# Multi-level Cross-attention Siamese Network For Visual Object Tracking

**Jianwei Zhang[1], Jingchao Wang[1], Huanlong Zhang[2*], Mengen Miao[1], Zengyu Cai[3], and Fuguo Chen[4]**

[1] College of Software Engineering, Zhengzhou University of Light Industry, Zhengzhou 450000, China
[e-mail: ing@zzuli.edu.cn]
[2] College of Electric and Information Engineering, Zhengzhou University of Light Industry
Zhengzhou 450000, China
[e-mail: zhl_lit@163.com]
[3] College of Computer and Communication Engineering, Zhengzhou University of Light Industry
Zhengzhou 450000, China
[e-mail: mailczy@163.com]
[4]Ping Gao Group Co., Ltd., Pingdingshan 467001, China
[e-mail: zhuye500@163.com]
[*]Corresponding author: Huanlong Zhang

*Received April 19, 2022; revised October 31, 2022; accepted December 1, 2022; published December 31, 2022*

## Abstract

Currently, cross-attention is widely used in Siamese trackers to replace traditional correlation operations for feature fusion between template and search region. The former can establish a similar relationship between the target and the search region better than the latter for robust visual object tracking. But existing trackers using cross-attention only focus on rich semantic information of high-level features, while ignoring the appearance information contained in low-level features, which makes trackers vulnerable to interference from similar objects. In this paper, we propose a Multi-level Cross-attention Siamese network(MCSiam) to aggregate the semantic information and appearance information at the same time. Specifically, a multi-level cross-attention module is designed to fuse the multi-layer features extracted from the backbone, which integrate different levels of the template and search region features, so that the rich appearance information and semantic information can be used to carry out the tracking task simultaneously. In addition, before cross-attention, a target-aware module is introduced to enhance the target feature and alleviate interference, which makes the multi-level cross-attention module more efficient to fuse the information of the target and the search region. We test the MCSiam on four tracking benchmarks and the result show that the proposed tracker achieves comparable performance to the state-of-the-art trackers.

***Keywords:*** Computer vision, Object tracking, Cross-attention, Self-attention, Siamese network

**http://doi.org/10.3837/tiis.2022.12.011** **ISSN : 1976-7277**

## 1. Introduction

Visual object tracking is widely used in drones, intelligent cars, and intelligent monitoring [1]. Given the position of the target in the first frame, visual object tracking aims to predict the position and estimate the state of the target in subsequent frames. In recent years, visual object tracking has been significant progress. But due to occlusion and target appearance change, object tracking is still a challenging task.

The earliest object tracking researchers focus on correlation filter-based algorithms [2-4] because they could use the Fourier domain for calculations, reaching high tracking speed. However, the accuracy of tracking does not meet the requirements of industrial applications. In recent years, many deep trackers [5, 6] are emerged and gradually dominated the target tracking field. Among these deep trackers, Siamese-based trackers [7, 8, 9, 10, 11, 12] have received the attention of most researchers because they achieved a good balance between speed and accuracy. The most representative Siamese-based tracker SiamFC [7] was proposed in 2016 to learn a common representation of target tracking from offline learning and showed good tracking performance. However, the SiamFC did not provide a reliable method to estimate the target size. In the following work, SiamRPN [8] introduced the regional proposal network in the object detection field to the object tracking field for the first time, describing the tracking as a local detection. The tracker eliminates the necessity for traditional multi-scale testing and online fine-tuning. However, to be able to generate precise bounding frames, SiamRPN often requires careful design of anchors based on prior knowledge. To solve this problem, many researchers have proposed anchor-free trackers [9, 10, 11, 12]. These trackers all consist of a Siamese subnet and a classification regression subnet. The former is used to establish a similar relationship between template features and search area features, while the latter is used to distinguish between target and background and to generate precise bounding boxes. These trackers avoid hyperparameters associated with the candidate boxes so that the tracker does not need to elaborate hyperparameters to achieve good results. In these trackers, correlation plays an important role. The establishment of similar relationships between templates and search regions directly affects the accuracy of classification tasks and regression tasks. However, the traditional correlation operation can only establish the target-level similar relationship between the template and the search region. And when a non-rigid change occurs during the target tracking process, the tracker's ability to identify the target is reduced, which will lead to track failure.

To adapt to the change of targets, many researchers have introduced more advanced methods into Siamese trackers instead of the original correlation operations, like graph attention [13], which can establish a good part-level similarity between the target and the search area. Furthermore, recent works [14, 15, 16, 17, 18] introduce cross-attention into the tracking domain to establish the part-level independence between the target feature and the search region feature. The cross-attention is part of the Transformer [19], which is first proposed in natural language processing. Although these trackers using cross-attention achieve good results, they only pay attention to using high-level semantic features for cross-attention, resulting in a lot of appearance information being ignored, which reduces the tracker's ability to discriminate similar objects.


**Fig. 1.** Visualized comparisons of MCSiam with representative trackers ATOM [20], MDNet [21], SiamFC, and SiamRPN++ [22]. MCSiam produces more accurate bounding boxes in the face of non-rigid variations and similar object interference.

In this paper, inspired by Transformer and the idea of a Feature Pyramid Network(FPN), we proposed a novel multi-level cross-attention Siamese network for tracking. Specifically, to make full use of the semantic and appearance features of the target for tracking, we designed a multi-level cross-attention module to fuse the features of the target and the search region at different layers. In addition, we design a target-aware module to guide the tracker to distribute more resources to the useful information. It can alleviate the influence of background features on the tracking. As shown in **Fig. 1**, benefiting from our proposed method, MCSiam can generate more accurate bounding boxes in the face of similar object interference and non-rigid variations.

