template feature and the search feature. The position with the highest score is the target position.

3.2. Golden Threshold Stochastic Pooling. Zeiler and Fergus combined the advantages and disadvantages of maximum pooling and average pooling to propose stochastic pooling [44]. Zeiler believes that the maximum pooling is always to select the largest activation from the pooling area every time, completely excluding other activations except the maximum value. Stochastic pooling applies multinomial distribution and calculates the score probability of each response position to randomly select activation. In this way, nonmaximum activations could also be selected. We calculate the probability of each position i by normalizing the area activation, as shown in the following formula:

$$P_i = \frac{a_i}{\sum_{k \in R_j} a_k}, \tag{2}$$

where a_i is the activation value of position i and R_j represents the area j in the feature map. Multinomial distribution selects a location i within the region:

$$s_j = a_i, \quad \text{where } l \sim P\left(p_1, \ldots, p_{|R_j|}\right), \tag{3}$$

where s_j represents the final activation of region j, which is randomly selected by the probability calculated through each position of region j. The activation with the greater probability is more likely to be selected. Although this can ensure that the information is not lost to a certain extent, because of its randomness, it is possible to select a value with a small activation probability and lose important information. In the target tracking task, we should try to avoid this uncertainty. Therefore, we improve on the basis of stochastic pooling, sort the calculated probabilities, and filter out some activations with too small probability values by setting a threshold T (e.g., $T = 0.002$).

$$s_j = a_i, \quad \text{where } l \sim P\left(p_1, \ldots, p_{|R_j|}\right) > T. \tag{4}$$

The selection of T is set according to the ratio of the maximum activation part (e.g., $T = 0.618P_{\max}$). Meanwhile, to make reasonable use of the advantages of the maximum pooling layer to highlight important information, we pay more attention to the top ranked by the activation value, so that the random selection can fall in this range with a high probability, and weaker activations are inhibited. An example of golden threshold stochastic pooling is shown in Figure 2. The backpropagation process is similar to the maximum pooling backpropagation, and only the value of the position of the selected node that has been recorded by the forward propagation is retained, as shown in the following formula:

$$\frac{\partial L}{\partial x_i} = \begin{cases} 0, & \delta(i, j) = \text{false}, \\ \dfrac{\partial L}{\partial y_j}, & \delta(i, j) = \text{false}, \end{cases} \tag{5}$$

where x_i, y_i are the input node and output node and $\delta(i, j)$ is the decision function, which represents whether the input node i is selected as the maximum output by the output node j.

3.3. Channel Attention Module. According to the characteristics of the target tracking task, we designed an enhanced attention mechanism, as shown in Figure 3, which consists of a spatial self-attention module, a channel attention module, and a template search collaborative attention module. The spatial attention module is based on the correlation dependency structure of the pixels at the same

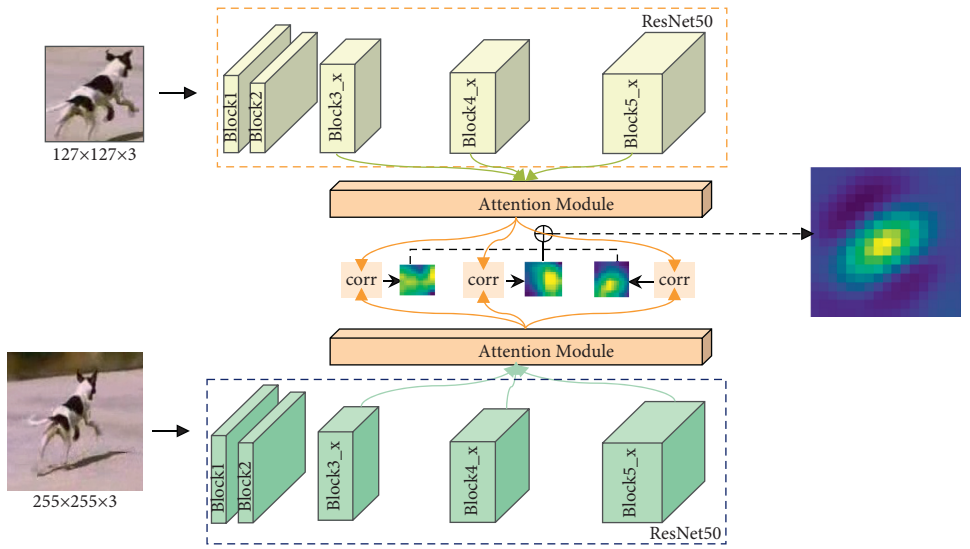


Figure 1: ESA-Siam network framework. Based on the Siamese benchmark network framework, ESA-Siam uses ResNet50 as the backbone network to do attention screening on the last three network blocks of the template branch and the search branch. After that, cross-correlation operations are performed on the template features and their respective search features, and then the fusion features are performed to obtain the final output feature map.

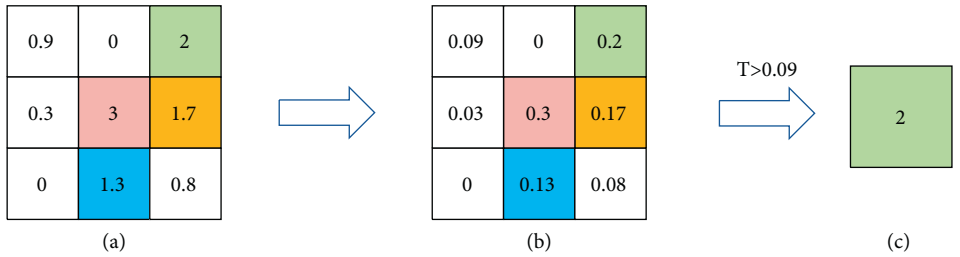


Figure 2: Example of golden threshold stochastic pooling. (a) Activation within a given pooling region. (b) Probability of activation. (c) Sampled activation. If the threshold $T > 0.09$ is set, the probability of selecting a nonmaximum value will increase.

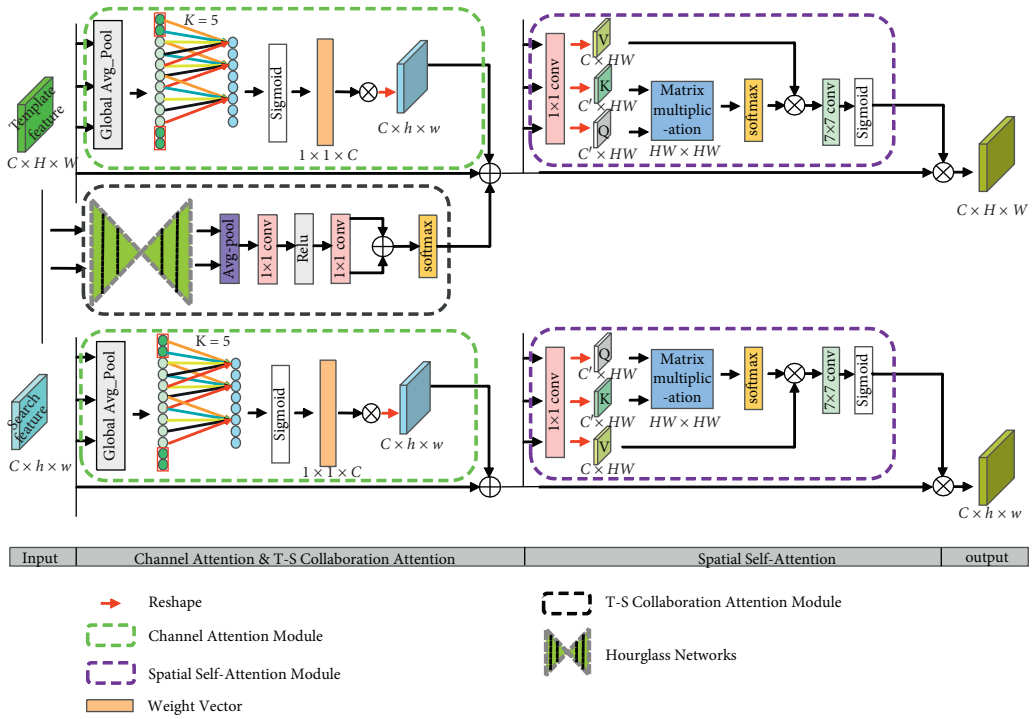


Figure 3: The details of the attention module. It consists of three parts, a channel attention module, a spatial self-attention module, and a template search collaborative attention module. The features extracted through the backbone network are input to the channel attention and T-SCAttn, and the features generated are finally input to the spatial self-attention module for optimization and improvement.

location in the feature map to characterize features. Spatial self-attention mechanism can capture the relationship between internal data and features to establish a correlation between any two locations. The feature of particular location can be weighted and summed through all location feature information. Channel attention can distinguish targets by the response of different channels to different targets. A channel with a high response value may represent the same target, and a higher response weight will be given, while a lower response weight will be given, so as to adjust the characteristic response adaptively. The template search collaborative attention module captures nonlocal semantic feature information globally and updates template features through the hourglass network [45].

