Visual tracking with multilevel feature, similarity attention, color constraint, and global redetection

Song Guiling®, Zhang Jingyi, Xue Feng, Lu Ru, Yang Qin and Ming Anlong

Abstract
Visual tracking is fundamental in computer vision tasks. The Siamese-based trackers have shown surprising effectiveness in recent years. However, two points have been neglected: firstly, few of them focus on fusing the image level and semantic level features in neural networks, which usually resulting in tracking failure when differentiating the target from other distractors of the same class. Secondly, the robustness of the previous redetection scheme is limited by simply expanding the search region. To address these two issues, we propose a novel multilevel feature-weighted Siamese region proposal network tracker, which employs a feature fusion module to construct discriminative feature embedding and a similarity-based attention module to suppress the distractors in the search region. Furthermore, a color-based constraint module is presented to further suppress the distractors with the same class to the target. Finally, a well-designed global redetection scheme is built to handle long-term tracking tasks. The proposed tracker achieves state-of-art performance on a series of popular benchmarks, including object tracking benchmark 2013 (0.699 in success score), object tracking benchmark 2015 (0.700 in success score), visual object tracking 2017 (0.470 in expected average overlap score), and visual object tracking (0.485 in expected average overlap score).

Keywords
Visual tracking, siameseRPN, color-based constraint module

Introduction
Visual tracking is a technique to track the target in an image sequence given the target’s bounding box in the first frame as the template. With the progress of computer vision, this task has received close attention in recent years and is widely used in intelligent applications, such as robotics, autonomous car, and video surveillance. In the development of visual tracking, many excellent works have been proposed. Most of them have addressed several challenging problems such as vague target, rotation, deformation, and illumination variation. Furthermore, other works in long-term (LT) visual tracking focus on overcoming the target disappearance in a video.

Recently, due to the trade-off between accuracy and speed, trackers based on the Siamese network are considered as the mainstream methods. Siamese-based trackers were proposed. They extract features of the template and a search frame by two convolution structures with shared weights and generate a similarity map between the template...
and this frame. What’s more, some improved Siamese-based trackers can be used to predict target deformation, or even directly output target segmentation, which have made significant achievements. However, two key issues remain. Firstly, the existing Siamese-based methods perform poorly in distinguishing the target and the distractors with the same class of the target. The reason is that conventional Siamese-based trackers utilize the backbone of the classification task. Thus, the high-level features fail to represent the difference between the target and other objects with the same class. Based on these features, the obtained similarity map is also unreliable to distinguish the target and others. As shown in Figure 1, the previous method has high responses on the distractors, that is, the referee, athletes marked in red, where the response indicates the similarity between these candidates and the template. Secondly, the previous redetection scheme simply expands the search region, thus limiting the performance of the LT tracking.

In this article, we argue that since the high-level cue of the target and a same-class object are similar, the low-level cue plays a key role in this case. In contrast, high-level cues are essential to distinguish the target from other similar appearance objects. Thus, the features of different levels are useful in different conditions. For this reason, we propose a multilevel feature-weighted Siamese region proposal network (MFW-SiamRPN). In contrast to the previous works, a feature fusion module (FFM) is designed to fuse all level features to a unified representation, which is used to encode the similarity between the template and each sliding window in the search region. Then, we design a similarity attention module. It utilizes channel attention to suppress channels that fail to distinguish template and distractors and utilizes spatial attention to suppress the similarities between template and distractors. Besides, to make full use of the color information lost in the deep network, a color-based constraint module (CCM) is proposed to suppress the network’s output. Finally, to track a target in LT scenarios, a global redetection scheme is proposed to detect the occurrence that the target is out of the view and redetect the target, which is simple but efficient.

In general, the main contributions can be summarized as follows:

- To distinguish the target and other distractors, we propose an MFW-SiamRPN (MFW-SiamRPN). It utilizes multilevel cues to template and search region and suppresses other distractors by a similarity attention module.
- A CCM, which takes full use of color information, is designed to constraint the network output.
- A well-designed global redetection scheme is proposed, which ensures the robustness of the proposed tracker and outperforms other Siamese-based trackers in LT tracking.
- Our method outperforms the existing state-of-the-art methods on the fashionable benchmarks: object tracking benchmark (OTB)-2013, OTB-2015, visual object tracking (VOT)-2017, VOT2018, and VOT-2018-LT. Moreover, abundant ablation studies verify our point of view and the effectiveness of each module.

**Related work**

**Review of Siamese-based trackers**

Recently, trackers based on the Siamese network draw great attention for their remarkable performance in both
accuracy and efficiency. These methods employ two identical networks initialized in the classification task (e.g. AlexNet\cite{Krizhevsky_2012}, Visual Geometry Group (VGG)\cite{Simonyan_2014}, and MobileNet\cite{Howard_2017}) to extract features from the template and search region and locate the target by matching the template. Generic Object Tracking Using Regression Networks (GOTURN)\cite{Reda_2017} directly concatenates the features of the template and search region and employs several fully connected layers to locate the target. Siamese Instance Search for Tracking (SINT)\cite{Cazacu_2019} constructs the vectors of the target and several candidates by region-of-interest pooling and then outputs the candidate most similar to the template. Fully convolutional (FC) Siamese networks (SiamFC)\cite{Li_2018} proposes a cross-correlation layer to obtain the similarity between the target and the search region, improving accuracy and efficiency. Based on SiamFC, RasNet\cite{Guiling_2020} proposes a residual attention module, which reduces the noise in the deep features. As shown in Figure 2(a) and (b), SiamRPN\cite{Li_2018, Sun_2019} and SiamRPN++\cite{Sun_2019} use a well-designed region proposal network (RPN)\cite{Ren_2015} to predict the position and deformation of the target simultaneously and outperform other Siamese-based trackers. However, since the backbones used in these methods are all pretrained in the classification task, the inherent difference between the two tasks, namely, classification and tracking, making the deep feature fail to represent the difference between the target and other objects with the same class. We propose MFW-SiamRPN and CCM to overcome this issue. As shown in Figure 2, different from others, all level features are fused to represent the template and the search region.

