*electronics*

MDPI

*Article*

# Siamese Visual Tracking with Spatial-Channel Attention and Ranking Head Network

**Jianming Zhang *, Yifei Liang, Xiaoyi Huang, Li-Dan Kuang and Bin Zheng**

School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410076, China; lyfei@stu.csust.edu.cn (Y.L.); xiaoyihuang@stu.csust.edu.cn (X.H.); kuangld@csust.edu.cn (L.-D.K.); zhengbin@csust.edu.cn (B.Z.)

* Correspondence: jmzhang@csust.edu.cn

**Abstract:** Trackers based on the Siamese network have received much attention in recent years, owing to its remarkable performance, and the task of object tracking is to predict the location of the target in current frame. However, during the tracking process, distractors with similar appearances affect the judgment of the tracker and lead to tracking failure. In order to solve this problem, we propose a Siamese visual tracker with spatial-channel attention and a ranking head network. Firstly, we propose a Spatial Channel Attention Module, which fuses the features of the template and the search region by capturing both the spatial and the channel information simultaneously, allowing the tracker to recognize the target to be tracked from the background. Secondly, we design a ranking head network. By introducing joint ranking loss terms including classification ranking loss and confidence&IoU ranking loss, classification and regression branches are linked to refine the tracking results. Through the mutual guidance between the classification confidence score and IoU, a better positioning regression box is selected to improve the performance of the tracker. To better demonstrate that our proposed method is effective, we test the proposed tracker on the OTB100, VOT2016, VOT2018, UAV123, and GOT-10k testing datasets. On OTB100, the precision and success rate of our tracker are 0.925 and 0.700, respectively. Considering accuracy and speed, our method, overall, achieves state-of-the-art performance.

**Keywords:** object tracking; Siamese network; attention mechanism; head network



**Citation:** Zhang, J.; Liang, Y.; Huang, X.; Kuang, L.-D.; Zheng, B. Siamese Visual Tracking with Spatial-Channel Attention and Ranking Head Network. *Electronics* **2023**, *12*, 4351. https://doi.org/10.3390/electronics12204351

Academic Editors: Yue Wu, Beiji Zou, Xiaoyan Kui and Weixin Si

Received: 12 September 2023
Revised: 10 October 2023
Accepted: 18 October 2023
Published: 20 October 2023



## 1. Introduction

Object tracking is important work in the field of computer vision [1,2]. With the advancements of deep learning, object tracking has a wide range of applications in human–computer interaction [3], intelligent driving [4], video surveillance [5], virtual reality [6], and other fields [7,8]. Although object tracking has made significant progress at present, it still faces challenges from two aspects: (1) the target itself, such as deformation, scale variation, etc; (2) the external environment, such as occlusion, low resolution, illumination variation, etc.

Due to its outstanding performance, deep learning has gotten much attention in recent years. The object tracker based on deep learning has also become the mainstream tracker at present, among which, the Siamese tracker is one of the most famous tracking frameworks. Although Siamese trackers have achieved a large improvement in performance, there is still a problem, which is that Siamese trackers are trained offline, meaning that the templates of Siamese trackers cannot be updated online. Therefore, when the tracker encounters distractors that bear a resemblance to the tracked target, it may struggle to maintain accurate tracking, leading to a decrease in overall precision. However, introducing an attention mechanism into most Siamese trackers can effectively address these issues. Specifically, we propose the Spatial Channel Attention Module (SCAM), which can output an enhanced fusion feature map by combining spatial and channel information from template and search

region features. Finally, we feed the output of SCAM into the ranking head network for classification and regression to achieve more robust tracking.

In object tracking, the loss function can make the predicted bounding box as close as possible to the ground-truth bounding box. By computing the ratio of intersection and union between the predicted bounding box and the ground-truth bounding box, the optimization of the regression branch of the model can be achieved by Intersection over Union (IoU) loss [9]. Cross entropy loss [10] is the most commonly used loss term of the classification branch. It guides the predicted category of the classification branch to be the same as the true category. However, in the tracking process, there is a mismatch between the predicted value of the classification branch and the predicted value of the regression branch. For example, a positive sample after a huge deformation, which has a low IoU, may not be selected as a result. In fact, considering the predicted values of both classification and regression branches is conducive to the selection of the final results. Inspired by RBO [11], we introduce joint ranking losses and design the ranking head network. Specifically, the classification ranking loss allows us to better eliminate distractors and select the correct target. The confidence&IoU ranking loss links the classification and regression branches and selects the most suitable predicted bounding box as the result, improving the accuracy of our tracker.

In summary, we propose a Siamese visual tracker with spatial-channel attention and a ranking head network. The contributions of our work are shown below:

(1) We propose SCAM. Specifically, we first calculate the spatial similarity matrix between two feature maps, and then we use this similarity matrix to filter the information in the search region's feature. We concatenate the filtered and original search region's features, and send it to the channel attention module. This approach is used to achieve spatial channel attention. This enhances the representation of fusion features and makes them more discriminative.

(2) We design a ranking head network. Specifically, we introduce joint ranking loss terms into our approach. We use the mutual guidance of classification confidence score and IoU to select the final result, which can solve the problem of mismatch between the predicted values of the classification branch and the regression branch. Through the ranking head network, we can obtain more precise results and achieve a more robust tracking performance.

(3) We train our tracker on ImageNet DET [12], ImageNet VID [12], COCO [13], YouTube-BB [14], and GOT-10k training set [15]. Excellent results have been achieved on five challenging datasets, including the GOT-10k testing set, UAV123 [16], OTB100 [17], VOT2016 [18], and VOT2018 [19]. Our code and data are available at https://github.com/csust7zhangjm/lyf2021 (accessed on 9 October 2023).

SCAM can enhance feature representation by combining spatial and channel information. It can be applied as a module in other tasks based on deep learning. The classification ranking loss can optimize the predicted values of the classification branch, and the confidence&IoU ranking loss can link the classification branch and regression branch to output the most suitable result. Therefore, in some tasks with multiple branches, the predicted values of multiple branches can be considered simultaneously, making the results of related tasks more accurate. The following is the structure of our paper. Section 2 presents the related work of this research, Section 3 describes SCAM and the joint ranking loss terms, Section 4 is the experimental proof of the proposed method, and Section 5 gives our conclusions.

## 2. Related Work

We will review the work related to object tracking in the following three areas: single-object Siamese tracking, the attention mechanism, and the head network.

### *2.1. Single Object Siamese Tracking*

The Siamese object tracker has received extensive attention. SiamFC [20] was the pioneer of Siamese object tracking. It converted the traditional tracking process into a similarity matching problem, and determined the location of subsequent tracking by calculating the similarity. The success of SiamFC inspired many subsequent trackers, and they applied Siamese networks to object tracking to achieve the most advanced performance. SiamRPN [21] introduced the Region Proposal Network (RPN) [22], which, in turn, transformed the similarity matching problem into an independent classification and regression problem, where classification and regression branches serve different purposes, with the former used for distinguishing the background and foreground, and the latter for locating bounding boxes. In addition, trackers such as DaSiamRPN [23] and SiamRPN++ [24] improved SiamRPN to achieve better performance, but the above RPN-based algorithms designed multi-scale anchor boxes to obtain accurate bounding boxes, which would undoubtedly consume a large amount of time and cost and bring a large computational burden.

In 2019, the anchor-free Siamese tracking algorithm was proposed. Unlike the anchor-based algorithm, the anchor-free algorithm directly calculated the position of the target. Since no anchor box was introduced in the process of determining the target position, the anchor-free Siamese object tracking algorithm could reduce the computational burden and improve the tracking speed for trackers. SiamBAN [25] and SiamCAR [26] were among the most outstanding algorithms, and their performance reached the advanced level at that time. Overall, compared to RPN-based algorithms, anchor-free tracking algorithms could reduce the computational burden and, to some extent, reduce computational time. Anchor-free Siamese object tracking is a trend for the future.

### *2.2. Attention Mechanism*

An important module in deep learning is the attention mechanism module, which is plug-and-play and very powerful [27]. SENet [28] established interdependence between channels by compressing feature maps. STMTrack [29] proposed a pixel-level correlation method; through the matrix reshape and matrix multiplication operation between the template and the search region's feature, we could obtain the weight information of the spatial position, and then the weight information was weighted with the template's feature to realize the spatial attention operation. CBAM [30] was a lightweight attention mechanism module that could perform attention operations on spatial information and channel information, making the model more concerned with the object itself. NLNet [31] was also a common attention module that improved the model's understanding of global contextual information by capturing global dependencies.

