applied sciences

MDPI

Article

SiamCAM: A Real-Time Siamese Network for Object Tracking with Compensating Attention Mechanism

Kai Huang [1], Peixuan Qin [1], Xuji Tu [2], Lu Leng [1] and Jun Chu [1,*]

[1] Jiangxi Key Laboratory of Image Processing and Pattern Recognition, Nanchang Hangkong University, Nanchang 330063, China; 2016083500103@nchu.edu.cn (K.H.); 1816085212011@nchu.edu.cn (P.Q.); leng@nchu.edu.cn (L.L.)
[2] College of Softerware, Nanchang Hangkong University, Nanchang 330063, China; 71068@nchu.edu.cn
* Correspondence: chuj@nchu.edu.cn

Abstract: The Siamese-based object tracking algorithm regards tracking as a similarity matching problem. It determines the object location according to the response value of the object template to the search template. When there is similar object interference in complex scenes, it is easy to cause tracking drift. We propose a real-time Siamese network object tracking algorithm combined with a compensating attention mechanism to solve this problem. Firstly, the attention mechanism is introduced in the feature extraction module of the template branch and search branch of the Siamese network to improve the feature representation of the network to the object. The attention mechanism of the search branch enhances the feature representation of both the target and the similar backgrounds simultaneously. Therefore, based on the above two-branch attention, we propose a compensated attention model, which introduces the attention selected by the template branch into the search branch, and improves the discriminative ability of the search branch to the object by using the feature attention weighting of the template branch to the object. Experimental results on three popular benchmarks, including OTB2015, VOT2018, and LaSOT, show that the accuracy and robustness of the algorithm in this paper are adequate. It improved occlusion cases, similar object interference, and high-speed motion. The processing speed on GPU reaches 47 fps, which can achieve real-time object tracking.

Keywords: object tracking; Siamese network; attention

check for updates

Citation: Huang, K.; Qin, P.; Tu, X.; Leng, L.; Chu, J. SiamCAM: A Real-Time Siamese Network for Object Tracking with Compensating Attention Mechanism. *Appl. Sci.* **2022**, *12*, 3931. https://doi.org/10.3390/app12083931

Academic Editors: Junchi Yan and Minghao Guo

Received: 23 February 2022
Accepted: 11 April 2022
Published: 13 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Visual object tracking has received widespread attention in the past few years due to its wide application in visual surveillance [1], robotics [2], human-computer interaction [3], and augmented reality [4]. With the state of an object in the initial frame as inference, tracking aims to predict the object's state in each subsequent frame. With the development of deep learning, object tracking algorithms based on deep learning have become the research focus. The deep tracking methods can be roughly divided into two categories, i.e., Discriminative Correlation Filter (DCF) based methods and object Siamese networks based methods [5–7].

The DCF-based algorithms [8–10] train correlation filters by extracting depth features and filtering subsequent frame images to obtain the most relevant object. DCF-based approaches treat the tracking problem as a binary classification problem. The trained classifier distinguishes the object and the background. It selects the candidate sample with the highest confidence as the prediction result. Classifiers typically trained using neural networks have a relative advantage in discriminating targets and interferences. e.g., the ATOM [8] (Accurate Tracking by Overlap Maximization) network introduces IOU-Net (Intersection Over Union Network) [9] based on the Siamese network to accurately estimate the target frame and improve the accuracy. The DiMP (Discriminative Model

Prediction) [10] algorithm also improves the discriminative ability and design template update. While introducing the online learning model, the iterative process of the loss function is optimized by the extremely fast gradient algorithm. The speed of online iteration is very high. The overall algorithm has reached SOTA in terms of robustness and speed performance with a significant improvement. However, the discriminative algorithm essentially regards the tracking as a binary classification problem, separating the object's relationship and background. When the background of adjacent frames changes drastically, the discriminant algorithm cannot accurately track the object. Moreover, the discriminant algorithm relies on the appearance model's construction. The discriminative algorithm relies on the construction of the appearance model, which requires the union of background and foreground information and the introduction of historical frame information.

The Siamese-based object tracking algorithm balances the tracking accuracy and speed, achieving real-time, high-precision target tracking. It is the mainstream algorithm in the current object tracking. Based on the pioneering SiamFC [11] and SINT [12], many methods try to improve the tracking performance of Siamese trackers. SiamFC is the earliest end-to-end Siamese network target tracking algorithm, which regards the tracking problem as a similarity matching problem. The same feature extraction network is used to extract features for the template and module search branches in the offline learning stage. Then the template feature is used as a window for sliding matching in the search area, and the highest response value is the object. However, SiamFC did not consider the scale problem when taking the bounding box and could not solve the problem of different scales of objects. Therefore, SiamRPN [13] utilizes the Region Proposal Network [14] (RPN) for joint high-quality foreground-background classification and bounding box regression learning. However, due to the imbalance of the negative sample categories in the training data, the Siamese network cannot distinguish objects similar to the target well.

Current algorithms usually solve this problem from two aspects: one type of algorithm expands positive samples by using translation, flipping, and other means, and at the same time increases various types of complex negative samples so that the network can discriminate. For example, in the training phase of DaSiamRPN [15], the ImageNet [16] and COCO detection datasets [17] are made into positive sample pairs through data amplification to expand the types of training data sets and improve the generalization ability of the tracker. Moreover, it extracts pictures in different categories, and the same category as negative samples generates various kinds of complex negative samples and improves the discriminative ability of the tracker. The other is to improve the feature expression ability of the feature extraction module. For example, SiamRPN++ [18] introduces deeper Resnet [19] as the backbone network. The extracted features have a more vital expressive ability, which improves the anti-interference ability of the tracker. However, SiamRPN++ is still susceptible to interference from similar objects in a complex background. The Siamese network is essentially a template matching process that takes the first frame of target information as a template and cannot judge interferers in the background similar to the object. Moreover, they use the same classification network for feature extraction. It is impossible to obtain features with different directivity between the template and the search area. As shown in Figure 1, when other objects similar to the target appear in the current frame, the features extracted by the feature extraction network are very similar. As a result, double peaks appear in the response map, it is impossible to resolve the correct target. The Siamese-based algorithms do not have an online update mechanism, so the drifts are generated, and the tracking fails.

These methods enhance the ability to express features and improve the ability to distinguish similar objects to a certain extent. However, these algorithms will still lose track when similar objects of the same type have interfered in the scene. SiamRPN++ proves through experiments that the same category has a high response in some specific feature channels. In contrast, the response is shallow in other channels, indicating that the channel can reflect object category information to a certain extent.

We propose a robust tracking algorithm based on compensating attention to solve the above problems. The channel-spatial attention mechanism is introduced into the double branches of the Siamese network to strengthen the high-response channel characteristics of specific target categories and at the same time, strengthen the dominant characteristics of a specific area of the object. Then we add a compensation attention model based on the dual-branch attention, introduce the attention of template branch selection into the search branch, and merge the attention of the search area and the template branch selection. It improves the feature discrimination ability of the search branch to the target. It solves the problem that the channel-space attention introduced by the search branch enhances the target characteristics while also enhancing the characteristics of similar objects near the target.

