This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3072778, IEEE Access



# Cooperative Use of Recurrent Neural Network and Siamese Region Proposal Network for Robust Visual Tracking

**XUECHEN ZHAO[1], YAOMING LIU[2], and GUANG HAN[2]**

[1]College of communication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing, 210003 China
[2]Engineering Research Center of Wideband Wireless Communication Technique, Ministry of Education, Nanjing University of Posts and Telecommunications, Nanjing, 210003 China

Corresponding author: GUANG HAN (e-mail: hanguang8848@163.com).

This work was supported by the Natural Science Foundation of China NSFC under Grants 61871445, 61302156; Key R & D Foundation Project of Jiangsu Province under Grant BE2016001-4.

**ABSTRACT** In order to solve the problems of unbalanced sample data and the lack of consideration of temporal information in existing Siamese-based trackers, this paper proposes a Siamese recurrent neural network and region proposal network (Siamese R-RPN), which can be trained in an end-to-end manner. Siamese R-RPN is consisted of Siamese network, recurrent neural network and region proposal network. Image features extracted by the Siamese network are strengthened by the channel and spatial attention mechanisms, and are sent to the RPN for classification and regression. Temporal information is processed by a recurrent neural network-based Long Short-Term Memory (LSTM) to predict the rough location of the target, it is mapped to the anchor feature map of the RPN for anchor selection. This makes the positive and negative samples participating in the training procedure to become more balanced and representative. Because of the collaborative use of temporal and spatial information, the tracker proposed in this paper has achieved state-of-the-art performance on three large tracking benchmarks—OTB 2015, VOT2016 and VOT 2018—where this verifies its effectiveness.

**INDEX TERMS** Object Tracking, Recurrent Neural Network LSTM, Siamese Network, Region Proposal Network, Attention Mechanism

## I. INTRODUCTION

Object tracking is widely used in such applications as video surveillance, intelligent transportation, autonomous driving and human-computer interaction [1]. Given the initial state of the target tracked in the first frame, object tracking can estimate the unknown state (such as position and scale) of the target in successive video frames. Although significant advancements in the area have been reported, no satisfactory method is available to cover all tracking scenarios due to the variety of the relevant scenes, changeable environment, and such complex conditions as deformation, occlusion, blurring and rapid movement caused by the motion of objects [2-4]. Object tracking is thus still considered as a challenging task.

Convolutional neural networks have been successfully applied to object detection and recognition because of their powerful feature representation capabilities [5], and this has inspired the introduction of deep learning to solve the challenges posed by object tracking [6-8]. Although this helps to improve the accuracy of tracking, deep learning-based target tracking algorithms are computationally expensive when extracting deep features or fine-tuning the network online. This makes it difficult for them to meet real-time requirements.

To satisfy real-time requirements, Siamese network [10], which refers to correlation filtering [9], has attracted considerable research attention. Siamese network is a special neural network architecture consisting of two or more weight-sharing sub-networks. Its core is mainly to convert target tracking into a matching problem, and learn a general similarity function offline from a large number of videos. A one-stage tracker Siamese-RPN [11] based on the Siamese network was recently proposed. It avoids the time-consuming pyramid needed to estimate the scale of the target by adding a region proposal network after the Siamese network [12] for enhancing tracking performance.

Although the above technique can yield satisfactory performance, Siamese-RPN often misjudges the target in case of interference from another target similar to the actual

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3072778, IEEE Access



one during the tracking process. This causes the tracking bounding box to easily drift to the interfering target or the background of the image. This may occur for three reasons: 1) The features extracted by the Siamese network are not fully utilized. Siamese-RPN or other Siamese-based trackers mostly use ResNet as feature extraction network. They use features extracted from the last layer of ResNet to distinguish the foreground (target) from the background. In the case that similar targets interfere with a given target, two targets may belong to the same type of object and have similar high-level semantic features. This makes it challenging to distinguish them using only features of the last layer of ResNet, even though it contains more semantic information than the other layers. 2) The number of positive and negative samples is unbalanced during training. The target of tracking occupies only a small part of the image, whereas the region proposal network (RPN) generates the region proposal as positive and negative samples in the entire image through the anchor mechanism. As a result, the number of positive samples is much smaller than that of negative samples. It then becomes challenging to fully train the Siamese network using an unbalanced number of samples. The negative samples obtained are mostly easy negative samples, which contain no or little interference and similar semantic information. In case of interference by similar semantic information to that sought, the performance of the tracker is significantly degraded [13]. 3) In the Siamese-RPN, feature extraction is performed based on the idea of target detection by transforming multi-frame target tracking tasks into single-frame target detection tasks. However, there are important differences between target tracking and target detection. Target detection focuses on identifying different classes of targets. The spatial features of a single image are fully mined and the relationship between multiple images is forgotten. In addition to the need to distinguish different categories of objects, target tracking needs to handle interference by objects belonging to the same class as the target. Moreover, target tracking is a temporal task where multiple images are linked. If only spatial features of a single image are considered, the trained tracking model is rendered limited.