The recently popular Transformer [32] was a powerful attention mechanism module, including a self-attention mechanism and cross-attention mechanism. Its application had enabled many algorithms to achieve state-of-the-art performance. Overall, the attention mechanism was applied to many deep learning tasks, and its introduction could undoubtedly bring performance improvements. However, we should also note that the introduction of attention mechanisms may also result in elevated computational complexity, and make the algorithm fail to achieve real-time performance.

### *2.3. Head Network*

The head network is a crucial component in object tracking. Its function is to predict the location of the target based on the input information. Nowadays, most of the mainstream trackers rely on anchor-free algorithms, which directly calculate the position of the bounding box. This method effectively reduces computation and boosts tracking performance. CornerNet [33] was an object detection method that used the two corner points of the bounding box as key points to achieve accurate object detection. ExtremeNet [34], on the other hand, was also an object detection method that predicted the center point and the four corner points of all objects simultaneously.

The Head network can be further divided into a classification branch and a regression branch. The function of the classification is to distinguish the foreground from the background, and the purpose of regression is to locate the bounding box's position. Some current trackers, such as SiamRPN, SiamFC++ [35], etc., determined the position of the bounding box through classification and regression branches in the head network. However, in the inference phase, mismatches between the predicted values of the classification and regression branches could interfere with the selection of results. PrDiMP [36] used probabilistic methods to model the labels of the targets, but this approach could worsen the discrepancy between classification and regression even more [4]. Therefore, if the mismatch between these two branches can be successfully resolved, our tracker's robustness can be further improved.

## 3. Methods

### *3.1. Overview*

Figure 1 shows our overall tracking process. In the feature extraction stage, ResNet-50 [37] was chosen as our feature extraction network. SiamBAN selects the feature maps of C3, C4, and C5 layers outputted by the feature extraction network, but we observe that the weight values of C5 in the classification and regression network are very small, and as such, we only used C3 and C4. The utilization of features from different layers can make varying impacts on the tracking process. The features extracted by the shallow layer contain richer spatial information, whereas the features extracted by the deep layer contain richer semantic information. Therefore, we select the output features of C3 and C4 for feature fusion, and our approach takes both spatial information and semantic information into consideration.

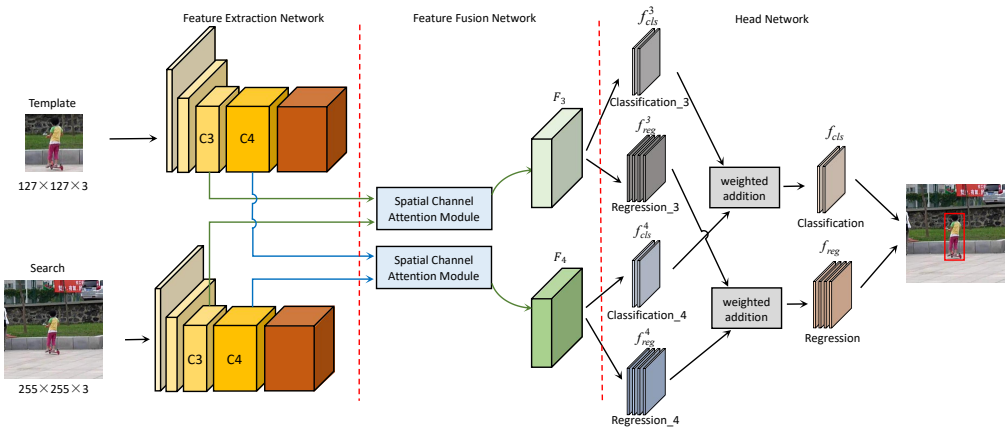


**Figure 1.** The structure of our proposed tracking algorithm. Our tracker consists of a feature extraction network, feature fusion network, and head network.

In the feature fusion network, our SCAM adopts pixel-wise correlation [29] instead of depth-wise cross-correlation, and utilizes a channel attention mechanism to enhance our feature's expression. In the classification and regression network, classification feature maps and regression feature maps from different layers are added with weights, and fed into our improved loss function, which further improves the accuracy of the tracker results.

### *3.2. Spatial Channel Attention Module*

The attention mechanism is helpful for most visual tasks. The inputs to SCAM are the feature maps output by feature extraction network. As shown in Figure 2, let $Z \in \mathbb{R}^{C\times h\times w}$ be the template's feature map, and $X \in \mathbb{R}^{C\times H\times W}$ be the search region's feature map. We use the Reshape operation for the template's feature $Z$ and the search region's feature $X$ to obtain feature maps $Z^1$, $Z^2$, and $X^1$, as shown in Equation (1):

$$\begin{aligned} Z^1 &= \text{Reshape1}(Z), \\ Z^2 &= \text{Reshape2}(Z), \\ X^1 &= \text{Reshape3}(X), \end{aligned} \tag{1}$$

where, $Z^1 \in \mathbb{R}^{C \times hw}$, $Z^2 \in \mathbb{R}^{hw \times C}$, and $X^1 \in \mathbb{R}^{C \times HW}$. The function of Reshape$i(.)$, $i \in \{1, 2, 3\}$, is to change the dimensions of $Z$ and $X$. Here, we utilize Reshape$i(.)$ to reduce the dimensions of $Z$ and $X$, so that their dimensions change from three dimensions to two dimensions.

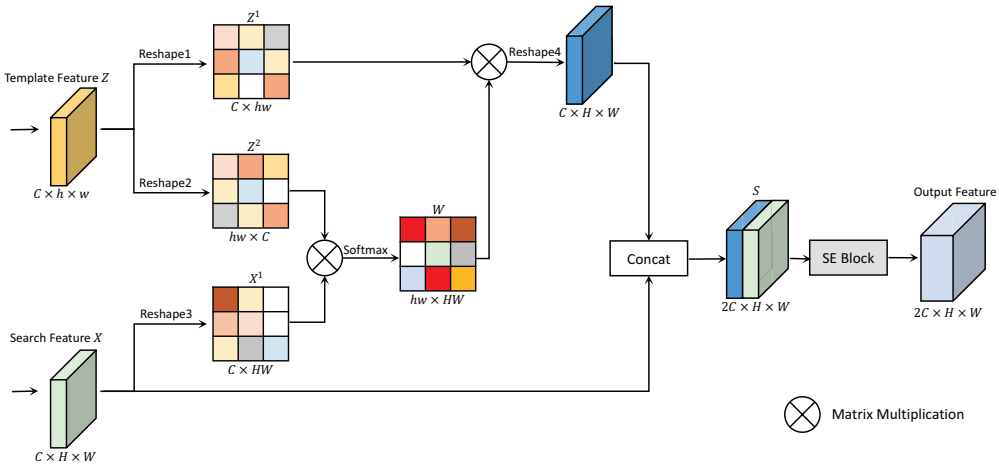


**Figure 2.** Our proposed spatial channel attention module. It consists of a spatial attention module and a channel attention module, taking template feature maps and search feature maps as input to perform attention operations.

Next, the similarity matrix $W \in \mathbb{R}^{hw \times HW}$ is obtained through matrix multiplication, and the matrix entry $W_{ij}$ is calculated according to Equation (2):

$$W_{ij} = \frac{\exp\left[\left(Z^2_{i.} \odot X^1_{.j}\right)/\sqrt{C}\right]}{\sum_{\forall k} \exp\left[\left(Z^2_{k.} \odot X^1_{.j}\right)/\sqrt{C}\right]}, \tag{2}$$

where $Z^2_{i.}$ represents $i$-th row on $Z^2$, $X^1_{.j}$ represents $j$-th column on $X^1$, $\odot$ denotes the vector dot-product operation, and $C$ represents the number of channels.

By concatenating the filtered and original search region's feature in the channel dimension, we can obtain the fusion feature $S$, as shown in Equation (3):

$$S = \text{Concat}\Big(\text{Reshape4}\Big(\Big(Z^1 \otimes W\Big)\Big), X\Big), \tag{3}$$

where Concat $(\cdot, \cdot)$ represents the concatenate operation.

The input of SCAM are C3 and C4's output features from the feature extraction network, and the output is the fusion feature $F_i$. Its operation is shown in Equation (4):

$$F_i = \text{SCAM}(\text{Z}_\text{i}, \text{X}_\text{i}), \tag{4}$$

where SCAM$(\cdot, \cdot)$ is the module we proposed, $Z_i$ represents the template's feature of the $i$-th layer, and $X_i$ represents the search region's feature of the $i$-th layer, $i \in \{3, 4\}$.