**LT tracking and redetection scheme**

In visual tracking, the target is usually occluded or moved out of the view. To address the issue, the LT tracking redetects the target when the target returns to the view of the camera. There are mainly two types of methods for LT tracking. Firstly, some methods redetect the target by expanding the search region in the frame. DaSiamRPN\cite{Si_2019} and SiamRPN++\cite{Sun_2019} establish the reliable confidence score by specially designed training samples but thoughtlessly enlarge the search region while losing the target, which limits the performance of the LT tracking. Secondly, a few methods construct a redetection scheme. Skimming-Perusal Long-term Tracking (SPLT)\cite{Zheng_2018} builds a global redetection scheme by training an additional network called skim-module and surpasses other methods that enlarge the search region. However, the additional skimming model requires feature maps of the whole search region and downsamples it to a vector, which brings high Graphics Processing Unit (GPU) cost and risk of low accuracy.

The redetection task could be divided into two steps, object detection and target identification. We find several simple but effective proposals to locate an object\cite{Dai_2015, Pan_2018, Alahi_2016, Wei_2016} and the excellent Siamese-based trackers could handle the target identification.\cite{Li_2018} In the section “Redetection scheme for LT tracking,” we apply object proposal to locate candidate objects and combine our CCM with the Siamese network to identify the target.

**Method**

Our method is described in the following order. In the section “Multilevel feature-weighted SiamereRPN,” considering that the low-level feature plays a vital role in distinguishing the target and other objects, the MFW-SiamRPN is presented, which weights features of all levels and then suppresses the similarity between the false positives and the target. In the section “Color-based constraint module,” to take color information into account, a CCM is presented to constrain the network’s output. In the section “Redetection scheme for LT tracking,” the redetection
scheme for LT tracking is proposed to address the issue that the target disappears or is fully occluded.

Multilevel feature-weighted SiameseRPN

Overview. We adopt SiamRPN\(^5\) as the basic tracker. Figure 3 shows the structure of our network. Given the first frame \(I_1\) and the current frame \(I_n\), the whole network takes a template \(z\) from frame \(I_1\) and a search region \(x\) from frame \(I_n\) as inputs, where \(w_T\) and \(h_T\) are the width and height of the target. And the template is centered on the target and has red, green, and blue (RGB) channels and a resolution of \(2w_T \times h_T\). Our network aims to locate the target in the search region \(x\). In the following paragraphs, we first introduce the entire network and then represent the proposed modules in detail.

Mathematically, let \(F\) denote the fusion of the backbone (see the section “Multilevel feature-weighted SiameseRPN”). Let \(A_s()\) denote the spatial attention and \(A_c()\) denote the channel attention. The weighted similarity can be formulated as follows

\[
f(z, x) = A_s(A_c(f(z, x)) \otimes f(z, x)) \otimes A_c(f(z, x)) \otimes f(z, x)
\]  

where \(\otimes\) denotes the element-wise multiplication. Finally, following the same way as in ref.\(^5\), we generate many anchor boxes in search region \(x\) as the candidates of the target. And the similarity \(f(z, x)\) is utilized to predict the probability that each box is the target by using two convolution branches, namely Classification (CLS) branch and Regression (REG) branch. The CLS branch predicts the similarity between each anchor boxes and the target (denoted by \(N_s\) score in the section “Color-based constraint module”). The REG branch predicts the deformation and displacement \([d_x, d_y, d_w, d_h]\) of each anchor box. Mathematically, the two branches can be formulated as follows

\[
\text{CLS}(z, x) = C_1(f(z, x)) \\
\text{REG}(z, x) = C_1(f(z, x))
\]

where \(C_1\) denotes a \(1 \times 1\) convolution operation, which adjusts the channels of \(f(z, x)\) to 2 (for CLS) and 4 (for REG). By using a SoftMax operation, \(\text{CLS}(z, x)\) is transformed into a similarity map of template and search region in the corresponding positions.

During training, we use the SoftMax loss function to supervise the CLS branch, while the REG branch is optimized by Smooth L1 loss function. The total training loss is
Spatial Attention

between channels, and the max-pooling is to gather another pooling. The average pooling is to capture the relation parallel pooling layers, namely average pooling and max down-sampled to one-eighth size of input images by two Res features of the first two stages, namely our network, the prediction is denoted as Figure 4.

Guiling et al.

Figure 4. The detail of our channel attention (left), spatial attention (middle), and similarity maps after attention (right).

the summation of them. Formally, for the CLS branch in our network, the prediction is denoted as \( G \in \mathbb{R}^{1 \times W \times H} \), the label is denoted as \( \hat{G} \in \mathbb{R}^{1 \times W \times H} \), their resolutions are same with similarity \( f(z, x) \). The SoftMax operation is utilized to calculate the probability of a pixel belong to the target: \( G(i, j) = \frac{G(i, j)}{G(i, j) + G(\mathbf{i}, J)} \), and the SoftMax loss can be formulated as follows

\[
I_{\text{softmax}} = -\sum_{i=1}^{W_z} \sum_{j=1}^{H_z} G(i, j) \times \log(G(i, j)) \quad (4)
\]

For the REG branch in our network, the prediction is denoted as \( L \in \mathbb{R}^{4 \times W \times H} \), each pixel \( x \) of \( L \) response is \([d_x, d_y, d_w, d_h]\) and the corresponding label is \( y \), the smooth L1 loss value at each corresponding position can be calculated as follows

\[
I_{\text{smooth}} = \begin{cases} 
0.5 \times (x - y)^2, & |x - y| < 1 \\
|x - y|, & |x - y| \geq 1 
\end{cases} \quad (5)
\]

Feature fusion module. Based on the motivation: both the low-level feature and the high-level feature are crucial to distinguish the target and others, a FFM is proposed to capture the feature from all levels. In detail, the stage-\( n \) side-outputs of the backbone are represented as \( \text{Res n} \). The features of the first two stages, namely \( \text{Res 1} \) and \( \text{Res 2} \), are down-sampled to one-eighth size of input images by two parallel pooling layers, namely average pooling and max pooling. The average pooling is to capture the relation between channels, and the max-pooling is to gather another important cue about distinctive object features. Then, these down-sampled features are concatenated with \( \text{Res 3, Res 4, and Res 5} \). Finally, a bottle-neck structure, namely a sequence of two \( 1 \times 1 \) convolutions and a \( 3 \times 3 \) convolution, is utilized to nonlinearly compress all the features and reduce the channel number to 512. Formally, the FFM can be formulated as follows

\[
f = \text{Cat}(\text{MP(Res1)}, \text{AP(Res1)}, \text{MP(Res2)}, \text{AP(Res2)}|\text{Res3, Res4, Res5})
\]

\[
F() = \text{relu}(C_1(\text{relu}(C_3(\text{relu}(C_1(f)))))
\]

where \( \text{MP()} \) denotes the max pooling operation, \( \text{AP()} \) denotes the average pooling operation, \( \text{Cat()} \) denotes the concatenation operation, \( C_s() \) denotes the \( 3 \times 3 \) convolution operation, \( \text{relu()} \) denotes the Rectified Linear Unit (ReLU) activation.