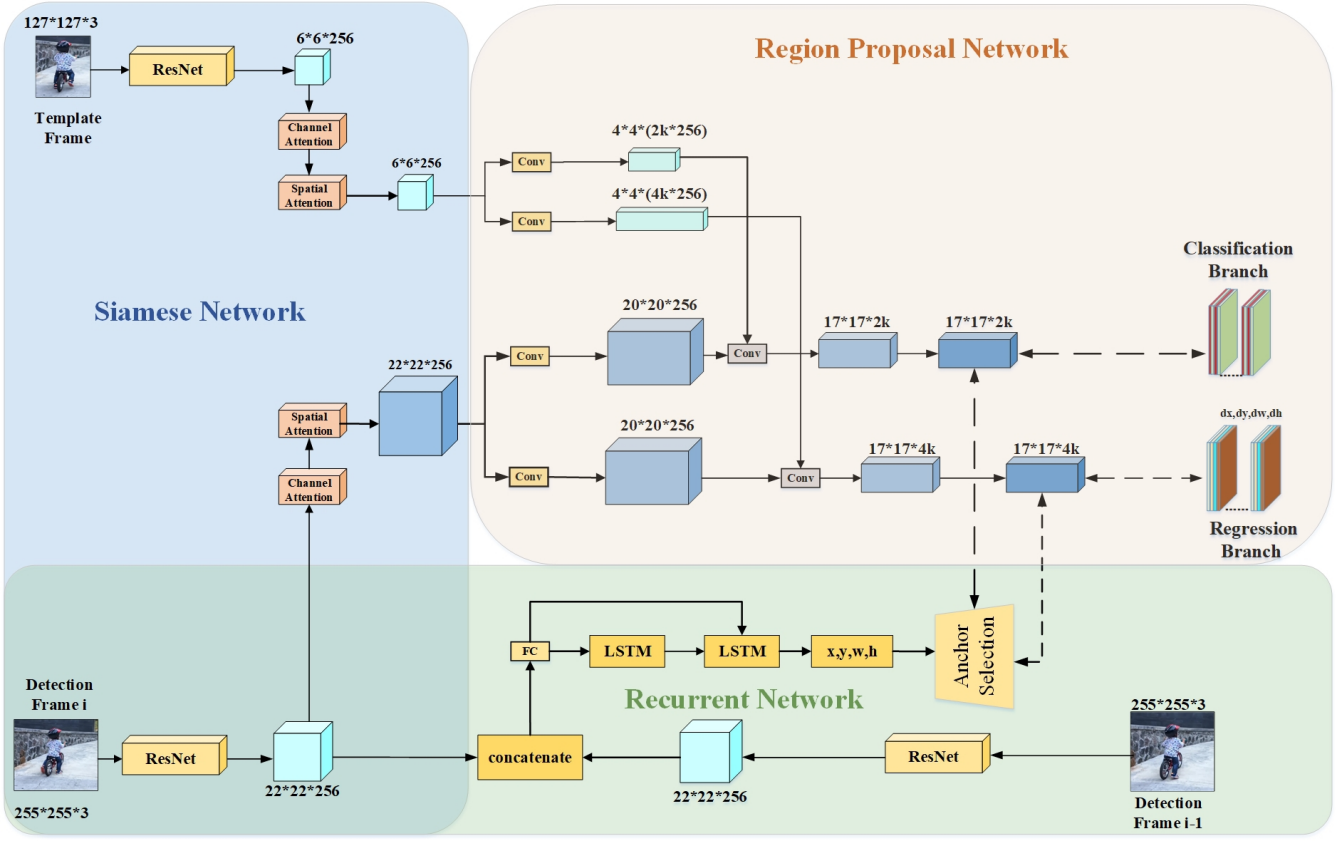



**Fig. 1. Architecture of Siamese R-RPN. The image features extracted by the Siamese network are input to the region proposal network (RPN) after being strengthened by the channel attention and spatial attention mechanisms. The outputs of the classification and regression branches are obtained through the correlation convolution operation. A two-layer LSTM is used in the recurrent neural network to predict the rough position of the target, then the rough location is mapped to the anchor feature map of the RPN for anchor selection, which is carried out to remove the invalid anchor and realize transition of position from coarse to fine. The dotted line indicates that the selected anchor points are mapped to the RPN, which reduces the classification and regression calculation of anchor box, and makes the accurate location of the subsequent target better. The classification branch outputs 2k channels, representing the classification probability of the foreground and the background in k anchor boxes. The regression branch outputs 4k channels, representing the four fine-tuned coordinates corresponding to the k anchor boxes.**

## II. RELATED WORK

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3072778, IEEE Access



As shown in Fig. 1, this section first introduces the Siamese-based tracker using a neural network, then discusses the effect of the anchor-based on detection, and finally describes the results of research on the attention mechanism in computer vision.

### A. Siamese-based tracker

In recent years, Siamese-based object tracker has attracted considerable research interest because of its excellent accuracy and real-time performance. The Siamese network usually contains two or more network branches with the same parameters, and determines the position of the target of tracking by mapping features extracted from paired images to the feature space for comparison. The network was first used in GOTURN [17] for target tracking, where it can be regarded as a deep regression method with a processing speed of up to 100 fps. The SINT algorithm proposed by Tao [18] transforms the target tracking problem into a matching problem, it is implemented through a neural network. However, a large number of candidate boxes are generated, and it need to be processed in each frame, which is time-consuming. In the same year, SiamFC [12] was proposed by Bertinetto. It is more practical than SINT. A Siamese convolutional neural network is used in it as feature extractor, the similarity between patches in the form of a sliding window is calculated through a simple correlation operation. Finally, the patch with the highest score is defined as the new location of the target. To locate the target in the search image, all possible locations can be exhaustively tested, the patch most similar in appearance to the target is selected. A balance between accuracy and speed is thus struck in SiamFC, it requires no model update. For work on the shortcomings of SiamFC without scale estimation, the interested reader can refer to the concept of Faster R-CNN [5]. The RPN (region proposal network) was first introduced to SiamFC by defining tracking as a local detection task [19] and replacing multi-scale detection with bounding box regression to obtain the maximum response bounding box. This yielded a high accuracy for fast single target tracking. The extractor-aware module introduced in DASiamRPN [20] was trained with high-quality sample pairs based on SiamRPN, the local-to-global search strategy was used when tracking failed. This enables the algorithm to be extended to deal with long-term tracking problems. To obtain a more accurate position of the target, SiamMask [21] used image segmentation to replace the rectangular target bounding box. In addition, Zhang [22] and Martin [23] updated the model to improve its performance. Besides, Zhong [57] proposed a hierarchical tracker that learns to move and track based on the combination of data-driven search at the coarse level, and coarse-to-fine verification at the fine level.

Since SiamFC was introduced, many improvements on it [20-26] had been proposed, but most of them were based on shallow networks such as AlexNet [27]. No study to date has examined improving the performance of the object tracker by enhancing the backbone network. Due the effect of the padding layer, the performance of the Siamese network tracker does not improve, but degrades when a deeper network is used. SiamRPN++ [28] was a recent a solution to this problem. Through a simple and effective spatial sensing sampling strategy, uniform sampling is performed in the range of 16 to 64 pixels from the center of the offset. This enables deep networks (ResNet [29]) to track. Multi-layer aggregation was also employed to further use the deeper features. The residual unit proposed in SiamDW [30] made for a deeper and wider network architecture for the Siamese tracker. Experimental results had shown that if the model is properly trained, the performance of the tracker can be substantially improved when using a deeper network. SiamBAN [58] and SiamCAR [59] used the anchor-free strategy to avoid complex anchor boxes settings and achieves good tracking performance

### B. Anchor-based Detection

Since the application of deep learning technology to target detection, many classic tools have been developed in the area, of which the two-stage R-CNN [31] and single-stage SSD [32] are representative. Such methods as Fast R-CNN, R-FCN [33] and RetinaNet [34] have been derived from them. These are all anchor-based methods. That is, they involve setting fixed anchors of different sizes and aspect ratios on the feature map, which contains all targets in the image, to carry out the subsequent classification and regression operations. However, the recent literature [35, 36] has highlighted certain disadvantages of anchors. First, a large imbalance between positive and negative samples occurs, speed of training decreases because of the excessive number of anchors. Second, the super-parameters of anchor size and aspect ratio need to be set manually, and they are not universal. Owing to the indeterminate nature of changes in the location and scale of the target, a target tracker trained with a fixed anchor often incurs a large calculation cost. In case of different targets of tracking, automatically adjusting the number and hyperparameters of anchors in a targeted manner for superior performance and faster processing speed has emerged as a direction of research.

### C. Attention Mechanism

The attention mechanism is originally designed for machine translation. It is now an important concept in neural networks. In 2015, Bahdanau et al. [37] proposed the attention mechanism to assign different weights to different parts of the input for the purpose of differentiation. Xu et al. [38] subsequently used the attention mechanism in computer vision, it was also used for image captioning. The residual attention network was proposed by Wang et al. [39] in 2017, it is based on image classification. It involved adding an attention mechanism to the residual network. Additional attention models can be extracted from feature maps of different depths by means of the residual connection, which improves classification accuracy while significantly reducing the amount of calculation required. Hu won the image classification task in the ImageNet Competition with SENet [40] in 2017. The core idea of SENet was to learn feature

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3072778, IEEE Access



weights according to network loss, so as to increase the weight of effective feature maps and reduce the impact of invalid maps on the results. Experiments had shown that feature map representation using the neural network can be enhanced by appropriately designing the attention mechanism. The performance of the network model can be significantly improved while incurring only a small increase in calculation

Inspired by the technologies of anchor-based detection and the attention mechanism, this paper proposes a framework called the Siamese R-RPN. As shown in Fig. 1, it is based on the Siamese-RPN with certain improvements. Spatial and channel attention mechanisms are introduced to process the spatial features extracted by the convolutional neural network to obtain comprehensive features of the image. This enhances the ability of the tracking model to distinguish between targets of the same class. Moreover, an improved recurrent neural network LSTM [16] is used to learn the sequence of video frames to obtain the temporal and motion-related information of the target of tracking, where this can help fine-tune the position of the target of tracking. Finally, the end-to-end network is trained as a whole to integrate the spatial – temporal information to accurately determine the position of the target.