Our major contributions are three folds:

- We proposed a novel Multi-level Cross-attention Siamese Network, which can aggregate different levels of target information and search region information for target positioning and **size estimation,** thereby improving the adaptability and discriminating ability of the tracker.
- We develop a target-aware module based on self-attention, which uses **self-attention** to enhance the important feature to highlight targets and suppress distractions.
- To demonstrate the MCSiam's effectiveness, we conducted a comprehensive ablation study and tested it on UAV123, TrackingNet, LaSOT, and NFS. Extensive experiments prove the performance of MCSiam is better than many advanced trackers.

## 2. Related work

### 2.1 Siamese-based Tracker

Siamese-based trackers have achieved good results in terms of speed and accuracy. However, the earliest proposed SiamFC hasn't an effective target size estimate strategy. Although it has been suggested to use multi-scale search region features for tracking, and then use the scale with the highest confidence as the final scale of the target. But this method not only has high time complexity but also high space complexity. To solve this problem, SiamRPN introduced

the regional proposal network into the tracking framework, describing the **tracking** task as a detection task for each frame. Use pre-set anchor boxes to locate the target and estimate its size. However, the placement of the anchor box involves a lot of hyperparameters, and the tracking effect is very sensitive to the setting of these hyperparameters. So, it often takes a long time to adjust these hyperparameters to achieve good results. To solve the problem of hyperparameter sensitivity, inspired by research in the field of object detection, SiamCAR [9], SiamFC++ [10], SiamBAN [12], and Ocean [11] proposed anchor-free tracking frameworks. Which used regression branches to directly regress the offset value of the target boundary relative to the classification position. Compared with anchor-based trackers, anchor-free trackers greatly simplify the tracking pipeline and improved the tracking speed. In addition, SiamGAT [13] introduced graph attention into the tracking framework to generate a better synthetic feature map by establishing similar relations between templates and search areas. It can establish the part-level similarity relationship between them, to better adapt to the change of target. To further improve the adaptability of the tracker to target changes, STMTrack [23] proposes a novel Siamese framework based on a space-time memory network, which consists of a memory branch and a query branch. Which uses multiple memory frames and foreground-background label maps to locate the target in the query frame and has a strong adaptive ability to appearance change of the target.

### 2.2 Transformer-based Tracker

The Transformer was proposed in machine translation. Due to its excellent parallelism and attention mechanism, it is widely used in natural language processing. In recent years, Transformer has also demonstrated excellent performance in computer vision tasks such as image recognition [24, 25], object detection [26, 27], and semantic segmentation [28, 29]. ViT [24] tried to use the original Transformer with as few changes as possible for image classification, and the pre-trained model on large-scale data sets achieved results that surpassed other advanced models based on a convolutional neural network. CvT[25] improves Vision Transformer(ViT) in performance and efficiency by introducing convolution into ViT to yield the best of both designs. DETR [26] proposes a new end-to-end object detection method, which simplifies the detection pipeline and uses the Transformer structure for detection tasks. Deformable DTER[27] mitigates the slow convergence and high complexity issues of DETR. It combines the best of sparse spatial sampling of deformable convolution and the relation modeling capability of the Transformer. MedT[28] has gated axial attention as its main building block for the encoder and uses LoGo strategy for training. Finally, it achieves a good performance over ConvNets. SETR[29] presented an alternative perspective for semantic segmentation by introducing the Transformer. Inspired by these works, some researchers introduce the Transformer into visual object tracking and have achieved advanced results. TransT [14] proposes a novel, simple, high-performance Transformer-based feature fusion network for object tracking. The network uses attention mechanisms for feature fusion, which can establish long-distance feature dependencies so that the tracker adaptively focuses on useful information. DTT [15] presents a new discriminative tracking approach based on encoder-decoder Transformer architecture, which can exploit the rich scene information for robust tracking. DualTFR [18] proposes a completely Transformer-based tracker that does not contain any convolutional neural network structures. They demonstrated the superiority of the entirely attention-based tracking paradigm over the traditional convolutional neural network-based tracking paradigm. But these trackers only use single-layer features for cross-attention and do not notice the importance of different target information of different layer features for tracking.

# 3. Multi-level Cross-attention Siamese Network

## 3.1 MCSiam Architecture

To obtain a simple and effective method to fuse semantic information and appearance information for objecting tracking, we design a multi-level cross-attention Siamese tracker. As illustrated in **Fig. 2**, the MCSiam consists of a backbone, target-aware module, multi-level cross-attention module, and prediction head.

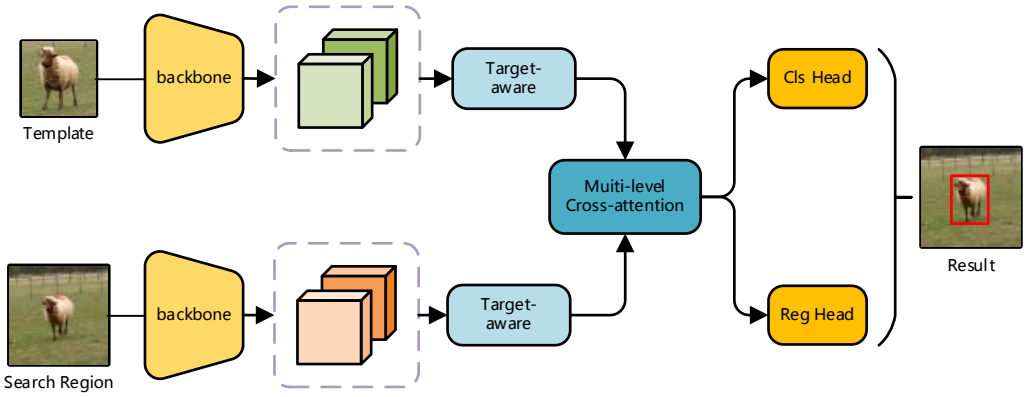


**Fig. 2.** Overall of MCSiam. It consists of a backbone, target-aware module, multi-level cross-attention module, and prediction head.

