


Article

SiamFFN: Siamese Feature Fusion Network for Visual Tracking

Jiahao Bao ^{1,2,3} , Menglong Yan ^{1,2,4}, Yiran Yang ^{1,2,3} and Kaiqiang Chen ^{1,2,*}¹ Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China² Key Laboratory of Network Information System Technology (NIST), Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China³ School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100190, China⁴ Jigang Defence Technology Company, Ltd., Jinan 250132, China

* Correspondence: chenkaqiang14@mails.ucas.ac.cn; Tel.: +86-188-1090-0679

Abstract: Siamese network-based trackers have developed rapidly in the field of visual object tracking recently. Many Siamese network-based trackers currently in use rely on result fusion to combine the classification result map and regression result map. However, these result maps are obtained from the multi-level feature map and are independent of each other. It is inappropriate and flawed to use result fusion. Additionally, classification module and regression module are independent of each other, which leads to feature misalignment. In this paper, we propose a feature-fusion approach that involves fusing similarity response maps using a novel scale attention mechanism and subsequently decoding the features. To reduce the feature misalignment and produce more precise tracking results, we suggest using Classification Supervised Regression Loss (CSRL), to train the model. Experiments conducted on three challenging benchmark datasets show that this method outperforms current models in terms of both performance and efficiency, running at 40 fps.

Keywords: visual object tracking; Siamese network; feature fusion; feature alignment



Citation: Bao, J.; Yan, M.; Yang, Y.; Chen, K. SiamFFN: Siamese Feature Fusion Network for Visual Tracking. *Electronics* **2023**, *12*, 1568. <https://doi.org/10.3390/electronics12071568>

Academic Editor: Oscar Deniz Suarez

Received: 6 March 2023

Revised: 24 March 2023

Accepted: 25 March 2023

Published: 27 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Object tracking is a basic challenge that involves predicting the target state in the video based on the initial state. It has several uses, including visual surveillance [1], pose estimation [2], and autonomous vehicles [3]. Therefore, it is a very active research direction. Despite the recent advances, various issues, such as scale variances, background clutters, scale variation and scale variation, continue to make it very challenging.

In recent years, Siamese network-based trackers [4–7] have shown encouraging progress. The pioneering method, SiamFC [4] utilizes the Siamese network architecture [8] to address the object tracking problem to the object tracking issue, establishing the ground-work for a series of later methods. Following this work, although several studies [9–11] focus on improving the feature representation of the Siamese model, the overall structure has remained mostly unchanged. In 2018, SiamRPN [5] introduced the region proposal network (RPN) [12]. Then, SiamRPN++ [6] presented a sampling strategy to successfully introduce ResNet to the Siamese tracker. In addition, it proposes a depth-wise crosscorrelation (DW-Xcorr) layer to produce multichannel similarity response maps. Since RPN relies on anchor points and a series of related hyperparameters, the model's generalization ability is severely reduced. The following research has focused on how to remove the effects of anchor points. Therefore, a series of anchor-free trackers are proposed. SiamBAN uses a per-pixel-prediction method to regress the bounding box from the similarity response map. Recently, a series of new trackers inspired by transformer [13] have been proposed, such as TransT [14].

To leverage multi-level features for prediction, existing Siamese network-based trackers [4–7] input the feature maps of the last three blocks of ResNet-50 [15] into the similarity

matching module to obtain three similarity response maps. After that, the three similarity response maps are input to the tracking head separately for prediction. The result maps obtained from the tracking head are fused in an adaptive manner. As shown in Figure 1a, these methods fuse different levels of result maps to obtain the final tracking results. However, these result maps represent the results obtained from multi-level features. Directly fusing them is inherently flawed. Instead, a better approach is to fuse the results obtained from different scales based on their corresponding weights. Moreover, result fusion may be effective, but it does not provide an explanation for why it works.