In terms of object tracking, most researchers have focused on ways to optimize the network structure and extract apparent features of the target. This paper proposes a Siamese recurrent neural network and region proposal network (Siamese R-RPN) that uses an improved recurrent neural network LSTM to process the temporal information of the target, which can achieve SOTA-level performance. The main contributions of this article are as follows:

1) A recurrent neural network is introduced to the Siamese-RPN. With this combination, the temporal and spatial information of the target is fused to improve the accuracy of the tracking model.

2) An anchor selection module is designed to improve the balance and representativeness of the training samples. The tracking model is fully trained by selecting anchors from coarse to fine.

3) The spatial and channel attention mechanisms are added to the Siamese network structure to fully mine the apparent characteristics of the target image, enhance the target's ability to resist intra-class interference and improve the robustness of the tracking model.

4) A joint training loss function is proposed to form an end-to-end network model to jointly train the Siamese network and the recurrent neural network LSTM, all the while it ensures that the tracking model satisfies the requirements of the real-time performance.

5) A numbers of test experiments are conducted on multiple benchmarks, and the results show that the proposed tracker delivers excellent performance.

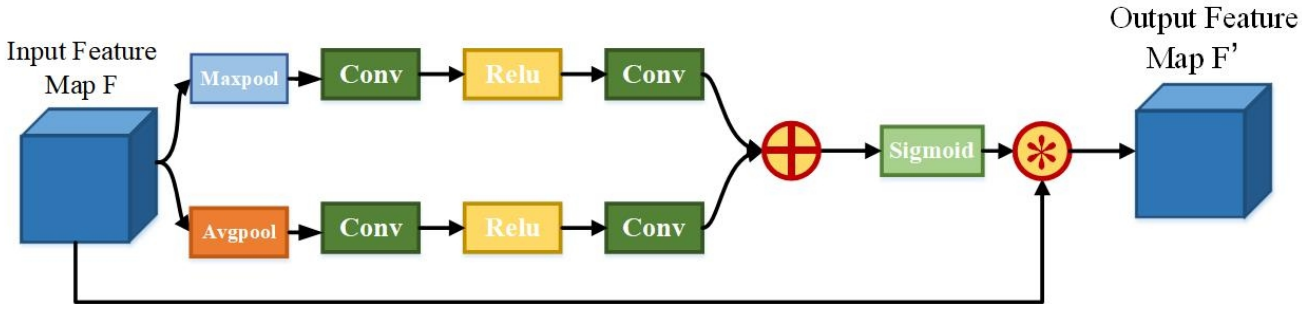



**Fig. 2.** **Framework of the channel attention mechanism, where * is the element point multiplication operation acting on the feature map at the channel level.**

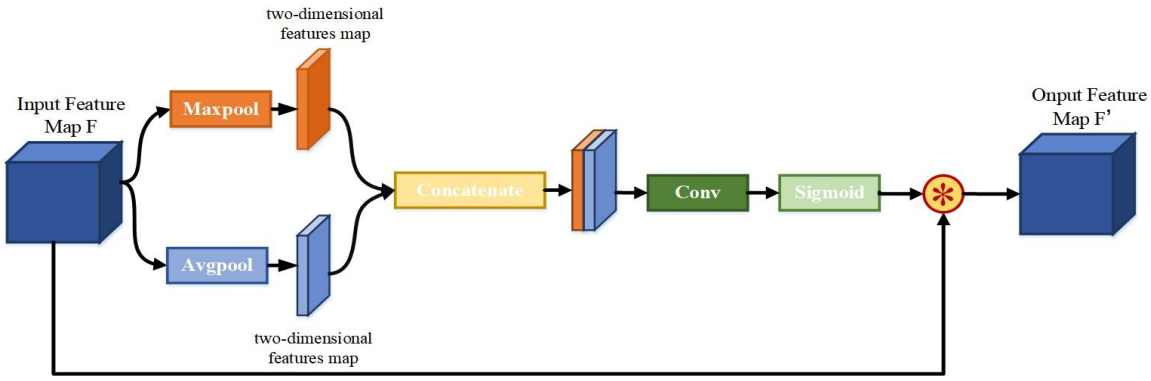



**Fig. 3.** **Framework of the spatial attention mechanism, where * is the element point multiplication operation acting on the feature map at the spatial level.**

## III. TRACKING ALGORITHM

Fig. 1 shows a diagram of the structure of the proposed Siamese R-RPN. It includes a Siamese network, a recurrent

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3072778, IEEE Access



neural network and a region proposal network. High-level feature information extracted from only the last layer of the neural network is used to distinguish the foreground from the background of the image in the traditional Siamese network. However, in the case of interference from similar targets, this highly abstract semantic information and the lack of fine-grained information that is conducive to detection make it difficult to locate the target for the tracker. In response to this problem, the channel and spatial attention mechanisms are added to the Siamese network to improve its identification capability, and it can enable the tracker to resist intra-class interference. In addition, the location of the tracking target is often uncertain. A target bounding box can be obtained by traversing the full image when a fixed anchor is used. However, a large number of candidate anchor boxes do not contain the target of tracking in operation, it has no effect on its final estimated position. This is a waste of computing resources, and affects the processing speed of the model. To solve this problem, an anchor selection module is designed by combining an improved recurrent neural network with the RPN to select the anchor from coarse to fine. While yielding the representative anchor box, this method significantly reduces the number of candidate anchor boxes. Finally, to improve the training speed of the network model, the recurrent neural network and RPN are jointly trained to realize an end-to-end network model.

### A. Channel and Spatial Attention Mechanisms

With the successful application of convolutional neural networks in a number of fields, a large number of network structures have been proposed, such as AlexNet, ResNet and Inception [41]. Experimental results have shown that the introduction of the attention mechanism to a neural network structure [42] can improve the model's ability to represent features.

The principle of the attention mechanism can be summarized by (1):

$$F' = \tau \bullet F \tag{1}$$

$F$ and $F'$ represent the input original feature map and the feature map enhanced by the attention mechanism, respectively, $\tau$ represents weights of the attention mechanism module after training. These weights correspond to values of the original feature maps one by one. In order to obtain faster training speed and better comprehensive performance improvement, we design a concise structured channel attention mechanism and spatial attention mechanism, as shown in Fig. 2 and Fig. 3

*1) Channel Attention Mechanism*

At the channel level, features of different channels have different feedback-related effects on the target. The output features of the channel convolution layers of different targets also have different weights, these dynamic values are related to the task target being executed. It is thus necessary to obtain the weight by making the most of the channel attention mechanism. As shown in Fig. 2, the input feature map is first compressed in the spatial dimension to gain two one-dimensional feature vectors. Considering the value of each pixel and the maximum value of local pixels in the feature map, the average pooling and maximum pooling operations are applied to each channel feature map to gain 1D feature vectors. These two feature vectors are then sent to the convolutional neural network. Following the application of the sigmoid function, the element addition is performed to output the weight. Finally, the feature map of the channel is processed using this weight:

$$\begin{aligned} F' &= \tau \bullet F \\ \tau &= \sigma\{\mathrm{conv}_2[\mathrm{conv}_1(\mathrm{Avg}(F), w_1), w_2] \\ &\quad + \mathrm{conv}_2[\mathrm{conv}_1(\mathrm{Max}(F), w_1), w_2)]\} \end{aligned} \tag{2}$$

$\mathrm{Avg}$ and $\mathrm{Max}$ represent the average pooling operation and the maximum pooling operation, respectively, $\sigma$ represents the sigmoid function, $w_1$ and $w_2$ represent the respective network weights of $\mathrm{conv}_1$ and $\mathrm{conv}_2$.