Since the low-level features contain little semantic information, such features could be used to distinguish two objects of the same class. In contrast, high-level features could be used to separate objects with different classes. Hence, multilevel features are all necessary to distinguish the target and other objects, and the fused features are better to represent all the objects than those in the previous works.

Attention based on similarity. As shown in Figure 3, the similarity \( f(z, x) \in \mathbb{R}^{C \times H_i \times W_i} \) is obtained by the depth-wise cross-correlation of the two fused features. To further enhance the difference between the template and other distractors, we capture the attention of the similarity. Specifically, two types of attention information are captured sequentially, namely channel attention and spatial attention. These two submodules are motivated by Convolutional Block Attention Module (CBAM). Firstly, the channel attention module is utilized to capture the inter-channel relationship of similarity \( f(z, x) \). As shown in Figure 4, it uses the average pooling operation to aggregate the spatial information and uses the max pooling operation to capture the discriminative cue in similarity matching. Then, a shared convolution block, which is constructed by three \( 1 \times 1 \) convolutions, is utilized to fuse both pooled features. Finally, a sigmoid function is used to generate channel weights \( A_s(f(z, x)) \in \mathbb{R}^{C \times 1 \times 1} \). Formally, the channel weights can be formulated as follows
The area with blue in (a) is the region of the target (T), while the rest surrounding area is background (B), (b) is a similarity map generated by MFW-SiamRPN, (c) is the target probability map of x, and (d) is the search region x, the distractors with high NA scores are in red bounding boxes and the selected target is in green bounding boxes. MFW: multilevel feature-weighted; RPN: region proposal network.

\[
A_t(X) = \text{Sig}(C_1(\text{relu}(C_1(\text{relu}(C_1(\text{MP}(X)))))))
\]
\[
+ (C_1(\text{relu}(C_1(\text{relu}(C_1(\text{AP}(X)))))))
\]  

where \(\text{Sig}()\) denotes the sigmoid function, and \(X\) denotes the input feature. The spatial attention module \(A_t()\) is utilized to capture the interspatial relationship of similarity. Specifically, the feature map is firstly passed through a channel-wise max-pooling layer and a channel-wise average pooling layer to be down-sampled to two feature maps with one channel, respectively. For the input feature \(X\) with \(C\) channels, the channel-wise max pooling and average pooling can be formulated as follows

\[
X_{\text{avgpool}}(i, j) = \text{mean}(X(0 : C, i, j))
\]
\[
X_{\text{maxpool}}(i, j) = \text{max}(X(0 : C, i, j))
\]

Then, these two one-channel feature maps are concatenated and passed through a 3 x 3 convolution and a sigmoid to generate the spatial attention

\[
A_t(X) = \text{Sig}(C_3(\text{Cat}(X_{\text{avgpool}}, X_{\text{maxpool}})))
\]

Finally, the output feature map can be formulated as follows

\[
f(z, x) = A_t(A_t(f(z, x)) \otimes f(z, x))
\]

where the bolded \(f(z, x) \in \mathbb{R}^{C \times H \times W}\) denotes the feature after attention.

For the channel attention, since each channel’s discriminative ability in \(f(z, x)\) is different, the channel attention gives high weights to the channels of similarity \(f(z, x)\) with intense discrimination. Thus, compared to directly applying the original similarity to match the target, the weighted similarity better distinguishes the target and other distractors. For the spatial attention, since the pooled features contain each pixel’s discriminative channel, the spatial attention module weights all pixels, which enhances the response of the target and suppresses that of the distractors. Thus, our similarity attention enhances the difference between target and distractors in feature representation.

**Color-based constraint module**

Due to the normalization operation in Convolutional Neural Network (CNN), normalized color in different search regions and templates ranges variously, resulting in limited color information expression. To make full use of the color information, we propose the CCM, which takes the boxes predicted by the network as inputs and computes the color-based similarities between these boxes and the target. In detail, CCM contains three steps.

Firstly, assuming that the template \(z\) consists of two disjoint regions: the rectangle region of the target with size \(w_T \times h_T\) (i.e. T) and the rest of the template \(z\) is denoted as background area (i.e. B), this module constructs the color histograms of the two regions, respectively. Specifically, we present one pixel in the template with RGB channels by a triplet \(i \in \mathbb{R}^3\). Then, the RGB values of all pixels are reduced by 16 times: \(i = [i/16] \in \mathbb{R}^1\). Finally, since each value of \(i\) is an integer from 0 to 15, we count the number of occurrences of each triplet in region \(T\) to construct the normalized color histogram \(H_T \in \mathbb{R}^{16 \times 16 \times 16}\). In the same way, the histogram \(H_B \in \mathbb{R}^{16 \times 16 \times 16}\) is constructed corresponding to region \(B\).

Secondly, according to the color histograms \(H_T\) and \(H_B\), the probability that a pixel \(i\) in search region \(x\) belongs to target is obtained by the Bayes rule (see additional materials for detailed formula)

\[
P(i \in \exists | T, B, i) = \frac{H_T([i/16])}{H_T([i/16]) + H_B([i/16])}
\]

where \(\exists\) denotes the class of the target. Based on Eq. (11), the probabilities of all pixels in \(x\) are obtained to form a probability map \(S\), as shown in Figure 5(c).