Through the classification and regression network, we get fusion feature $F_i$'s classification feature map $f^i_{cls}$ and regression feature map $f^i_{reg}$, and we perform weighted addition operations on the results from different layers, as shown in Equation (5):

$$\begin{aligned} f_{cls} &= \sum_{i=3}^{4} \beta^i_{cls} f^i_{cls}, \\ f_{reg} &= \sum_{i=3}^{4} \beta^i_{reg} f^i_{reg}, \end{aligned} \tag{5}$$

where $f_{cls}$ and $f_{reg}$ are the final classification and regression feature maps, respectively. In addition, $\beta^i_{cls}$ and $\beta^i_{reg}$ are the weights obtained after optimization together with the network.

### *3.3. Ranking Head Network*

**Regression loss**. For every position within the regression's feature map $f_{reg}$, we can find the corresponding region in the search region's feature map $X$. Our method can predict the bounding box at each position through regression. Figure 3 shows the loss function we used. Our regression loss is shown in Equation (6):

$$L_{reg} = 1 - \frac{y_{label\_reg} \cap y_{reg}}{y_{label\_reg} \cup y_{reg}}, \tag{6}$$

where $y_{label\_reg}$ represents the ground-truth bounding box and $y_{reg}$ represents the predicted bounding box.

**Classification loss**. In the classification branch, we usually use cross entropy loss as the classification loss function in the head network of our tracker. Our classification loss is shown in Equation (7):

$$\begin{aligned} L_{cls} = \lambda_{pos} \frac{1}{N_{pos}} \sum_{i \in P_{pos}} CE\left(y^i_{cls}, y_{label\_cls}\right) \\ + \lambda_{neg} \frac{1}{N_{neg}} \sum_{i \in P_{neg}} CE\left(y^i_{cls}, y_{label\_cls}\right), \end{aligned} \tag{7}$$

where $CE$ represents the cross entropy loss, and $N_{pos}$ and $N_{neg}$ represent the number of positive samples and negative samples, respectively. $P_{pos}$ and $P_{neg}$ represent the positive and negative sample sets, respectively. $y_{label\_cls}$ represents the classification label, and we use $y_{cls}$ to denote the output of the classification branch. $\lambda_{pos}$ and $\lambda_{neg}$ are control parameters; we set them to 0.5 here.

**Classification ranking loss**. In our experiment, we will filter out negative samples with lower classification confidence scores, but some negative samples do have higher classification confidence scores. We refer to these negative samples as hard negative samples. These hard negative samples may have an appearance that closely resembles the tracking target. In the process of tracking, these hard negative samples are difficult to distinguish when performing classification. In order to give our tracker a stronger ability to distinguish hard negative samples, we introduce classification ranking loss as Equation (8), and our goal is to minimize the influence of these hard negative samples on the tracking results:

$$L_{rk-cls} = \frac{1}{\gamma} \log\left(1 + \exp\left(\left(E_{neg} - E_{pos} + \delta\right) \times \gamma\right)\right), \tag{8}$$

where $E_{neg}$ represents the expectation of negative samples, and it is obtained by using weighted addition method for the classification confidence score of each negative sample [11]. $E_{pos}$ is as well. $\gamma$ and $\delta$ are two parameters; we set them here to 4 and 0.5, respectively.

**Confidence & IoU ranking loss**. The branches of the tracker are relatively independent. This sometimes leads to a mismatch between the classification and the regres-

sion branches. In order to address this problem, as shown in Figure 4, we introduce the confidence&IoU ranking loss. The positive sample that ranks high in both classification confidence score and IoU is selected as the final result. We hope to improve the accuracy of our tracking results in this way. The loss function is shown in Equation (9):

$$\begin{aligned} L_{rk-ci} = & \frac{1}{N_{pos}} \sum_{i,j \in P_{pos}, p_i^{IoU} > p_j^{IoU}} \exp\left(-\alpha\left(p_i^{conf} - p_j^{conf}\right)\right) \\ & + \frac{1}{N_{pos}} \sum_{i,j \in P_{pos}, p_i^{conf} > p_j^{conf}} \exp\left(-\alpha\left(p_i^{IoU} - p_j^{IoU}\right)\right), \end{aligned} \tag{9}$$

where $p_i^{conf}$ represents the classification confidence score of each positive sample, and $p_i^{IoU}$ represents the predict bounding box of each positive sample. $\alpha$ is a parameter; we set it to 3.

Finally, our overall loss function is shown in Equation (10):

$$L_{\text{total}} = L_{reg} + L_{cls} + \lambda_{rk-cls} L_{rk-cls} + \lambda_{rk-ci} L_{rk-ci}. \tag{10}$$

where $\lambda_{rk-cls}$ is set to 0.5 and $\lambda_{rk-ci}$ is set to 0.25.

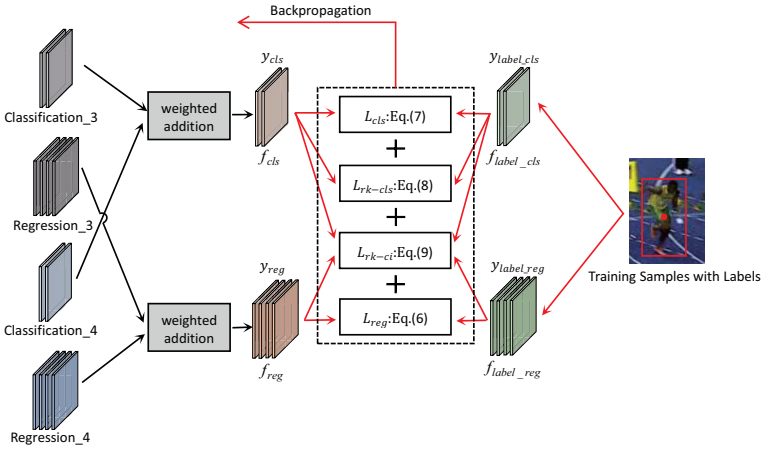


**Figure 3.** Our total loss terms. The total loss terms consists of four parts, including regression loss, confidence & IoU ranking loss, classfication ranking loss, and classfication loss.

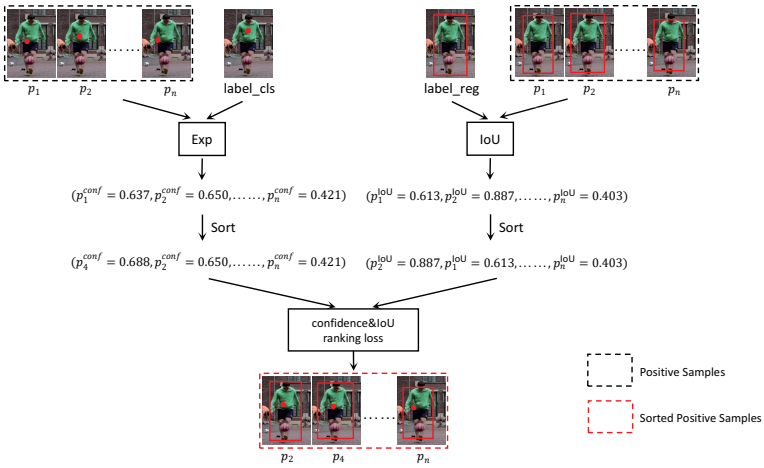


**Figure 4.** An illustration of confidence & IoU ranking loss. We hope that through the interconnection between classification and regression branches, those samples with a high classification confidence score and high IoU have a higher ranking.

## 4. Experiments

### *4.1. Implementation Details*

Our algorithm is based on CUDA 10.1, Pytorch 1.3.1, and Python 3.7, implemented on three 2080 Ti GPUs. ResNet-50 was chosen as our feature extraction network, which was trained on ImageNet and preserves its weights. We use images with a size of 127 × 127 pixels as input into our template feature extraction network and images with a size of 255 × 255 pixels as input into our search region feature extraction network. We train our tracker using five training datasets, including YouTube-BB, COCO, ImageNet DET, ImageNet VID, and GOT-10k training set. Also, we utilize the stochastic gradient descent algorithm to optimize our model. Considering our hardware environment, we set the batch size to 28. We trained for a total of 20 epochs. We used warming learning rates from 0.001 to 0.005 for the first epoch to the fifth epoch. When training the first epoch to the tenth epoch, we train only the classification and the regression branches in the head network. In training the eleventh epoch to the twentieth epoch, we make it possible for the backbone network to also be included in the training by fine-tuning the weights of the backbone network. In the testing phase, we select the first frame as our template and then perform similarity matching on the subsequent video sequences. We evaluate the performance of our proposed algorithm on OTB100, UAV123, VOT2016, VOT2018, and the GOT-10k testing set. We achieved satisfactory results on these datasets.