*2) Spatial Attention Mechanism*

Within the same feature map, different locations respond to targets differently. The difference between discrete pixels is enlarged when the feature map is spatially enhanced, so that pixel values sensitive to the target area increase and those irrelevant to it decrease. As shown in Fig. 3, a spatial attention mechanism is introduced to enhance the feature representation of the location of a specific area. The first step is to compress the feature map at the channel level. Two 2D feature maps are obtained by the average pooling and maximum pooling operations to process the input features in the dimensions of the channel. After the two feature maps are concatenated, the result is sent to the convolutional neural network, and the weight is output through the sigmoid function. Finally, the spatial feature map is fine-tuned according to the obtained weight:

$$\begin{aligned} F' &= \tau \bullet F \\ \tau &= \sigma\{\mathrm{conv}[\mathrm{cat}(\mathrm{Avg}(F); \mathrm{Max}(F)), w]\} \end{aligned} \tag{3}$$

$\mathrm{Avg}$ and $\mathrm{Max}$ represent the average pooling operation and the maximum pooling operation, respectively, $\sigma$ represents the sigmoid function, $\mathrm{cat}$ represents the concatenation operation, $w$ represents the network weight of $\mathrm{conv}$.

### B. Anchor Selection

Siam-RPN achieves good results mainly due to the design and use of the RPN module. As shown in Fig. 1, the RPN has a classification branch and a regression branch. The classification branch is used to identify the candidate area as foreground or background while the regression branch is used to regress the exact coordinates of the candidate area.

The core of RPN module is to generate anchor boxes by mapping the anchor feature map (each pixel in the feature map is an anchor) to the original image. According to the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3072778, IEEE Access



fixed area scaling factor and aspect ratio, each anchor corresponds to multiple rectangular boxes of different sizes. The classification branch outputs 17 × 17 × 2k, where the channel number "2" represents the classification of the foreground and the background; the regression branch outputs 17 × 17 × 4k, where the channel number "4" represents the corresponding offset positional size of the anchor box on the original image. The number of resulting anchor boxes is 17 × 17 × k.

During training, the obtained anchor boxes are selected to obtain more representative positive and negative samples [43, 44], especially negative samples (hard samples) that are difficult to distinguish. The IOU threshold is used in the traditional method to select the anchor boxes according to values of the overlapping area of each anchor box and the ground truth box. However, it is difficult to find a representative high-value sample for the anchor box by screening in this way, especially for negative samples, it leads to considerable redundancy. To solve this problem, an improved recurrent neural network LSTM is introduced to design the anchor selection module.

The core idea of anchor selection is that the improved recurrent neural network LSTM is used to process temporal information to predict the state information of the target in the next frame, including position-related and scale-related information. The confidence (the value is zero or one) of the anchor is calculated through state information. According to this value, the anchor is preliminarily selected. Zero is eliminated and one is reserved.

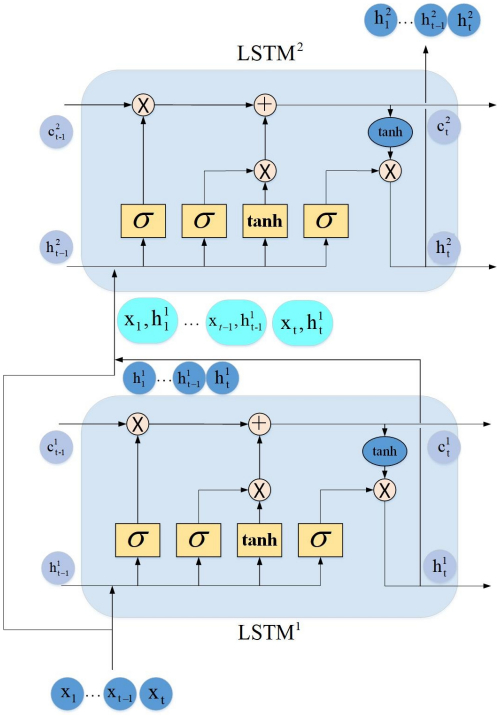



**Fig. 4. Structural diagram of two-layer LSTM.**

According to the temporal characteristics of the object tracking task, an improved recurrent neural network is used to extract the temporal information of the video sequences to predict the state of the tracking target. As shown in Fig. 4, a two-layer LSTM is used, where each layer has 1,024 units. The input $x_t$ and output $h_t^1$ of the first layer of the LSTM are subjected to a concatenation operation to obtain $[x_t, h_t^1]$ as the input to its second layer LSTM. Compared with the single-layer LSTM, two-layer LSTM [45] can capture more complex information on the motion of the target [46]. The equation below expresses how LSTM extracts, saves and outputs temporal information through the gate operation for each frame of the image. t represents the $t$-th frame of the image, $x_t$ and $h_{t-1}$ represent the input to the LSTM and the output of its previous frame, $W$ , $R$ and $P$ represent the corresponding weight matrices, b represents the bias term, $\sigma$ represents the sigmoid function, $\otimes$ represents the element point multiplication operation. Forward propagation output is used to return the current coordinates $h_t$ and save important memory-related information $c_t$.

$$
\begin{aligned}
z_t &= \tanh(W_z x_t + R_z h_{t-1} + b_z) \\
i_t &= \sigma(W_i x_t + R_i h_{t-1} + P_i c_{t-1} + b_i) \\
f_t &= \sigma(W_f x_t + R_f h_{t-1} + P_f c_{t-1} + b_f) \\
c_t &= i_t \otimes z_t + f_t \otimes c_{t-1} \\
o_t &= \sigma(W_o x^t + R_o h_{t-1} + P_o c_t + b_o) \\
h_t &= o_t \otimes \tanh(c_t)
\end{aligned}
\tag{4}
$$

The output of the second layer of the LSTM is sent to a fully connected layer to output the predicted position of the target $[x, y, w, h]$. This position is mapped to the feature map to obtain the coordinates $[x', y', w', h']$. This is regarded as the boundary of anchor selection, and all anchors beyond this boundary are discarded while those within it are retained. This helps realize anchor positioning from coarse to fine and improve the representativeness of the anchor box. The IOU threshold principle is used to further filter the fine anchors. The anchor box with IOU> 0.7 is defined as a positive sample and that with IOU <0.3 as a negative sample. Thirty of the positive and negative samples are randomly selected for training.

The above process can be expressed by (5):

$$
\begin{aligned}
p_s &= \mathrm{Fc}(h_t) \\
C_i &= \begin{cases} 0 & A_{ci} \not\subset p_s \\ 1 & A_{ci} \subset p_s \end{cases} \\
A_f &= \sum_i A_{ci} * C_i \\
A_{train} &= \sum_{i=1}^{30} A_{f-p}(\mathrm{IOU} > 0.7) + \sum_{i=1}^{30} A_{f-n}(\mathrm{IOU} < 0.3)
\end{aligned}
\tag{5}
$$

$h_t$ represents the output of the recurrent neural network at time $t$ and $p_s$ represents the predicted state obtained

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3072778, IEEE Access



after going through a fully connected layer, including $[x, y, w, h]$ . $A_{ci}$ represents the *i*-th coarse anchor, $C_i$ corresponds to the confidence of $A_{ci}$ , $A_f$ is the fine anchor obtained from coarse to fine, $*$ represents the element multiplication operation, $A_{f-p}$ and $A_{f-n}$ are the positive and negative samples corresponding to the fine anchor, respectively, $A_{train}$ is the positive and negative sample pair which is finally sent to the model for training.