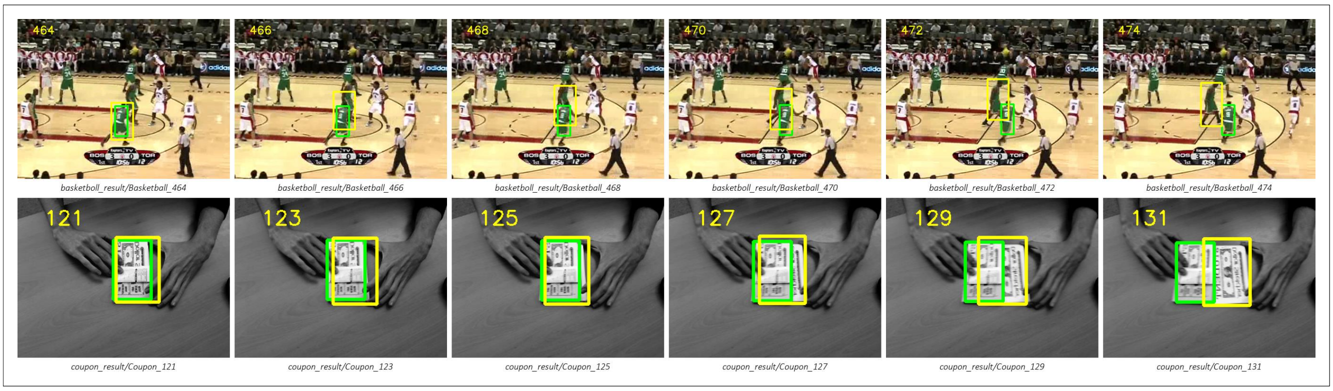
464
466
468
470
472
474
basketboll_result/Basketball_464
basketboll_result/Basketball_466
basketboll_result/Basketball_468
basketboll_result/Basketball_470
basketboll_result/Basketball_472
basketboll_result/Basketball_474
121
123
125
127
129
131
coupon_result/Coupon_121
coupon_result/Coupon_123
coupon_result/Coupon_125
coupon_result/Coupon_127
coupon_result/Coupon_129
coupon_result/Coupon_131

Figure 1. Drift problem leads to wrong target, when there is similar object interference in complex scenes. Green box is ground true and yellow box is the result of a Siamese-based tracker.

2. Related Work

2.1. Siamese Network-Based Trackers

In recent years, Siamese network-based tracking algorithms have treated tracking as a metric problem. They have attracted much attention for their excellent trade-offs in terms of speed and accuracy. Bertinetto et al. [11] first introduced SiamFC for visual tracking, aiming to learn the similarity metric between target templates and search regions and localize objects using correlation operations. With the success of RPN on object detection, Li et al. applied RPN to the Siamese Network framework to solve the multi-scale problem of objects. Although the SiamRPN network has improved performance in tracking, it cannot use the deeper backbone network and still uses the external AlexNet network [20]. To use the powerful depth features extracted by deep networks, SiamRPN++ and SiamDW address the challenges of introducing deep networks into the Siamese framework, resulting in a significant performance improvement. Zhu et al. introduced the hard negative mining technique in DaSiamRPN [15] to overcome the data imbalance issue by including more semantic negative pairs into the training process. In order to solve the shortcomings of the algorithms for target representation and running speed, SiamMask [21] is used as an integrated processing framework for visual object tracking and video object segmentation tasks.

The Siamese-based tracker does not perform any model updates or a simple linear update strategy, so the performance completely depends on the general matching ability of the Siamese network. However, the appearance of objects in the presence of tracking often varies greatly, and failure to update the model causes the tracker to fail. Guo et al. proposed the DSiam [22] tracker and designed dynamic transformation matrices. Yang et al. proposed the MemTrack [23] that dynamically writes and reads previous templates to cope with target appearance variations. And Zhang et al. proposed the UpdateNet [24] which learns a generic function that computes the updated template based on an initial ground-truth template, accumulated template, and the predicted template in the current frame.

Recently,There have been several notable transformer trackers, such as TransT [25], DualTFR [26], and STARK [27], which introduce the transformer to tracking framework for stronger information interaction and achieve compelling results. SiamBAN [28] and SiamCAR [29] employed full convolutional networks to directly classify objects and regress their bounding boxes at each spatial position, removing the problematic hyperparameter adjustments of anchor points. Although the performance of the existing Siamese network tracking algorithms has improved significantly, the feature representation capability of the backbone network is still lacking, and the ability to discriminate similarity interferers is not robust enough.

2.2. Real-Time Trackers

The tracking speed is a crucial metric to evaluate the trackers, especially to meet the real-time requirements. However, evaluating tracking speed is not simply a matter of quizzing speed. Many key factors come into play, including feature extraction, model update methods, and programming languages. In DCF-based trackers, MOSSE [30] runs at 669 fps on the CPU. Some other SOTA algorithms also achieve the same requirement of real-time tracking, such as ECO [31], ATOM [8], and DiMP [10]. For Siamese network-based algorithms, the network structure itself has a relatively large impact on the speed. For example, the tracking speed of SamFC [11] is 86 fps, which is far from the speed of MOSSE, but it also achieves real-time performance. The subsequent proposed DaSiamRPN [15] and SiamRPN [13] achieve tracking speeds of 160 fps on the GPU. The tracking speed of SiamRPN++ [18] is 35 fps, which barely meets the real-time requirement. For the tracker Siam R-CNN [32], combined with R-CNN, the tracking speed is only 4.7 fps due to its complex structural design. The proposed tracking speed is 47 fps, which meets the real-time requirement.

2.3. Attention Mechanisms

Attentional mechanisms were first proposed in the neuroscience neighborhood and later extended to other fields such as image classification, target detection, tracking, and human re-identification. For the object tracking task, RASNet [33] is an end-to-end residual attention twin network framework designed to learn generic feature representations and simultaneously have adaptive discriminators. The algorithm draws on the idea of residual network learning. It decomposes the spatial attention mechanism into prior and residual attention with strong individual adaptation capability.

Zeng [34] adds two attention modules to the framework of MDNet [35] to extract better features, a spatial attention mechanism, and a channel attention mechanism, respectively. It enables the network to extract discriminative and robust features or context through fused spatial channel attention. SA-Siam [36] considers a channel attention module in the semantic branch of their framework to improve the discrimination ability. IMG-Siam [37] considers the channel attention mechanism in the Siamese network to improve the matching model. Therefore, it separates the foreground and background of the object template in the template branch, extracts features and introduces attention separately, and then merges them to enhance the matching model. SiamFRN [38] introduces end-to-end features refine-based object tracking framework to improve the target representation utilizing semantic features. SCSAtt [39] employed channel attention and spatial attention mechanisms together to improve tracking performance with end-to-end learning.