Thirdly, we rerank the boxes given by the MFW-SiamRPN. Specifically, assuming that \(A = (x_A, y_A, h_A, w_A)\) denotes a bounding box with the same representation of other articles. Thus, the target probability of box \(A\) is stated as the mean probability of all pixels inside \(A\)

\[
TS_A = \frac{1}{h_A \times w_A} \sum_{i \in A} S(i)
\]
where $S(i)$ denotes the value of pixel $i$ on the probability map $S$.

Then, we combine this color-based probability with the score given by the network

$$SC_A = (N_A \times (1 - \alpha) + TS_A \times \alpha) \times \exp(1 - D_{xy} / 255) \times P_A$$

(13)

where $N_A$ is the similarity score predicted by the CLS branch of our MFW-SiamRPN, $D_{xy}$ is the distance between box $A$ and the center of region $x$, $P_A$ is the deformation penalty of the bounding box $A$, which is referred to ref. Then $\alpha$ is a constant to balance the Siamese network and the CCM. By utilizing Eq. (13), a more reliable target prediction is obtained than utilizing the network only, as shown in Figure 5(d).

**Redetection scheme for LT tracking**

To extend our network for LT tracking, we propose a redetection scheme, as shown in Figure 6. Since the score of the target SC given by CCM is reliable, the redetection scheme is activated if $SC < 0.7$, where $SC_2$ denotes the target score in the second frame of the input video. The redetection scheme is composed of three steps: probability map-generating, object proposal, and target selection. Initially, the whole image is taken as the input, and a target probability map of this image is generated by Eq. (13) in CCM. Secondly, we employ the object proposal method, i.e. Edge boxes, to obtain various bounding boxes with the same size as the target in the template or in the last frame. For clarity, we denote the bounding box as $A$, and its objectness score predicted by Edge boxes as $OS(A)$. Finally, the boxes with the top 200 objectness scores are retained to the third step. Thirdly, all the bounding boxes with 200 are reranked by the color-based probability. In more detail, by utilizing the target probability map given by the first step and utilizing Eq. (13), the target probability $TS_A$ is obtained. Then, the score for LT tracking $SL_A$ is formulated to rerank all the boxes

$$SL_A = (OS_A \times (1 - \beta) + TS_A \times \beta) \times \exp \left(1 - \frac{D_{xy}}{\sqrt{w_I \times h_I}} \right)$$

(14)

where $OS_A$ is the objectness score of a bounding box $A$, $TS_A$ is the CCM score in the section “Color-based constraint module,” $\beta$ is a constant to balance the object proposal Module and the CCM. $D_{xy}$ is the distance between the bounding box and the center of an image, $w_I$ and $h_I$ are the width and height of an image, respectively. By top selection of the $SL_A$ score, the top five candidate bounding boxes are left. These five candidates have a high probability of including objects and are similar to the target in color. Finally, five search regions centered on these candidates are sent into our Siamese network to predict the target’s location and size.

The whole redetection scheme in LT tasks could be seen in Figure 6. When the target disappears from the search region, it will obtain a low SC score and the tracker activates the global redetection scheme. The object proposal method helps our tracker locate candidate objects and the CCM helps our tracker filter out objects that are not similar to the target in color. Consequently, the Siamese network predicts the bounding box of the target precisely.

**Experiments and evaluations**

**Implementation details**

The training pairs of our Siamese network come from three data sets. Large-scale video detection data set (ImageNet Large Scale Visual Recognition Challenge [ILSVRC] 2015) contains 4417 videos of 30 different objects and over 1 million images of 1000 different objects. Microsoft Common Object in Context (COCO) data set contains 328,000 images of 91 different objects. YouTube-BoundingBoxes data set contains 210,000 videos of 13 different objects. All labeled objects in these images and videos are cropped with the target centered in it and double the size of $\sqrt{w_T \times h_T}$. After the crop operation, these template
patches are resized to 127 x 127 pixels. The search regions are cropped twice the size of template patches in the same ways.

The proposed Siamese network is trained for 20 epochs with a batch size of 20 and an initial learning rate of $10^{-2}$ by using the stochastic gradient descent (SGD) solver. The momentum and weight decay are set to 0.1 and 0.5, respectively. Parameters of the backbone are frozen in the first 10 epochs and unlocked in the last 10 epochs to adapt to tracking tasks.

### Evaluation on OTB data set

OTB2013 consists of 50 labeled videos and brings challenging scenes such as rotation, blur, and occlusion. OTB2015 is an extension of OTB2013, including 100 labeled sequences, and contains more complicated scenes. The precision score and the success score are used to evaluate the performance.

The precision score represents the ratio of predicted results (the distance between the predicted center and ground-truth center is within 20 pixels). The success score shows the Area Under Curve (AUC) of the successful plot. The x-axis of the success plot indicates the IOU threshold between the predicted and ground-truth result (varies from 0 to 1), whereas the y-axis shows the proportion of successful frames, which satisfied the IOU threshold. One-pass evaluation is employed for the evaluation.

Our tracker is compared with several state-of-art Siamese-based trackers, including SiamFC, parallel tracking and verifying (PTAV), RasNet, Correlation Filter based Network (CFNet), SiamRPN, DaSiamRPN, SiamRPN++, Siamese Cascaded Region Proposal Network (C-RPN), Discriminative Model Prediction (DIMP), GradNet, and Meta-Learning Tracker (MLT). Table 1 presents the precision scores and success scores of Siamese-based trackers mentioned earlier. As presented in Table 1, our tracker outperforms these state-of-art Siamese-based trackers over two benchmarks. Specifically, our tracker achieves the best precision (0.930 on OTB2013 and

### Table 1. Precision and success results on OTB2013 (left) and OTB2015 (right). Red bold type indicates the best performance, and blue bold type indicates the second-best performance (the same below).