The traditional channel attention mechanism uses a multilayer perceptron (MLP) method to calculate the weight of each channel. This method increases a great number of parameters due to the use of a large number of fully connected layers, reduces the calculation speed, and affects the real-time performance of the algorithm. We designed and introduced the ECA module in ECA-Net [46] to avoid the negative impact of dimensionality reduction using a fully connected layer, and at the same time, proper cross-channel interaction can significantly reduce the model parameters. This strategy adopts one-dimensional convolution to realize and uses the feature of convolution operation weight sharing. The size of the convolution kernel k in one-dimensional convolution is obtained through adaptive calculation. The specific calculation formula is as follows:

$$k = \psi(C) = \left|\frac{\log_2(C)}{\gamma} + \frac{b}{\gamma}\right|. \tag{6}$$

In general, the number of channels is always the power of 2. We set $r = 2$, $b = 1$. Through the adaptive convolution kernel size k to completethe cross-channel information interaction, so that the layer with a largernumber of channels can interact more between channels.

Compared with using multilayer perceptrons to connect to each other, the parameter number is significantly reduced, to ensure the real-time nature of the algorithm. As shown in Figure 3, the channel attention module (C-Attn) squeezes the input feature map F, and after global average pooling, a feature vector $f = (f_1, f_2, \ldots, f_c)$ is obtained as the input of the one-dimensional convolutional layer, where $f_i \in \mathbb{R}$. Then, we get the weight vector $P = (p_1, p_2, \ldots p_c)$ from the sigmoid function, where $p_i \in \mathbb{R}$. The input feature F is elementwise multiplied with the weight vector P. Finally, we get the features $F_A^C \in \mathbb{R}^{C\times h\times w}$ filtered by the channel attention.

3.4. Spatial Self-Attention Module. The self-attention space module is a supplement to channel attention, as shown in Figure 3. Channel attention and spatial self-attention work in series. The output of the channel attention is the input of the spatial self-attention module. $F_A^C \in \mathbb{R}^{C\times h\times w}$ is input to an independent 1×1 convolution and passes through three conversion functions to obtain three feature vectors $Q, K \in \mathbb{R}^{C'\times HW}$, $V \in \mathbb{R}^{C\times HW}$. We transpose vector Q and then perform matrix multiplication with vector K. We can generate a spatial self-attention feature map through the columnwise softmax operation as follows:

$$\beta_{i,j} = \exp\frac{\left(Q_i^T \cdot K_j\right)}{\sum_{i=1}^{\mathrm{WH}} \exp\left(Q_i^T \cdot K_j\right)}, \tag{7}$$

where $\beta_{i,j}$ represents the weight between the i-th location region and the j-th location region. The result $\beta_{i,j}$ is elementwise multiplied with vector V. Then, we performed a 7×7 convolution operation. A sigmoid activation is performed for generating a feature vector with weights $\Omega = (\omega_1, \omega_2, \ldots, \omega_c)$, where $\omega_i \in \mathbb{R}^{C\times HW}$. After that, the input feature is elementwise multiplied with Ω. Finally, we get the final output feature $F_A^S \in \mathbb{R}^{C\times h\times w}$ with high similarity to the target by the following formula:

$$X_A^S = \alpha\Omega F, \tag{8}$$

where α is a hyperparameter. We initialize it to 0.0001 and then gradually increase it to give more weight, which can adapt to simple tasks at the beginning and face more complex tasks later.

3.5. Template Search Collaborative Attention Module (T-SCAttn). . We designed a template search collaborative attention (T-SCAttn) module to implicitly update template features. We combine the context information of the target in the search image with the template feature and use the context information to improve the accuracy of target positioning. Search branch and template branch are complementary. T-SCAttn is composed of two components, where one is used to perform multiscale information interaction between template features and search features extracted by the backbone network. We use the stacked hourglass network (as shown in Figure 4) to generate multiscale template information X_t^{T-S} and multiscale search information X_s^{T-S}. The hourglass network does not change the size of the feature map. Another component is used to perform attention filtering on features. Inspired by CBAM, we only use global average pooling and one-dimensional convolution to get the context information of the feature map. We first perform a 1×1 convolution to reduce the number of channels to one channel. Then, we apply the ReLU activation function and one-dimensional convolution to filter the context information feature map and apply the softmax layer. The result of the T-SCAttn is elementwise added with the output of channel attention module and input feature. We can get it according to the following formula:

$$X^{T-S} = \mathrm{Softmax}\left(\mathrm{AvgPool}\left(X_t^{T-S}\right) + \mathrm{AvgPool}\left(X_s^{T-S}\right)\right), \tag{9}$$

where X^{T-S} is the output of the T-SCAttn module.

3.6. Network Structure and Algorithm. The proposed network is based on the Siamese network and an enhanced attention mechanism. The proposed network framework can be described as

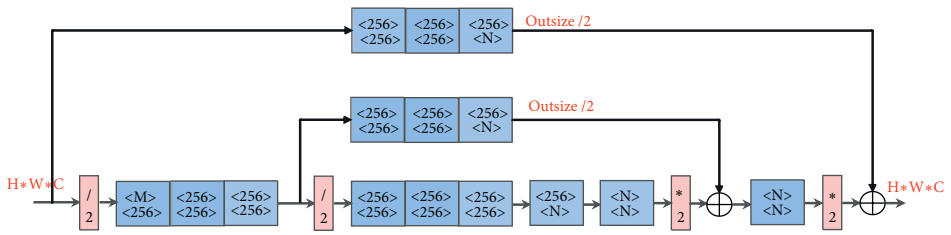


Figure 4: Structure of stacked hourglass networks. The size of input feature map is $H * W * C$. /2 represents downsampling, and $*$ 2represents upsampling, using nearest neighbor upsampling. The upper < > content in the blue box represents the number of input channels, and the lower < > represents the number of output channels. ⊕ represents elementwise addition. The size of the output feature map is $H * W * C$. Stacked hourglass network has nothing to do with the input size, and it only needs to provide the number of input and output channels. Also, it can gradually extract deeper features.

$$m(T,S) = [\eta(\mu(\psi(Z)) + \Delta(Z,X))] * [\eta(\mu(\psi(X)))] + b, \tag{10}$$

where Z is a template image, X is the search patch, $\psi(\cdot)$ represents the backbone network, $\mu(\cdot)$ denotes the channel attention module, $\eta(\cdot)$ denotes the spatial self-attention module, and $\Delta(\cdot)$ represents the template search collaborative attention module. The output of $\psi(\cdot)$ is fed to the channel attention module as $\mu(\psi(Z))$. Then, the output forwards input to the spatial self-attention module as $\eta(\mu(\psi(Z)))$.

The backbone network uses ResNet50, and the network structure and the corresponding operations of each layer are shown in Table 1. The network is divided into five blocks, and the number of residual blocks from the second block to the fifth block is (3, 4, 6, 3). To avoid the resolution of the feature map extracted by the network from being too small, the last three blocks are not downsampled but replaced by the dilated convolution, which increases the receptive field while not making the feature map resolution too small. We train on positive and negative samples to construct a loss function. We get the optimal model parameters by minimizing the loss function. A positive sample is expressed as a point that does not exceed a certain pixel distance from the center. If one sample point exceeds the distance range, it is treated as a negative sample. The single point loss function is defined as shown in the following formula:

$$l(y,v) = \log(1 + \exp(-yv)), \tag{11}$$

where $y \in (+1,-1)$ indicates the ground-truth label of the sample and v represents actual score of the template image and search image. We use the average loss value of all location points to represent the loss during training as shown in the following formula:

$$L(y,v) = \frac{1}{D}\sum_{u \in D} l(y[u], v[u]), \tag{12}$$

where D represents the score map, u is the search position, and $v[u]$ represents the score for each position. We use stochastic gradient descent (SGD) during training to find the global minimum of the loss function as shown in the following formula:

$$\arg\min_{\theta} E(L[y, f(z,x,\theta)]), \tag{13}$$

where θ refers to the network parameters and E represents the mathematical expectation.

We also describe the training algorithm and testing algorithm of the proposed network framework, as shown in Algorithms 1 and 2.

4. Experiments and Results

We evaluate the proposed tracker algorithm ESA-Siam on five benchmark datasets, including OTB50, OTB100, VOT2016, VOT2018, and LaSOT. We compared 18 state-of-the-art tracking methods, including SiamDW [47], DSiam, HCF [48], CSR-DCF [49], GradNet [50], Staple [51], fDSST [52], UpdateNet [53], RASNet, SAMF, SiamRPN, DeepSRDCF, SRDCF, CFNet, MDNet, C-COT, ECO, and SiamFC.

4.1. Implementation Details. We use ResNet50 trained offline on GOT10K [54] as the backbone network. The GOT10K dataset contains more than 10,000 video clips of real moving objects and more than 1.5 million manually labeled bounding boxes, covering more than 560 categories. According to SiamFC, we set the search image size during training and testing to $127 \times 127 \times 3$ and the template image size to $255 \times 255 \times 3$. We use stochastic gradient descent (SGD) optimizer with momentum set to 0.9 to minimize equation (13). During training, the initial learning rate is set to 0.01, the L2 penalty item (weight_decay) is set to $5e-4$, and the learning rate is exponentially decayed until 10^{-5}. The batch_size is 8, and the training epochs are 50. We set the threshold $T = 0.618P_{\max}$ in equation (4) and use three scale ratios [0.9638, 1, 1.0375] to scale the search patch. We set the initial value of the hyperparameter in equation (8) to 0.0001 and then increase it to $10^{-1} \sim 10^{-2}$ exponentially. Our method is implemented based on Python3.8, Cuda10.2, and Pytorch1.6. The experiment was performed on a machine with a CPU model of Intel(R)Core(TM)i5-9400F CPU @2.90 GHz, a graphics card of NVIDIA GeForce RTX 2070s, and a memory of 32 GB DDR4 RAM. The average tracking speed of the proposed tracker was 60 frames per second (FPS). The loss change during training is shown in Figure 5. The Y-axis is the loss value, and the X-axis is the number of training batches.