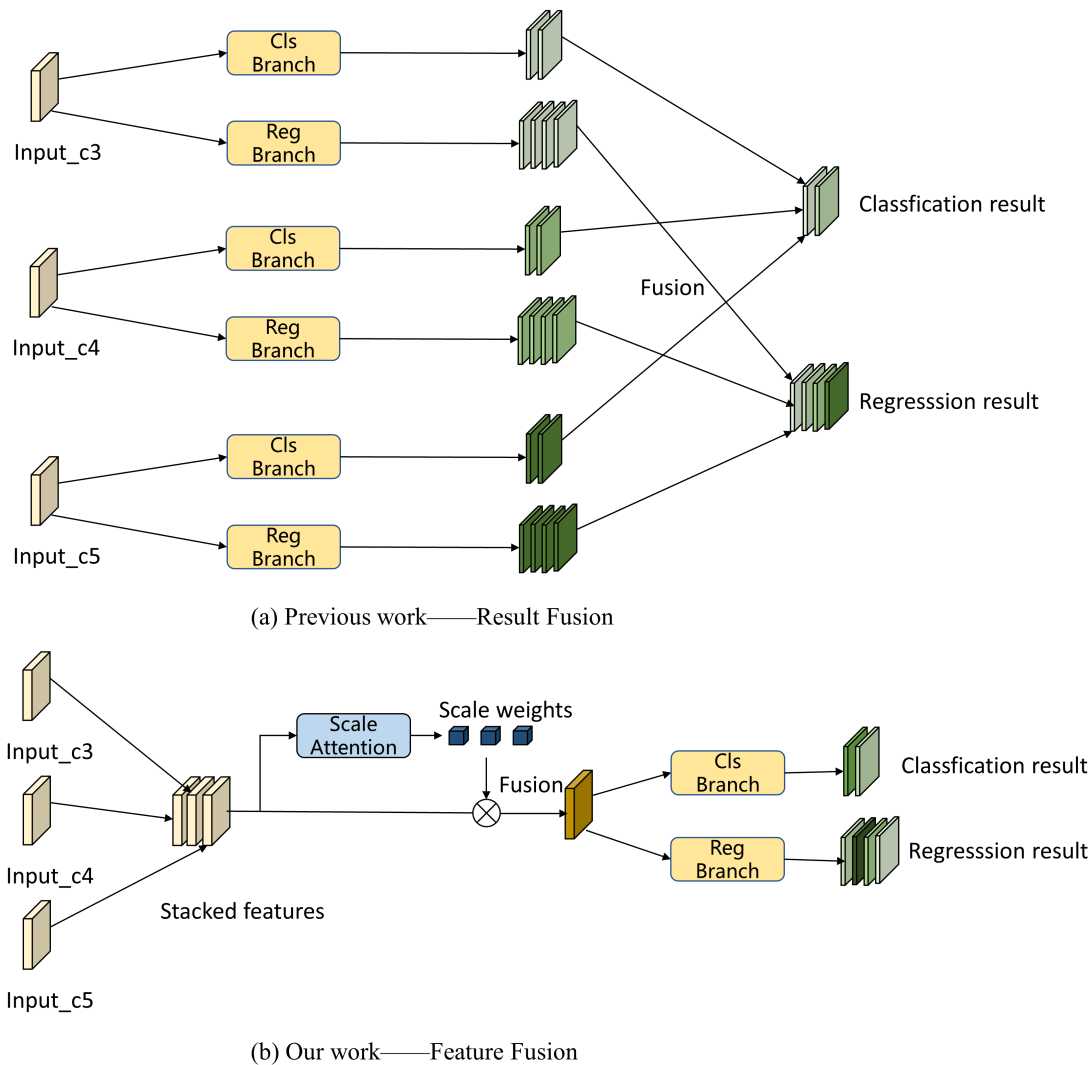


Figure 1. Comparison of previous work and our work. (a) Previous works use result fusion to fuse the result maps and obtain the tracking results. The classification branch and regression branch first decode the features of the similarity response map to obtain the result map, and then fuse it. (b) Our work uses a novel scale attention to fuse the multi-level similarity response maps. Subsequently, the two branches decode it to produce the tracking results.

Therefore, we propose a more valid and explainable method named Siamese Feature Fusion Network (SiamFFN) for handling this. The framework consists of backbone, similarity matching module, and feature fusion head. Unlike the previous tracking head, feature fusion head fuses the feature maps. As shown in Figure 1b, a scale attention is proposed to fuse multi-level similarity response maps based on their semantic importance. Then, the fused similarity response map is feature decoded to obtain the tracking results.

Moreover, the classification and regression branches are separate, leading to feature misalignment in the resulting output feature maps. Siamese network-based trackers lack a direct structural connection between the classification and regression branches, which are optimized independently of each other. However, the regression branch generates the corresponding prediction bounding box based on the classification result map during the tracking phase. As a result, there is a large number of inconsistent predictions during the inference stage, where the predictions often have high classification scores but less accurate regression bounding boxes. As shown in Figure 2, the blue bounding box has a higher classification score than the red bounding box, leading to the blue bounding box being outputted as the final tracking result. However, the red bounding box is more accurate in terms of tracking results. Therefore, we use a Classification Supervised Regression Loss (CSRL), which enables joint optimization of both two branches.