SATIN [40] is a target tracking framework based on keypoint detection. The algorithm uses the idea of centroid detection borrowed from CenterNet [41], transposing the keypoint detection method to target tracking and introducing a spatial channel attention mechanism to enhance the representational and discriminatory capabilities of the feature extraction network. Similarly, we introduce spatial attention mechanisms to each branch of the Siamese Network separately and add a compensating attention model to branch attention. By introducing the attention mechanism to enhance the representation and discriminative ability of the network.

3. Approach

We use SiamRPN++ as the base network. Channel and spatial attention are introduced in the specific layer of the backbone network of its template branch and search branch, enhancing the feature expression ability. The channel attention weight of the template branch is overlapped and added to the search branch. The compensatory attention of the search area is combined with the attention of the template branch selection. The feature attention weight of the template branch to the object is used to improve the feature discrimination ability of the search branch to the object. The model structure is shown in Figure 2.

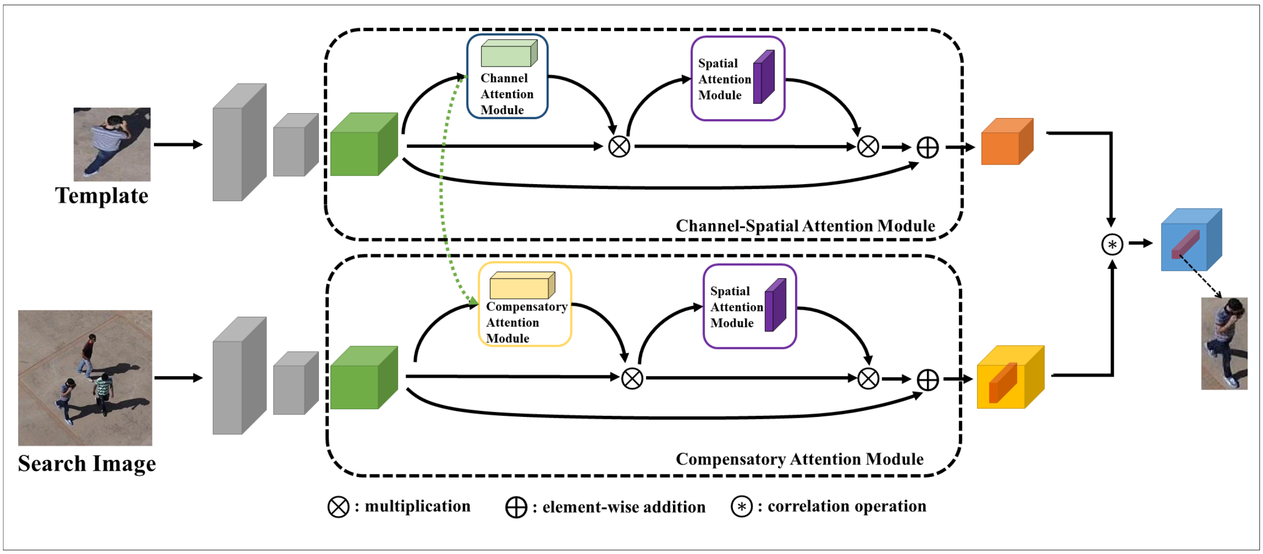
Template
Search Image
Channel Attention Module
Spatial Attention Module
Channel-Spatial Attention Module
Compensatory Attention Module
Spatial Attention Module
Compensatory Attention Module
⊗ : multiplication
⊕ : element-wise addition
⊛ : correlation operation

Figure 2. The architecture of our approach. On the template branch, the channel-spatial attention module is the same as the CBAM module. The pale green part is the channel attention matrix, and the purple is the spatial attention matrix. On the search branch, the pale yellow is our compensatory attention matrix, and the purple is likewise spatial attention matrix.

3.1. The Attention Model of Siamese Network

The Siamese network extracts image features through the migration classification network, and the in-depth features of the image are biased towards semantic information. SiamRPN++ uses ResNet [19] as the backbone network to extract features. Experiments confirm that different channels respond differently to specific object categories, indicating that deep features can learn semantic information about object bias. The dual-branch structure of the Siamese network is different from the input image information, and the features extracted by the two branches have different concerns in dimensions such as channel and space. We use the attention mechanism to filter the image information and learn the importance of the object in the search branch during the feature extraction process. In other words, let the network extract target features that are more effective for the target tracking task, ignoring the background information. Because the information of a particular category of the image responds differently to different channels of the feature map, we use channel attention to enhance the characterization information of the object and search branch objects and weigh the channels with a stronger dependence on the object to obtain features maps with a more significant response. When the translation invariance of the feature extraction network is destroyed, the network is easy to learn the positional bias, which is the response to the object in the center is the greatest. In order to eliminate the positional bias caused by the data distribution of different datasets, we use the spatial attention mechanism to learn the location of the object, suppress the positional bias, and strengthen the representation of the physical location information of the object in the image.

We introduce the CBAM (Convolutional Block Attention Module) [42] attention model. The attention mechanism model is a relatively lightweight and embeddable module. It obtains a new feature map by calculating, weighting channel, and spatial attention on a specific layer of the feature extraction network.

Channel attention. The channel-wise attention is employed to select proper channels that adaptively facilitate the current tracking task. Given the input feature map $Z^i \in \mathbb{R}^{H \times W \times C}$, where W, H, and C indicate the width, height, and channel number of feature maps, respectively. We first apply max-pooling and average-pooling operations through the spatial axis to get two 1D channel feature descriptors denoted as $Z^i_{max} \in \mathbb{R}^{1 \times 1 \times C}$ and $Z^i_{avg} \in \mathbb{R}^{1 \times 1 \times C}$. Then multiple layers of the perceptron are applied to each pooled feature descriptor to create a one-dimensional channel attention map $W^i_{cz} \in \mathbb{R}^{1 \times 1 \times C}$.

$$W^i_{cz} = sigmoid(MLP(Z^i_{max}) \oplus MLP(Z^i_{avg}))$$

where $\oplus$ denotes the element-wise addition and MLP means the multi-layer perception. The multi-layer perception is composed of a channel reduction layer, which its $\frac{C}{r} \times C$, and a ReLU activation, and a channel increasing layer, which its $C \times \frac{C}{r}$ and a sigmoid activation.

The Channel-Refined feature map $F^i_z \in \mathbb{R}^{H \times W \times C}$ weighted with the channel attention can be calculated sequentially as,

$$F^i_z = Z^i \otimes W^i_{cz}$$

where $\otimes$ indicates multiplication, respectively.

Spatial attention. We use spatial attention to emphasize areas of information that adequately represent the current target object. Given the channel-refined feature map $F^i_z \in \mathbb{R}^{H \times W \times C}$, We first apply max-pooling and average-pooling operations through the channel axis to get two 2D spatial feature descriptors denoted as $F^i_{max} \in \mathbb{R}^{H \times W \times 1}$ and $F^i_{max} \in \mathbb{R}^{H \times W \times 1}$. Spatial feature descriptors are then concatenated and fed into a single convolutional layer with sigmoid activation. We can get a 2D spatial attention map $W^i_{sz} \in \mathbb{R}^{H \times W \times 1}$,

$$W^i_{sz} = sigmoid(conv([F^i_{max}, f^i_{avg}]))$$

where $[F^i_{max}, f^i_{avg}] \in \mathbb{R}^{H \times W \times 2}$ denotes the concatenation of global average-pooled and max-pooled feature descriptors. *conv* means the single convolutional layer with a sigmoid activation, which is a 7×7 convolutional layer.