| Methods  | Backbones | OTB2013 Precision | OTB2013 Success | OTB2015 Precision | OTB2015 Success |
|----------|-----------|-------------------|----------------|-------------------|----------------|
| SiamFC   | AlexNet   | 0.809             | 0.608          | 0.771             | 0.582          |
| PTAV     | AlexNet   | 0.895             | 0.663          | 0.859             | 0.635          |
| RasNet   | AlexNet   | 0.872             | 0.670          | 0.693             | 0.642          |
| CFNet    | AlexNet   | 0.766             | 0.651          | —                 | —              |
| SiamRPN  | AlexNet   | —                 | —              | 0.851             | 0.637          |
| DaSiamRPN| AlexNet   | —                 | —              | 0.880             | —              |
| SiamRPN++| Res50     | —                 | —              | 0.914             | 0.696          |
| SiamDW   | Res50     | 0.920             | 0.670          | 0.900             | 0.670          |
| C-RPN    | Res50     | —                 | 0.675          | —                 | —              |
| DIMP     | AlexNet   | —                 | —              | 0.663             | —              |
| GradNet  | AlexNet   | —                 | —              | 0.861             | 0.639          |
| MLT      | AlexNet   | —                 | 0.621          | —                 | 0.611          |
| Ours     | Res50     | 0.930             | 0.699          | 0.915             | 0.700          |

Figure 7. Comparisons of our MFW-SiamRPN tracker with state-of-the-art Siamese-based trackers in challenging scenes of OTB2015 benchmark. Our tracker shows more robustness on scenarios with similar distractors, occlusion, flurries, illumination variation, and so on. MFW: multilevel feature-weighted; OTB: object tracking benchmark; RPN: region proposal network.
For the success score, our tracker superiorly performs against other evaluated methods (0.699 on OTB2013 and 0.7 on OTB2015). Compared with the AlexNet-based methods, our tracker benefits from the Res50 backbone and outperforms DaSiamRPN by 3.5% on the precision of OTB2015. Note that our method only achieves a little improvement over SiamRPN++, which is due to a lot of gray-scale videos in OTB 2015, and no color cue can be utilized. Besides, compared with other methods, whose backbone are deep CNNs, our tracker benefits from the FFM, the attention module, and the CCM. The res50 backbone extracts features that help to distinguish the target from the background. Low-level cues utilized by the FFM help our tracker distinguish the target from distractors with the same class, while high-level cues help our tracker distinguish the target from distractors with a similar appearance. The CCM replenishes color information, which is lost in normalization operation. The performance of our tracker proves the effectiveness of these modules.

**FC:** fully convolutional; **OTB:** object tracking benchmark; **RPN:** region proposal network; **PTAV:** parallel tracking and verifying.

**Evaluation on VOT data set**

VOT2017 and VOT2018 consist of 60 sequences and show more difficult scenes than OTB benchmarks. We use the most-watched expected average overlap (EAO) score on the baseline benchmark as the evaluation standard. EAO is an estimator of the average overlap a tracker is expected to attain on a large collection of short-term sequences. Besides, we also choose the accuracy (average overlap when tracking successfully) and the Robustness (failure times) as evaluation standards.

As shown in Table 2, our tracker is compared with the state-of-art tracking algorithms, including SiamFC, SiamRPN, DaSiamRPN, SiamRPN++, SiamDW, C-RPN, GradNet, DIMP, UpdateNet,13 Our approach shows the best EAO score (0.470 on VOT2017 and 0.485 on VOT2018). SiamRPN++ and DIMP have the same backbone as our tracker. These three Res50 based trackers have similar accuracy scores. Moreover, DIMP shows less tracking failures (0.153) followed by our tracker (0.201). As to the most concerned EAO score on VOT2018, our tracker outstands the DIMP by 4.5% and SiamRPN++ by 7.1%, respectively, which sufficiently proves the effectiveness of the proposed method.

**Evaluation on the VOT2018-LT data set**

We choose the VOT2018-LT data set with a lot of completely lost scenes to show the superiority of our redetection scheme. This data set includes totally 146,847 frames with 35 sequences of various objects. Each sequence contains at least 1000 images and average 12 LT object disappearances. The main evaluation protocol of the VOT2018-LT data set is F1 score which is defined by

\[
F1(\theta) = \frac{2 \times \text{Pr}(\theta) \times \text{Re}(\theta)}{\text{Pr}(\theta) + \text{Re}(\theta)}
\]  

where \(\theta\) is a given threshold. \(\text{Pr}(\theta), \text{Re}(\theta)\) are the precision, recall, and F1 score under the corresponding threshold, respectively. The threshold varies from 0 to 1 to find the maximum F1 score.

### Table 2. Evaluation results on VOT2017 (left) and VOT2018 (right).

| Trackers | Backbones | VOT2017 | | | | VOT2018 | | | |
|----------|-----------|---------|---------|---------|---------|---------|---------|---------|---------|
|          |           | A (\%)  | R (\%)  | EAO (\%) | A (\%)  | R (\%)  | EAO (\%) |
| SiamFC   | AlexNet   | 0.502   | 0.585   | 0.188    | 0.585   | 0.276   | 0.383    |
| SiamRPN  | AlexNet   | 0.490   | 0.460   | 0.244    | 0.560   | 0.340   | 0.326    |
| DaSiamRPN| AlexNet   | —       | —       | —        | 0.600   | 0.234   | 0.414    |
| SiamRPN++| Res50     | —       | —       | —        | —       | —       | —        |
| SiamDW   | Res50     | 0.520   | 0.410   | 0.301    | 0.597   | 0.153   | 0.440    |
| C-RPN    | Res50     | —       | —       | 0.289    | —       | —       | —        |
| GradNet  | AlexNet   | 0.507   | 0.375   | 0.247    | —       | —       | —        |
| DIMP     | AlexNet   | —       | —       | —        | 0.629   | —       | —        |
| UpdateNet| AlexNet   | —       | —       | —        | —       | —       | —        |
| Ours     | Res50     | 0.620   | 0.210   | 0.470    | 0.601   | 0.201   | 0.485    |

### Table 3. Evaluation results on VOT2018-LT (for all metrics, larger is better).

| Tracker      | F1 | Precision | Recall |
|--------------|----|-----------|--------|
| SiamFC       | 0.433 | 0.636 | 0.328 |
| DaSiamRPN-LT | 0.607 | 0.627 | 0.588 |
| PTAVplus     | 0.481 | 0.595 | 0.404 |
| SPLT         | 0.616 | 0.633 | 0.600 |
| SiamRPN++LT  | 0.629 | —     | —      |
| Ours         | 0.641 | 0.633 | 0.625 |
Accordingly, we choose some trackers specifically optimized for LT tracking to compare their performance, including, SiamFC, DaSiamRPNLT, SiamRPN++LT, PTAVplus, and SPLT. The evaluation result is placed in Table 3.