First, the template and search region are extracted using the backbone to obtain their high-level and low-level features. Second, use the object-aware module to enhance the target information and suppress others information in the features. Third, the enhanced template feature and search region feature of different layers are input into the multi-layer cross-attention module for fusion and then obtain the synthetic feature map including their interdependence, which contains rich appearance information and semantic information. Finally, input the synthetic feature map to the prediction head to estimate the state of the target.

## 3.2 Feature extraction

As illustrated in **Fig. 3**, in this paper, we adopt a modified ResNet50 [30] pre-trained on ImageNet as the backbone. Compare with AlexNet [31], the ResNet is easier to optimize and performs better.

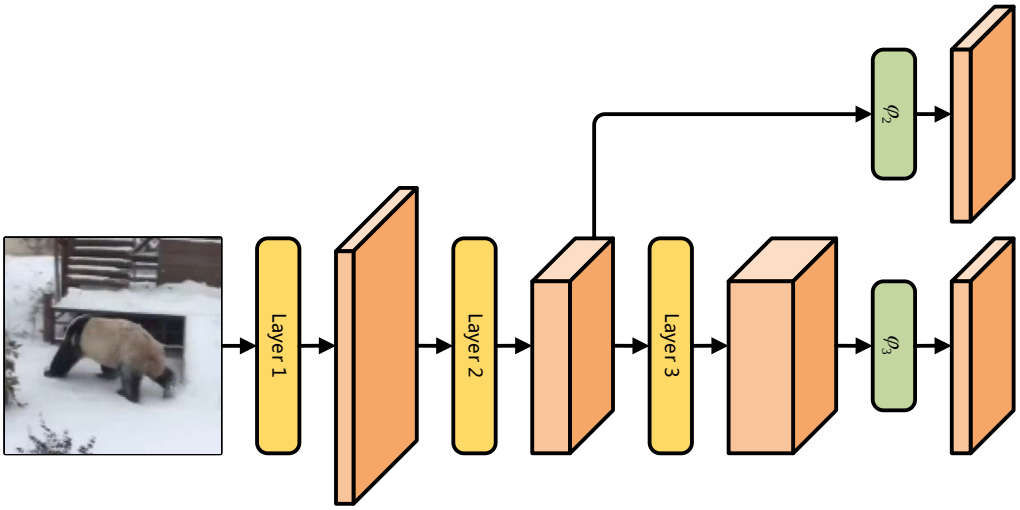


**Fig. 3.** The pipeline of the backbone.

In the tracker we proposed, we only use the first three layers of ResNet for feature extraction and abandon the fourth and fifth layers. In addition, we added two convolutional layers with kernel size $1 \times 1$ (represented by $\varphi_2$ and $\varphi_3$) at the end of the ResNet50 to compress the features with 512 channels output by the second layer and the features with 1024 channels output by the third layer. After the feature extraction of the backbone network, we will obtain the template features $T_{layer2}$, $T_{layer3}$ and the search area features $S_{layer2}$, $S_{layer3}$. Among them, $T_{layer2}$and $S_{layer2}$ contain a lot of appearance information and a little semantic information, but $T_{layer3}$ and $S_{layer3}$ are the opposite.

### 3.3 Target-aware Module

In the traditional correlation-based tracking method, the template is simply cropped from the target area as the target feature, and used to locate the target in the search region. However, this approach results in a lot of background information about the target being included in the template feature. If these background features are not processed, the interdependence between the target and the search region will be disturbed, and the tracking performance will be degraded. To alleviate the interference caused by background features in templates, we design a self-attention-based object-aware module.

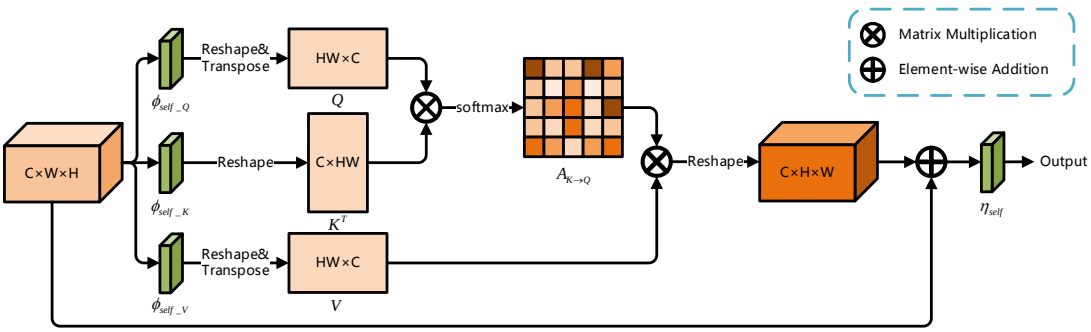


**Fig. 4.** The pipeline of the target-aware module.

As shown in **Fig. 4**, firstly, we compute the features $Q$, $K$, and $V$ using the input $X$, $\phi_{self\_Q}$, $\phi_{self\_K}$ and $\phi_{self\_V}$ is are three convolutional layers with different parameters and a convolution kernel of $1 \times 1$.

$$Q = Transpose(Reshape(\phi_{self\_Q}(X)))$$
$$K = Reshape(\phi_{self\_K}(X))$$
$$V = Transpose(Reshape(\phi_{self\_V}(X)))$$

Then compute the similarities between every two positions to obtain a similarity matrix. And then, we input the similarity matrix to a softmax layer and obtain a spatial weight matrix $A_{i,j}$. The process is defined as:

$$A_{ij} = \frac{exp(Q_i K_j^T)}{\sum_{s=1}^{s=W\times H} exp(Q_i K_s^T)}$$

Where $i \in \{1,2,\ldots,W \times H\}$ and $j \in \{1,2,\ldots,W \times H\}$. Then, We retrieve target information from V using the weight matrix $A_{i,j}$.

$$\overline{Y_i} = A_{i,1} \times V_1 + A_{i,2} \times V_2 + \ldots + A_{i,W\times H} \times V_{W\times H} (i \in \{1,2,\ldots,W \times H\})$$

Then, reshape the $\bar{Y}$ to the shape of $X$. Since there may be information loss in the whole calculation process, we add the original input as the residual term to the enhanced feature $\bar{Y}$, which can prevent the loss of important information.