Finally, the attention-refined feature map $F^i \in \mathbb{R}^{H \times W \times C}$ weighted with both the channel attention and spatial attention can be calculated sequentially as,

$$F^i = Z^i \oplus (Z^i \otimes W^i_{cz} \otimes W^i_{sz})$$

where $\otimes$ and $\oplus$ indicate multiplication and element-wise addition, respectively. The structure of the attention model is shown in Figure 3.

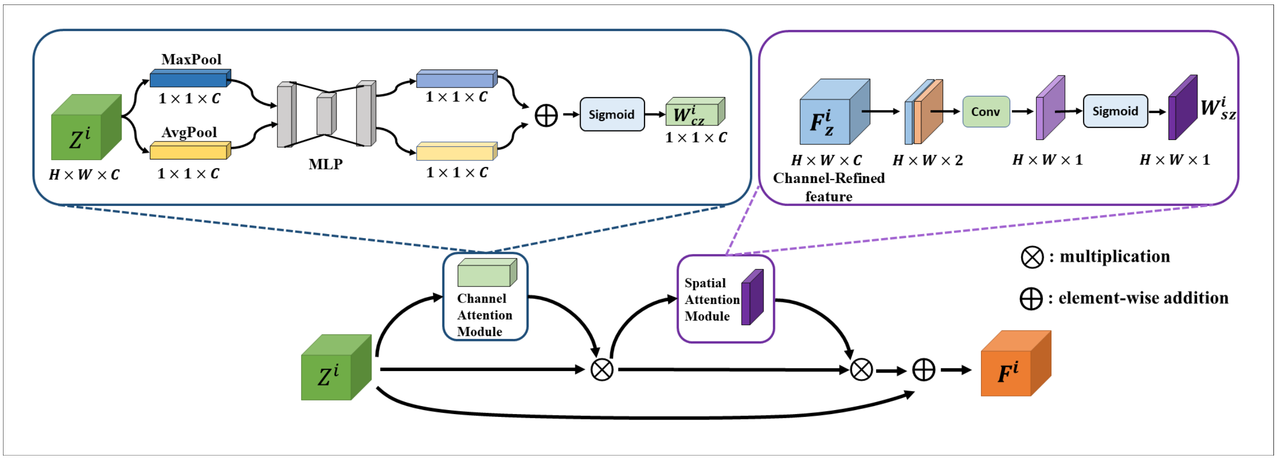

Figure 3. The Attention Model of Siamese Network.

3.2. The Compensatory Attention Model

There are identical and independent template branches and search branches in the Siamese structure. The template branch extracts only the target features, and the search branch contains both the target and the context. The presence of distractors in the search branch during tracking can affect the judgment of the target by the attention mechanism. Because the search branch has some generalizations to different targets when learning feature attention. It is impossible to distinguish between distractors and targets. Attention to different objects of the same category is usually enhanced simultaneously. Thus we propose to compensate for the autonomously selected channels of search branch attention with the channels of template branch attention selection. The structure of the compensatory attention model is shown in Figure 4.

As shown above, the template branch network has layer i features of $Z^i \in \mathbb{R}^{b\times C\times W\times H}$, and the search branch layer i feature of $X^i \in \mathbb{R}^{b\times C\times W\times H}$. The following is an example of X^i to give Z^i and X^i channel attention calculation. We compute the average pooling $X_a^i \in \mathbb{R}^{b\times C\times 1\times 1}$ and maximum pooling $X_m^i \in \mathbb{R}^{b\times C\times 1\times 1}$ on the channel dimension separately using X^i. They are downsampled by the fully connected layer, denoted as $X_a^{i}{}', X_m^{i}{}' \in \mathbb{R}^{b\times \frac{C}{16}\times 1\times 1}$. The same operation is performed to raise the dimension after activation, denoted as $X_a^{i}{}'', X_m^{i}{}'' \in \mathbb{R}^{b\times C\times 1\times 1}$. After adding them together to get the pooled feature map X_p^i, the channel attention weight W_x^i is obtained after activation and expansion, i.e.,

$$X_p^i = X_a^{i}{}'' + X_m^{i}{}'', X_p^i \in \mathbb{R}^{b\times C\times 1\times 1}$$

$$W_x^i = \sigma(x_p^i)$$

Similarly, the attention weight W_z^i of the template branch feature is obtained. Combination W_x^i and W_z^i as the final attention weights for the search branch, defined as

$$W_{sup}^i = f(W_x^i, W_z^i)$$

where $f(\cdot)$ can use sum $f_{add}(\cdot)$, dot product $f_{mul}(\cdot)$, and concatenate $f_{cat}(\cdot)$. According to the experimental results, the best results can be obtained by connecting the operations. The specific experiments are described in the experimental section of Section 4.

Finally, W_z^i and W_{sup}^i are weighted on the Z^i, and X^i feature maps to obtain a new feature map with reassigned channel weights in the layer i. The specific steps of the compensated attention strategy applied to the search branch are the following Algorithm 1.

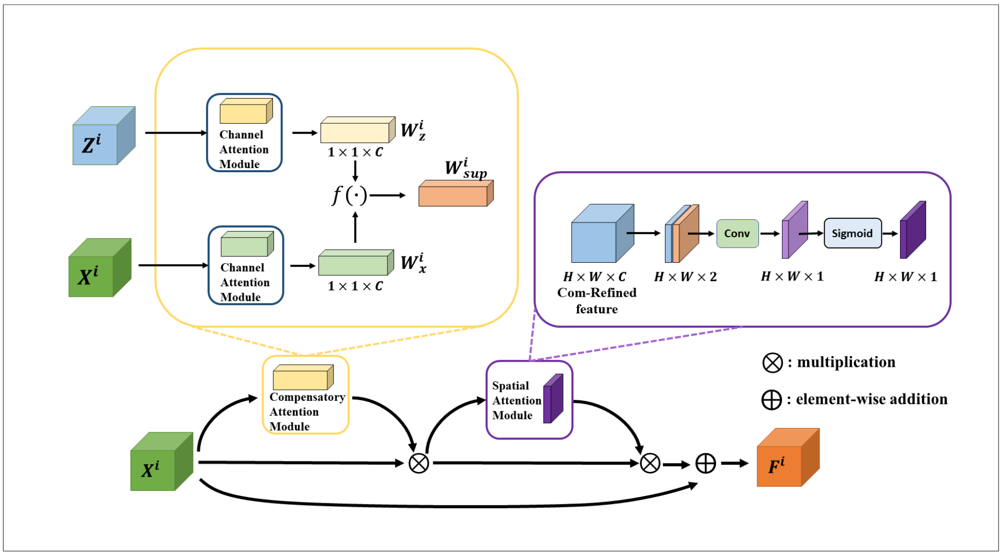

Figure 4. Compensated attention model for search branch.