### C. End-to-end network training

As shown in Fig. 1, the training of the proposed Siam R-RPN involves two parts: one is the recurrent neural network LSTM, the other is the classification and regression branch of RPN. The following details the end-to-end training of the network through the joint loss function.

*1) RPN network loss function*

The loss function in Faster R-CNN is used to train the RPN network in this paper. Softmax loss is used to supervise the classification branch. For the regression branch, when the position and size of the input anchor box is similar to the ground truth box, the transformation between them can be regarded as linear, the anchor box can be fine-tuned using linear regression to achieve the precise position of the target. A smooth L1 loss function and normalized distance are applied to train the regression branch. $\mathrm{A_X, A_y, A_W, A_h}$ represent the coordinates of the center, length and width of the anchor box, $\mathrm{T_x, T_y, T_w, T_h}$ represent the center coordinates, length and width of the target bounding box in the dataset, respectively. Then, the normalized distance is expressed as (6).

$$\delta(0)=\frac{\mathrm{T_x}-\mathrm{A_x}}{\mathrm{A_w}}, \delta(1)=\frac{\mathrm{T_y}-\mathrm{A_y}}{\mathrm{A_h}}, \delta(2)=\ln\frac{\mathrm{T_w}}{\mathrm{A_w}}, \delta(3)=\ln\frac{\mathrm{T_h}}{\mathrm{A_h}} \tag{6}$$

Smooth L1 loss is expressed as:

$$\mathrm{smooth_{L1}}(\mathrm{x},\sigma)=\begin{cases}0.5\sigma^2\mathrm{x}^2, & |\mathrm{x}|<\dfrac{1}{\sigma^2}\\ |\mathrm{x}|-\dfrac{1}{2\sigma^2}, & |\mathrm{x}|\geq\dfrac{1}{\sigma^2}\end{cases} \tag{7}$$

Finally, the loss function of the RPN network is given by:

$$\mathrm{L_{RPN}}=\mathrm{L_{cls}}+\lambda\mathrm{L_{reg}} \tag{8}$$

where $\lambda$ is a hyperparameter used to balance the relationship between $L_{\mathrm{cls}}$ and $L_{reg}$ , $L_{cls}$ represents the cross-entropy loss function, $L_{reg}$ is as shown in (9):

$$\mathrm{L_{reg}}=\sum_{\mathrm{i}=0}^{3}\mathrm{smooth_{L1}}[\delta(\mathrm{i}),\sigma] \tag{9}$$

*2) Recurrent neural network LSTM loss function*

During training, the output of the target bounding box $G_{pred}=[G_x, G_y, G_w, G_h]$ of the recurrent neural network LSTM, and the output of the target bounding box $[\mathrm{T_x}, \mathrm{T_y}, \mathrm{T_w}, \mathrm{T_h}]$ in the corresponding dataset are directly fed into the smooth L1 loss function. The loss function of the recurrent neural network LSTM is shown in (10):

$$\mathrm{L_{LSTM}}=\sum_{\mathrm{i}=0}^{3}\mathrm{smooth_{L1}}[\mathrm{G(i)},\sigma] \tag{10}$$

Finally, the joint loss function of the Siamese R-RPN is shown in (11):

$$\begin{aligned}\mathrm{loss}&=\mathrm{L_{RPN}}+\mu\mathrm{L_{LSTM}}\\&=\mathrm{L_{cls}}+\lambda\mathrm{L_{reg}}+\mu\mathrm{L_{LSTM}}\\&=\mathrm{L_{cls}}+\lambda\sum_{\mathrm{i}=0}^{3}\mathrm{smooth_{L1}}[\delta(\mathrm{i}),\sigma]+\mu\sum_{\mathrm{i}=0}^{3}\mathrm{smooth_{L1}}[\mathrm{G(i)},\sigma]\end{aligned} \tag{11}$$

where $\mu$ is a hyperparameter that controls the relationship between the loss functions of the recurrent neural network and the loss functions of the RPN network.

## IV. Experimental results and analysis

### A. End-to-end network training

1) Backbone

The backbone network used in this article is ResNet. Inspired by SiamDW, ResNet is operated as follows: the network step size is changed from 16 pixels to 8 pixels, receptive field is expanded by using dilated convolution to maintain the padding unchanged. The impact of padding on network training accuracy is eliminated by cropping feature maps. Under ensuring the depth of the network, entire target area can still be captured by each anchor, avoiding strong center deviation to the target, and solving the problem of padding destroying absolute translation invariance in deep neural networks.

2) Training

The Siamese R-RPN is pre-trained on the ImageNet-1k classification task, and then on the GOT-10K dataset. GOT-10K is a target tracking dataset released by the Chinese Academy of Sciences [47]. It contains more than 10,000 videos divided into more than 560 categories. The bounding boxes of the objects are all manually labeled for a total of more than 1.5 million. During training, video slice processing is performed on video streams of the dataset, where the slice length is 20 frames. First frames of each video slice is used to crop an image of size 127 × 127 centered on the target as template frame. Meanwhile, the same video slice is traversed, and two consecutive images which are cropped of size 255×255 centered on the target are selected as the input of the detection frame and the recurrent neural network. Finally, such three images (template frame and two consecutive images) are combined as a training sample. In this way, a large number of combinations of training sample are generated to train the Siamese network.

SGD is used to train the network from end to end. We set the momentum of SGD to 0.9, the learning rate is dropped from 0.01 to 0.0001. We iteratively train it for 50 epochs, where each epoch can sample 20,000 combinations. The

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3072778, IEEE Access



proposed model is trained on two NVIDIA Titan 1080Ti GPUs.

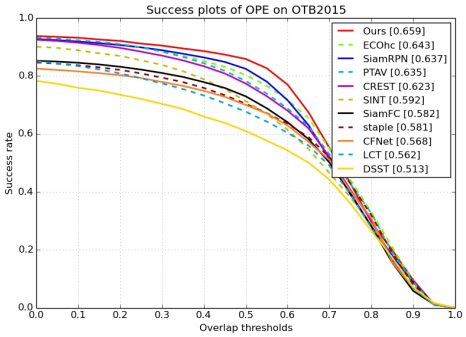
### B. Analysis of experimental results

The experiments are conducted on three challenging tracking benchmarks: OTB2015 [48], VOT2016 [49] and VOT2018 [14]. In the experiments, the run speed of Siamese R-RPN can reach 32 FPS.

*1) Experiments on OTB2015*

OTB2015 contains 100 tracking video sequences, providing a fair and standard test platform for tracking algorithms. Success plots and precision plots are used as evaluation indicators. The former indicate the ratio of successful frames with overlap being greater than the given threshold to all frames, when the threshold changes from zero to one. The latter represents the ratio of video frames where the distance between the center of the predicted target and the true center is smaller than the given threshold. In this experiment, our algorithm is tested, along with SiamRPN [11], SiamFC [12], CFNet [50], SINT [18], Staple [51], ECO-HC [52], CREST [53], PTAV [54], LCT [55] and DSST [56] for comparative verification and evaluation.