### *4.2. Results on OTB100 Benchmark*

There are 100 fully annotated sequences in OTB100 dataset. Each sequence in the dataset is annotated with challenges, including occlusion, deformation, motion blur, etc. On OTB100, we use the precision and the success rate to test the performance of our tracker. On OTB100, we compare our tracker with nine excellent trackers such as SiamBAN [25], SiamCAR [26], SiamFC++ [35], DaSiamRPN [23], SRDCF [38], SiamFC [20], CFNet [39], SiamRPN++ [24], and SiamDW [40].

We can see from Figure 5 that our tracker achieves satisfactory performance on the OTB100 dataset. Compared with SiamDW, SiamCAR, SiamBAN, and SiamRPN++, our tracker improves the accuracy by 0.3%, 1.6%, 1.6%, and 2.2%, respectively. At the same time, our trackers have the highest success rate, reaching up to 0.700. Figures 6 and 7 show the names of the 11 challenges and the results of our tracker on 11 challenges.

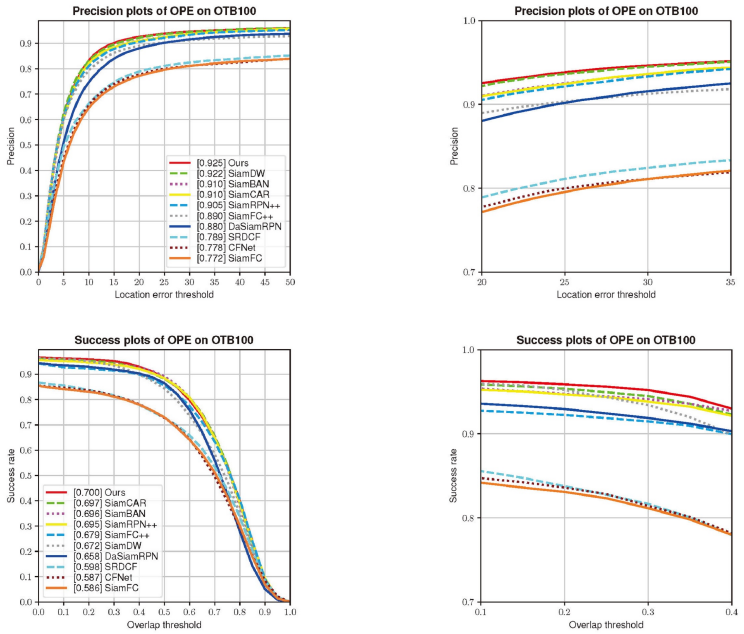


**Figure 5.** The left shows the precision and success rates of our tracker and the comparison tracker on the OTB100. The right image is a zoomed localized region of the left image.