The experimental results show that the features enhanced by the compensated attention method in this paper have more substantial expressive power and a more fantastic response to the target than the features extracted directly from the original text.

Figure 5 shows the feature map after enhancing the compensated attention method. It can be observed from the figure that the features extracted from the limbs of the ants are more explicit after adding the attention mechanism in the fourth column, which shows that the improved feature extraction network extracts more detailed features from the objects.

Algorithm 1 Compensated Attention Strategy for Searching Branches.

```
Input: $X^i, W_z^i$
Output: $W_{sup}^i$
for i = 1 to N do
    $X_{wh}^a = avgpooling(X^i)$.
    $X_{wh}^m = maxpooling(X^i)$.
    $X_{wh}^{pool} = X_{wh}^a + X_{wh}^m$.
    $W_x^i = \sigma(X_{wh}^{pool})$.
    $W_{sup}^i = f(W_x^i, W_z^i)$.
    if $f(\cdot)$ is add then
        $W_{sup}^i = f_{add}(W_x, W_z) = \sum_{c=1}^{C}(W_x^{i,c} + W_z^{i,c})$.
    else if $f(\cdot)$ is multiply then
        $W_{sup}^i = f_{mul}(W_x, W_z) = \sum_{c=1}^{C}(W_x^{i,c} * W_z^{i,c})$.
    else if $f(\cdot)$ is concat then
        $W_{sup}^i = f_{cat}(W_x, W_z) = \sum_{c=1}^{C}[W_x^{i,c} \dots W_z^{i,c}]$.
    end if
end for
```

X^i is the layer i feature map of the search frame W_z^i is the attention weight of the template frame features. C is the number of feature map's channels. N is the number of layers.

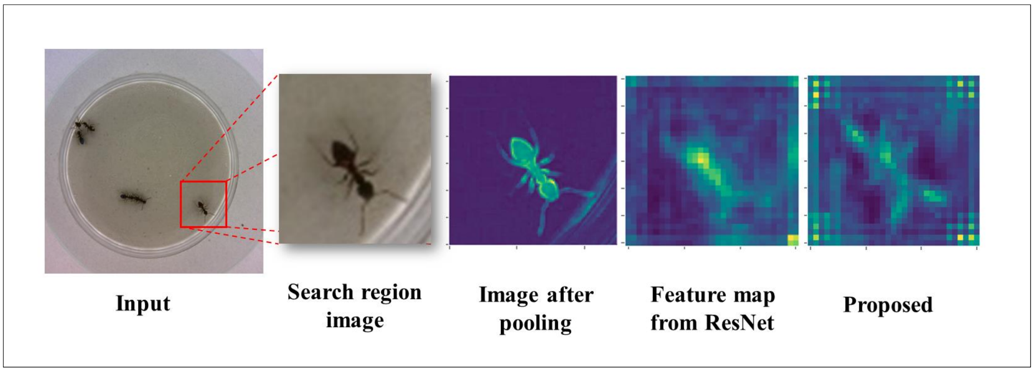

Figure 5. Feature extraction comparison chart.

A graph of the response of the compensated attention-enhanced features of this paper with the features extracted by SiamRPN++ to the target is given in Figure 6. From the figure, we can find that SiamRPN++ also generates responses at the locations of similar objects. However, the algorithm in this paper does not have responses at the locations of interfering objects, so the resistance of this paper to interfering objects is significantly improved.

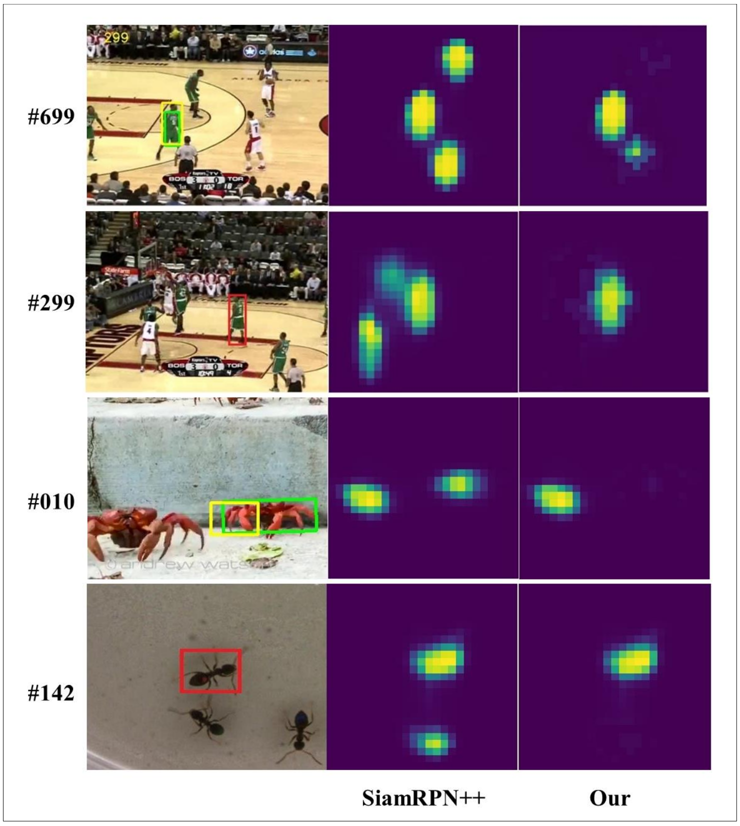

Figure 6. Comparison of response plots with SiamRPN++ algorithm.

4. Experiment

Our proposed tracking algorithm is evaluated on the generic tracking datasets VOT2018 [43], OTB2015 [44], and LaSOT [45]. At the same time, ablation experiments are performed on different components. Some of the state-of-the-art tracking algorithms, such as the Siamese family of algorithms, DiMP, ATOM, and the algorithms in this paper, are selected for comparison. Then, the effectiveness of different domain attention is evaluated at different layers of the network and in different combinations. Our method is trained with stochastic gradient descent (SGD). The experimental environment is 128G of running memory and 4 GTX 1080ti. We use synchronized SGD over 4 GPUs with 128 pairs per minibatch (32 pairs per GPU), which takes 12 h to converge. We use a warmup learning rate of 0.001 for the first 5 epochs to train the RPN braches. The Siamese network framework used is the PySOT [46] framework based on PyTorch 0.4.1 [47]. For the last 10 epochs, the whole network is end-to-end trained with a learning rate exponentially decayed from 0.005 to 0.0005. The parameters of the whole network are released for training later. One iteration processes 64 images with a CPU work thread of 1. The whole training is supported by the YouTubeBB [48], GOT10k [49], VOT2018, and DET [50] datasets.