| Number | Network                | Backbone | L1  | L2  | L3  | L4  | L5  | Att | Ta | Re | OTB2015 | VOT2018 | LT          |
|--------|------------------------|----------|-----|-----|-----|-----|-----|-----|----|----|---------|----------|-------------|
| i      | SiamRPN                | AlexNet  | 0.637| 0.326|     |     |     |     |    |    | 0.607   | 0.326    | 0.607       |
| ii     | SiamRPN               |          | 0.670| 0.360|     |     |     |     |    |    |         |          |             |
| iii    | SiamRPN               |          |     |     |     |     |     |     |    |    |         |          |             |
| iv     | SiamRPN++             | Res50    |     |     |     |     |     |     |    |    | 0.601   | 0.128    | 0.621       |
| v      | SiamRPN++             |          |     |     |     |     |     |     |    |    | 0.610   | 0.189    |             |
| vi     | SiamRPN++             |          |     |     |     |     |     |     |    |    | 0.669   | 0.331    |             |
| vii    | SiamRPN++             |          |     |     |     |     |     |     |    |    | 0.678   | 0.374    |             |
| viii   | SiamRPN++             |          |     |     |     |     |     |     |    |    | 0.646   | 0.320    |             |
| ix     | SiamRPN++             | Res50    |     |     |     |     |     |     |    |    | 0.696   | 0.414    | 0.629       |
| x      | SiamRPN++             |          |     |     |     |     |     |     |    |    | 0.690   | 0.421    |             |
| xi     | SiamRPN++             |          |     |     |     |     |     |     |    |    |         |          |             |
| xii    | Ours                   | Res50    |     |     |     |     |     |     |    |    | 0.690   | 0.420    | 0.631       |
| xiii   | Ours                   | Fused by FFM |     |     |     |     |     |     |    |    | 0.696   | 0.431    |             |
| xiv    | Ours                   | Fused by FFM |     |     |     |     |     |     |    |    | 0.699   | 0.461    |             |
| xv     | Ours                   | Fused by FFM |     |     |     |     |     |     |    |    |         |          |             |
| xvi    | Ours                   | Fused by FFM |     |     |     |     |     |     |    |    |         |          |             |

OTB: object tracking benchmark; VOT: visual object tracking; FFM: feature fusion module; CCM: color-based constraint module; RPN: region proposal network; LT: long-term; L1–L5: Res1–Res5, respectively; Att: the proposed attention module; Ta: the proposed CCM; Re: the proposed Re-detection scheme.

Ablation studies

In this section, we firstly present the positive function of the MFW-SiamRPN and CCM in short-term tracking tasks. Then, the analysis of LT tasks is brought out to show the contribution of the redetection scheme in Table 4.

Analysis of FFM. To pursue the influence of our FFM on the tracking result, the features of the first two layers are trained with a single output structure. The comparison among iv–viii shows that by independent operation, the features of Res1 and Res2 perform terribly in both OTB2015 and VOT2018 benchmarks. ix and xii show that the introduction of features from Res1 and Res2 can improve the performance on the VOT2018 benchmark by 0.6% but reduce the performance on OTB2015 by 0.6%, which is due to the inefficient features fusion. The performance gap between independent operation (iv–viii) and collaborative operation (ix, xii) of multilevel features confirms our conclusion that all features of the backbone should be taken into consideration. From ix, xii, and xiii, we find that simply using the average outputs of multilayers is not an appropriate solution. The FFM gains an improvement of 1.1% over the average type baseline, which fully proves the effectiveness of the proposed FFM.

Analysis of attention structure. From Table 4, we can also deduce the advantages of our attention structure. The xiii and xiv present that, through our attention structure, the feature-fused Res50 tracker gets a competitive performance (0.461 on EAO) 4.7% better than SiamRPN++ and 3.0% than the fused baseline. What’s more, to investigate the impact of different attention types, some additional attention evaluations are proposed in Table 5. Evaluations in Table 5 show that by changing the order of channel attention and spatial attention, there is a slight drop of 0.4% on the success score in OTB2015, which is in contrast to an increase of 0.1% on EAO in VOT2018. In the VOT2018 benchmark, channel attention and spatial attention bring an increase of 0.9% and 2.4%, respectively.

Table 4. Ablation study of our tracker on OTB2015 (success score), VOT2018 (EAO score), and VOT2018-LT (F1 score).

| Number | Network                | Backbone | L1  | L2  | L3  | L4  | L5  | Att | Ta | Re | OTB2015 | VOT2018 | LT          |
|--------|------------------------|----------|-----|-----|-----|-----|-----|-----|----|----|---------|----------|-------------|
| i      | SiamRPN                | AlexNet  | 0.637| 0.326|     |     |     |     |    |    | 0.607   | 0.326    | 0.607       |
| ii     | SiamRPN               |          | 0.670| 0.360|     |     |     |     |    |    |         |          |             |
| iii    | SiamRPN               |          |     |     |     |     |     |     |    |    |         |          |             |
| iv     | SiamRPN++             | Res50    |     |     |     |     |     |     |    |    | 0.601   | 0.128    | 0.621       |
| v      | SiamRPN++             |          |     |     |     |     |     |     |    |    | 0.610   | 0.189    |             |
| vi     | SiamRPN++             |          |     |     |     |     |     |     |    |    | 0.669   | 0.331    |             |
| vii    | SiamRPN++             |          |     |     |     |     |     |     |    |    | 0.678   | 0.374    |             |
| viii   | SiamRPN++             |          |     |     |     |     |     |     |    |    | 0.646   | 0.320    |             |
| ix     | SiamRPN++             | Res50    |     |     |     |     |     |     |    |    | 0.696   | 0.414    | 0.629       |
| x      | SiamRPN++             |          |     |     |     |     |     |     |    |    | 0.690   | 0.421    |             |
| xi     | SiamRPN++             |          |     |     |     |     |     |     |    |    |         |          |             |
| xii    | Ours                   | Res50    |     |     |     |     |     |     |    |    | 0.690   | 0.420    | 0.631       |
| xiii   | Ours                   | Fused by FFM |     |     |     |     |     |     |    |    | 0.696   | 0.431    |             |
| xiv    | Ours                   | Fused by FFM |     |     |     |     |     |     |    |    | 0.699   | 0.461    |             |
| xv     | Ours                   | Fused by FFM |     |     |     |     |     |     |    |    |         |          |             |
| xvi    | Ours                   | Fused by FFM |     |     |     |     |     |     |    |    |         |          |             |