Table 1: Network structure and operations corresponding to each network block (block represents network block, Gold-SPool represents golden stochastic pooling, dilation represents dilated convolution, ResNet in Figure 1 includes Block1, Block2, Block3, Block4, and Block5, and "—" represents no operation).

Block	Operation	Template size	Search size
—	—	$127 \times 127 \times 3$	$255 \times 255 \times 3$
Block1	$7 \times 7,\ 64,\ 3 \times 3$Gold-SPool, s = 2	$31 \times 31 \times 64$	$62 \times 62 \times 64$
Block2	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$15 \times 15 \times 256$	$31 \times 31 \times 256$
Block3 + dilation	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$15 \times 15 \times 512$	$31 \times 31 \times 512$
Attention	—	$15 \times 15 \times 256$	$31 \times 31 \times 256$
Block4 + dilation	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$15 \times 15 \times 1024$	$31 \times 31 \times 1024$
Attention	—	$15 \times 15 \times 512$	$31 \times 31 \times 512$
Block5 + dilation	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 024 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$15 \times 15 \times 2048$	$31 \times 31 \times 2048$
Attention	—	$15 \times 15 \times 1024$	$31 \times 31 \times 1024$

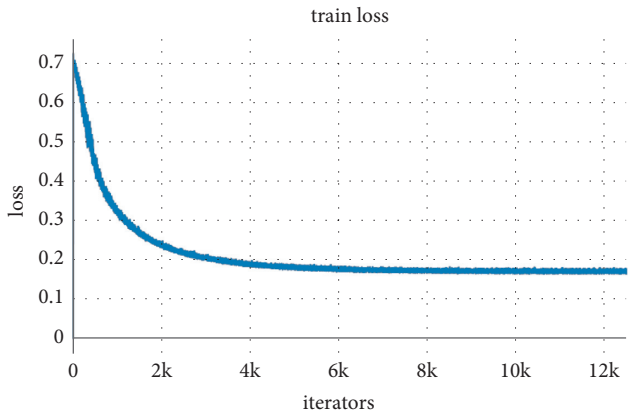


Figure 5: The loss changes during training.

4.2. Experiments on OTB50 and OTB100. The proposed method is evaluated on the OTB50 and OTB100 benchmark datasets. OTB50 has 50 video sequences, and OTB100 has 100 video sequences. The OTB dataset evaluation tool evaluates the tracking algorithm through two indicators: precision plot and success plot. The evaluation standard of tracking accuracy is the percentage of the number of frames with the center position error within T1 (the experiment is set to 20) pixels to the number of frames in the entire video sequence. The tracking success rate refers to the percentage of the frame number of the entire video sequence whose intersection ratio IoU (Intersection over Union) is greater than the threshold T2 (experimentally set to 0.5) between the target frame predicted by the algorithm and the real target frame, as shown in the following equation:

$$\mathrm{IoU} = \frac{\mathrm{Box}_t \cap \mathrm{Box}_g}{\mathrm{Box}_t \cup \mathrm{Box}_g}, \tag{14}$$

where Box_t represents the area of the area enclosed by the target prediction bounding box and Box_g represents the area enclosed by the target real bounding box.

As shown in Figure 6, the tracking accuracy and tracking success rate of ESA-Siam on the OTB50 dataset are 0.85 and 63.30, respectively, which are 4% and 4.77% higher than the state-of-the-art algorithm SiamRPN. Compared with the reliable channel-based method, CSR-DCF has increased by 11% and 10.04%.

It can be seen from Figure 7 that the tracking accuracy and tracking success rate of ESA-Siam on the OTB100 dataset are 0.863 and 65.04, respectively. Compared with the basic algorithm SiamFC, the tracking success rate has increased by 6.72%. It is 9% and 6.65% higher than that of the CFNet algorithm that combines correlation filtering and SiamFC, respectively, and 1.8% and 2.13% higher than that of SiamRPN. Meanwhile, the tracking success rate of ESA-Siam on the OTB100 dataset is 2.7% higher than that of SiamDW which also introduced the ResNet50 network. In addition, the performance of ESA-Siam is also better than that of CSR-DCF, a reliable channel-based method, and

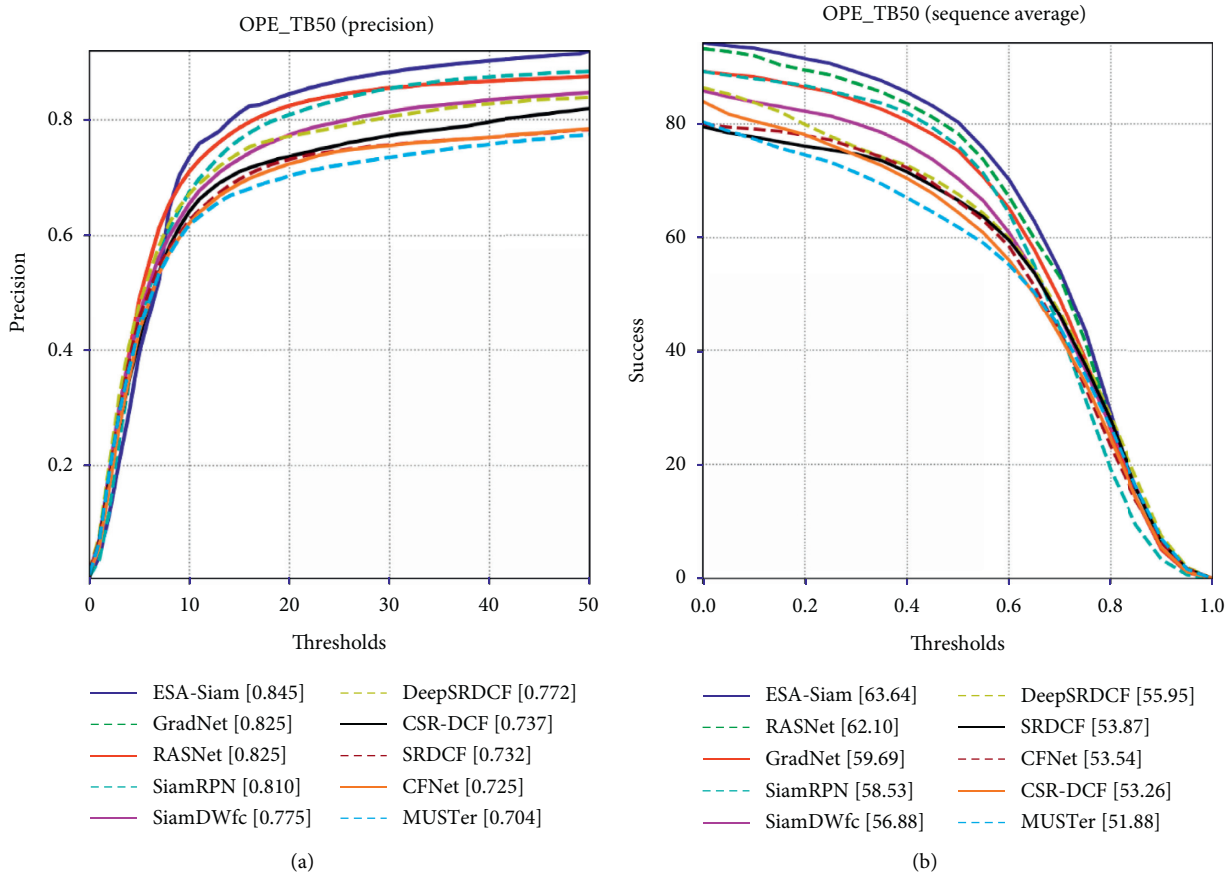


Figure 6: Test results of different trackers on the OTB50 dataset: (a) precision plot of OPE on OTB50; (b) success plot of OPE on OTB50.

RASNet, a method that integrates attention. ESA-Siam integrates the hybrid attention mechanism while improving the downsampling method of ResNet50 and uses the T-SCAttn module to implicitly update the template features, which can select features that are more discriminative to the target, so it can improve the robustness of the algorithm.

To verify the robustness of the ESA-Siam algorithm, we further carried out experiments on 11 tracking challenges on the OTB100 dataset, including Background Clutter (BC), Deformation (DEF), Fast Motion (FM), In-Plane Rotation (IPR), Illumination Variation (IV), Low Resolution (LR), Motion Blur (MB), Occlusion (OCC), Out-of-Plane Rotation (OPR), Out-of-View (OV), and Scale Variation (SV). As shown in Figure 8, we mainly evaluated the success of OPE on OTB100. We have observed that the ESA-Siam algorithm has won the championship in IV, IPR, LR, OCC, and so on. In other challenges such as SV, BC, and so on, ESA-Siam also has achieved great tracking performance.

Quantitative analysis of the algorithm was done as described in the previous section, in order to further verify the effectiveness of ESA-Siam. At the same time, a challenging sequence was selected from the OTB dataset for qualitative testing of the algorithm. Meanwhile, it was compared with CFNet and the related filtering algorithm DeepSRDCF combined with deep learning features, SiamFC and CSR-DCF. In the comparative experiment, six video sequences of Bird2, Human9, KiteSurf, Matrix, Singer2, and Dancer2 were selected. These six video sequences include IPR, OPR, LR, OCC, IV, DEF, FM, BC, and other challenges. Figure 9 shows the tracking effect comparison of the five algorithms including ESA-Siam. In these challenging sequences, the ESA-Siam algorithm has achieved better tracking results.