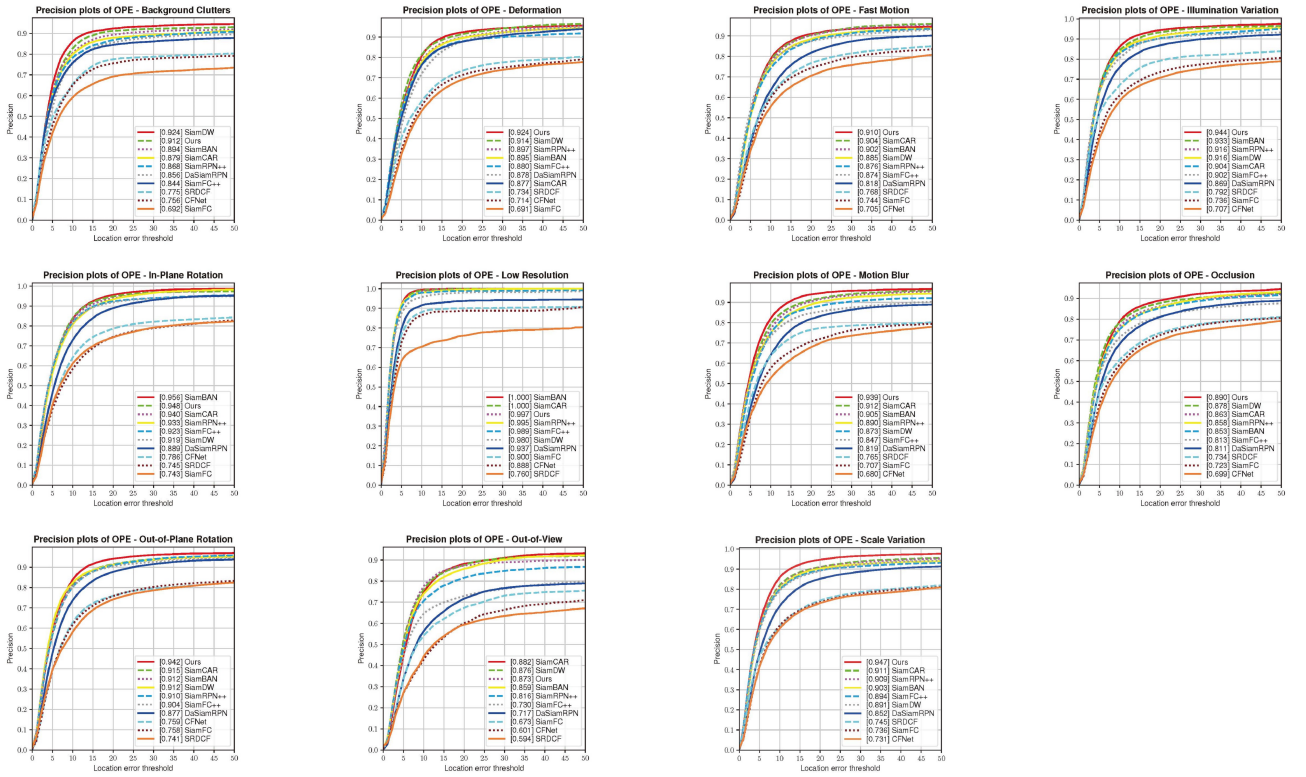


**Figure 6.** The precision of our tracker and comparison trackers on the 11 challenges of the OTB100.

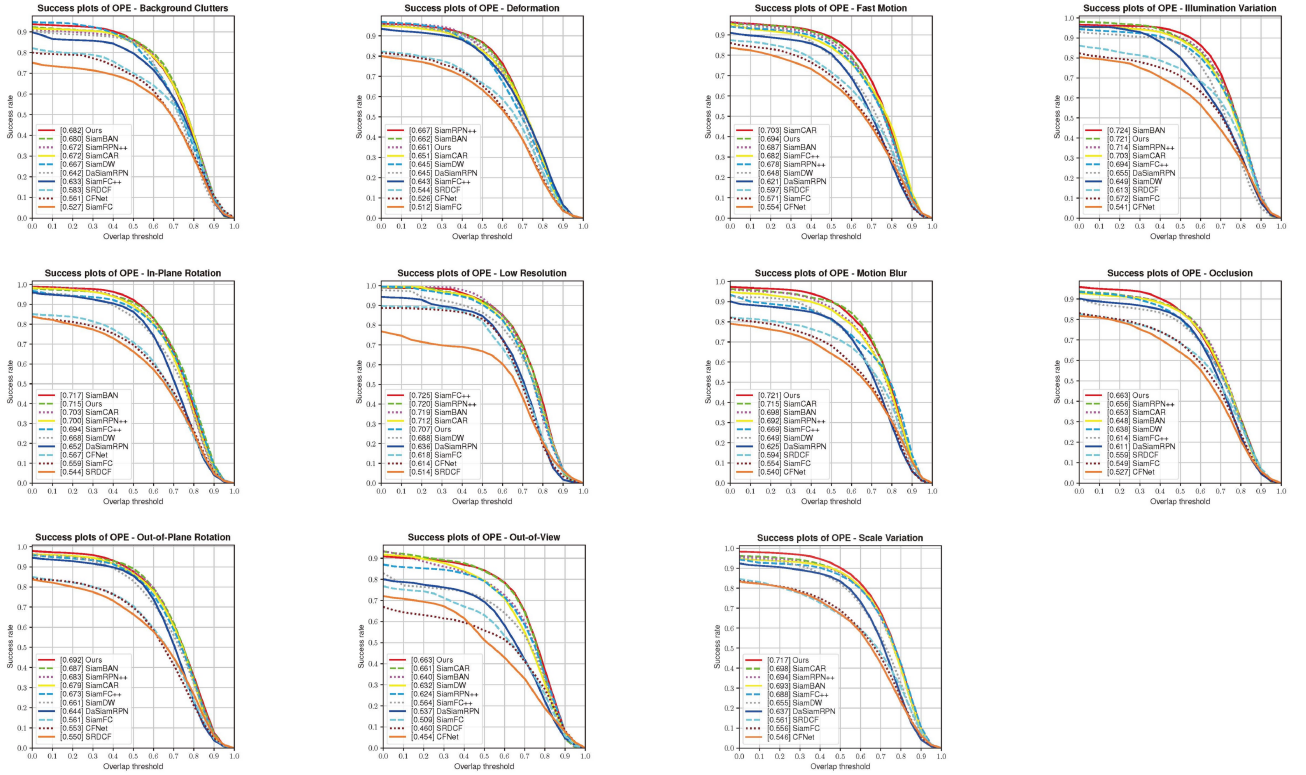


**Figure 7.** The success rate of our tracker and comparison trackers on the 11 challenges of the OTB100.

### *4.3. Results on UAV123 Benchmark*

Another excellent dataset in the field of object tracking is UAV123. There are 123 video sequences included in UAV123, all of which are captured from a low-altitude aerial perspective. Among them, the 1st video to the 103rd video are stable, the 104th video to the 115th video are unstable, and the 116th video to the 123rd video are synthetic. On UAV123, we compare our tracker with eight excellent trackers, SiamCAR, ECO [41], ECO-HC [41], SiamRPN, SiamRPN++, STRCF [42], TADT [43], and DaSiamRPN. Figure 8 clearly demonstrates that compared to SiamRPN++, SiamCAR, and DaSiamRPN, our tracker improves the precision by 5.3%, 3.6%, and 7.8%, respectively. In addition, our tracker has achieved a success rate of 0.631, which is 1.2% higher than SiamCAR. Both are the best among the eight compared trackers. Figures 9 and 10 show the names of the 12 challenges and the results of our tracker on 12 challenges.