4.1. Analysis

We evaluate the algorithms in this paper using two small datasets, OTB2015, VOT2018, and one large dataset LaSOT. OTB2015 and VOT2018 have been developed for a long time, have a well-developed assessment system, and have more authoritative assessment data in small datasets. The LaSOT is a large target tracking dataset containing 1400 video sequences, 70 categories, and over 3.52 million precise bounding boxes. Visualizing bounding box annotations makes the LaSOT dataset a large and finely annotated dataset.

Table 1 illustrates the experimental results of the OTB dataset. As can be seen from Table 1, the algorithm in this paper is the highest in both accuracy and precision on the OTB dataset, surpassing the mainstream depth correlation filtering algorithms. It is nearly 0.01 higher than the baseline algorithm SiamRPN++ and more than 0.02 than other mainstream algorithms. The algorithm's performance reaches 0.703 and 0.920 after appropriate tuning of the baseline network parameters. a comparison of the algorithm's performance with the mainstream algorithm after fine-tuning is shown in Figure 7.

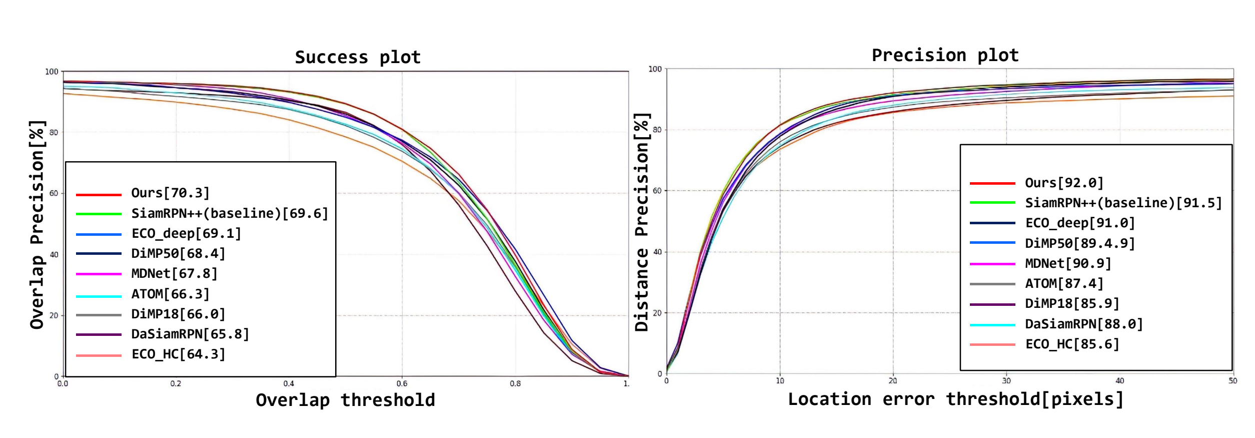

Figure 7. Comparison chart of OTB dataset.

Table 1. Experimental results of our algorithm and other algorithms on the OTB2015 dataset. The bolded values in the table are the best performance.

Tracker	Success	Precision
SiamFC [11]	0.587	0.772
MDNet [35]	0.678	0.909
DaSiamRPN [15]	0.659	0.880
ATOM [8]	0.667	0.879
DiMP50 [10]	0.684	0.894
Siam R-CNN [32]	0.701	0.891
SiamRPN++ [18]	0.696	0.910
SA-Siam [36]	0.657	0.865
RASNet [33]	0.642	-
SCSAtt [39]	0.641	0.855
SiamFRN [38]	0.636	0.840
IMG-Siam [37]	0.638	0.846
Ours	**0.701**	**0.916**

Table 2 shows the comparison results on the VOT2018 dataset. Compared to the baseline network SiamRPN++, the algorithm in this paper improves by nearly 0.02. Due to the difference in experimental equipment and parameter settings, there is an inevitable error in reproducing SiamRPN++ and the original paper. The algorithm for comparison with this paper is the result of our reproduction. However, there is still a gap compared with the mainstream depth correlation filtering algorithms DiMP and ATOM, which have better performance. Compared to OTB2015, the OTB2018 dataset scene is more complex. The related filter series algorithm will perform online fine-tuning during tracking. Its ability to update the image based on its contextualization information during tracking is suitable for objects with similar interference.

Table 2. Experimental results of our algorithm with other algorithms on the VOT2018 dataset. The bolded values in the table are the best performance.

Tracker	Accuracy ↑	Robust ↓	EAO ↑
C-COT [51]	0.49	0.32	0.267
ECO [31]	0.48	0.28	0.276
DaSiamRPN [15]	0.57	0.33	0.326
ATOM [8]	0.59	0.20	0.401
DiMP18 [10]	0.59	**0.17**	0.402
SiamMask [21]	0.60	0.32	0.380
SA-Siam [36]	0.50	0.45	0.236
SCSAtt [39]	0.54	0.22	0.250
SiamFRN [38]	0.52	0.35	0.199
SiamRPN++ [18]	0.60	0.23	**0.414**
Ours	**0.60**	0.22	0.395

In addition, the average frame speed of different algorithms on VOT2018 is evaluated in this paper, and the results are shown in Table 3. The algorithm in this paper is better in robustness and other data than the Siamese series papers. The speed is not reduced too much, indicating that this paper can improve the accuracy without sacrificing speed. The comparison between the Siamese family of algorithms and other algorithms also reflects the apparent advantage of Siamese networks in terms of speed, although not accuracy.

Table 4 illustrates the comparison between the algorithm in this paper and the mainstream algorithm on the LaSOT dataset. From the table, it can be observed that the algorithm of this paper performs well in long sequence videos, and compared with the Siamese network series of algorithms, our algorithm improves the ability of the Siamese network to track targets with severe deformation in videos with a long time and many frames, which further confirms the substantial improvement of the algorithm of this paper in feature discrimination ability.

Table 3. Experimental results of the speed of our algorithm running with other algorithms on the VOT2018 dataset. The bolded values in the table are the best performance.

	Ours	ATOM [8]	SiamRPN++ [18]	DiMP50 [10]	SiamMaskk [21]	ECO [31]
Speed(fps)	**47**	30	48	18	56	4

Note: Speed indicates the number of frames processed by the algorithm per second.

Table 4. Experimental results of our algorithm and other algorithms on the LaSOT dataset. The bolded values in the table are the best performance.

	ECO [31]	SiamFC [11]	MDNet [35]	DaSiamRPN [15]	ATOM [8]	DSiam [22]	SiamRPN++ [18]	Ours
Success	32.4	33.6	36.7	41.5	**51.5**	33.3	49.5	51.3
Precision	33.8	42.0	46.0	49.6	57.6	40.5	56.9	**58.8**

Note: Success indicates accuracy, precision indicates precision.

4.2. Visualization

Our algorithm can still accurately identify targets and calculate target frames in many complex scenes. Figure 8 shows the tracking results of our algorithm compared with mainstream algorithms such as the SiamRPN++ algorithm on some video sequences. The figure below shows that our algorithm outperforms the SiamRPN++ algorithm on all these video frames. The first four video sequences are taken from the OTB dataset. The last two video sequences are taken from the VOT dataset.