4.3. Experiments on VOT2016, VOT2018, and LaSOT. We also tested the methods on the VOT2016, VOT2018, and LaSOT datasets according to the three indicators expected average overlap (EAO), accuracy (A), and Robustness (R). Among them, EAO can be used as an index for comprehensive performance evaluation of the algorithm. The calculation of EAO is related to the accuracy and robustness. First, the average of per-frame overlaps Φ_{N_s} in the length N_s of the video sequence is defined as

$$\Phi_{N_s} = \frac{1}{N_s}\sum_{i=1}^{N_s}\Phi_i, \tag{15}$$

where Φ_i is the accuracy rate between the predicted target frame and the real target frame. EAO is defined as

$$\Phi = \frac{1}{N_{hi} - N_{lo}}\sum_{N_s=N_{lo}}^{N_{hi}}\Phi_{N_s}. \tag{16}$$

Table 2 shows the comparison of the test results of each method on VOT2016. We compared 9 algorithms including ESA-Siam, HCF, SAMF, SiamFC, SRDCF, MDNet,

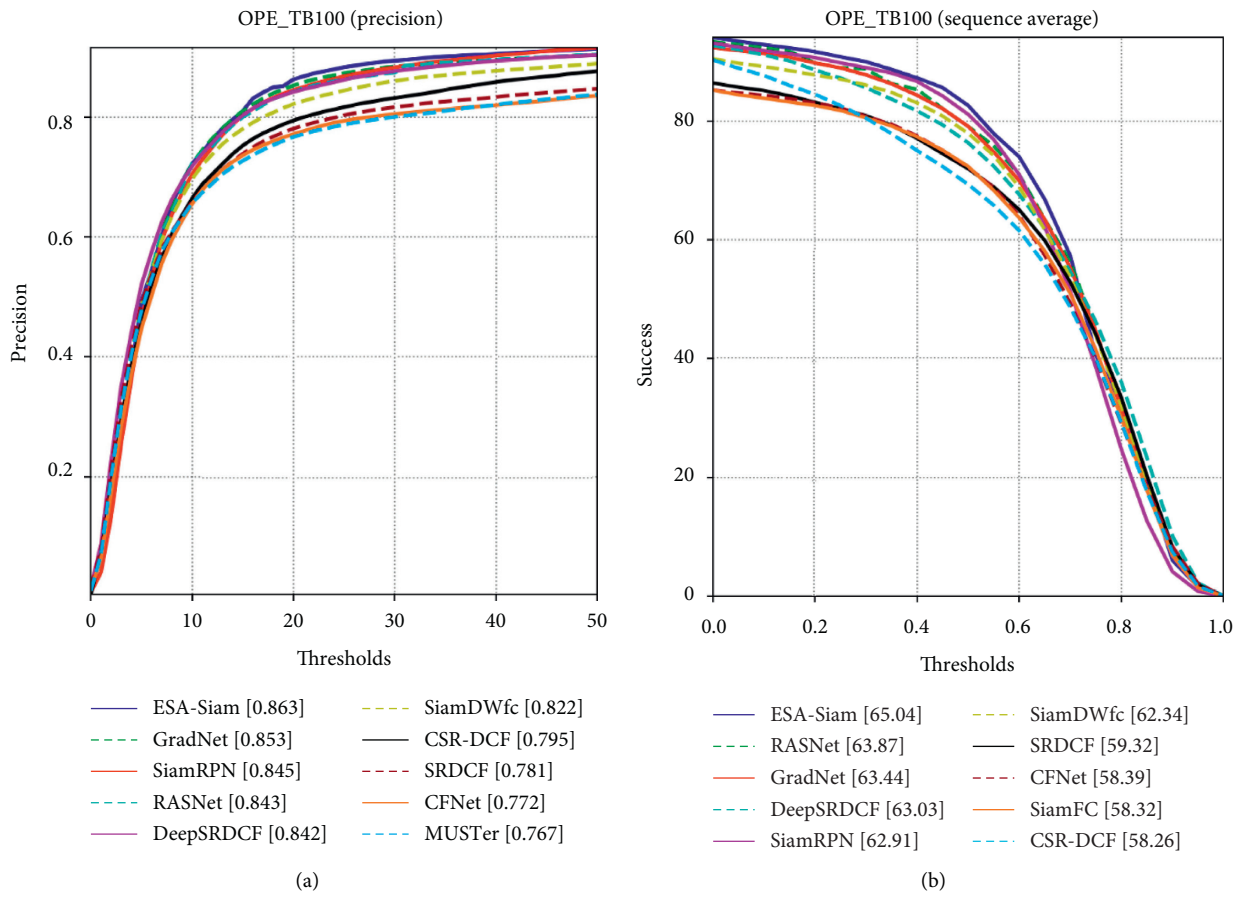


Figure 7: Test results of different trackers on the OTB100 dataset: (a) precision plot of OPE on OTB100; (b) success plot of OPE on OTB100.

DeepSRDCF, Staple, and C-COT. ESA-Siam has achieved good results on the indicators of accuracy and robustness.

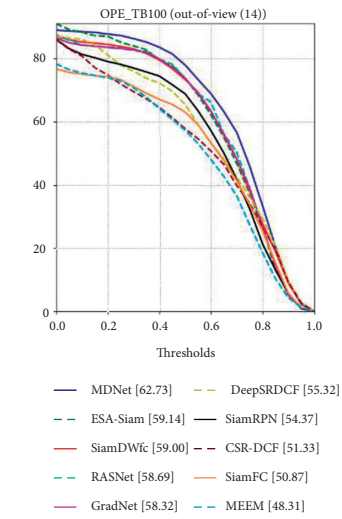
Table 3 shows the comparison of the tracking results of each method on the VOT2018 test set. Our proposed method achieves accuracy of 0.618, robustness of 0.223, and EAO of 0.411. On OVT2018, our method achieved the highest scores on the three evaluation indicators. Compared with the state-of-the-art SiamRPN, our method has a significant improvement of 7.5% and 5.6% in EAO and accuracy, respectively.

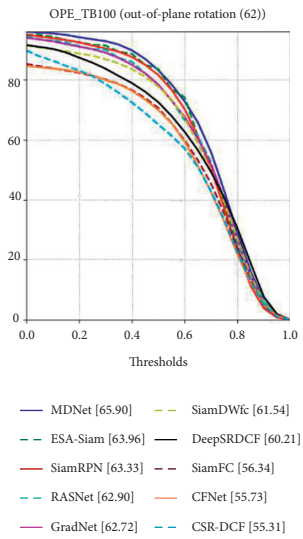
Table 4 shows the comparison of the test results of each algorithm on the LaSOT dataset. Our method achieved the highest scores in both success rate and standardization accuracy. Compared with MDNet, the standardization accuracy has increased from 0.461 to 0.515 (with 0.413 $\longrightarrow$ 0.450).

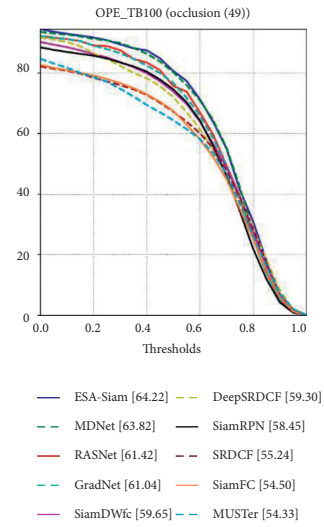
In addition, we selected three complex challenge scenarios (including Deformation, Motion Blur, and Partial Occlusion) on the LaSOT dataset to evaluate the proposed method with 9 existing state-of-the-art algorithms as shown in Figure 10. The experimental results show that the proposed method has achieved the champion results in tracking precision and success rate. In terms of success rate, the proposed method improved by nearly 6 percentage points over the second place. The proposed method can better deal with the challenging scenes in real life, such as Full Occlusion or Partial Occlusion, Target Deformation, and so on.

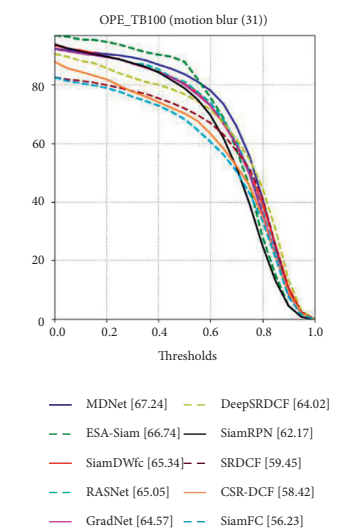
4.4. Ablation Study. We conducted extensive ablation studies with ESA-Siam on OTB100 to verify the effectiveness of its various components. Two indicators are used to evaluate the work of each component: one is tracking accuracy and the other is tracking success rate. We name the methods of using different components. The component using spatial self-attention is named S-Attn, the component using channel attention is named C-Attn, and the component using template search feature is named T-S-Attn. In addition, we compare the evaluation results of each component with the benchmark algorithm SiamFC and CSR-DCF, as shown in Figure 11. Experimental results verify the effectiveness of various components of ESA-Siam. Compared with SiamFC, the performance of the channel attention module C-Attn has increased by 7.6% on precision (with 58.32 $\longrightarrow$ 63.95 on success). DeletedOn the other hand, the introduction of the template-searchcollaboration attention module T-S-Attn is 5.83% higher in success rate and5.9% higher in accuracy than CSR-DCF, which is also based on channel weighting.