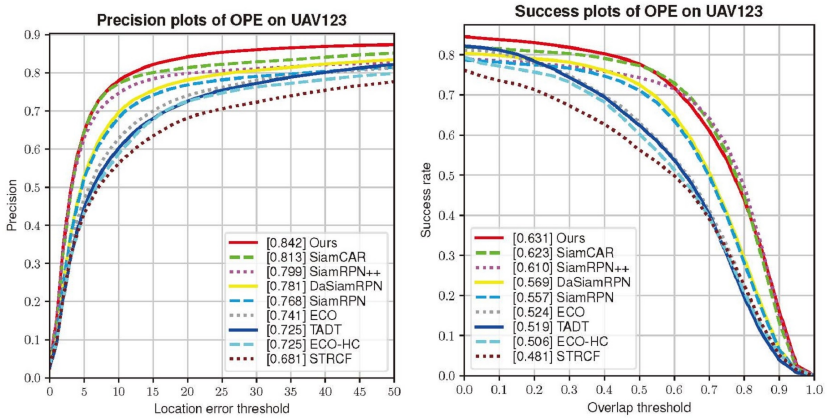


**Figure 8.** The precision and success rates of our tracker and comparison trackers on UAV123.

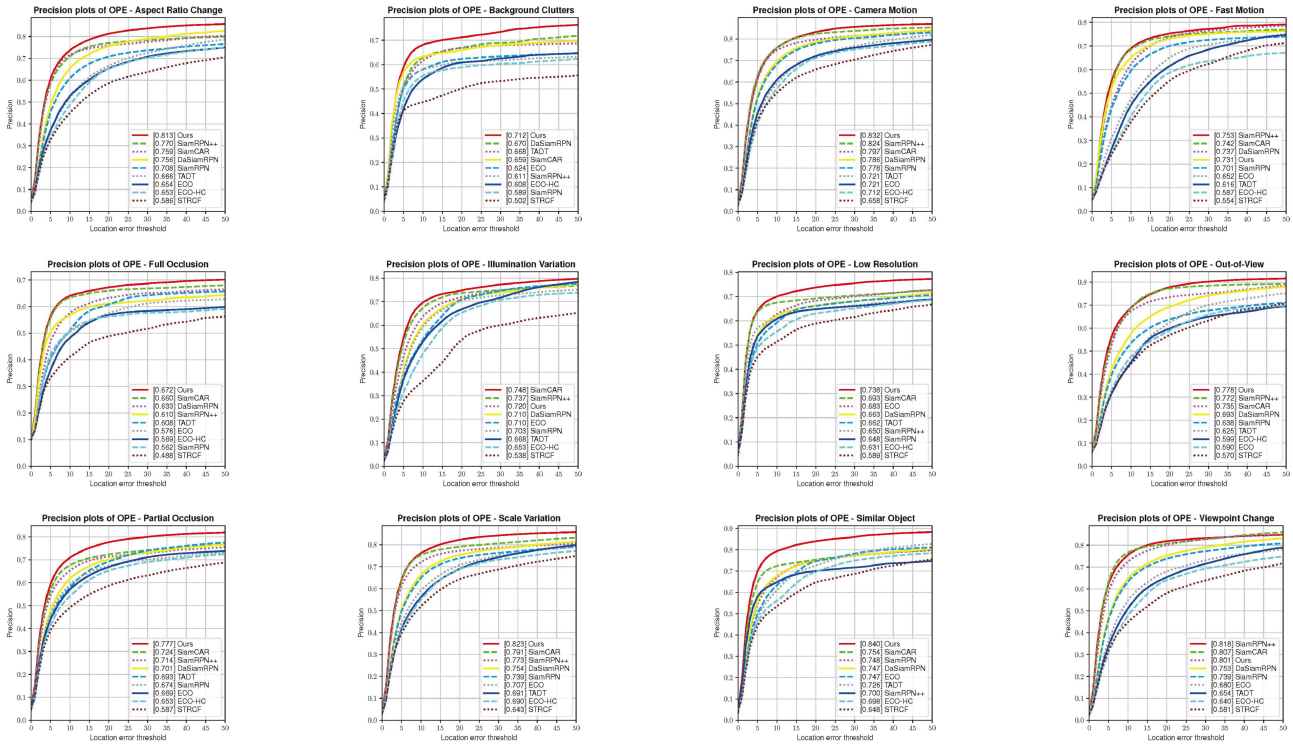


**Figure 9.** The precision of our tracker and comparison trackers on the 12 challenges of the UAV123.

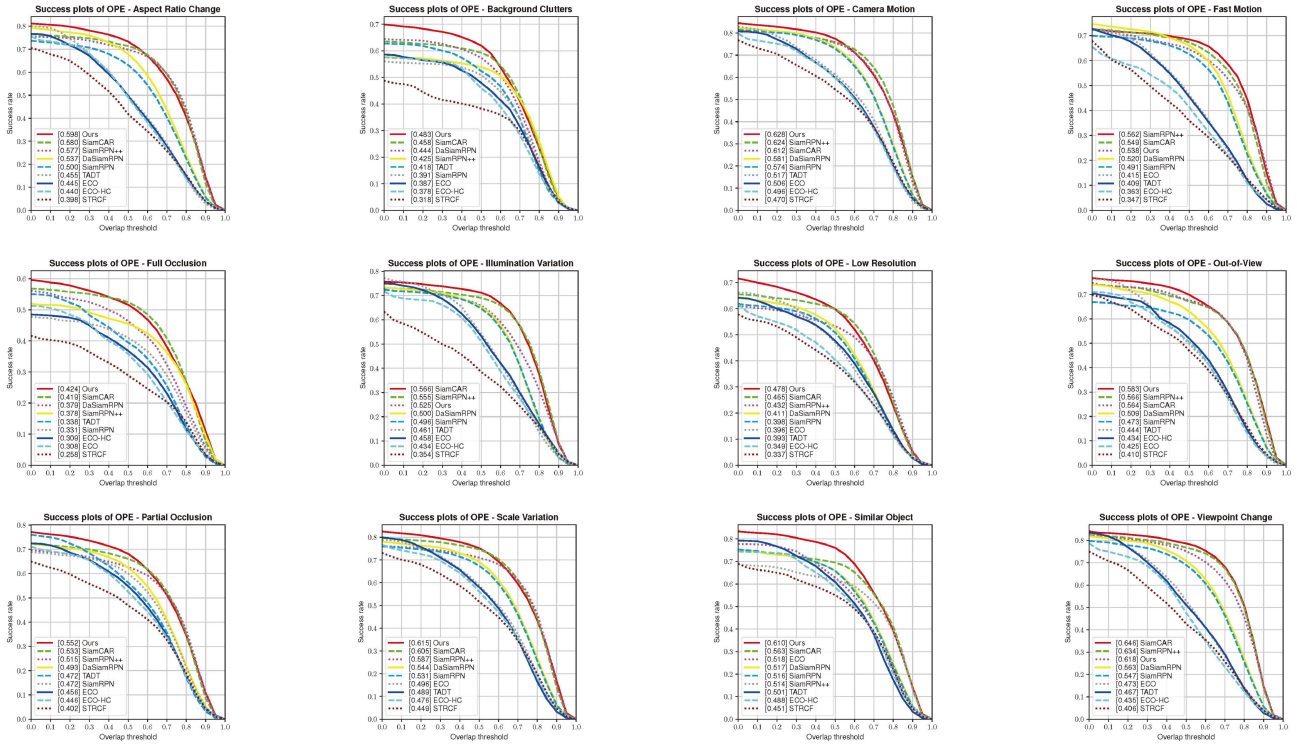


**Figure 10.** The success rate of our tracker and comparison trackers on the 12 challenges of the UAV123.

### *4.4. Results on VOT2016 Benchmark*

The VOT dataset holds a position of authority within the area of object tracking. There are 60 types of videos in the VOT2016 dataset. Before the emergence of VOT, the mainstream tracking strategy was to use the first frame of the video sequence to initialize the tracker, and then until the end of the video. However, due to the presence of distracting objects, the tracker is likely to lose its localization of the target. Therefore, when the tracker loses the target, the VOT is delayed back 5 frames. In VOT2016, when we evaluate the performance of a tracker, we use the expected average overlap (EAO). In this way, we can balance the accuracy and robustness of the tracker at the same time. We compare our tracker with other advanced trackers, including SiamRPN, DaSiamRPN, ECO-HC, MCCT-H [44], SiamRPN++, SiamMask [45], SiamR-CNN [46], ECO, and MCCT [44]. The data in Table 1 clearly show that compared with SiamR-CNN, SiamRPN++, and SiamMask, our tracker improves by 15.2%, 21.2%, and 24.7% in EAO, and our robustness is the lowest among these algorithms at only 0.084, which is enough to prove that our tracker can achieve stable tracking in complex environments. Figure 11 shows the names of the six challenges and the results of our tracker on six challenges.

**Table 1.** Evaluation of our tracker and other trackers on VOT2016. E is EAO, A represents accuracy, R denotes robustness. Higher values of EAO and A represent greater accuracy, so we use ↑. The smaller the R value, the greater the immunity to interference, so we use ↓. The three best results are highlighted in bold, bold and italic, and italic.

| | MCCT-H | ECO-HC | SiamRPN | ECO | MCCT | DaSiam-RPN | SiamMask | SiamRPN++ | SiamR-CNN | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| E↑ | 0.299 | 0.322 | 0.337 | 0.374 | 0.393 | 0.401 | 0.425 | *0.437* | ***0.460*** | **0.530** |
| A↑ | 0.570 | 0.542 | 0.578 | 0.555 | 0.579 | 0.609 | *0.634* | ***0.644*** | **0.645** | *0.634* |
| R↓ | 0.331 | 0.303 | 0.312 | 0.200 | *0.186* | 0.224 | 0.214 | 0.219 | ***0.172*** | **0.084** |

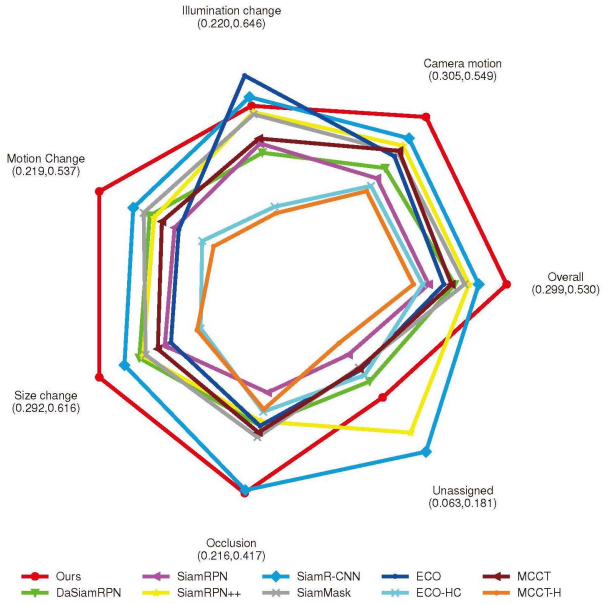


**Figure 11.** Evaluation of our tracker and comparison trackers on the six challenges of the VOT2016.

### *4.5. Results on VOT2018 Benchmark*

VOT2018 is one of VOT's latest object tracking datasets. Compared to VOT2016, the video sequences in VOT2018 exhibit a higher level of complexity and contain more difficult challenges. VOT2018 and VOT2016 use the same evaluation criteria, and we conducted comparative experiments on VOT2018, comparing state-of-the-art trackers such as SiamCAR, SiamMask, SiamFC++, SiamKPN [47], SiamRPN++, SiamR-CNN, SiamRPN, DaSiamRPN, and ATOM [48]. As shown in Table 2, compared with SiamCAR, SiamKPN, and SiamFC++, our tracker improves by 1.7%, 0.5%, and 0.9% in the average overlap rate. In terms of robustness, our tracker performs the best among the eight compared trackers. The results on the six challenges are shown in Figure 12.

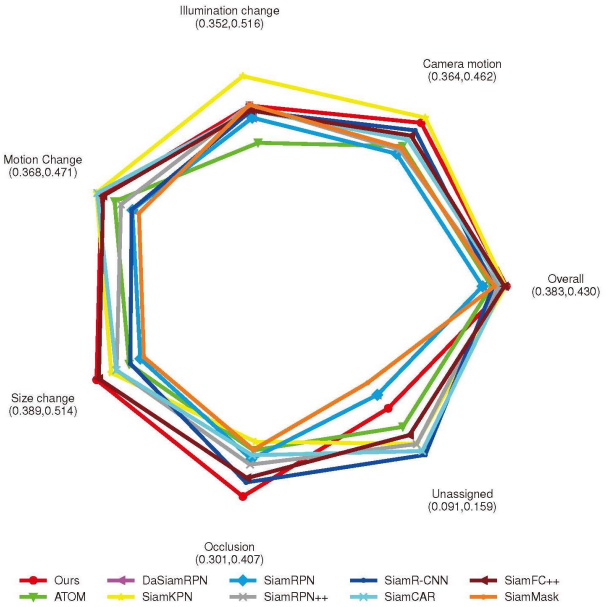


**Figure 12.** Evaluation of our tracker and comparison trackers on the six challenges of the VOT2018.

**Table 2.** Evaluation of our tracker and other trackers on VOT2018. E is EAO, A represents accuracy, R denotes robustness. Higher values of EAO and A represent greater accuracy, so we use ↑. The smaller the R value, the greater the immunity to interference, so we use ↓. The three best results are highlighted in bold, bold and italic, and italic.

| | DaSiam-RPN | ATOM | SiamR-CNN | SiamMask | Siam-RPN++ | SiamCAR | SiamFC++ | SiamKPN | SiamRPN | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| E↑ | 0.383 | 0.400 | 0.405 | 0.406 | 0.415 | 0.423 | *0.426* | ***0.428*** | 0.383 | **0.430** |
| A↑ | 0.586 | 0.590 | **0.612** | *0.598* | ***0.601*** | 0.578 | 0.583 | 0.596 | 0.586 | 0.587 |
| R↓ | 0.276 | 0.203 | 0.220 | 0.248 | 0.234 | 0.197 | ***0.173*** | *0.187* | 0.276 | **0.164** |

### *4.6. Results on GOT-10k Benchmark*

GOT-10k is a large dataset which was published by the Chinese Academy of Sciences that contains numerous video sequences. There are 560 kinds of moving objects in the training set, with 87 different motion patterns between them. The total video sequence exceeds 10,000 and there are more than 1,500,000 manually labeled bounding boxes. The GOT-10k testing set has a total of 420 video sequences. These video sequences contain a total of 31 different motion categories and 84 different object categories. There are three indicators in the GOT-10k testing set, including average overlap (AO), success rate ($SR_{0.5}$, $SR_{0.75}$), and frames per second (FPS). Table 3 clearly shows the results of our tracker compared to SiamCAR, SiamMask, DaSiamRPN, SiamDW, SiamRPN, SiamRPN++, and ATOM.