Table 5. Evaluation results on different types of attention.

| Attention type          | OTB2015 (success) | VOT2018 (EAO) |
|-------------------------|-------------------|---------------|
| None                    | 0.696             | 0.431         |
| Channel only            | 0.696             | 0.440         |
| Spatial only            | 0.696             | 0.455         |
| Due (SAC)               | 0.699             | 0.461         |
| Due (parallel)          | 0.695             | 0.462         |

SAC: spatial attention after channel attention; Parallel: two kinds of attention run parallel like RasNet3; OTB: object tracking benchmark; VOT: visual object tracking; EAO: expected average overlap.
Analysis of color-based constraint module. The xiv and xv in Table 4 prove the advantages of our CCM. Our tracker gains an improvement of 2.4% on the VOT2018 benchmark. However, our CCM does not work on gray sequences and brings little improvement in OTB2015. Furthermore, we infer that the absence of color information should be a generality problem for all Siamese networks. The ii, x, and xv in Table 4 confirm this speculation. Our CCM brings an improvement of 3.4% for DaSiamRPN and 0.7% for SiamRPN++.

Analysis of redetection module. The comparison of xv and xvi in Table 4 shows that the MFW-SiamRPN tracker with our global redetection scheme has an improvement of 2.6% on the VOT2018-LT benchmark. What’s more, any tracker that lacks a global redetection process could benefit from our redetection scheme on LT benchmarks. The comparison of x and xi in Table 4 shows that SiamRPN++ with our global redetection scheme gains a 0.2% improvement. At the same time, ii and iii in Table 4 show that our global redetection scheme brings an improvement of 1.4% for SiamRPN.

Conclusion
In this article, we present a novel MFW-SiamRPN to track the template in a video. Our contributions are motivated by two intuitions. Firstly, too much attention to high-level features leads to ignorance of the capability of low-level features to differentiate between same-class objects. For this reason, the FFM is designed to aggregate features from the lowest level to the highest level. Then, similarity attention is proposed to select more discriminative features and suppress the similarity between distractors and the target. And the CCM is proposed to make full use of the color information lost in the deep network. As a result, our method tracks the target stably even when same-class objects interfere. Secondly, to track the target robustly in an LT video, a global redetection scheme is proposed to precept the target’s condition and redetect the target. Numerous experiments on several data sets demonstrate the effectiveness of our method.

Author contribution
Song Guiling and Zhang jingyi are contributed equally to this work, and the contributions of other authors are in the order of the author sequence.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iD
Song Guiling https://orcid.org/0000-0001-9476-5650

References
1. Bertinetto L, Valmadre J, Henriques J, et al. Fully-convolutional Siamese networks for object tracking. In: ECCV workshops, Amsterdam, The Netherlands, 11–14 October 2016, pp. 850–865.
2. Valmadre J, Bertinetto L, Henriques J, et al. End-to-end representation learning for correlation filter based tracking. In: IEEE conference on computer vision and pattern recognition (CVPR), Honolulu, HI, USA 21–26 July, 2017, pp. 5000–5008. IEEE.
3. Wang Q, Teng Z, Xing J, et al. Learning attentions: residual attentional Siamese network for high performance online visual tracking. In: IEEE conference on computer vision and pattern recognition (CVPR), Salt Lake City, UT, USA, 18–23 June, 2018, pp. 4854–4863. IEEE.
4. Fan H and Ling H. Parallel tracking and verifying: a framework for real-time and high accuracy visual tracking. In: IEEE international conference on computer vision (ICCV), Venice, Italy, 22–29 October, 2017, pp. 4130–4144. IEEE.
5. Li B, Junjie Y, Wu W, et al. High performance visual tracking with Siamese region proposal network. In: IEEE/CVF conference on computer vision and pattern recognition (CVPR), Salt Lake City, UT, USA, 18–23 June, 2018, pp. 8971–8980. IEEE.
6. Zhu Z, Wang Q, Li B, et al. Distactor-aware Siamese networks for visual object tracking. In: European conference on computer vision (ECCV), Munich, Germany, 8–14 September, 2018, pp. 101–117.
7. Li B, Wu W, Wang Q, et al. SiamRPN++: evolution of Siamese visual tracking with very deep networks. In: IEEE/CVF conference on computer vision pattern recognition, Long Beach, CA, USA, 16–20 June 2019, pp. 4282–4291.
8. Zhang Z and Peng H. Deeper and wider Siamese networks for real-time visual tracking. In: IEEE/CVF conference on computer vision and pattern recognition (CVPR), Long Beach, CA, USA, 15 June, 2019, pp. 4586–4595.
9. Fan H and Ling H. Siamese cascaded region proposal networks for real-time visual tracking. In: IEEE/CVF conference on computer vision and pattern recognition (CVPR), Long Beach, CA, USA, 15 June 2019, pp. 7944–7953.
10. Bhat G, Danelljan M, Gool LV, et al. Learning discriminative model prediction for tracking. In: IEEE/CVF international conference on computer vision (ICCV), Seoul, South Korea, 22 April 2019, pp. 6181–6190.
11. Li P, Chen B, Ouyang W, et al. Gradnet: gradient-guided network for visual object tracking. In: IEEE/CVF
12. Choi J, Kwon J, and Lee K. Deep meta learning for real-time target-aware visual tracking. In: *IEEE/CVF international conference on computer vision (ICCV)*, Seoul, South Korea, 22 April 2019, pp. 6161–6170.

13. Zhang L, Gonzalez-Garcia A, Weijer J, et al. Learning the model update for Siamese trackers. In: *IEEE/CVF international conference on computer vision (ICCV)*, Seoul, South Korea, 22 April 2019, pp. 4009–4018.