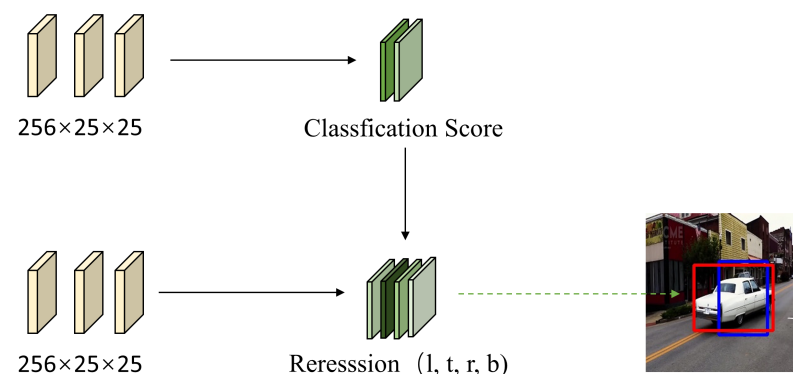


Figure 2. The flow chart of the tracing head during inference. The blue bounding box has a higher classification score than the red bounding box, but the red bounding box has significantly more accurate tracking results.

In summary, this letter makes three contributions:

1. We design a Siamese Feature Fusion Network (SiamFFN), which uses the way of feature fusion to obtain the final tracking results. It is more scientific and effective than the previous approach of using result fusion.
2. We use a Classification Supervised Regression Loss (CSRL) to alleviate the misalignment between two branches. This enables the model to produce more accurate predictions.
3. An empirical evaluation on multiple challenging benchmark datasets validates the superior performance of SiamFFN over several state-of-the-art trackers, demonstrating its effectiveness in achieving top-performing results.

The rest of this article is structured as follows. Section 2 provides a review of three parts: Siamese networks based tracker, attentional mechanisms and feature alignment. Section 3 describes Siamese Feature Fusion Network (SiamFFN) and Classification Supervised Regression Loss (CSRL). Section 4 presents a comprehensive evaluation of the performance of SiamFFN compared to other state-of-the-art trackers. Additionally, we conduct ablation experiments to prove the effectiveness of our SiamFFN and CSRL. Finally, we draw conclusions.

2. Related Work

This section focus on three aspects: Siamese network-based trackers, attentional mechanisms and feature alignment.

2.1. Siamese Network-Based Trackers

Siamese network-based trackers [4–7] have made significant breakthroughs in visual object tracking recently. As one of the pioneering works, SiamFC [4] extracts features using a modified AlexNet [16], which removes the padding and fully connected layers

In addition, SiamFC proposes the cross-correlation (Xcorr) layer for correlation operation. Then, the researchers go on to build some revised Siamese methods [9–11] on the basis of this Siamese framework. DSiam [9] proposes dynamic Siamese networks which can learn target appearance changes and background suppression. RASNet [10] introduces spatial attention and channel attention mechanisms. However, because these trackers are all built on SiamFC's framework, they can only achieve multi-scale search by inputting images at multiple scales to account for scale variation. Such processing requires additional computational effort and the accuracy achieved is not very high.

Then, the region proposal network (RPN) [12] is introduced by SiamRPN [5]. RPN is made up of two branches: a classification branch using to distinguish between the target's foreground and background and a regression branch using to regress the bounding box. In addition, SiamRPN also introduces the up-channel cross correlation (Up-Xcorr) layer. After that, SiamRPN++ [6] deepens the Siamese network. It removes the stride from the last two blocks of ResNet [15] and adds the dilated convolution [17]. Apart from this, SiamRPN++ also proposes a depth-wise cross correlation (DW-Xcorr) layer, which has become a popular way to calculate similarity.

SiamRPN++ refines the basic framework of Siamese network-based trackers, and most of the subsequent trackers are improved with this framework. C-RPN [18] proposes to solve the class imbalance problem by cascading a series of RPNs in a Siamese network from deep layers to shallow layers. Some other studies concludes that RPN must rely on a huge number of hyperparameters related to the anchors, which considerably decreases the tracker's generalization performance. To solve them, the anchor-free method is proposed, such as SiamBAN [7], SiamFC++ [19] and Ocean [20]. In this way, they can remove the inconvenient anchor hyperparameters. Recently, transformer [13] gains popularity in computer vision. Transformer is completely built on the attention mechanism, with no levels of convolutional or recurrent neural networks. Based on it, TransT [14] proposes a novel attention-based feature fusing network.