As showed in Fig. 5, the proposed method achieves satisfactory results in terms of both success plots and precision plots. Siam-RPN trained with pre-defined anchor parameters achieves 0.637 on the success plot and 0.850 on the precision plot, whereas our Siam R-RPN achieves 0.659 on the former and 0.871 on the latter. The recurrent neural network LSTM is used in Siam R-RPN for anchor selection. While reducing the number of anchors, it improves performance. Compared with Siam-RPN, its corresponding success and precision plots are higher by 2.2% and 2.1%, respectively, it proves that using temporal information to select anchors can improve the balance and representativeness of the positive and negative samples, which can fully train the tracking model such that it can accurately track the target.



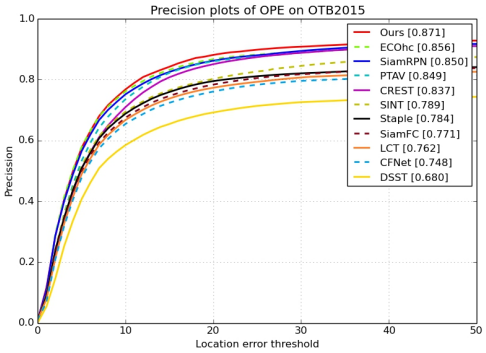




**Fig. 5. Success and precision plots on OTB2015.**

*2) Experiments on VOT2016*

VOT2016 contains 60 challenging tracking videos designed to evaluate the short-term tracking effect of algorithms. To balance accuracy and robustness, the expected average overlap (EAO) is used to evaluate the overall performance of the tracking algorithms, the normalized speed (EFO) is used to evaluate their operating efficiency.

TABLE I COMPARISON BETWEEN THE PROPOSED METHOD AND STATE-OF-THE-ART TRACKERS ON VOT2016 DATASET

| Tracker | EAO | Accuracy | Failure | EFO |
|---|---|---|---|---|
| Ours | **0.375** | **0.598** | **0.80** | 7.3 |
| C-RPN | 0.363 | 0.594 | 0.95 | 9.30 |
| SiamRPN | 0.344 | 0.560 | 1.08 | 23.30 |
| C-COT | 0.331 | 0.539 | 0.85 | 0.50 |
| TCNN | 0.325 | 0.554 | 0.96 | 1.10 |
| ECO-HC | 0.322 | 0.540 | 1.08 | 15.13 |
| SSAT | 0.321 | 0.577 | 1.04 | 0.50 |
| MLDF | 0.311 | 0.490 | 0.83 | 1.20 |
| Staple | 0.295 | 0.544 | 1.35 | 13.14 |
| EBT | 0.291 | 0.470 | 0.90 | 3.00 |
| STAPLEp | 0.286 | 0.557 | 1.32 | **44.80** |

Based on the four indicators of EAO, accuracy, failure and EFO, the comparative results are shown in Table I. Although EFO of our algorithm is not as high as that of Siam-RPN, its EAO and accuracy improve by 3.1% and 3.8% compared with Siam-RPN, respectively, and its failure index decreases by 0.28. Therefore, under the premise of satisfying the requirements of real-time performance, the overall performance of the tracking algorithm is improved.

*3) Experiments on VOT2018*

VOT2018 [14] contains 60 tracking videos with a total of 21,356 frames. During the tracking process, if the area of overlap between the tracking bounding box and the ground-truth bounding box is less than a given threshold, this is regarded as tracking failure, and the boxes are re-initialized to the correct position five frames after the failure. The main indicators for evaluating performance are EAO, accuracy and robustness.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3072778, IEEE Access



TABLE II COMPARISON BETWEEN THE PROPOSED METHOD AND STATE-OF-THE-ART TRACKERS ON VOT2018 DATASET

| Tracker | EAO | Accuracy | Robustness |
|---|---|---|---|
| Ours | **0.398** | **0.594** | 0.243 |
| LADCF | 0.389 | 0.510 | 0.159 |
| DaSiamRPN | 0.384 | 0.586 | 0.280 |
| MFT | 0.383 | 0.276 | **0.140** |
| RCO | 0.376 | 0.507 | 0.155 |
| SPM | 0.338 | 0.580 | 0.300 |
| ASRCF | 0.328 | 0.490 | 0.234 |
| ECO | 0.276 | 0.480 | 0.280 |

As shown in Table II, Siam R-RPN is tested and evaluated on the VOT2018, and the results are compared with seven SOTA trackers. Although the robustness index of the proposed method is not the best, its EAO and accuracy is improved. Compared with DaSiamRPN, they are improved by 1.4% and 0.8%, respectively. Robustness is declined by 3.7%. The results show that the overall performance of the proposed tracker is improved.

*4) Analysis of Ablation Experiments*

To verify the influence and effect of different parts of Siam R-RPN on the performance of the tracking algorithm, the ablation experiment on OTB2015 is analyzed based on the AUC (area under curve of success plots) and precision.

TABLE III ABLATION STUDY OF THE PROPOSED TRACKER ON OTB2015

| AlexNet | ResNet-50 | Attention Mechanism | LSTM | RPN | OTB2015 AUC | OTB2015 Precision |
|---|---|---|---|---|---|---|
| √ | | | | √ | 0.606 | 0.804 |
| | √ | | | √ | 0.638 | 0.852 |
| | √ | √ | | √ | 0.646 | 0.859 |
| | √ | | √ | √ | 0.653 | 0.866 |
| | √ | √ | √ | √ | **0.659** | **0.871** |
| √ | | √ | √ | √ | 0.622 | 0.821 |

*1. Backbone network analysis*

The performance of the tracker is closely related to the features extracted by the backbone neural network. The experimental results of the first and second rows in Table III show that the ResNet-50 is used for the baseline neural network, compared with the use of AlexNet, AUC and precision are increased by 3.2% and 4.8%, respectively. After the problem that the padding destroys the invariance of absolute translation in deep neural networks is solved, the features extracted by it help improve the accuracy and robustness of the proposed tracker compared with shallow neural networks.

*2. Effect of attention mechanism*

The second and third rows in Table III show that by adding the attention mechanism, the AUC and precision of the tracker are improved by 0.8% and 0.7%, respectively. As a result, features enhanced by the attention mechanism can help the RPN network better classify and return the position of the target of tracking.

*3. Effect of anchor selection module*

A recurrent neural network LSTM is added to the Siam-RPN with ResNet-50 as the backbone network for anchor selection. The second and fourth rows in Table III show that through the collaborative use of LSTM and convolutional neural networks, the AUC and precision of the tracker are increased by 1.5% and 1.4%, respectively. However, the AUC and precision when only the attention mechanism is used, are increased by only 0.8% and 0.7%, respectively. Anchor selection is implemented in Siam R-RPN using LSTM. It can improve the balance and representativeness of tracking model in terms of positive and negative samples, which enable it to accurately locate the target. Moreover, coarse to fine selection, which helps avoid the situation where the target frame drifts to the image background, the overall performance of the tracker is improved.

*4. Universal analysis*

To verify that the designed modules have an effect on different neural networks, ablation experiments are conducted on AlexNet and ResNet-50. The second and fifth rows in Table III show that once the attention mechanism and LSTM are added to the tracker based on ResNet-50, the AUC and precision are increased by 2.1% and 1.9%, respectively. The first and sixth rows show that once the attention mechanism and LSTM are added to the tracker based on AlexNet, the AUC and precision are increased by 1.6% and 1.7%, respectively. Thus, each proposed module can improve the performance of the tracker on different backbone neural networks.