14. Yan B, Zhao H, Wang D, et al. ‘Skimming-perusal’ tracking: a framework for real-time and robust long-term tracking. In: *IEEE/CVF international conference on computer vision (ICCV)*, Seoul, Korea (South), 22 April 2019, pp. 2385–2393.

15. Wang Q, Zhang L, Bertinetto L, et al. Fast online object tracking and segmentation: a unifying approach. In: *IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, Long Beach, CA, USA, 16–20 June 2019, pp. 1328–1338. IEEE.

16. Gao J, Wang X, Qing J, et al. Tracking-by-fusion via Gaussian process regression extended to transfer learning. *IEEE Trans Pattern Anal Mach Intellig (TPAMI)* 2018; 42(4): 939–955.

17. Li D, Yu Y, and Chen X. Object tracking framework with Siamese network and re-detection mechanism. *EURASIP J Wirel Commun Netw* 2019; 261: 1–14.

18. Luukezic A, Zajc LC, Vojir T, et al. Now you see me: evaluating performance in long-term visual tracking. Volume abs/1804.07056, 2018, pp. 294–301.

19. Kristan M, Leonardis A, Matas J, et al. The sixth visual object tracking VOT 2018 challenge results. In: *The European conference on computer vision (ECCV)* workshops, Munich, Germany, 8–14 September 2018, pp. 3–53.

20. Portaz M, Kohl M, Chevallet JP, et al. Object instance identification with fully convolutional networks. In: *IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, Las Vegas, NV, USA, 27–30 June, 2016. pp. 1420–1429.

21. Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks. In: *IEEE conference on computer vision and pattern recognition (CVPR)*, Portland, Oregon, USA, 23–28 June, 2013, pp. 2411–2418.

22. Wu Y, Lim J, and Yang M. Object tracking Benchmark. *IEEE Trans Pattern Anal Mach Intellig* 2015; 37(9): 1834–1848.

23. Zhang L, Gonzalez-Garcia A, Weijer J, et al. Learning the model update for Siamese trackers. In: *IEEE/CVF international conference on computer vision (ICCV)*, Seoul, South Korea, 22 April 2019, pp. 911–920.

24. Chen J, Yuan Z, Peng J, et al. DASNet: dual attentive fully convolutional Siamese networks for distractor-aware tracking. *Signal Process Image Commun* 2018; 64: 107–117. S0923596518302121.

25. Longchao Y and Peilin. Robust real-time visual object tracking via multi-scale fully convolutional Siamese networks. In: *Multimedia tools and applications*, Springer, 13 April 2018.

26. Held D, Thrun S, and Savarese S. Learning to track at 100 fps with deep regression networks. In: *European conference on computer vision (ECCV)*, Amsterdam, The Netherlands, 8–16 October, 2016, pp. 749–765.

27. Tao R, Gavves E, and Smeulders AM. Siamese instance search for tracking. In: *IEEE conference on computer vision and pattern recognition (CVPR)*, Las Vegas, NV, USA, 27–30 June, 2016, pp. 1420–1429.

28. Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks. In: *IEEE Trans Pattern Anal Mach Intellig*, 2015; 39(6): 1137–1149.

29. Kristan M, Leonardis A, Matas J, et al. The sixth visual object tracking VOT 2018 challenge results. In: *The European conference on computer vision (ECCV)* workshops, Munich, Germany, 8–14 September 2018, pp. 3–53.

30. Lukezic A, Zajc LC, Vojir T, et al. Now you see me: evaluating performance in long-term visual tracking. Volume abs/1804.07056, 2018, pp. 294–301.

31. Howard AG, Zhu M, Chen B, et al. MobileNets: efficient convolutional neural networks for mobile vision applications. Volume abs/1704.04861, 2017, pp. 1282–1291.

32. Ports M, Kohl M, Chevallet JP, et al. Object instance identification with fully convolutional networks. *Multimedia Tools Appl* 2019, 78(3): 2747–2764.

33. Portaz M, Kohl M, Chevallet JP, et al. Object instance identification with fully convolutional networks. *Multimedia Tools Appl* 2019, 78(3): 2747–2764.

34. Tao R, Gavves E, and Smeulders AM. Siamese instance search for tracking. In: *IEEE conference on computer vision and pattern recognition (CVPR)*, Las Vegas, NV, USA, 27–30 June, 2016, pp. 1420–1429.

35. Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks. In: *IEEE Trans Pattern Anal Mach Intellig*, 2015; 39(6): 1137–1149.

36. Dollar P and Zitnick CL. Fast edge detection using structured forests. *IEEE Trans Pattern Anal Mach Intellig*, 2014; 37(8): 1558–1570.

37. Chen J, Yuan Z, Peng J, et al. DASNet: dual attentive fully convolutional Siamese networks for change detection of high resolution satellite images. *IEEE J Select Topics Appl Earth Observe Remote Sensing* 2021; 14: 1194–1206.

38. Yang K, Song H, and Zhang K. Deeper Siamese network with multi-level feature fusion for real-time visual tracking. *Electr Lett* 2019, 55(13): 742–745.

39. Wu Y, Lim J, and Yang M. Online object tracking: a benchmark. In: *IEEE conference on computer vision and pattern recognition (CVPR)*, Portland, Oregon, USA, 23–28 June, 2013, pp. 2411–2418.
41. Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge. *Int J Comput Vision (IJCV)* 2015; 115: 211–252.

42. Woo S, Park J, Lee JY, et al. CBAM: convolutional block attention module. In: *European conference on computer vision (ECCV)*, Munich, Germany, 8–14 September 2018, pp. 3–19.

43. Lin TY, Maire M, Belongie S, et al. Microsoft COCO: common objects in context. In: *European conference on computer vision (ECCV)*, Zurich, Switzerland, 6–12 September 2014, pp. 740–755.

44. Real E, Shiens J, Mazzocchi S, et al. Vanhoucke, ‘YouTube-BoundingBoxes: a large high-precision human-annotated data set for object detection in video.’ In: *IEEE conference on computer vision and pattern recognition (CVPR)*, Honolulu, HI, USA, 21–26 July, 2017. pp. 7464–7473.