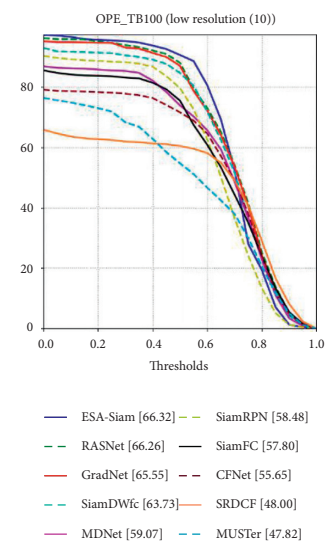
In addition, we also conducted experiments on the gold stochastic pooling method and other components in OVT2016. Table 5 shows the performance changes of the tracker with the integration of different components. As

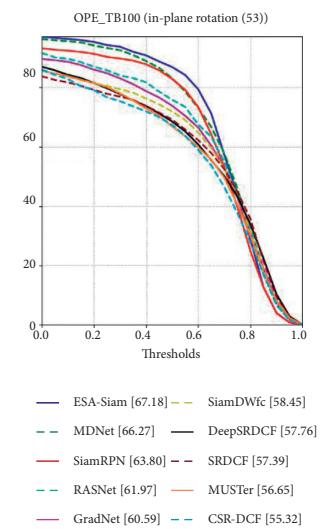


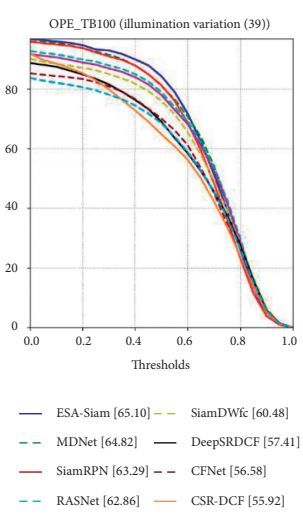


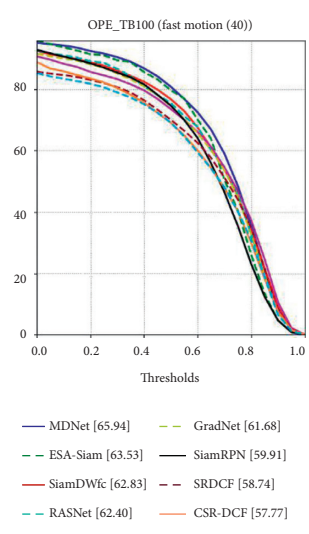


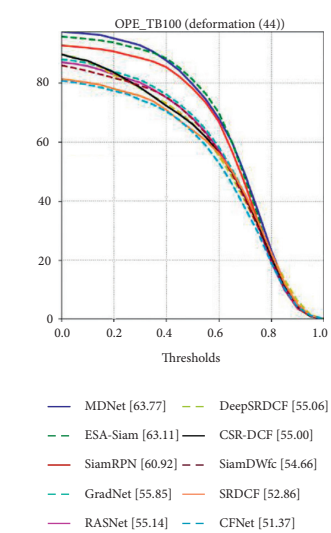






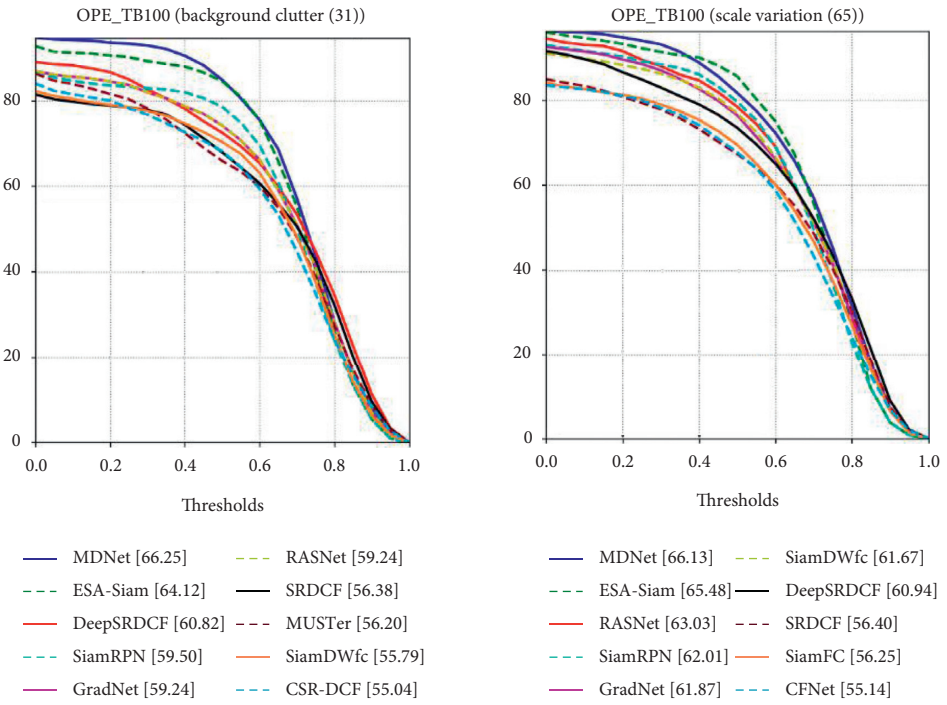






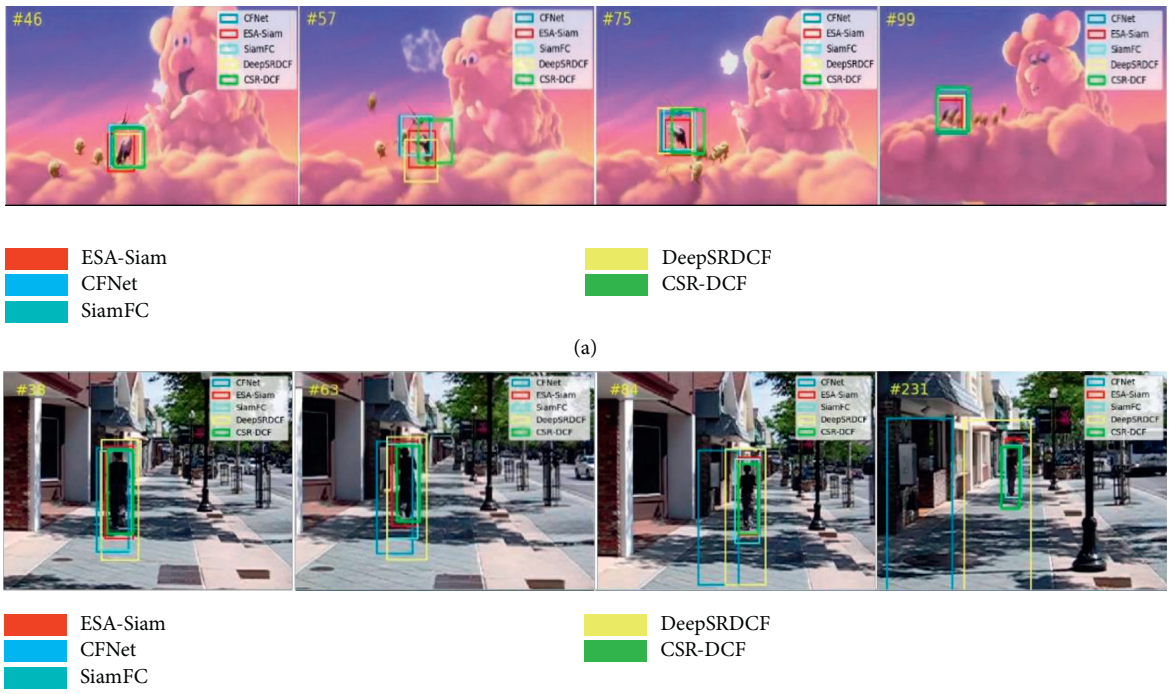
(a)

Figure 8: Continued.



(b)

Figure 8: Comparison of the tracking success rate of 11 challenge sequences on the OTB100 dataset by 10 different algorithms.



(a)

(b)

Figure 9: Continued.