*5) Analysis of Qualitative Experiments*

This section qualitatively compares the performance of the proposed method with Siam-RPN and Siam-FC on some examples of OTB2015, as is shown in Fig. 6. Compared with the other two trackers, Siam R-RPN uses the attention mechanism to strengthen the extracted features, thus it has a strong resistance to intra-class interference. The first, second and third lines of Fig. 6 show that object tracking tasks performed in complex scenes often encounter interference from similar targets. Siam-RPN and Siam-FC are able to detect objects appearing in the image, but they could not distinguish whether the detected object is the target of tracking, and often misjudge it. Siam R-RPN is able to accurately locate and track the target because of its fuller use of the features. Moreover, the results in the fourth, fifth and sixth rows show that the use of anchor selection helps prevent the tracking bounding box from drifting into the background, leading to strong anti-occlusion ability. In

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3072778, IEEE Access



addition, the results in the seventh row show that Siam R-RPN can accurately locate the tracking target, and provide the target bounding box in case of rotational deformation of the target.

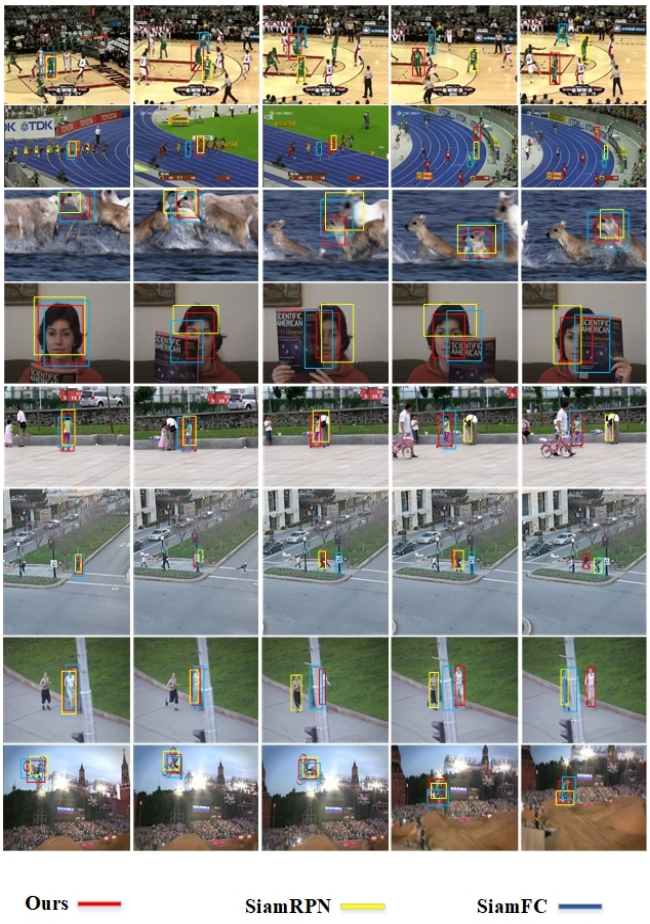



**Fig. 6. Qualitative results of the proposed method in comparison with other trackers on OTB2015.**

In order to better analyze the experimental results, visual analysis is performed on the correlation convolution response maps, which are represented by heat maps, as shown in Fig. 7. In the heat maps, the score is expressed in accordance with the color depth, the red color represents the highest score, which denotes the target location predicted by the tracking algorithm.

Through the frame-by-frame inspection of the heat maps, when similar target interference, similar background interference and target occlusion are encountered during the tracking process, the performance of Siam-RPN will be greatly reduced, and the correlation convolution response maps are difficult to accurately predict the location of the tracking target. Siamese R-RPN can solve these challenges well. By observing the heat map, we find that Siam-RPN obtains target candidate boxes from all anchor points, but the motion range of the tracking target is limited, and the actual number of effective anchor points is fixed, thus processing all anchor points has defects. On the one hand, a large number of calculations will slow down the speed; on the other hand, invalid anchor points will contain interference information, which will affect the accurate tracking results. However, Siamese R-RPN uses temporal information to predict rough location of the target and select anchor points to eliminate a large number of invalid anchor points, this can solve these challenges and achieve better performance.

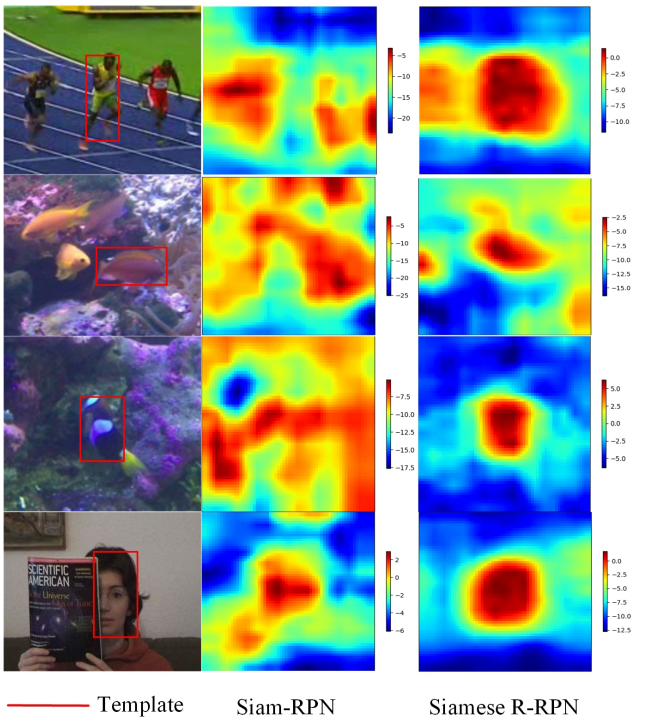



**Fig. 7. Visualization of the correlation convolution response maps generated by SiamRPN and Siamese R-RPN on VOT 2016.**

## V. Conclusion

A Siamese recurrent neural network and region proposal network (Siam R-RPN) trained in an end-to-end manner is proposed in this paper. Channel and spatial attention mechanisms are introduced to the feature extraction of Siam R-RPN for improving its feature representation. The recurrent neural network LSTM is used to generate the bounding box for the position of the tracking target, which is mapped to the anchor feature map of the RPN network for anchor selection to remove invalid anchors. Finally, the finely tuned anchor box is sent to the classification and regression branches to obtain the accurate position of the target. This paper demonstrates a means of cooperatively using temporal and spatial information in target tracking tasks. Its excellent tracking performance shows the potential of recurrent neural networks for target tracking. However, fully utilizing and mining the temporal information needs to be further studied.