During the whole calculation process, we calculated the dependencies between any two points in the entire feature map. Compared with the convolution operation, the receptive field is more flexible, and the feature is enhanced according to the information of the feature itself. It can highlight important target information in the feature map and suppress other features.

### 3.4 Multi-level Cross-attention Module

Most of the traditional Siamese-based tracking relies on cross-correlation operation and its related variants. But it can only establish target-level interdependencies between templates and search areas. As a result, the tracker cannot adapt to the non-rigid changes of the target. To solve this problem, many researchers use cross-attention to replace the original cross-correlation. However, they only used high-level semantic features for cross-attention operations and did not pay attention to the appearance information contained in the low-level features, resulting in a lack of the tracker's ability to discriminate similar objects.

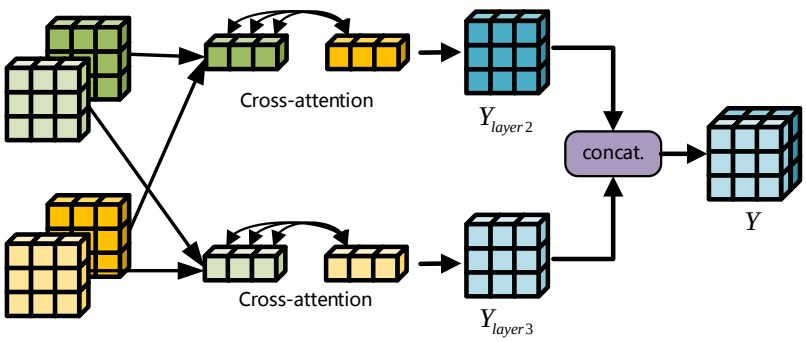


**Fig. 5.** Overall of the multi-level cross-attention module.

To aggregate the rich semantic information and appearance information of different level features, in this paper, we design a multi-level cross-attention module. Its structure is shown in **Fig. 5**, which can aggregate features of different layers and then combine them for target state estimation.

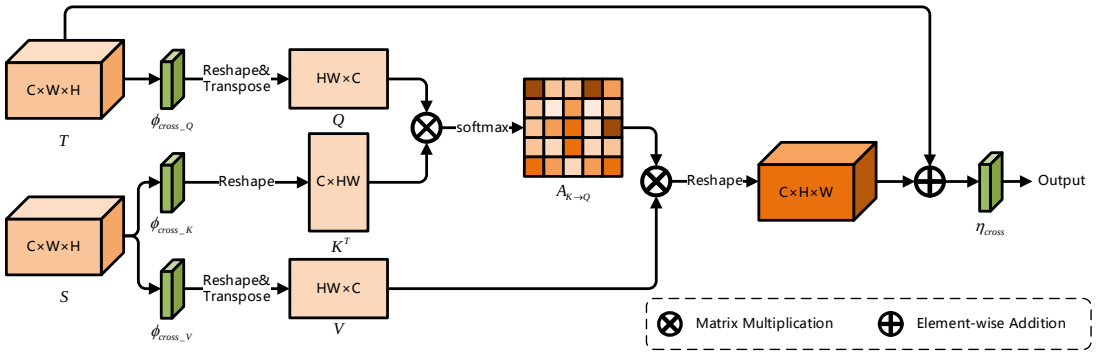


**Fig. 6.** The pipeline of cross-attention.

The pipeline of cross-attention is shown in **Fig. 6**. The inputs of it consist of two tensors, which are from the template branch and the search branch. First, we compute the features $Q$, $K$, and $V$, the process is defined as:

$$Q = Transpose(Reshape(\phi_{cross\_Q}(T)))$$
$$K = Reshape(\phi_{cross\_K}(S))$$
$$V = Transpose(Reshape(\phi_{cross\_V}(S)))$$

Second, we compute the similarity matrix like the target-aware module. Different from the target-aware module, which computes the similarities in one feature. In this module, the similarity matrix consists of the similarities between each position of the template and search region. Then, important information is extracted from the template features according to the similarity matrix. In the end, we use element-wise addition to fuse the extracted target information with the search region information. These calculation processes are similar to the calculation methods in the target perception module, and will not be repeated here. Readers can refer to the content of the previous section.

After performing cross-attention on the target and search region features of the corresponding layer, we combine the cross-attention results in the channel dimension in a concatenated manner. The process can be defined as:

$$Y = Concat(Y_{layer2}, Y_{layer3})$$

The $Concat()$ donates the concatenated operation, $Y_{layer2} \in \mathbb{R}^{C\times H\times W}$ represent the result of cross-attention between the $T_{layer2}$ and $S_{layer2}$, $Y_{layer3} \in \mathbb{R}^{C\times H\times W}$ represent the result of cross-attention between the $T_{layer3}$ and $S_{layer3}$, and $Y \in \mathbb{R}^{2C\times H\times W}$ is the output of the multi-level cross-attention module. $Y$ consists of many dependencies of semantic information and appearance information.

### 3.5 Prediction head

In our proposed tracker, the prediction head includes a classification head and a regression head, each consisting of a multilayer perceptron. Their input is the synthetic feature map $Y$ generated by the multi-level cross-attention nodule. Different between the classification head

and the regression head, the output $R_{cls} \in \mathbb{R}^{1\times H\times W}$ of the former represents the probability that each location in the search area is a target, the output $R_{reg} \in \mathbb{R}^{4\times H\times W}$ of the latter represents the normalized coordinates concerning the search region size.