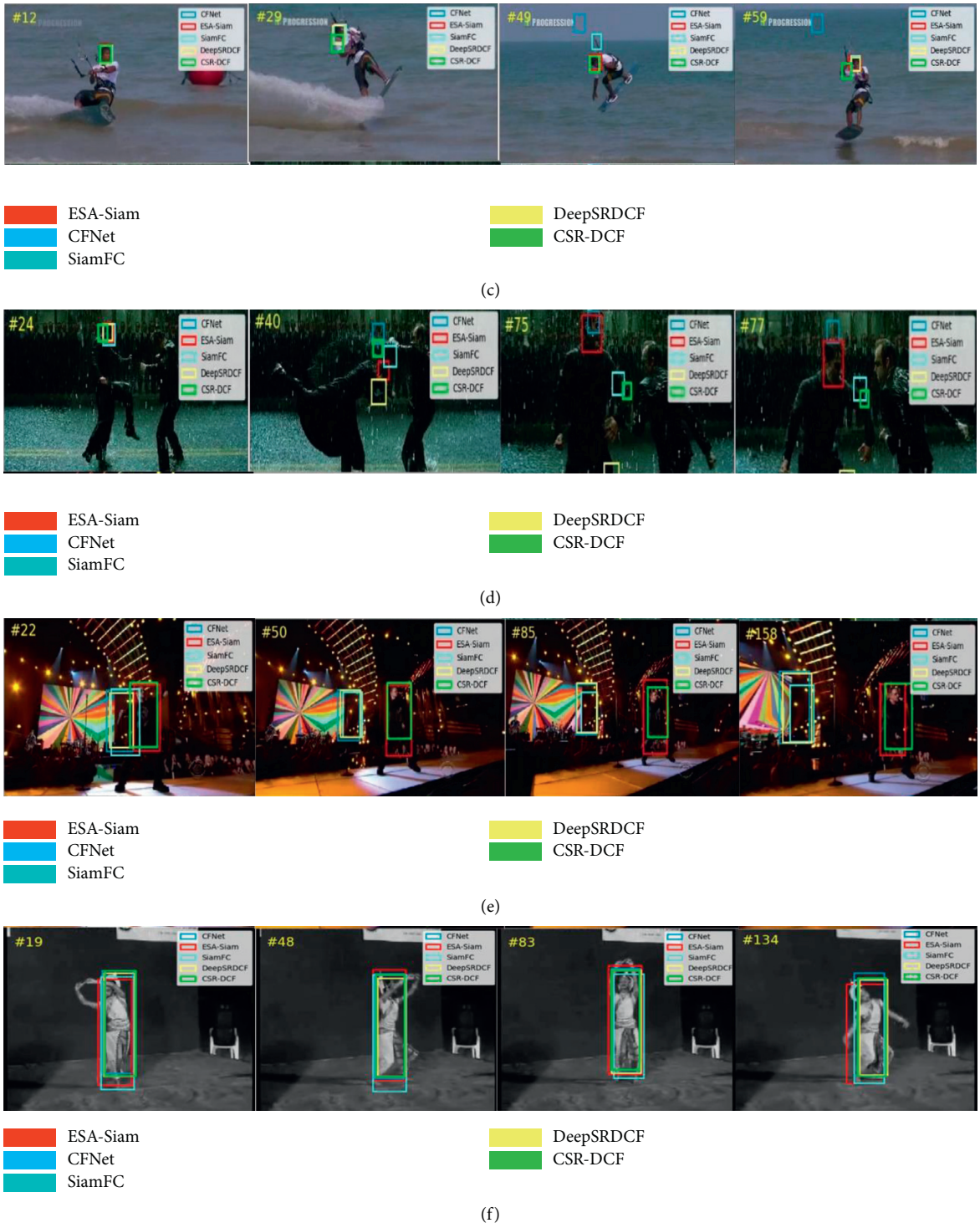


Figure 9: Qualitative comparison of tracking results of various algorithms on challenges, such as (a) Bird2, (b) Human9, (c) KiteSurf, (d) Matrix, (e) Singer2, and (f) Dancer2.

shown in Table 5, simply adding single channel attention and spatial attention to the benchmark SiamFC cannot effectively improve its tracking performance. Integrating channel attention and spatial attention can increase by 7.8% with EAO. If combined with the template search attention module, EAO can increase by 11.8%. In addition, gold stochastic pooling components can also be effectively applied to the twin network to improve the tracking performance. As shown in Table 6, we conducted a series of experiments to discuss the impact of different gold stochastic pooling thresholds on the tracking performance. When $T = 0$, it means that gold stochastic pooling degenerates to

Input: random initialization the network parameters θ, golden threshold stochastic pooling T, spatial self-attention parameters of α. Template Z and search patch X from GOT10K.
Preprocessing: crop and resize Z and X and set optimizer, loss function, and learning rate adjustment strategy.
While *epoch > 0 and input video dataset is not empty* **do**
Get template Z and corresponding bounding box;
Get search patch X and corresponding bounding box;
Compute $\psi(Z)$, $\psi(X)$ by the backbone network;
Compute $\mu(\psi(Z))$, $\mu(\psi(X))$ by the channel attention module;
Compute $\eta(\mu(\psi(Z)))$, $\eta(\mu(\psi(X)))$ by the spatial self-attention module;
Create sample positive and negative labels;
Compute $\Delta(Z, X)$ and update template;
Computer response map of Z nad X;
Computer loss and update parameters;
Optimize loss to minimize.
end

Algorithm 1: Offline training of the proposed framework.

Input: test video; initial frame and bounding box of initial frame;
Compute $\psi(Z)$ by the backbone network;
Compute $\mu(\psi(Z))$ by the channel attention module;
Compute $\eta(\mu(\psi(Z)))$ by the spatial self-attention module;
Preprocessing: crop and resize X and set three different scale patches X_1, X_2, X_3.
While *test video is not empty* **do**
Get search patch X and corresponding bounding box;
Compute $\psi(X)$ by the backbone network;
Compute $\mu(\psi(X))$ by the channel attention module;
Compute $\eta(\mu(\psi(X)))$ by the spatial self-attention module;
Upsampling feature map X to 272×272;
Locate target center in feature map X by finding peak;
Computer the offset of the upsampled map relative to the feature map;
Computer the offset of the feature map relative to original image;
Update target size and corresponding bounding box;
end

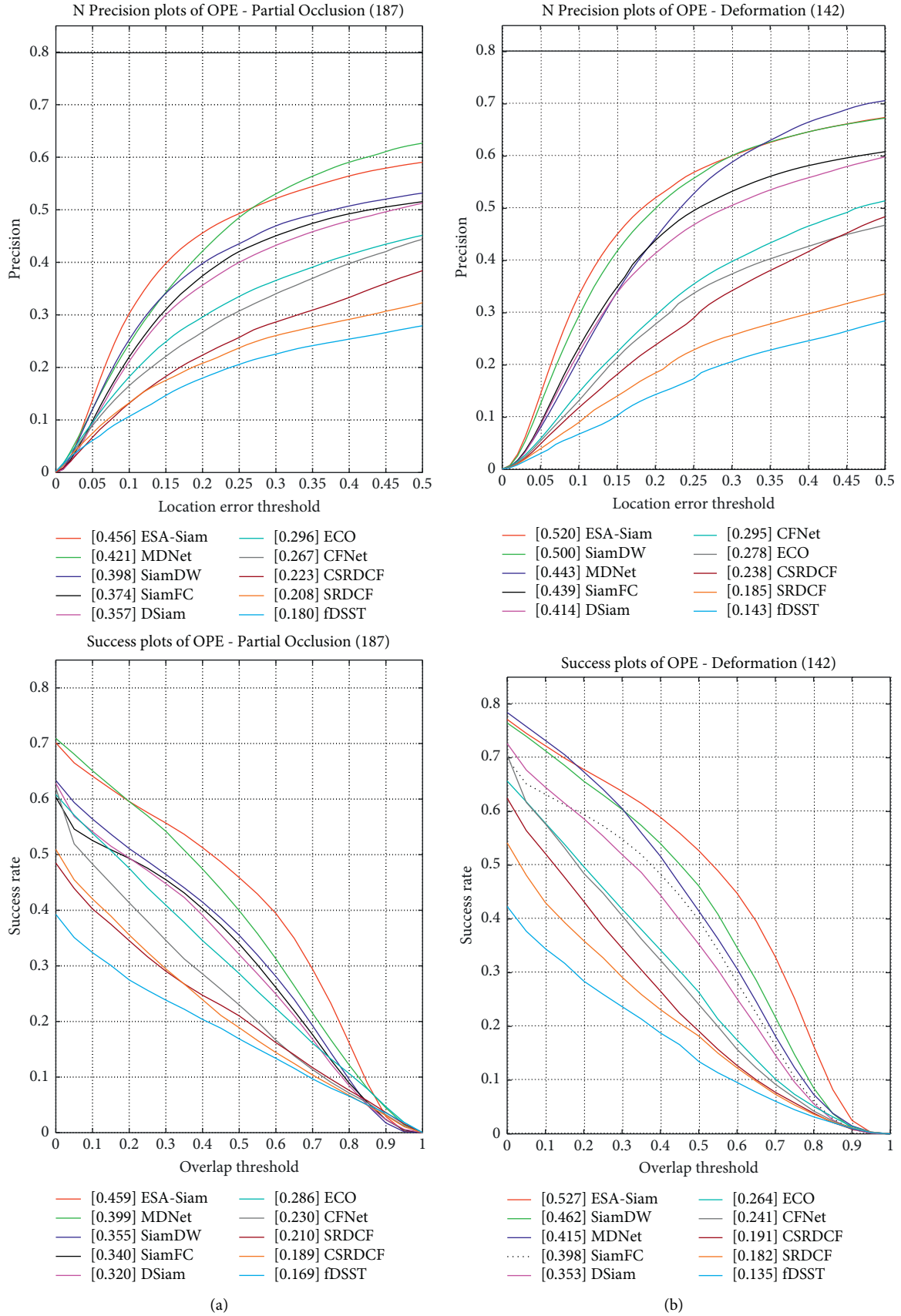
Algorithm 2: Inference of the proposed framework.

Table 2: Results on VOT2016 on expected average overlap (EAO), accuracy (A), and robustness (R).