However, most of the existing Siamese network-based trackers [6,7] use result fusion to obtain the final tracking results. These result fusion trackers are not structurally sound. In contrast to them, our Siamese Feature Fusion Network (SiamFFN) uses feature fusion in its structural design. Specifically, we design a novel scale attention to fuse multi-level feature maps according to their semantic importance. Furthermore, we use a Classification Supervised Regression Loss (CSRL) to facilitate feature alignment between the two branches. The related work about them is reviewed in section B and section C.

2.2. Attentional Mechanisms

Attention mechanism can be described as an algorithm for dynamic weight modification based on the input image features. It excels at a lot of visual tasks since its debut. SENet [21] presents the idea of squeeze-and-excite (SE) block as the first approach. GSoP-Net [22] adds a global second-order pooling to the fundamental global average pooling to simulate higher-order statistics. ECANet [23] works to minimize the complexity of the excitation module by determining channel interaction using a 1D convolution. Recently, transformer [13] has achieved remarkable performance in various tasks by utilizing self-attention mechanism.

Visual object tracking also makes use of attention. RASNet [10] applies the attention method suggested by SENet to Siamese network-based trackers. Its main technique involves leveraging the attention mechanism to enhance the representation of feature maps. Based on transformer influence, TransT [14] proposes a novel attention-based feature fusing network.

Our proposed scale attention differs from previous approaches in the following ways: (1) The motivations are different. Our scale attention is proposed to fuse the similarity response map, while most of the previous approaches use attentional mechanisms to improve the representation of features. (2) The implementations are different from previous work [24]. We apply global average pooling to extract the spatial information from the

feature map, and then integrate all channels together by convolution. Finally, we obtain the scale weights based on semantic information by activating with ReLU and hard Sigmoid. We can accordingly perform better feature fusion on the similarity response map.

2.3. Feature Alignment

In most of the object detectors [12,25,26], there is the problem of feature misalignment. To solve this problem, IoU-Net [27] proposes to use the predicted IoU as localization confidence. PISA [28] suggests a Classification-Aware Regression Loss (CARL), with higher classification score gradients in samples with greater regression losses. Harmonic loss [29] proposes that classification and regression branches can oversee each other's optimization during training.

The same problem exists in most of the tracking heads. SiamFC++ [19] uses a new branch to predict the centerness [30]. SiamRCR [31] suggests a technique for combining classification and regression losses. However, it still needs to add an additional localization branch to predict the localization accuracy. Recently, some studies shift their focus towards addressing this issue by employing loss functions. Following it [32], we use Classification Supervised Regression Loss (CSRL) to optimize the joint classification branch and regression branch.

3. Method

In this section, a novel feature fusion approach is proposed to design the tracker's tracking head, and our model is called Siamese Feature Fusion Network (SiamFFN). Moreover, we suggest using a Classification Supervised Regression Loss (CSRL) to solve the problem of feature misalignment.

3.1. Siamese Feature Fusion Network

Our baseline model, SiamBAN is a simple yet effective Siamese tracking framework. SiamBAN's tracking head first decodes the features of the three input similarity response maps. After that, the three obtained results are adaptively fused. These three sets of result maps are obtained from multi-level similarity response maps, and they are independent of each other. The use of result fusion is a simple way to combine results, but its effectiveness is limited.

Our SiamFFN is mainly built on the SiamBAN. As shown in Figure 3, SiamFFN consists of Siamese network backbone, similarity matching module, and feature fusion head. The Siamese network was built around the ResNet-50 [15]. To output multi-level features, we eliminate the downsampling operation from the last two blocks and add atrous convolution [33]. Multi-level feature maps are then fed into the similarity-matching module, resulting in three similarity response maps. In the feature fusion head, a novel scale attention mechanism is used to fuse the multi-level similarity response maps. As shown in Figure 4, three input similarity response map F_3 , F_4 , F_5 are concatenated along a new dimension:

$$S = \text{stack}(F_3, F_4, F_5) \quad (1)$$

in which S is the fused similarity response map and *stack* concatenating operation.

For better integration, we propose a scale attention mechanism which can capture the semantic significance of multi-level features. Specifically, global average pooling is first used to obtain the maximum value of each similarity response map. After that, all channels are integrated together by a convolution layer to finally acquire three numbers. Finally, the ReLU function is used to activate it. For any input similarity response map F_i , the scale attention π_s is as follows:

$$\pi_s(F_i) = \delta(\text{Conv}(\text{Avg}(F_i))) \quad (2)$$

in which *Conv* denotes the convolution layer, *Avg* denotes average pooling and δ denotes ReLU function.