**Table 3.** Evaluation of our tracker and other trackers on GOT-10k. ↑ indicates that a higher value is better. The three best results are highlighted in bold, bold and italic, and italic.

| | SiamDW | DaSiamRPN | SiamRPN++ | SiamCAR | ATOM | SiamRPN | SiamMask | Ours |
|---|---|---|---|---|---|---|---|---|
| AO↑ | 0.416 | 0.444 | 0.517 | ***0.569*** | *0.556* | 0.483 | 0.453 | **0.618** |
| $SR_{0.5}$↑ | 0.475 | 0.536 | 0.616 | ***0.670*** | *0.634* | 0.581 | 0.550 | **0.722** |
| $SR_{0.75}$↑ | 0.144 | 0.220 | 0.325 | ***0.415*** | *0.402* | 0.270 | 0.248 | **0.491** |
| FPS↑ | *66.67* | **134.40** | 3.18 | 17.21 | 20.71 | ***97.55*** | 15.37 | 50.91 |

### *4.7. Ablation Experiment*

We performed five sets of ablation experiments on UAV123. We use the backbone of Figure 1 to replace the backbone in SiamBAN, and make the modified SiamBAN as our baseline. As shown in Table 4, our method has increases in the success rate and precision by 2.3% and 3.7%, respectively. In order to verify the ability of our proposed SCAM to distinguish similar objects, we evaluated the SOB challenge on UAV123, as shown in Figure 13. The experiments prove that our proposed SCAM has good discrimination ability for similar objects.

**Table 4.** We verified the validity of each part on UAV123. $\Delta s$ represents increases in success plot and $\Delta p$ represents increases in precision plot.

| Method | Success | $\Delta s$ | Precision | $\Delta p$ |
|---|---|---|---|---|
| baseline | 0.608 | – | 0.805 | – |
| baseline + SCAM | 0.620 | +1.2% | 0.817 | +1.2% |
| baseline + SCAM + $L_{rk-cls}$ | 0.624 | +1.6% | 0.823 | +1.8% |
| baseline + SCAM + $L_{rk-ci}$ | 0.622 | +1.4% | 0.827 | +2.2% |
| baseline + SCAM + $L_{rk-cls}$ + $L_{rk-ci}$ | 0.631 | +2.3% | 0.842 | +3.7% |

### *4.8. Real-Time Analysis*

We conducted a speed test on three 2080Ti GPUs. Table 5 shows that our tracker achieves real-time tracking on five datasets. We also measured the speed of our tracker and other advanced trackers on GOT-10k. As shown in Table 3, the speed of our tracker reached 50.91 FPS, achieving a compromise between accuracy and real-time performance.

**Table 5.** The speed of our tracker on different datasets. ↑ indicates that a higher value is better.

| | OTB100 | UAV123 | VOT2016 | VOT2018 | GOT-10k |
|---|---|---|---|---|---|
| FPS↑ | 51.1 | 63.1 | 48.5 | 59.4 | 50.91 |

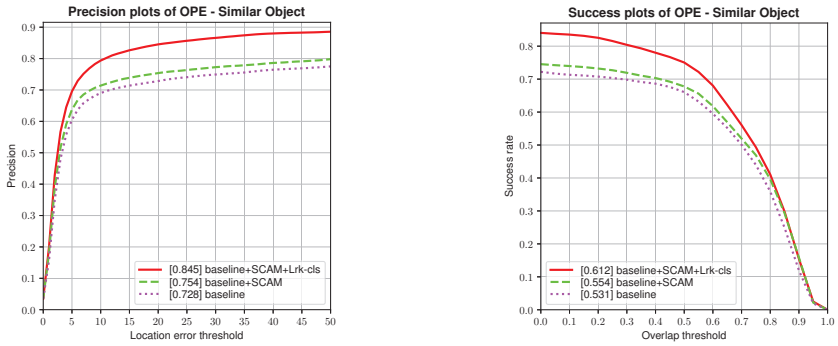


**Figure 13.** Discrimination ability for similar objects on UAV123.

### *4.9. Experimental Summary*

In Section 4, we validate our proposed algorithm from four different perspectives. Specifically, in Section 4.1, we introduce the corresponding training strategies and the experimental environment. From Section 4.2 to Section 4.6, we evaluate the effectiveness of our tracker on five datasets: OTB100, UAV123, VOT2016, VOT2018, and the GOT-10k testing set. In comparison to mainstream trackers, our tracker achieves an excellent performance. In Section 4.7, we perform ablation experiments on our proposed SCAM and the introduced joint ranking loss terms on the UAV123 dataset, and the experimental results prove that the method we propose is effective. Finally, in Section 4.8, we test the speed of our tracker on three 2080Ti GPUs, and the results emphasize its real-time capability. The visualization of our algorithm is shown in Figure 14.

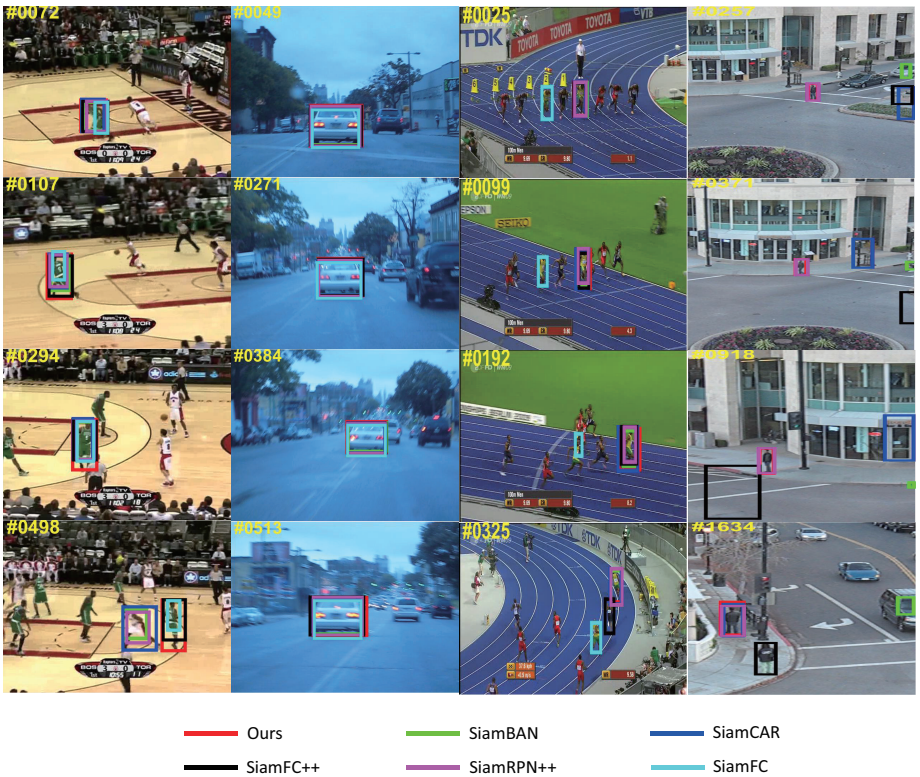


**Figure 14.** Visualization results of our tracker and other comparative trackers in four video sequences of the OTB100.

When evaluating the performance of trackers on different datasets, differences in tracker performance may be due to different proportions of video sequences with the same challenge. For example, OTB100 has a large number of video sequences containing an occlusion challenge, so when a tracker can solve the problem of target occlusion, it will have a high success rate on OTB100.

## 5. Conclusions

Overall, we proposed a Siamese visual tracker with spatial-channel attention and a ranking head network, and trained our tracker using five authoritative datasets. Our proposed SCAM can not only fuse template's feature and search region's feature, but can also establish long-term dependencies between spatial position information and channel information. The introduced confidence&IoU ranking loss and classification ranking loss loss can link the classification and the regression branches, use the classification confidence score and IoU to guide the selection of the final result, and improve the performance of the tracker. Overall, our proposed SCAM can combine information from both the spatial and channel to achieve feature enhancement. The joint ranking loss terms we introduce can consider both classification confidence score and IoU to output the most suitable result.

In addition, we also conducted a series of experiments on OTB100, VOT2016, VOT2018, UAV123, and GOT-10k. On the OTB100 dataset, our tracker achieved a success rate of 0.700, which is the best of all trackers. Our tracker achieved a precision of 0.842 on UAV123. The EAO of our tracker is 0.530 on VOT2016 and 0.430 on VOT2018, respectively, which are the best among all trackers. Our tracker's AO reached 0.618 on GOT-10k. Therefore, our tracker achieves decent performance on these datasets.

However, our tracker's performance is not satisfactory when the video has a low resolution or the target is out of view. We are considering introducing an online update module in future work; the introduction of an online update module could improve the stability of our tracker in the face of complex environments. These issues are the direction of our future efforts.

**Author Contributions:** Conceptualization, J.Z.; methodology, Y.L.; software, Y.L.; validation, L.-D.K.; formal analysis, J.Z. and Y.L.; investigation, B.Z.; data curation, X.H.; writing—original draft preparation, Y.L.; writing—review and editing, J.Z. and L.-D.K.; visualization, X.H.; supervision, J.Z.; project administration, B.Z.; funding acquisition, J.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Natural Science Foundation of China under Grant 61972056, and the Scientific Research Fund of Hunan Provincial Education Department under Grants 21B0287 and 22B0341.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Publicly available datasets were analyzed in this study. These datasets can be found at http://cvlab.hanyang.ac.kr/tracker_benchmark/datasets.html (accessed on 1 August 2022), https://www.votchallenge.net/challenges.html (accessed on 9 October 2023) and http://got-10k.aitestunion.com/downloads (accessed on 9 October 2023).