## REFERENCES

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," in *ACM Computing Surveys*, vol. 38, no. 4, pp. B1–B45, 2006.
[2] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," in *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015.
[3] Z. Zhu, W. Wu, W. Zou, and J. Yan, "End-to-end flow correlation tracking with spatial-temporal attention," arXiv preprint arXiv: 1711.01124, 2017.
[4] Zhu, G. Huang, W. Zou, D. Du, and C. Huang, "UCT: Learning unified convolutional networks for real-time visual tracking," In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, Oct. 2017.
[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," In *Advances in Neural Information Processing Systems,* pp. 91–99, 2015.
[6] G. Han, H. Du, J. Liu, N. Sun and X. Li, "Fully Conventional Anchor-Free Siamese Networks for Object Tracking," in *IEEE Access*, vol. 7, pp. 123934-123943, 2019.
[7] L. Zhou, X. Yao and J. Zhang, "Accurate Positioning Siamese Network for Real-Time Object Tracking," in *IEEE Access*, vol. 7, pp. 84209–84216, 2019.
[8] X. Li *et al.*, "Target-aware deep tracking," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1369–1378, 2019.
[9] Y. Ming and Y. Zhang, "ADT: Object Tracking Algorithm Based on Adaptive Detection," in *IEEE Access*, vol. 8, pp. 56666–56679, 2020.
[10] Y. Zha, M. Wu, Z. Qiu, S. Dong, F. Yang and P. Zhang, "Distractor-Aware Visual Tracking by Online Siamese Network," in *IEEE Access*, vol. 7, pp. 89777–89788, 2019.
[11] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with Siamese region proposal network," In *CVPR*, 2018.
[12] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional Siamese networks for object tracking," In *ECCVW*, 2016. 1, 3, 4, 6, 7, 8.
[13] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," In *ICCV*, 2017. 2.
[14] M. Kristan *et al.*, "The sixth visual object tracking VOT2018 challenge results," In *ECCV Workshops*, 2018.
[15] Chaudhari S, Polatkan G, Ramanath R, et al. "An attentive survey of attention models," arXiv preprint arXiv: 1904.02874, 2019.
[16] Karpathy A, Johnson J, Fei-Fei L. "Visualizing and understanding recurrent networks," arXiv preprint arXiv: 1506.02078, 2015.
[17] D. Held, S. Thrun, and S. Savarese, "Learning to track at 100 fps with deep regression networks," In European Conference on Computer Vision, pp. 749–765, 2016.
[18] R. Tao, E. Gavves, and A. W. M. Smeulders, "Siamese instance search for tracking," *CVPR*, pp. 1420–1429, 2016.
[19] L. Huang, X. Zhao, and K. Huang, "Bridging the gap between detection and tracking: A unified approach," *Proceedings of the IEEE International Conference on Computer Vision*, 2019.
[20] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, and W. Hu, "Distractor-aware Siamese networks for visual object tracking," In *ECCV*, 2018.
[21] Q. Wang *et al.*, "Fast online object tracking and segmentation: A unifying approach," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.
[22] L. Zhang *et al.*, "Learning the model update for Siamese trackers," *Proceedings of the IEEE International Conference on Computer Vision*, 2019.
[23] G. Bhat *et al.*, "Learning discriminative model prediction for tracking," *Proceedings of the IEEE International Conference on Computer Vision*, 2019.
[24] H. Fan and H. Ling, "Siamese cascaded region proposal networks for real-time visual tracking," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.
[25] G. Wang *et al.*, "SPM-tracker: Series-parallel matching for real-time visual object tracking," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.
[26] B. Yan *et al.*, "'Skimming-Perusal'Tracking: A framework for real-time and robust long-term tracking," *Proceedings of the IEEE International Conference on Computer Vision*, 2019.
[27] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," In *International Conference on Neural Information Processing Systems*, pp. 1097–1105, 2012.
[28] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "SiamRPN++: Evolution of Siamese visual tracking with very deep networks," In *CVPR*, 2019.
[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," In *CVPR*, 2016.
[30] Zhang Z, Peng H, "Deeper and wider siamese networks for real-time visual tracking," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4591–4600, 2019.
[31] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
[32] W. Liu *et al.*, "SSD: Single shot multibox detector," *European Conference on Computer Vision*. Springer, Cham, 2016.
[33] Dai J, Li Y, He K, et al, "R-fcn: Object detection via region-based fully convolutional networks," In *Advances in neural information processing systems*, pp. 379–387, 2016.
[34] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.,* pp. 2980–2988, 2017.
[35] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," In *Proc. Eur. Conf. Comp. Vis.,* pp. 734–750, 2018.
[36] Tychsen-Smith L, Petersson L, "Denet: Scalable real-time object detection with directed sparse sampling," In *Proceedings of the IEEE international conference on computer vision*, pp. 428–436, 2017.
[37] Bahdanau D, Cho K, Bengio Y. "Neural machine translation by jointly learning to align and translate," arXiv preprint arXiv: 1409.0473, 2014.
[38] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," *International Conference on Machine Learning*, pp. 2048–2057, 2015.
[39] F. Wang *et al.*, "Residual attention network for image classification," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3156–3164, 2017.
[40] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, 2018.
[41] Szegedy C, Liu W, Jia Y, et al, "Going deeper with convolutions," In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.
[42] Woo S, Park J, Lee J Y, et al, "Cbam: Convolutional block attention module," In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 3–19, 2018.
[43] Law H, Deng J, "Cornernet: Detecting objects as paired keypoints," In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 734–750, 2018.
[44] Tychsen-Smith L, Petersson L, "Denet: Scalable real-time object detection with directed sparse sampling," In *Proceedings of the IEEE international conference on computer vision*, pp. 428–436, 2017.
[45] K. Greff, R. K. Srivastava, J. Koutn´ık, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," arXiv preprint arXiv: 1503.04069, 2015.
[46] D. Gordon, A. Farhadi, and D. Fox, "Re3: Real-time recurrent regression networks for object tracking," arXiv preprint arXiv: 1705.06368, 2017.
[47] L. Huang, X. Zhao, and K. Huang, "GOT-10k: A large high-diversity benchmark for generic object tracking in the wild," 2018.
[48] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.,* vol. 37, no. 9, pp. 1834–1848, 2015.
[49] M. Kristan *et al.*, "The visual object tracking VOT2016 challenge results," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pp. 777–823, 2016.
[50] J. Valmadre, L. Bertinetto, J. F. Henriques, A. Vedaldi, and P. H. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 2805–2813, 2017.
[51] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 1401–1409, 2016.

[52] M. Danelljan *et al.*, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR),* pp. 6638–6646, 2017.
[53] Y. Song, C. Ma, L. Gong, J. Zhang, R. W. Lau, and M.- H. Yang, "Crest: Convolutional residual learning for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV),* pp. 2555–2564, 2017.
[54] H. Fan and H. Ling, "Parallel tracking and verifying: A framework for real-time and high accuracy visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, pp. 5486–5494, 2017.
[55] C. Ma, X. Yang, C. Zhang, and M.-H. Yang, "Long-term correlation tracking," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 5388–5396, 2015.
[56] M. Danelljan, G. Hger, F. S. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference,* Sep. 2014.
[57] Zhong, B., Bai, B., Li, J., Zhang, Y., & Fu, Y, "Hierarchical tracking by reinforcement learning-based searching and coarse-to-fine verifying," in *IEEE Transactions on Image Processing*, 28(5), pp. 2331–2341, 2018.
[58] Chen, Z., Zhong, B., Li, G., Zhang, S., & Ji, R. "Siamese box adaptive network for visual tracking," *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* pp. 6668–6677, 2020.
[59] Guo, D., Wang, J., Cui, Y., Wang, Z., & Chen, S, "SiamCAR: Siamese fully convolutional classification and regression for visual tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* pp. 6269–6277, 2020.

**XUECHEN ZHAO** is pursuing the B.S. degree in the Communication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China. Her interests include single target tracking based on deep learning, video analysis and computer vision.

**YAOMING LIU** received the B.S. degree from Nantong University in 2017. Currently, he is pursuing the M.S. degree in the Signal and Information Processing, Nanjing University of Posts and Telecommunications, Nanjing, China. His current research interests include deep learning and computer vision.

**GUANG HAN** received the B.S. degree from Shandong University of Technology in 2004, and M.S. and Ph.D. degrees from Nanjing University of Science and Technology, in 2006 and 2010, respectively. Since 2010, he has been with Nanjing University of Posts and Telecommunications, Nanjing, China, where he is currently an associate professor in the Engineering Research Center of Wideband Wireless Communication Technology, Ministry of Education. His current research interests include pattern recognition, video analysis, computer vision and machine learning.