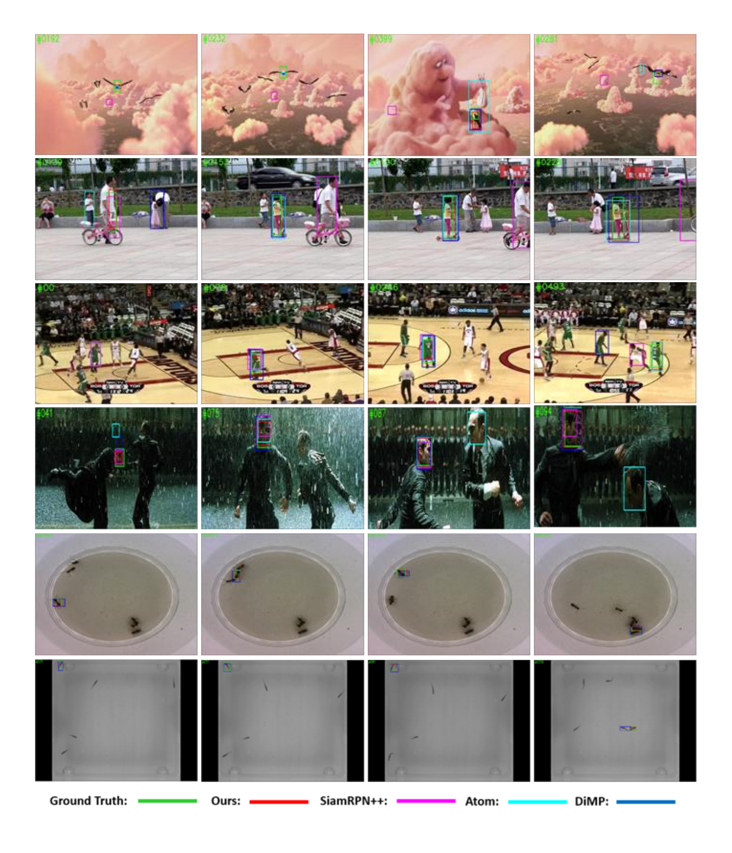

Figure 8. Visualization of our algorithm compared with other state-of-art algorithms.

4.3. Ablation Experiments

Ablation experiments were performed on the OTB2015 dataset for different components and combinations of approaches, and SiamRPN++ was chosen for the base network. To avoid different feature extraction networks being affected by different parameters and network layers, the ablation experiments of the attention mechanism were all performed on the same feature extraction network. ResNet is selected here. Table 5 shows the ablation experiments of applying different types of attention to the feature extraction network and adding attention at different positions. The attention mechanism is combined in a default weighted approach.

The performance of the tracker is degraded by first using channel attention directly on Layer4 of ResNet. The different layers are tested, and the reasons for failure are analyzed. It is analyzed that SiamRPN++ already suppresses the category that is not the target when performing the deep inter-correlation operation. This operation plays a specific role in channel selection, so there is no corresponding effect of further enhancement of the channel of the target response.

Then our algorithm combines different types of attention mechanisms in the channel and spatial domains, and the enhancement of the target features is somewhat improved. The tracking effect is also improved, and the accuracy is increased from 89.7 to 89.9. However, the effect was slight. After experimenting with different network layers for analysis, the analysis concluded that ResNet has deep network layers. The features extracted from different layers are different types of image features. The lower layer features are more of some region, line, and color features. In comparison, the deeper layer features are more oriented to get the semantic information of the image, which is why the feature pyramid can achieve better results in combining different layers of features. We analyzed the effect of feature information noticed by different features layers on the experimental results and found that deeper semantic information was more effective than shallow feature attention. Therefore, we improved the performance of the tracker to 90.8 after fusing the information of different layers which combined the attentional enhancement of Layer1, Layer3, and Layer4. The experimental results are shown in Table 5.

After we determined that the combined attention mechanism was effective for feature extraction, we conducted ablation experiments on the combined way of compensating attention, as shown in Table 6.

Three combinations of template branch and search branch attention were selected for experimentation: weighted, dot product, and connected. The connection achieved the best effect, which improved the tracker performance by almost one percentage point to 91.6; the other two methods also improved the experimental results, 91.0 and 91.5, respectively. It shows that the feature attention of the template branch does have a compensating effect on the search branch.

Table 5. Ablation experiments on dataset OTB2015-1 (Baseline for SiamRPN++ network architecture, C: channel attention, S: spatial attention,L1, L3, L4: ResNet network Layer1, Layer3, Layer4, respectively; right-hand metric is accuracy).

Baseline	**C**	**S**	**L1**	**L3**	**L4**	**OTB2015**	
						Precision	**Success**
√						0.897	0.674
√	√				√	0.889 ↓	0.670
√	√	√			√	0.899 ↑	0.681
√	√	√	√			0.895 ↓	0.673
√	√	√	√	√		0.903 ↑	0.680
√	√	√	√	√	√	0.908 ↑	0.687

Table 6. Ablation experiments on dataset OTB2015-2 (Baseline for SiamRPN++ network architecture, Base+Attn: multi-domain attention mechanism applied on two branches of the feature extraction network, M: compensation combination in the form of the dot product, A: weighted, C: connected, the right-hand metric for accuracy).

Baseline	**Base+Attn**	**M**	**A**	**C**	**OTB2015**	
					Precision	**Success**
√	√				0.908	0.688
√	√	√			0.910	0.695
√	√		√		0.915	0.698
√	√			√	0.916	0.701

5. Conclusions

The Siamese network will track a similarity matching problem of double branch feature with no interaction between the two branch independent structures. We propose a Siamese network architecture based on a compensated attention mechanism to enable feature extraction network processing to extract more discriminative features. We introduce a channel-space attention mechanism to secondary process the features extracted from the search branch by fusing the self-attentive features of the search region and the attention of the template branch selection. Thus, the attention mechanism's erroneous enhancement of the distractors is eliminated, allowing the response value gap between the target and distractor terms to be increased during the later inter-correlation operation. Our Siamese network architecture improves the attention of the feature extraction network to the target features, enabling the entire twin network to improve the discriminative power of the target, achieving a guaranteed real-time speed while improving the tracker performance. However, this new Siamese network still does not solve the problem that the template is stale and cannot adapt to the drastic changes of the target, and how to further improve the ability of the model to judge complex samples will be investigated in the future.

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

Funding: This research was funded by the National Natural Science Foundation of China (NSFC) grant number 62162045, Jiangxi Science and Technology Support Plan Project grant number 20192BBE 50073, National Natural Science Foundation of China (61866028), Technology Innovation Guidance Program Project (Special Project of Technology Cooperation, Science and Technology Department of Jiangxi Province) (20212BDH81003) and Postgraduate Innovation Special Fund Project of Nanchang Hangkong University grant number YC2021133.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declared no potential conflict of interest with respect to the research, authorship, and/or publication of this article.