## 4. Experiments

### 4.1 Implementation Details

The training set of MCSiam includes TrackingNet [32], LaSOT [33], GOT-10k [34], and COCO [35]. We test MCSiam on four benchmarks including UAV123 [36], NFS [37], LaSOT, and TrackingNet. The backbone parameters are initialized using ImageNet-pretrained ResNet-50. The tracker is trained for 40 epochs. The learning rate is set to be 1e-5 for the pre-trained backbone, and 1e-4 for the others. The learning rate decays by a factor of 10 at the 32nd epoch. The entire training process took about 80 hours.

### 4.2 Evaluation

**UAV123.** The UAV123 dataset mainly consists of 123 drone videos. In this dataset, there are many cases where the target disappears completely, so many more difficult challenges are brought to the tracker, which can better test the performance of the tracker when the target disappears. We compare the MCSiam with ECO [38], SiamRPN, DaSiamRPN [39], UPDT [40], SiamRPN++, SiamCAR, and MetaRTT [41], the result is shown in **Table 1**. MCSiam achieves the best success score of 62.5% and the best precision score of 79.1%.

**Table 1.** The results of MCSiam comparison with other trackers on UAV123.

| Tracker | Success | Precision |
|---|---|---|
| ECO | 52.5 | 74.1 |
| SiamRPN | 55.7 | 76.8 |
| DaSiamRPN | 56.9 | 78.1 |
| UPDT | 55.0 | - |
| SiamRPN++ | 61.0 | 75.2 |
| SiamCAR | 61.2 | 76.0 |
| MetaRTT | 56.9 | 80.9 |
| Ours | 62.5 | 79.1 |

**NFS.** The NSF dataset contains 100 video sequences with fast-moving objects, which are annotated with axis-aligned bounding boxes. We compare MCSiam with UPDT, DiMP [42], ATOM, SiamBAN, PrDiMP [43], SiamRCNN [44], and KYS [45], and the result is shown in **Table 2**. The MCSiam we proposed achieves the best success score of 64.9%.

**Table 2.** The results of MCSiam comparison with other trackers on NFS.

| **Tracker** | UPDT | DiMP | ATOM | SiamBAN | PrDiMP | SiamRCNN | KYS | Ours |
|---|---|---|---|---|---|---|---|---|
| **Success** | 62.2 | 62.0 | 59.0 | 59.4 | 63.5 | 63.9 | 63.5 | 64.9 |

**LaSOT.** LaSOT is a long-term object tracking benchmark, which contains a total of 1400 video sequences, with an average of 2512 video frames per video sequence. It includes 70 video categories with 20 video sequences in each category. In these longer video sequences, there are more challenges such as occlusion, object change, etc. We compare MCSiam with DiMP, LTMU [46], ATOM, DaSiamRPN, SiamRPN++, SiamMask [47], MDNet, SiamFC, ECO, CFNet [48], and KCF [2], and the result is shown in **Fig. 7**. The MCSiam we proposed

achieves the best success score of 57.5%, precision score of 59.2%, and normalized precision of 60.7%. Compare whit the advanced Siamese-based Tracker DaSiamRPN, we achieved a 4.9% increase in success score. Compared with the advanced discriminative tracker DiMP, we achieved a 2.1% improvement in precision score. These results demonstrate the effectiveness of MCSiam.