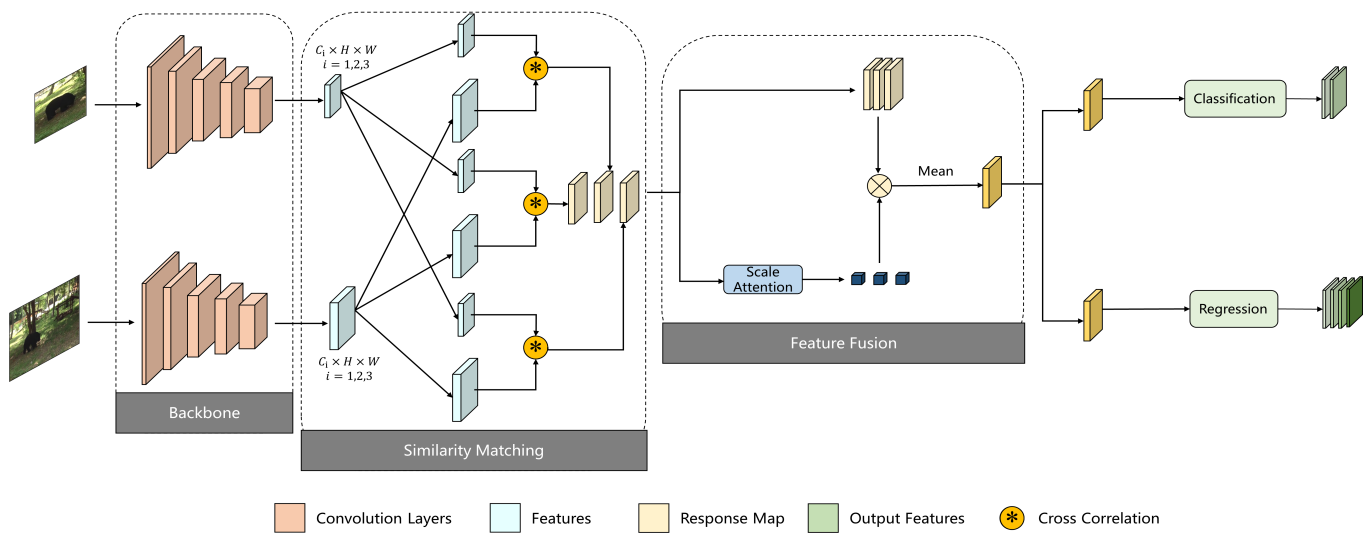


Figure 3. The overall architecture of the proposed SiameSE Feature Fusion Network (SiamFFN). The modified ResNet-50 outputs three different levels of feature maps. DW-Xcorr is responsible for calculating the similarity between them and outputting three similarity response maps. Then, we use a novel scale attention to fuse multi-level feature maps according to their semantic importance. Finally, it is decoded to obtain the final classification and regression maps.

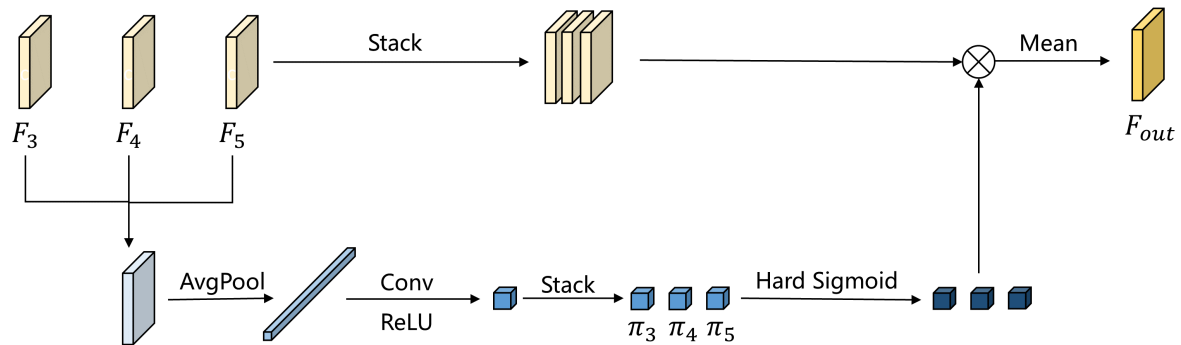


Figure 4. Feature fusion approach.

Similarly, the different levels of scale attention are concatenated together and reactivated by the hard sigmoid function. The equation is as follows:

$$W = \text{hard_sigmoid}(\text{stack}(\pi_s(F_3), \pi_s(F_4), \pi_s(F_5))) \quad (3)$$

$$\text{hard_sigmoid} = \max(0, \min(1, \frac{x+1}{2})) \quad (4)$$

Then, we multiply the similarity response map and scale attention of the corresponding level. Finally, the fused similarity response map is obtained by averaging them. The specific formula is as follows:

$$F_{out} = \text{mean} \sum_{i=3,4,5} S_i W_i \quad (5)$$

in which *mean* denotes averaging operation.