References

1. Xing, J.; Ai, H.; Lao, S. Multiple human tracking based on multi-view upper-body detection and discriminative learning. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 1698–1701.
2. Yuan, D.; Li, Q.; Yang, X.; Zhang, M.; Sun, Z. Object-Aware Adaptive Convolution Kernel Attention Mechanism in Siamese Network for Visual Tracking. *Appl. Sci.* **2022**, *12*, 716. [CrossRef]
3. Luo, S.; Li, B.; Yuan, X.; Liu, H. Robust Long-Term Visual Object Tracking via Low-Rank Sparse Learning for Re-Detection. *Appl. Sci.* **2021**, *11*, 1963. [CrossRef]
4. Perez-Cham, O.E.; Puente, C.; Soubervielle-Montalvo, C.; Olague, G.; Aguirre-Salado, C.A.; Nuñez-Varela, A.S. Parallelization of the honeybee search algorithm for object tracking. *Appl. Sci.* **2020**, *10*, 2122. [CrossRef]
5. Kristan, M.; Leonardis, A.; Matas, J.; Felsberg, M.; Pflugfelder, R.; Kämäräinen, J.K.; Danelljan, M.; Zajc, L.Č.; Lukežič, A.; Drbohlav, O.; et al. The eighth visual object tracking VOT2020 challenge results. In Proceedings of the European Conference on Computer Vision, Virtual, 23–28 August 2020; pp. 547–601.
6. Cheng, S.; Zhong, B.; Li, G.; Liu, X.; Tang, Z.; Li, X.; Wang, J. Learning to Filter: Siamese Relation Network for Robust Tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 15–19 June 2021; pp. 4421–4431.
7. Zhang, Z.; Liu, Y.; Wang, X.; Li, B.; Hu, W. Learn to match: Automatic matching network design for visual tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Virtual, 11–17 October 2021; pp. 13339–13348.
8. 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, 16–20 June 2019; pp. 4660–4669.
9. Jiang, B.; Luo, R.; Mao, J.; Xiao, T.; Jiang, Y. Acquisition of localization confidence for accurate object detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 784–799.
10. Bhat, G.; Danelljan, M.; Gool, L.V.; Timofte, R. Learning discriminative model prediction for tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 6182–6191.
11. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 850–865.
12. Tao, R.; Gavves, E.; Smeulders, A.W. Siamese instance search for tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, USA, 17–22 June 2016; pp. 1420–1429.
13. 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.
14. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99. [CrossRef] [PubMed]
15. 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.
16. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Fontainebleau Resort, Miami Beach, Florida, USA, 20–25 June 2009; pp. 248–255.
17. 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 European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.

18. 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, 16–20 June 2019; pp. 4282–4291.
19. 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, New York, NY, USA, 17–22 June 2016; pp. 770–778.
20. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
21. 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, 16–20 June 2019; pp. 1328–1338.
22. Guo, Q.; Feng, W.; Zhou, C.; Huang, R.; Wan, L.; Wang, S. Learning dynamic siamese network for visual object tracking. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1763–1771.
23. Yang, T.; Chan, A.B. Learning dynamic memory networks for object tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 152–167.
24. Zhang, L.; Gonzalez-Garcia, A.; Weijer, J.V.D.; Danelljan, M.; Khan, F.S. Learning the model update for siamese trackers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 4010–4019.
25. Chen, X.; Yan, B.; Zhu, J.; Wang, D.; Yang, X.; Lu, H. Transformer tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–15 June 2021; pp. 8126–8135.
26. Xie, F.; Wang, C.; Wang, G.; Yang, W.; Zeng, W. Learning Tracking Representations via Dual-Branch Fully Transformer Networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Virtual, 11–17 October 2021; pp. 2688–2697.
27. Yan, B.; Peng, H.; Fu, J.; Wang, D.; Lu, H. Learning spatio-temporal transformer for visual tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Virtual, 11–17 October 2021; pp. 10448–10457.
28. 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, Virtual, 14–19 June 2020; pp. 6668–6677.
29. 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, Virtual, 14–19 June 2020; pp. 6269–6277.
30. Bolme, D.S.; Beveridge, J.R.; Draper, B.A.; Lui, Y.M. Visual object tracking using adaptive correlation filters. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 2544–2550.
31. 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.
32. 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, Virtual, 14–19 June 2020; pp. 6578–6588.
33. Wang, Q.; Teng, Z.; Xing, J.; Gao, J.; Hu, W.; Maybank, S. Learning attentions: Residual attentional siamese network for high performance online visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4854–4863.
34. Zeng, Y.; Wang, H.; Lu, T. Learning spatial-channel attention for visual tracking. In Proceedings of the 2019 IEEE/CIC International Conference on Communications in China (ICCC), Changchun, China, 11–13 August 2019; pp. 277–282.
35. Nam, H.; Han, B. Learning multi-domain convolutional neural networks for visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, USA, 17–22 June 2016; pp. 4293–4302.
36. He, A.; Luo, C.; Tian, X.; Zeng, W. A twofold siamese network for real-time object tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4834–4843.
37. Qin, X.; Fan, Z. Initial matting-guided visual tracking with siamese network. *IEEE Access* **2019**, *7*, 41669–41677. [CrossRef]
38. Rahman, M.; Ahmed, M.R.; Laishram, L.; Kim, S.H.; Jung, S.K. Siamese high-level feature refine network for visual object tracking. *Electronics* **2020**, *9*, 1918. [CrossRef]
39. Rahman, M.M.; Fiaz, M.; Jung, S.K. Efficient visual tracking with stacked channel-spatial attention learning. *IEEE Access* **2020**, *8*, 100857–100869. [CrossRef]
40. Gao, P.; Yuan, R.; Wang, F.; Xiao, L.; Fujita, H.; Zhang, Y. Siamese attentional keypoint network for high performance visual tracking. *Knowl.-Based Syst.* **2020**, *193*, 105448. [CrossRef]
41. Zhou, X.; Wang, D.; Krähenbühl, P. Objects as points. *arXiv* **2019**, arXiv:1904.07850.
42. 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.
43. Kristan, M.; Leonardis, A.; Matas, J.; Felsberg, M.; Pflugfelder, R.; Cehovin 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.
44. Wu, Y.; Lim, J.; Yang, M.H. Online object tracking: A benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2411–2418.

45. Fan, H.; Lin, L.; Yang, F.; Chu, P.; Deng, G.; Yu, S.; Bai, H.; Xu, Y.; Liao, C.; Ling, H. Lasot: A high-quality benchmark for large-scale single object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 5374–5383.
46. Zhang, F.; Wang, Q.; Chen, Z. PySOT: SenseTime Research Platform for Single Object Tracking. 2019. GitHub. Available online: https://github.com/STVIR/pysot (accessed on 30 December 2021).
47. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 8026–8037.
48. 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.
49. 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.* **2019**, *43*, 1562–1577. [CrossRef] [PubMed]
50. 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]
51. Danelljan, M.; Robinson, A.; Khan, F.S.; Felsberg, M. Beyond correlation filters: Learning continuous convolution operators for visual tracking. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 472–488.