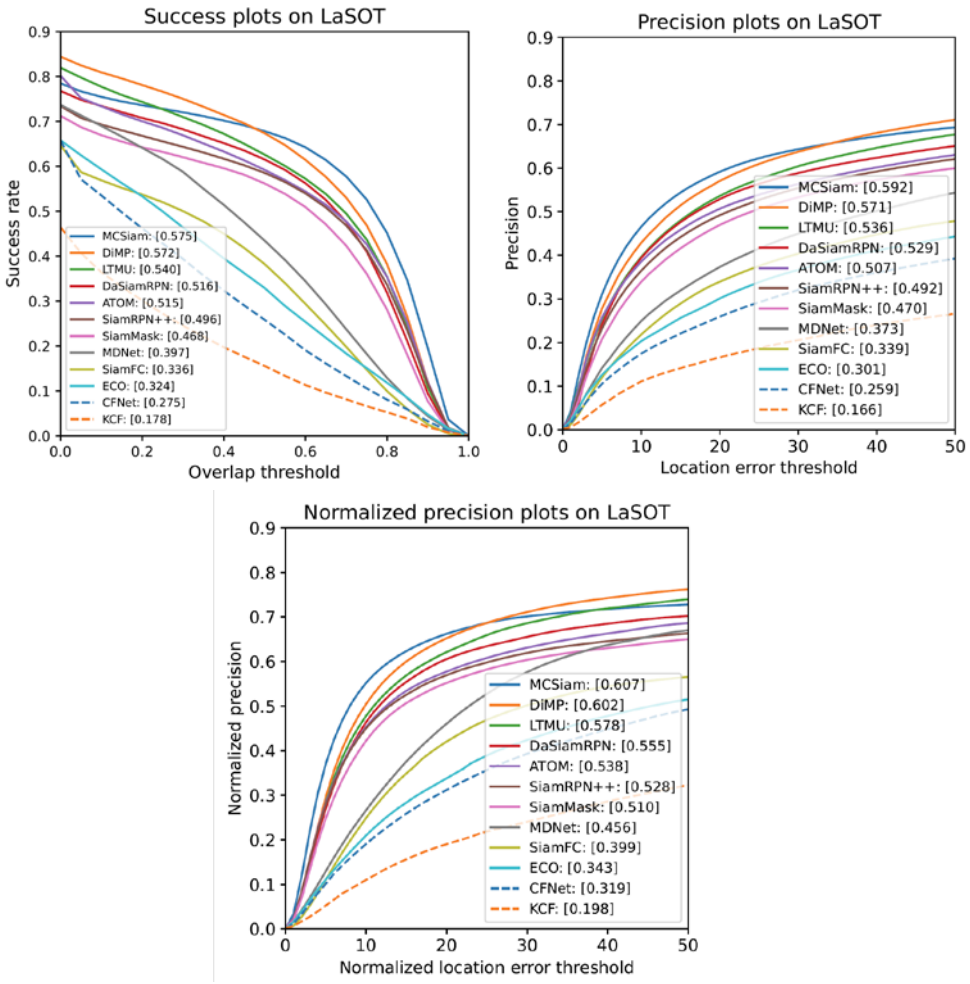


**Fig. 7.** Plots show comparisons of MCSiam with other advanced trackers on the LaSOT.

**TrackingNet.** The TrackingNet dataset is currently the largest object tracking benchmark, which contains more than 30,000 video sequences. The dataset tracks a wide variety of objects, with an average duration of 16.6 seconds per video sequence and a total duration of up to 140 hours. The test set of TrackingNet contains 511 videos. As shown in **Table 3**, we compare MCSiam with ATOM, SiamRPN++, DiMP, GlobalTrack [49], D3S [50], KYS, SiamAttn [51], SiamFC++, SiamGAT, AutoMatch [52], and LightTrack [53]. This Table shows that the MCSiam achieves the best success score of 76.5%, precision score of 73.1%, and normalized precision of 82%.

**Table 3.** The results of MCSiam comparison with other trackers on TrackingNet.

| Tracker | Success | Precision | Normal Precision |
|---|---|---|---|
| ATOM | 70.3 | 64.8 | 77.1 |
| SiamRPN++ | 73.3 | 69.4 | 80.0 |
| DiMP | 74.0 | 68.7 | 80.1 |
| GlobalTrack | 70.4 | 65.6 | 75.4 |
| D3S | 72.8 | 66.4 | 76.8 |
| KYS | 74.0 | 68.8 | 80.0 |
| SiamAttn | 75.2 | - | 81.7 |
| SiamFC++ | 75.4 | 70.5 | 80.0 |
| SiamGAT | 0.753 | - | - |
| AutoMatch | 76.0 | 72.6 | - |
| LightTrack | 73.3 | 70.8 | 78.9 |
| Ours | 76.5 | 73.1 | 82 |

### 4.3 Ablation Studies

**Discussion on the multi-level cross-attention module.** To demonstrate the effectiveness of our proposed multi-level cross-attention module, we compare trackers using different features for cross-attention on the LaSOT dataset, and the results are shown in **Table 4**.

**Table 4.** Quantitative comparison results of our tracker and its variants with different cross-attention methods on LaSOT and TrackingNet.

| Tracker | LaSOT | | | TrackingNet | | |
|---|---|---|---|---|---|---|
| | Suc. | Prec. | Norm. Prec. | Suc. | Prec. | Norm. Prec. |
| **layer2** | 55.4 | 54.7 | 58.4 | 73.7 | 68.5 | 79.3 |
| **layer3** | 57.1 | 58 | 60.1 | 75.9 | 72 | 81.4 |
| **layer2 & layer3** | 57.5 | 59.2 | 60.7 | 76.5 | 73.1 | 82 |

From **Table 4**, it can be found that when only single-layer features are used, the effect of layer 3 is better than that of layer 2. When using two layers of features at the same time, the effect is better than using a single layer of features. Due to the limited equipment, in this paper, we only conduct experiments on the second and third layers. However, the experimental results show that the performance of tracking can be improved when using multi-layer features for cross-attention.

**Discussion on the target-aware module.** To verify the effectiveness of our proposed object-aware method, we compare the tracker equipped with the object-aware module and the tracker without the object-aware module on LaSOT, and the results are shown in **Table 5**.

**Table 5.** Quantitative comparison results of our tracker with and without target-aware module on LaSOT and TrackingNet.

| Tracker | LaSOT | | | TrackingNet | | |
|---|---|---|---|---|---|---|
| | Suc. | Prec. | Norm. Prec. | Suc. | Prec. | Norm. Prec. |
| **w target-aware** | 57.5 | 59.2 | 60.7 | 76.5 | 73.1 | 82 |
| **w/o target-aware** | 50.8 | 49.7 | 54.3 | 72.2 | 66.8 | 77.8 |

As can be seen from **Table 5**, our proposed object-aware method plays an important role in tracking and significantly improves the performance of the tracker. Compared with the tracker without the object-aware module, the tracker with the object-aware module obtains more than 5% performance improvement.

## 5. Conclusion

We propose a new tracking framework based on multi-level cross-attention. This framework abandons the cross-correlation operation in the traditional Siamese network and adopts cross-attention to establish the interdependence between the target and the search region. And by combining the interdependencies between different level layers, the detailed information and semantic information of features at different levels are aggregated in the synthetic feature map. Furthermore, we also introduce an object-aware module based on self-attention mechanism to enhance object features and always interfere with features. Extensive experiments show that our proposed tracker achieves competitive performance.

## Acknowledgement

This work is supported by the National Natural Science Foundation of China (62072416, 61873246, 62102373), Program for Science & Technology Innovation Talents in Universities of Henan Province (21HASTIT028), Zhongyuan Science and Technology Innovation Leadership Program (214200510026), and Natural Science Foundation of Henan Province (202300410495).

## References

[1] Marvasti-Zadeh S M, Cheng L, Ghanei-Yakhdan H, et al., "Deep learning for visual tracking: A comprehensive survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 5, pp. 3943-3968, 2022. Article (CrossRef Link)

[2] J.F. Henriques, R. Caseiro, P. Martins, et al., "High-speed tracking with kernelized correlation filters," *IEEE Trans Pattern Anal Mach Intell*, vol. 37, no. 3, pp. 583-596. Mar. 2015. Article (CrossRef Link)

[3] D.S. Bolme, J.R. Beveridge, B.A. Draper, et al., "Visual object tracking using adaptive correlation filters," in *Proc. of CVPR 2010*, pp. 2544-2550, 2010. Article (CrossRef Link)

[4] M. Danelljan, G. Hager, F.S. Khan, et al., "Learning spatially regularized correlation filters for visual tracking," in *Proc. of ICCV 2015*, pp. 4310-4318, 2015. Article (CrossRef Link)