Our proposed feature fusion approach fully considers the semantic features of different levels and fuses them according to their importance.

3.2. Classification Supervised Regression Loss

The tracking head usually contains a classification branch and a regression branch. In 208 the inference phase, the tracking head finds the points with the highest classification scores in the classification map. According to its coordinates, the corresponding bounding

box information is found in the regression map. However, the lack of a connection between the two branches can result in feature misalignment problems. Therefore, we use a Classification Supervised Regression Loss (CSRL), which receives the inspiration from corrective loss [32].

For regression branch, the IoU_Loss [34] is denoted as:

$$IoU_Loss = 1 - IoU \quad (6)$$

where IoU denotes Intersection over Union.

The Cross Entropy (CE_Loss) is used for the classification branch, and is denoted as:

$$CE_Loss = - \sum_i y_i \log p_i \quad (7)$$

in which y_i denotes true value and p_i denotes prediction value.

As previously stated, the optimization between the two branches in the previous method was separate. We observe the tracking phase of existing methods and discover that the majority of them use classification branches to trigger regression branches. Therefore, we use a Classification Supervised Regression Loss. For positive samples x_i , the formula is as follows:

$$L_{pos} = CE_Loss + e^{-CE_Loss} IoU_Loss \quad (8)$$

In particular, we place a variable associated with classification loss in front of regression loss. As a result, a sample with a better classification score will obtain a large weight of the regression loss. Therefore, the two branches will produce more consistent prediction outputs throughout the inference phase, resulting in extraordinarily high localization accuracy.

4. Experiments

We conduct a comprehensive experimental evaluation of the our Siamese Feature Fusion Network (SiamFFN). OTB-2015 [35], VOT2016 [36] and UAV20L [37] are three tracking benchmarks we used in the experiments. To begin, we introduce the datasets used in our experiments and provide details about the training process implementation. Following that, three tracking benchmarks and the associated evaluation metrics are given. At last, we will discuss the comparison and ablation experiments.

4.1. Dataset

In this research, we use GOT-10k [38], COCO [39], ImageNet VID [40] and ImageNet DET [40] to train our SiamFFN. On several well-known tracking benchmarks, including OTB-2015 [35], VOT2016 [36] and UAV20L [37], we test our model. We will first give a quick overview of these datasets.

GOT-10k [38] contains 10,000 videos and contains over 1.5 million manually annotated bounding boxes. It is constructed based on the WordNet structure, which is used to ensure the category balance in the videos.

COCO [39] is a large-scale dataset suitable for various image tasks, containing over 330 K images with annotations for 220 K images. It includes 1.5 million targets, 80 target classes, and 91 material classes.

ImageNet [40] consists of 14,197,122 images and is a large computer vision dataset. It has many sub-datasets with different divisions. Among them, ImageNet VID has a total of 30 categories.

OTB-2015 [35] consists of 100 videos of 22 object categories. The video length of OTB-2015 dataset varies from 71 to 3872 frames, with an average resolution of 356×530 .

VOT2016 [36] includes 60 sequences. Each sequence is labeled by different attributes for each frame, including IV, MOC, SCO, ARC, OCC, and FCM. Sequences are usually 757×480 pixels in quality, with frame sizes varying from 48 to 1507 pixels.

UAV20L [37] consists of 20 long videos depicting 5 distinct object classes created with a flying simulator. The lowest frame count for these sequences is 1717 and the highest frame count is 5527.

4.2. Evaluation Criteria

OTB-2015 [35] and UAV20L [37] use the precision plot and success plot.

Precision Plot. Average Euclidean distance is used to calculate the central location error for each video frame. Depending on the threshold value, different percentage values can be obtained, which allow us to plot the precision values.

Success Plot. In each frame, R_b denotes the predicted bounding box and R_{gt} represents the ground truth. We can calculate the overlapping area between them by following formula:

$$OS = \frac{|R_b \cap R_{gt}|}{|R_b \cup R_{gt}|}. \quad (9)$$

VOT2016 [36] employs three measures in accordance with the VOT evaluation protocols: A, R, and EAO. A (accuracy) indicates the average overlap between the ground truth and the bounding area projected by the tracker during its effective tracking. R (robustness) is used to calculate how many times the tracker misses a subject while tracking. EAO (expected average overlap) estimates the average overlap anticipated by the tracker over a lot of short-term sequences that share the same visual characteristics.