Trackers	Accuracy ↑	Robustness ↓	EAO ↑
HCF	0.450	0.396	0.220
SAMF	0.503	0.443	0.226
SiamFC	0.532	0.461	0.235
SRDCF	0.535	0.419	0.247
MDNet	0.541	0.337	0.257
DeepSRDCF	0.528	0.326	0.276
Staple	0.544	0.378	0.295
C-COT	0.539	0.238	0.331
ESA-Siam	**0.622**	**0.231**	**0.353**

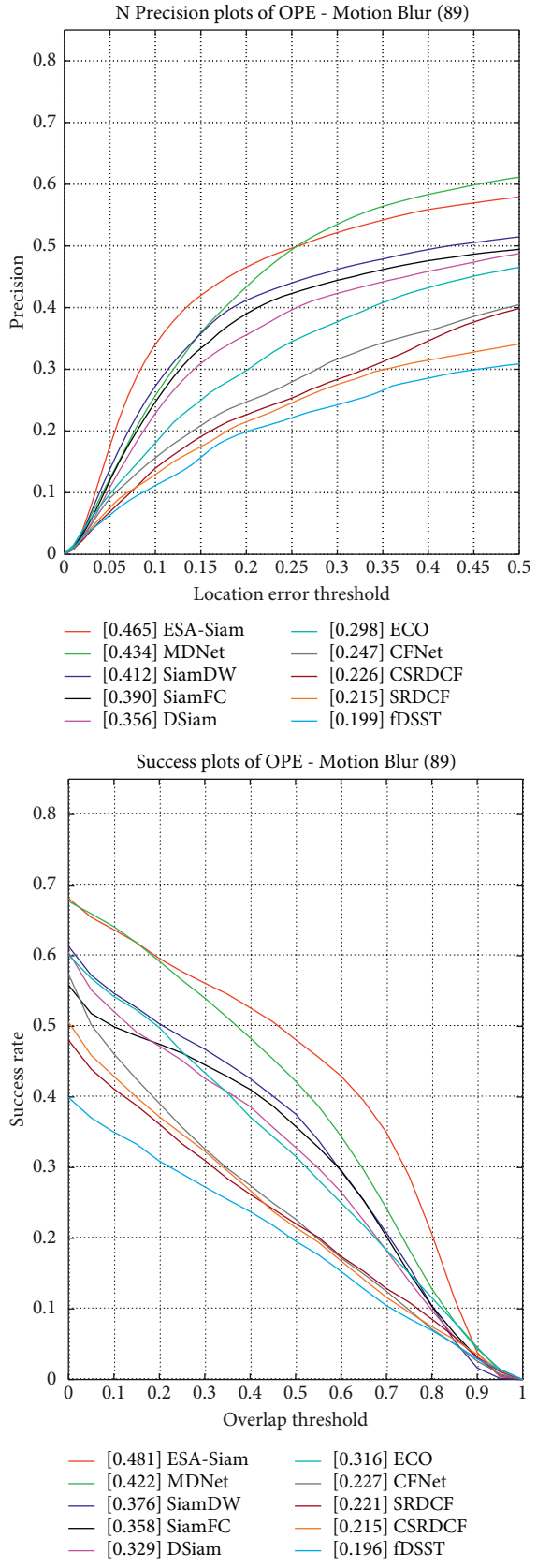
The values in bold highlights the algorithm with the first performance ranking, which can be seen intuitively.

normal stochastic pooling. When the value of t is greater, the probability of selecting the largest activation is higher. We observe that ESA-Siam can achieve the best performance when $T = 0.618P_{\max}$. Due to the mathematical peculiarity of 0.618 in stochastic pooling, we named it the golden stochastic pooling.



(a) (b)

Figure 10: Continued.



(c)

Figure 10: The evaluation results of the proposed method on complex challenge scenarios on the LaSOT dataset. (a) Partial occlusion. (b) Deformation. (c) Motion Blur.

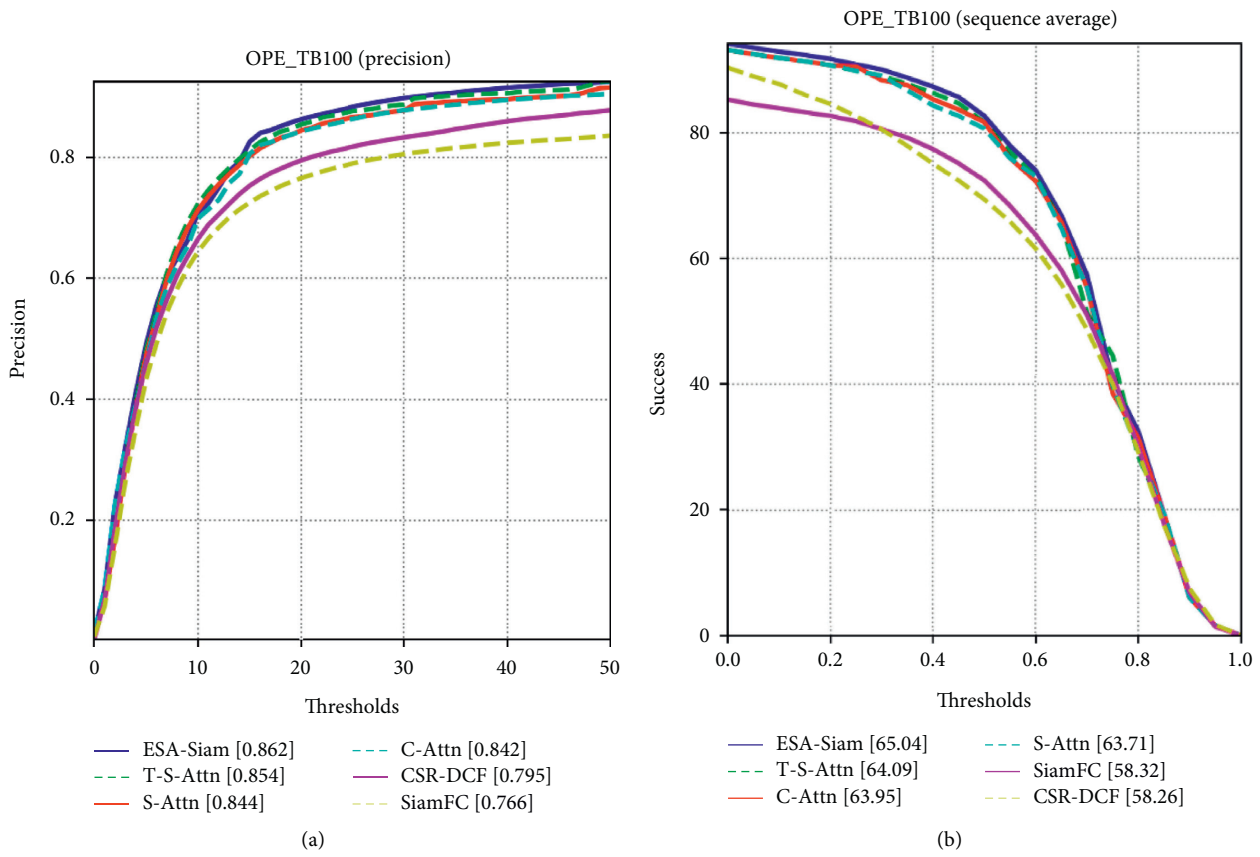


Figure 11: Precision and success plots of OPE on OTB100: (a) precision plots of OPE on OTB100; (b) success plots of OPE on OTB100. The performance of each component of ESA-Siam (C-Attn, S-Attn, and T-S-Attn) is better than that of the benchmark algorithms SiamFC and CSR-DCF.

Table 3: Results on VOT2018 on expected average overlap (EAO), accuracy (A), and robustness (R).

Trackers	Accuracy ↑	Robustness ↓	EAO ↑	Speed
Staple	0.530	0.688	0.169	13.4
SiamFC	0.503	0.585	0.188	84
DSiam	0.512	0.646	0.196	46.6
UpdateNet	0.518	0.454	0.244	10.5
CSR-DCF	0.491	0.356	0.256	12.7
C-COT	0.494	0.318	0.267	14.1
ECO	0.484	0.280	0.280	77.6
DeepSRDCF	0.489	0.293	0.293	20.5
SiamRPN	0.562	0.276	0.336	76.8
ESA-Siam	**0.618**	**0.223**	**0.411**	60

The values in bold highlights the algorithm with the first performance ranking, which can be seen intuitively.

Table 4: Results on LaSOT with success and normalized precision (Norm.Pr).

Trackers	Success	Norm.Pr
SiamFC	0.382	0.420
DSiam	0.362	0.405
CFNet	0.258	0.312
CSR-DCF	0.224	0.254
SiamDW	0.397	0.435
ECO	0.329	0.338
SRDCF	0.245	0.248
fDSST	0.196	0.208
MDNet	0.435	0.461
ESA-Siam	**0.501**	**0.495**

The values in bold highlights the algorithm with the first performance ranking, which can be seen intuitively.

Table 5: Ablation study on VOT2016 (base: SiamFC; GSP: gold stochastic pooling; CA: channel attention; SA: spatial self-attention; T-SA: template search collaboration attention).

Method	$A\uparrow$	$R\downarrow$	EAO↑	ΔEAO (%)
Base	0.532	0.461	0.235	—
Base + GSP	0.547	0.433	0.241	+0.6
Base + GSP + CA	0.557	0.367	0.250	+1.5
Base + GSP + SA	0.562	0.362	0.253	+1.8
Base + GSP + T-SA	0.577	0.291	0.270	+3.5
Base + GSP + CA + SA	0.606	0.273	0.313	+7.8
Base + GSP + CA + SA + T-SA	0.622	0.231	0.353	+11.8

Table 6: Performance of the proposed ESA-Siam on OTB100 dataset using different thresholds for stochastic pooling.