[5] J. Choi, J. Kwon, and K.M. Lee, "Deep meta learning for real-time target-aware visual tracking," in *Proc. of ICCV 2019*, pp. 911-920, Nov. 2019. Article (CrossRef Link)

[6] H.X. Li, Y. Li, and F. Porikli, "Deeptrack: Learning discriminative feature representations online for robust visual tracking," *IEEE Trans. Image Proc.*, vol. 25, pp. 1834-1848, Apr, 2016. Article (CrossRef Link)

[7] L. Bertinetto, J. Valmadre, J.F. Henriques, et al., "Fully-convolutional siamese networks for object tracking," in *Proc. of ECCV 2016*, pp. 850-865, 2016. Article (CrossRef Link)

[8] B. Li, J. Yan, W. Wu, et al., "High performance visual tracking with siamese region proposal network," in *Proc. of CVPR 2018*, pp. 8971-8980, 2018. Article (CrossRef Link)

[9] D. Guo, J. Wang, Y. Cui, et al., "Siamcar: Siamese fully convolutional classification and regression for visual tracking," in *Proc. of CVPR 2020*, pp. 6268-6276, 2020. Article (CrossRef Link)

[10] Y.D. Xu, Z.Y. Wang, Z.X. Li, et al., "Siamfc plus plus : Towards robust and accurate visual tracking with target estimation guidelines," in *Proc. of the AAAI Conference on Artificial Intelligence*, vol. 37(07), pp. 12549-12556, 2020. Article (CrossRef Link)
[11] Z. Zhang, H. Peng, J. Fu, et al., "Ocean: Object-aware anchor-free tracking," in *Proc. of ECCV 2020*, pp. 771-787, 2020. Article (CrossRef Link)
[12] Z. Chen, B. Zhong, G. Li, et al., "Siamese box adaptive network for visual tracking," in *Proc. of CVPR 2020*, pp. 6667-6676, 2020. Article (CrossRef Link)
[13] D. Guo, Y. Shao, Y. Cui, et al., "Graph attention tracking," in *Proc. of CVPR 2021*, pp. 9538-9547, 2021. Article (CrossRef Link)
[14] X. Chen, B. Yan, J. Zhu, et al., "Transformer tracking," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2021*, pp. 8122-8131, 2021. Article (CrossRef Link)
[15] B. Yu, M. Tang, L. Zheng, et al., "High-performance discriminative tracking with transformers," in *Proc. of 2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9836-9845, 2021.Article (CrossRef Link)
[16] N. Wang, W. Zhou, J. Wang, et al., "Transformer meets tracker: Exploiting temporal context for robust visual tracking," in *Proc. of CVPR 2021*, pp. 1571-1580, 2021. Article (CrossRef Link)
[17] B. Yan, H. Peng, J. Fu, et al., "Learning spatio-temporal transformer for visual tracking," in *Proc. of ICCV 2021*, pp. 10428-10437, 2021. Article (CrossRef Link)
[18] F. Xie, C. Wang, G. Wang, et al., "Learning tracking representations via dual-branch fully transformer networks," in *Proc. of ICCV 2021*, pp. 2688-2697, 2021. Article (CrossRef Link)
[19] A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention is all you need," *Advances in Neural Information Processing Systems*, pp. 5998-6008, 2017.
[20] M. Danelljan, G. Bhat, F.S. Khan, et al., "Atom: Accurate tracking by overlap maximization," in *Proc. of CVPR 2019*, pp. 4655-4664, 2019. Article (CrossRef Link)
[21] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proc. of CVPR 2016*, pp. 4293-4302, 2016. Article (CrossRef Link)
[22] B. Li, W. Wu, Q. Wang, et al., "Siamrpn++: Evolution of siamese visual tracking with very deep networks," in *Proc. of CVPR 2019*, pp. 4277-4286, 2019. Article (CrossRef Link)
[23] Z. Fu, Q. Liu, Z. Fu, et al., "Stmtrack: Template-free visual tracking with space-time memory networks," in *Proc. of CVPR 2021*, pp. 13769-13778, 2021. Article (CrossRef Link)
[24] A. Dosovitskiy, L. Beyer, A. Kolesnikov, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020. Article (CrossRef Link)
[25] H. Wu, B. Xiao, N. Codella, et al., "Cvt: Introducing convolutions to vision transformers," in *Proc. of ICCV 2021*, pp. 22-31, 2021. Article (CrossRef Link)
[26] N. Carion, F. Massa, G. Synnaeve, et al., "End-to-end object detection with transformers," in *Proc. of ECCV 2020*, pp. 213-229, 2020. Article (CrossRef Link)
[27] X. Zhu, W. Su, L. Lu, et al., "Deformable detr: Deformable transformers for end-to-end object detection," in *Proc. of International Conference on Learning Representations*, 2020.
[28] J.M.J. Valanarasu, P. Oza, I. Hacihaliloglu, et al., "Medical transformer: Gated axial-attention for medical image segmentation," in *Proc. of International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 36-46, 2021. Article (CrossRef Link)
[29] S. Zheng, J. Lu, H. Zhao, et al., "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proc. of CVPR 2021*, pp. 6877-6886, 2021. Article (CrossRef Link)
[30] K.M. He, X.Y. Zhang, S.Q. Ren, et al., "Deep residual learning for image recognition," in *Proc. of CVPR 2016*, pp. 770-778, 2016. Article (CrossRef Link)
[31] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, 2012.
[32] M. Mueller, A. Bibi, S. Giancola, et al., "Trackingnet: A large-scale dataset and benchmark for object tracking in the wild," in *Proc. of ECCV 2018*, pp. 300-317, 2018. Article (CrossRef Link)
[33] H. Fan, H. Ling, L. Lin, et al., "Lasot: A high-quality benchmark for large-scale single object tracking," in *Proc. of CVPR 2019*, pp. 5369-5378, 2019. Article (CrossRef Link)