4.3. Implementation Details

Our approach is implemented under PyTorch 1.8.0 framework on a Intel(R) Xeon(R) Silver 4210R CPU (2.40 GHz) along with a Nvidia Geforce RTX 3090 GPU. The backbone is modified ResNet-50 [15]. To train SiamFFN, we use Classification Supervised Regression Loss. The entire network is trained for 20 epochs using Stochastic Gradient Descent (SGD) with a momentum of 0.9 and a batch size of 80. During the first five warm-up epochs, the learning rate varies from 0.001 to 0.005. For the remaining 15 epochs, the learning rate ranges from 0.005 to 0.00005.

4.4. Comparison on Public Benchmarks

OTB-2015 [35]. We evaluate our tracker against nine state-of-the-art methods including TCTrack++ [41], TransT [14], Stark [42], SiamBAN [7], SiamRPN [5], SiamDW [43], DaSiamRPN [44], SiamFC [4] and DeepSRDCF [45]. Figure 5 illustrates that our tracker outperforms existing state-of-the-art trackers, achieving results of 0.694 and 0.903 on the success plot and precision plot, respectively.

VOT2016 [36]. As shown in Table 1, SiamFFN achieves the best EAO (0.501) and Robustnes (0.131) on the VOT2016 dataset while PrDiMP achieves the best Accuracy. Compared to the SiamBAN, our tracker has a reduction of 0.13 on Robustnes. This shows that our tracker is able to have better robustness compared to the previous tracker while maintaining good tracking accuracy.

Table 1. Tracking results on VOT2016 dataset.

Tracker	EAO	Accuracy	Robustness
SiamFC [4]	0.277	0.549	0.382
SiamRPN [5]	0.344	0.560	1.12
SiamRPN++ [6]	0.370	0.580	0.240
ECO [46]	0.374	0.546	11.67
ATOM [47]	0.424	0.617	0.190
SiamR-CNN [48]	0.461	0.645	0.173
PrDiMP [49]	0.476	0.652	0.140
SPM [50]	0.481	0.610	0.206
SiamBAN [7]	0.491	0.627	0.144
Ours	0.501	0.625	0.131

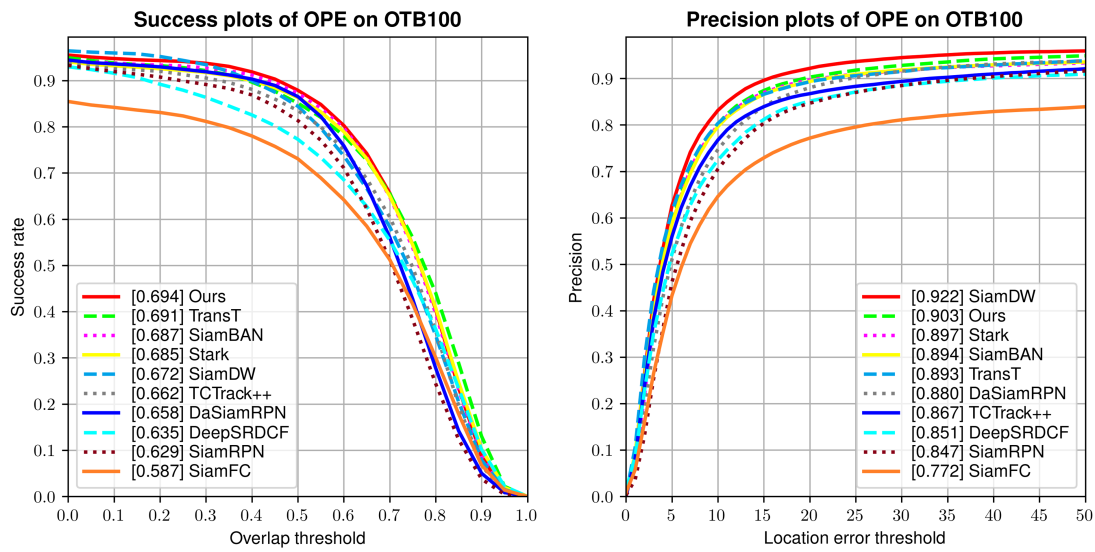


Figure 5. Tracking results on OTB-2015.

UAV20L [37]. Our tracker is compared to the nine best-performing trackers on UAV20L, including HiFT [51], SiamAPN++ [52], SiamAPN [53], SiamBAN [7], SiamRPN [5], SiamFC [4], BACF [54], ECO [46] and STRCF [55]. As shown in Figure 6, our tracker outperforms most other state-of-the-art trackers, achieving scores of 0.575/0.747 on the success plot and precision plot, respectively. HiFT borrows a idea of transformer and adds position encoding in the process of feature fusion, so the effect is better than our tracker on precision plot.