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

*Terminology*

AO: average overlap; EAO: expected average overlap; FPS: frames per second; IoU: intersection over union; RPN: region proposal network; SCAM: spatial channel attention module; $SR_{0.5}$ and $SR_{0.75}$: success rate.

*Algorithm*

CBAM: [30]; CornerNet [33]; DaSiamRPN: [23]; ExtremeNet: [34]; NLNet: [31]; PrDiMP: [36]; RBO [11]; SiamFC: [20]; SiamRPN: [21]; SiamRPN++: [24]; SiamBAN: [25]; SiamCAR: [26]; SENet: [28]; STMTrack: [29]; SiamFC++: [35].

## References


1. Zhang, J.; Feng, W.; Yuan, T.; Wang, J.; Sangaiah, A.K. SCSTCF: Spatial-channel selection and temporal regularized correlation filters for visual tracking. *Appl. Soft Comput.* **2022**, *118*, 108485. [CrossRef]
2. Zhang, J.; Jin, X.; Sun, J.; Wang, J.; Sangaiah, A.K. Spatial and semantic convolutional features for robust visual object tracking. *Multimed. Tools Appl.* **2020**, *79*, 15095–15115. [CrossRef]
3. Zhang, J.; Sun, J.; Wang, J.; Li, Z.; Chen, X. An object tracking framework with recapture based on correlation filters and Siamese networks. *Comput. Electr. Eng.* **2022**, *98*, 107730. [CrossRef]
4. Zhang, J.; Xie, X.; Zheng, Z.; Kuang, L.D.; Zhang, Y. SiamOA: Siamese offset-aware object tracking. *Neural Comput. Appl.* **2022**, *34*, 22223–22239. [CrossRef]
5. Zhang, J.; Sun, J.; Wang, J.; Yue, X.G. Visual object tracking based on residual network and cascaded correlation filters. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 8427–8440. [CrossRef]
6. Sidenmark, L.; Parent, M.; Wu, C.H.; Chan, J.; Glueck, M.; Wigdor, D.; Grossman, T.; Giordano, M. Weighted Pointer: Error-aware Gaze-based Interaction through Fallback Modalities. *IEEE Trans. Vis. Comput. Graph.* **2022**, *28*, 3585–3595. [CrossRef] [PubMed]
7. de Curtò, J.; de Zarzà, I.; Calafate, C.T. Semantic scene understanding with large language models on unmanned aerial vehicles. *Drones* **2023**, *7*, 114. [CrossRef]
8. de Curtò, J.; de Zarzà, I.; Roig, G.; Calafate, C.T. Summarization of Videos with the Signature Transform. *Electronics* **2023**, *12*, 1735. [CrossRef]
9. Yu, J.; Jiang, Y.; Wang, Z.; Cao, Z.; Huang, T. Unitbox: An advanced object detection network. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 516–520.
10. Yi-de, M.; Qing, L.; Zhi-Bai, Q. Automated image segmentation using improved PCNN model based on cross-entropy. In Proceedings of the 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, IEEE, Hong Kong, China, 20–22 October 2004; pp. 743–746.
11. Tang, F.; Ling, Q. Ranking-based Siamese visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 8741–8750.
12. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
13. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; Proceedings, Part V 13; Springer: Berlin/Heidelberg, Germany, 2014; pp. 740–755.
14. Real, E.; Shlens, J.; Mazzocchi, S.; Pan, X.; Vanhoucke, V. Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5296–5305.
15. Huang, L.; Zhao, X.; Huang, K. GOT-10k: A Large High-Diversity Benchmark for Generic Object Tracking in the Wild. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 1562–1577. [CrossRef] [PubMed]
16. Mueller, M.; Smith, N.; Ghanem, B. A Benchmark and Simulator for UAV Tracking. In Proceedings of the Computer Vision–ECCV, Amsterdam, The Netherlands, 11–14 October 2016; pp. 445–461.
17. Wu, Y.; Lim, J.; Yang, M.H. Object tracking benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1834–1848. [CrossRef] [PubMed]
18. Kristan, M.; Leonardis, A.; Matas, J.; Felsberg, M.; Pflugfelder, R.; Čehovin, L.; Vojir, T.; Häger, G.; Lukežič, A.; Fernández, G.; et al. The Visual Object Tracking VOT2016 Challenge Results. In Proceedings of the Computer Vision—ECCV 2016 Workshops, Amsterdam, The Netherlands, 11–14 October 2016; pp. 777–823.
19. Kristan, M.; Leonardis, A.; Matas, J.; Felsberg, M.; Pflugfelder, R.; Čehovin Zajc, L.; Vojir, T.; Bhat, G.; Lukezic, A.; Eldesokey, A.; et al. The sixth visual object tracking vot2018 challenge results. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018; pp. 3–53.
20. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In Proceedings of the Computer Vision—ECCV 2016 Workshops, Amsterdam, The Netherlands, 8–10 October 2016; Proceedings, Part II 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 850–865.
21. Li, B.; Yan, J.; Wu, W.; Zhu, Z.; Hu, X. High performance visual tracking with siamese region proposal network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8971–8980.
22. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; Volume 28.
23. Zhu, Z.; Wang, Q.; Li, B.; Wu, W.; Yan, J.; Hu, W. Distractor-aware siamese networks for visual object tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 101–117.

24. Li, B.; Wu, W.; Wang, Q.; Zhang, F.; Xing, J.; Yan, J. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4282–4291.
25. 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, Seattle, WA, USA, 13–19 June 2020; pp. 6668–6677.
26. 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, Seattle, WA, USA, 13–19 June 2020; pp. 6269–6277.
27. Zhang, J.; Huang, H.; Jin, X.; Kuang, L.D.; Zhang, J. Siamese visual tracking based on criss-cross attention and improved head network. *Multimed. Tools Appl.* **2023**. [CrossRef]
28. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
29. Fu, Z.; Liu, Q.; Fu, Z.; Wang, Y. Stmtrack: Template-free visual tracking with space-time memory networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13774–13783.
30. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
31. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 8–14 September 2018; pp. 7794–7803.
32. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
33. Law, H.; Deng, J. Cornernet: Detecting objects as paired keypoints. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 734–750.
34. Zhou, X.; Zhuo, J.; Krahenbuhl, P. Bottom-up object detection by grouping extreme and center points. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 850–859.
35. Xu, Y.; Wang, Z.; Li, Z.; Yuan, Y.; Yu, G. Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12549–12556.
36. Danelljan, M.; Gool, L.V.; Timofte, R. Probabilistic regression for visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 7183–7192.
37. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
38. Danelljan, M.; Hager, G.; Shahbaz Khan, F.; Felsberg, M. Learning spatially regularized correlation filters for visual tracking. In Proceedings of the IEEE International Conference on Computer Vision, Washington, DC, USA, 7–13 December 2015; pp. 4310–4318.
39. Valmadre, J.; Bertinetto, L.; Henriques, J.; Vedaldi, A.; Torr, P.H. End-to-end representation learning for correlation filter based tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2805–2813.
40. Zhang, Z.; Peng, H. Deeper and wider siamese networks for real-time visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4591–4600.
41. Danelljan, M.; Bhat, G.; Shahbaz Khan, F.; Felsberg, M. Eco: Efficient convolution operators for tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6638–6646.
42. Li, F.; Tian, C.; Zuo, W.; Zhang, L.; Yang, M.H. Learning spatial-temporal regularized correlation filters for visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4904–4913.
43. Li, X.; Ma, C.; Wu, B.; He, Z.; Yang, M.H. Target-aware deep tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1369–1378.
44. Wang, N.; Zhou, W.; Tian, Q.; Hong, R.; Wang, M.; Li, H. Multi-cue correlation filters for robust visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4844–4853.
45. Wang, Q.; Zhang, L.; Bertinetto, L.; Hu, W.; Torr, P.H. Fast online object tracking and segmentation: A unifying approach. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1328–1338.
46. Voigtlaender, P.; Luiten, J.; Torr, P.H.; Leibe, B. Siam r-cnn: Visual tracking by re-detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6578–6588.
47. Li, Q.; Qin, Z.; Zhang, W.; Zheng, W. Siamese keypoint prediction network for visual object tracking. *arXiv* **2020**, arXiv:2006.04078.
48. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. Atom: Accurate tracking by overlap maximization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4660–4669.