[34] L., Huang, X. Zhao, K. Huang, "Got-10k: A large high-diversity benchmark for generic object tracking in the wild," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 43, no. 5 pp. 1562-1557, 2021. Article (CrossRef Link)
[35] T.-Y. Lin, M. Maire, S. Belongie, et al., "Microsoft coco: Common objects in context," in *Proc. of ECCV 2014*, pp. 740-755, 2014. Article (CrossRef Link)
[36] M. Mueller, N. Smith and B. Ghanem, "A benchmark and simulator for UAV tracking," in *Proc. of ECCV 2016*, pp. 445-461, 2016. Article (CrossRef Link)
[37] H.K. Galoogahi, A. Fagg, C. Huang, et al., "Need for speed: A benchmark for higher frame rate object tracking," in *Proc. of ICCV 2017*, pp. 1134-1143, 2017. Article (CrossRef Link)
[38] M. Danelljan, G. Bhat, F.S. Khan, et al., "Eco: Efficient convolution operators for tracking," in *Proc. of CVPR 2017*, pp. 6931-6939, 2017. Article (CrossRef Link)
[39] Z. Zhu, Q. Wang, B. Li, et al., "Distractor-aware siamese networks for visual object tracking," in *Proc. of ECCV 2018*, pp. 103-119, 2018. Article (CrossRef Link)
[40] G. Bhat, J. Johnander, M. Danelljan, et al., "Unveiling the power of deep tracking," in *Proc. of ECCV 2018*, pp. 493-509, 2018. Article (CrossRef Link)
[41] Jung, I., You, K., Noh, H., Cho, M., & Han, B., "Real-time object tracking via meta-learning: Efficient model adaptation and one-shot channel pruning," in *Proc. of AAAI 2018*, vol. 34, no. 07, pp. 11205-11212, 2020. Article (CrossRef Link)
[42] G. Bhat, M. Danelljan, L. Van Gool, et al., "Learning discriminative model prediction for tracking," in *Proc. of ICCV 2019*, pp. 6181-6190, 2019. Article (CrossRef Link)
[43] M. Danelljan, L. Van Gool and R. Timofte, "Probabilistic regression for visual tracking," in *Proc. of CVPR 2020*, pp. 7181-7190, 2020. Article (CrossRef Link)
[44] P. Voigtlaender, J. Luiten, P.H.S. Torr, et al., "Siam R-CNN: Visual tracking by re-detection," in *Proc. of CVPR 2020*, pp. 6577-6587, 2020. Article (CrossRef Link)
[45] G. Bhat, M. Danelljan, L. Van Gool, et al., "Know your surroundings: Exploiting scene information for object tracking," in *Proc. of ECCV 2020*, pp. 205-221, 2020. Article (CrossRef Link)
[46] K.N. Dai, Y.H. Zhang, D. Wang, et al., "High-performance long-term tracking with meta-updater," in *Proc. of CVPR 2020*, pp. 6297-6306, 2020. Article (CrossRef Link)
[47] Q. Wang, L. Zhang, L. Bertinetto, et al., "Fast online object tracking and segmentation: A unifying approach," in *Proc. of CVPR 2019*, pp. 1328-1338, 2019. Article (CrossRef Link)
J. Valmadre, L. Bertinetto, J. Henriques, et al., "End-to-end representation learning for correlation filter based tracking," in *Proc. of CVPR 2017*, pp. 5000-5008, 2017. Article (CrossRef Link)
[48] L. Huang, X. Zhao and K. Huang, "Globaltrack: A simple and strong baseline for long-term tracking," in *Proc. of the AAAI Conference on Artificial Intelligence*, pp. 11037-11044, 2020. Article (CrossRef Link)
[49] A. Lukezic, J. Matas and M. Kristan, "D3s – a discriminative single shot segmentation tracker," in *Proc. of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7131-7140, 2020. Article (CrossRef Link)
[50] Y. Yu, Y. Xiong, W. Huang, et al., "Deformable siamese attention networks for visual object tracking," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6727-6736, 2020. Article (CrossRef Link)
[51] Z. Zhang, Y. Liu, X. Wang, et al., "Learn to match: Automatic matching network design for visual tracking," in *Proc. of the IEEE/CVF International Conference on Computer Vision*, pp. 13339-13348, 2021.
[52] B. Yan, H. Peng, K. Wu, et al., "LightTrack: Finding lightweight neural networks for object tracking via one-shot architecture search," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15175-15184, 2021. Article (CrossRef Link)

**Jianwei Zhang** received his Ph.D. degree in computer application technology from PLA Information Engineering University in 2010. He is a professor at the Zhengzhou University of Light Industry. His research interests include video object tracking and network security.

**Jingchao Wang** was born in Nanyang, Henan, China. He is currently pursuing a degree with the Zhengzhou University of Light Industry, Zhengzhou, China. His research interests include deep learning and visual tracking.

**Huanlong Zhang** from the School of Aeronautics and Astronautics, Shanghai Jiao Tong University, China, in 2015. He is currently an Associate Professor with the College of Electric and Information Engineering, Zhengzhou University of Light Industry, Henan, Zhengzhou, China. His research interests include pattern recognition, machine learning, image processing, computer vision, and intelligent human-machine systems.

**Mengen Miao** was born in Nanyang, Henan, China. He is currently pursuing a degree with the Zhengzhou University of Light Industry, Zhengzhou, China. His research interests include deep learning and visual tracking.

**Zengyu Cai** received his master degree in computer application technology from Northeast Normal University, Changchun, China, in 2006. He is an associate professor at the Zhengzhou University of Light Industry. His research interests include computer vision, plan recognition, and information security.

**Fuguo Chen** is a PhD student at the School of Electrical Engineering, Xi 'an Jiaotong University and a senior engineer at Pinggao Group Co., LTD. His research direction is research on key technologies of condition monitoring, evaluation and diagnosis of intelligent high-voltage switchgear.