Threshold	Precision	Success
ESA-Siam + $T = 0.367$	0.772	58.38
ESA-Siam + $T = 0.433$	0.797	59.76
ESA-Siam + $T = 0.525$	0.827	62.45
ESA-Siam + $T = 0.618$	**0.862**	**65.04**
ESA-Siam + $T = 0.780$	0.834	63.22
ESA-Siam + $T = 0.822$	0.830	62.31

The values in bold highlights the algorithm with the first performance ranking, which can be seen intuitively.

5. Conclusion

We propose an enhanced visual attention Siamese network that can update template features online for visual tracking. We introduce a template search collaboration attention module that can implicitly update target features online and combine the channel attention and spatial self-attention modules in the computationally efficient ECA module.Based on the Siamese network, combining with the visual attention mechanism can ensure that the algorithm is simple and efficient. ESA-Siam can keep the tracking speed in real-time and make the algorithm more robust. The algorithm we proposed can be applied to scenes disturbed by background, such as video surveillance, vehicle tracking, and UAV tracking.

Data Availability

The data used to support the findings of this study are included within the article.

Disclosure

The funders had no role in the design of the study; in the collection, analyses, or interpretation of the data; in the writing of the manuscript; or in the decision to publish the results.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

ZWQ and ZG were responsible for conceptualization, original draft preparation, and review and editing. ZWQ, ZG, ZZG, and LQ validated the study. ZWQ and ZZG were responsible for formal analysis and funding acquisition. ZZG visualized the study. ZWQ was responsible for methodology, project administration, and resources. LQ curated the data and supervised the study. ZG was responsible for investigation. All authors have read and agreed to the published version of the manuscript.

Acknowledgments

This study was partially supported by the National Key Research and Development Program of China (grant nos. 2018AAA0100400 and 2019QY1604), National Natural Science Foundation of China (grant no. U1836217), Open Platform Innovation Foundation of Hunan Provincial Education Department (grant no. 20K046), Scientific Research Project of Hunan Provincial Department of Education (grant no. 17C0479), and Special Fund Support Project for the Construction of Innovative Provinces in Hunan (2019GK4009).

References

[1] M. Elhoseny, "Multi-object detection and tracking (MODT) machine learning model for real-time video surveillance systems," *Circuits, Systems, and Signal Processing*, vol. 39, no. 2, pp. 611–630, 2020.

[2] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deepdriving: learning affordance for direct perception in autonomous driving," in *Proceedings of the IEEE international conference on computer vision*, pp. 2722–2730, Santiago, Chile, December 2015.

[3] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11923–11932, Seattle, WA, USA, June 2020.

[4] M. Felsberg, A. Berg, G. Hager et al., "The thermal infrared visual object tracking VOT-TIR2015 challenge results," in *Proceedings of the ieee international conference on computer vision workshops*, pp. 76–88, Santiago, Chile, December 2015.

[5] E. Real, J. Shlens, S. Mazzocchi, X. Pan, and V. Vanhoucke, "Youtube-boundingboxes: a large high-precision human-annotated data set for object detection in video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5296–5305, Honolulu, HI, USA, July 2017.

[6] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2014.

[7] 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*, pp. 4310–4318, Santiago, Chile, December 2015.

[8] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1396–1404, Honolulu, HI, USA, July 2017.

[9] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proceedings of the British Machine Vision Conference*, Nottingham, England, September, 2014.

[10] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integratio," in *Proceedings of the European conference on computer vision*, pp. 254–265, Zurich, Switzerland, September 2014.

[11] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional siamese networks for object tracking," in *Proceedings of the European conference on computer vision*, pp. 850–865, Amsterdam, Netherlands, October 2016.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[13] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8971–8980, Salt Lake City, UT, USA, June 2018.

[14] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. Hilaire Torr, "Fast online object tracking and segmentation: a unifying approach," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1328–1338, Long Beach, CA, USA, June 2019.

[15] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xiang, and J. Yan, "Siamrpn++: evolution of siamese visual tracking with very deep networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4282–4291, Long Beach, CA, USA, June 2019.

[16] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, and W. Hu, "Distractor-aware siamese networks for visual object tracking," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 101–117, Munich, Germany, September 2018.

[17] Q. Liu, X. Li, Z. He, N. Fan, D. Yuan, and H. Wang, "Learning deep multi-level similarity for thermal infrared object tracking," *IEEE Transactions on Multimedia*, vol. 23, pp. 2114–2126, 2020.

[18] D. Yuan, X. Chang, P.-Y. Huang, Q. Liu, and Z. He, "Self-supervised deep correlation tracking," *IEEE Transactions on Image Processing*, vol. 30, pp. 976–985, 2020.

[19] D. Yuan, N. Fan, and Z. He, "Learning target-focusing convolutional regression model for visual object tracking," *Knowledge-Based Systems*, vol. 194, Article ID 105526, 2020.

[20] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," *ICML deep learning workshop*, vol. 2, 2015.

[21] R. R. Varior, M. Haloi, and G. Wang, "Gated siamese convolutional neural network architecture for human re-identification," in *Proceedings of the European Conference on Computer Vision*, pp. 791–808, Amsterdam, Netherlands, October 2016.

[22] J. Mueller and A. Thyagarajan, "Siamese recurrent architectures for learning sentence similarity," *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, no. 1, 2016.

[23] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, "Learning dynamic siamese network for visual object tracking," in *Proceedings of the IEEE international conference on computer vision*, pp. 1763–1771, Venice, Italy, October 2017.

[24] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," in *in Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017)*, pp. 5998–6008, Long Beach, CA, USA, 2017.

[25] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: convolutional block attention module, Computer Vision_ECCV 2018," in *Lecture Notes in Computer Science*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., vol. 11211, Springer, Cham, Switzerland, 2018.

[26] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," 2018, https://arxiv.org/abs/1803.02155.

[27] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4854–4863, Salt Lake City, UT, USA, June 2018.

[28] Z. Zhu, W. Wu, W. Zou, and J. Yan, "End-to-end flow correlation tracking with spatial-temporal attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 548–557, Salt Lake City, UT, USA, June 2018.

[29] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: a benchmark," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2411–2418, Porland, OR, USA, June 2013.

[30] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015.

[31] M. P Kristan, A. Lebeda, J. Mates et al., "The visual object tracking VOT2016 challenge results," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshop*, pp. 777–823, Amsterdam, Netherland, October 2016.

[32] M. Kristan, A. Leonardis, J. Matas et al., "The sixth visual object tracking vot2018 challenge results," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, p. 0, Munich, Germany, September 2018.

[33] H. Fan, L. Lin, F. Yang et al., "Lasot: a high-quality benchmark for large-scale single object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5374–5383, Long Beach, CA, USA, June 2019.

[34] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2544–2550, San Francisco, CA, USA, June 2010.

[35] M. Danelljan, A. Robinson, F. Shahbaz Khan, and M. Felsberg, "Beyond correlation filters: learning continuous convolution operators for visual tracking," in *Proceedings of the European conference on computer vision*, pp. 472–488, Amsterdam, Netherlands, October 2016.

[36] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2805–2813, Honolulu, HI, USA, July 2017.

[37] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4293–4302, Las Vegas, NV, USA, June 2016.

[38] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proceedings of the IEEE international conference on computer vision workshops*, pp. 58–66, Santiago, Chile, December 2015.

[39] M. Danelljan, G. Bhat, F. Shahbaz Khan, and M. Felsberg, "Eco: efficient convolution operators for tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6638–6646, Honolulu, HI, USA, July 2017.

[40] A. He, C. Luo, X. Tian, and W. Zeng, "A twofold siamese network for real-time object tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4834–4843, Salt Lake City, UT, USA, June 2018.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, Las Vegas, NV, USA, June 2016.

[42] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, Salt Lake City, UT, USA, June 2018.

[43] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7794–7803, Salt Lake City, UT, USA, June 2018.

[44] M. D. Zeiler and R. Fergus, "Stochastic pooling for regularization of deep convolutional neural networks," 2013, https://arxiv.org/abs/1301.3557.

[45] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proceedings of the European conference on computer vision*, pp. 483–499, Amsterdam, Netherlands, October 2016.

[46] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: efficient channel attention for deep convolutional neural networks," in *Proceedings of the 2020 IEEE. CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Seattle, WA, USA, June 2020.

[47] Z. Zhang and H. Peng, "Deeper and wider siamese networks for real-time visual tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4591–4600, Long Beach, CA, USA, June 2019.

[48] C. Ma, J.-B. Huang, X. Yang, and M. H. Yang, "Hierarchical convolutional features for visual tracking," in *Proceedings of the IEEE international conference on computer vision*, pp. 3074–3082, Santiago, Chile, December 2015.

[49] A. Lukezic, T. Vojir, L. Čehovin Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6309–6318, Honolulu, HI, USA, July 2017.

[50] P. Li, B. Chen, W. Ouyang, D. Wang, X. Yang, and H. Lu, "GradNet: Gradient-guided network for visual object tracking," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6162–6171, Seoul, South Korea, April 2019.

[51] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: complementary learners for real-time tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1401–1409, Las Vegas, NV, USA, June 2016.

[52] M. Danelljan, G. Häger, F. S. Khan, and S. Felsberg, "Discriminative scale space tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 8, pp. 1561–1575, 2016.

[53] L. Zhang, A. Gonzalez-Garcia, J. V. D. Weijer, M. Danelljan, and F. Shahbaz Khan, "Learning the model update for siamese trackers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4010–4019, Seoul, South Korea, April 2019.

[54] L. Huang, X. Zhao, and K. Huang, "Got-10k: a large high-diversity benchmark for generic object tracking in the wild," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 5, pp. 1562–1577, 2021.