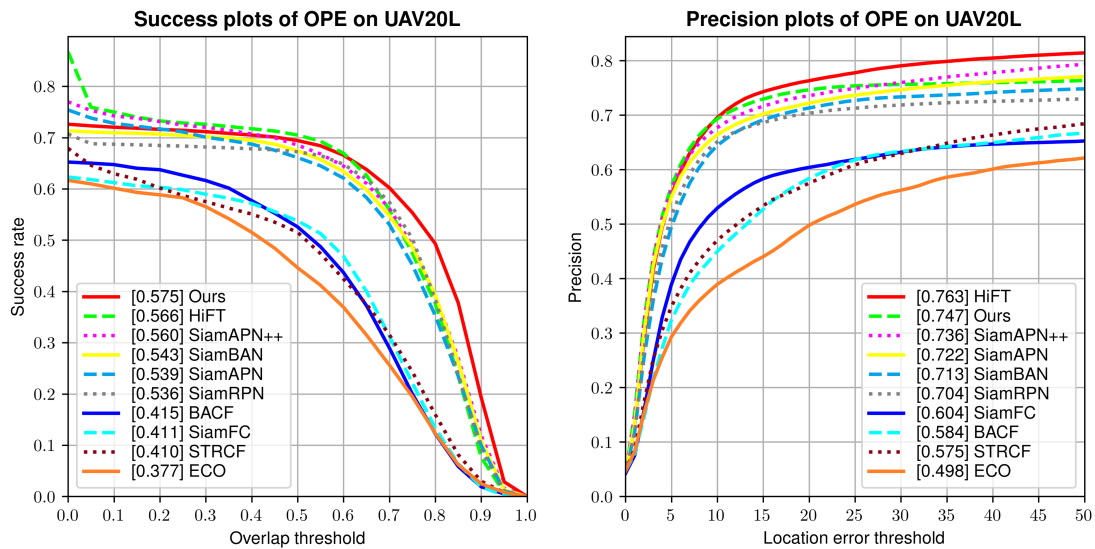


Figure 6. Tracking results on UAV20L.

Overall, our tracker outperforms the competition on several tracking benchmark datasets, including the traditional target tracking datasets OTB-2015 and VOT2016, as well as the UAV aerial photography datasets UAV20L. To show our tracker's superiority, we compare its performance on several datasets with that of the baseline tracker. As shown in Figure 7, we perform validation on OTB-2015 and UAV20L dataset. In the visualizations, the ground truth bounding boxes are shown in green, while the tracking results generated by previous methods, SiamBAN and our proposed SiamFFN, are represented by blue and red bounding boxes, respectively. We have also conducted a computational complexity analysis, which is summarized in Table 2. Compared with other result fusion trackers, our proposed SiamFFN has a smaller number of parameters and requires less

computational effort. Furthermore, SiamFFN demonstrates comparable speed to SiamBAN, but outperforms SiamRPN++ in terms of processing time.

OTB-2015



UAV20L



Ground Truth SiamBAN SiamFFN

Figure 7. Visualization of tracking results on videos from different datasets.

Table 2. Analysis of Computational Complexity.

Trackers	Flops (G)	Params (M)	FPS
SiamFC	5.05	3.1	100
SiamRPN	9.23	22.63	160
SiamRPN++	59.56	53.95	35
SiamBAN	59.59	53.9	40
SiamFFN	57.02	47.64	40

4.5. Ablation Study

We conduct an ablation study on the SiamFFN and baseline tracker. To conduct better ablation experiments, we will not be utilizing the model provided by the authors of the research. Instead, we will be reproducing SiamBAN using the four training datasets that we used. Finally, we evaluate its performance on the UAV20L dataset.

As shown in Table 3, SiamBAN achieves 0.543/0.713 on success plot and precision plot. Our proposed SiamFFN achieves 0.561/0.720 on success plot and precision plot, which improves by 0.018/0.019. This is a good proof that SiamFFN using feature fusion is better than SiamBAN using result fusion. In addition, we train SiamBAN and SiamFFN using Classification Supervised Regression Loss, respectively. As shown in Table 3, SiamBAN's performance on the UAV20L dataset improved by 0.014/0.007 with the help of CSRL. After training SiamFFN with CSRL, it improves to 0.575/0.747 on success plot and precision plot. In addition, we evaluate these four trackers in terms of each attribute of the UAV20L dataset. As shown in Figure 8, our tracker exhibits superior performance across multiple attributes, suggesting the remarkable robustness of our model in tackling diverse challenges.

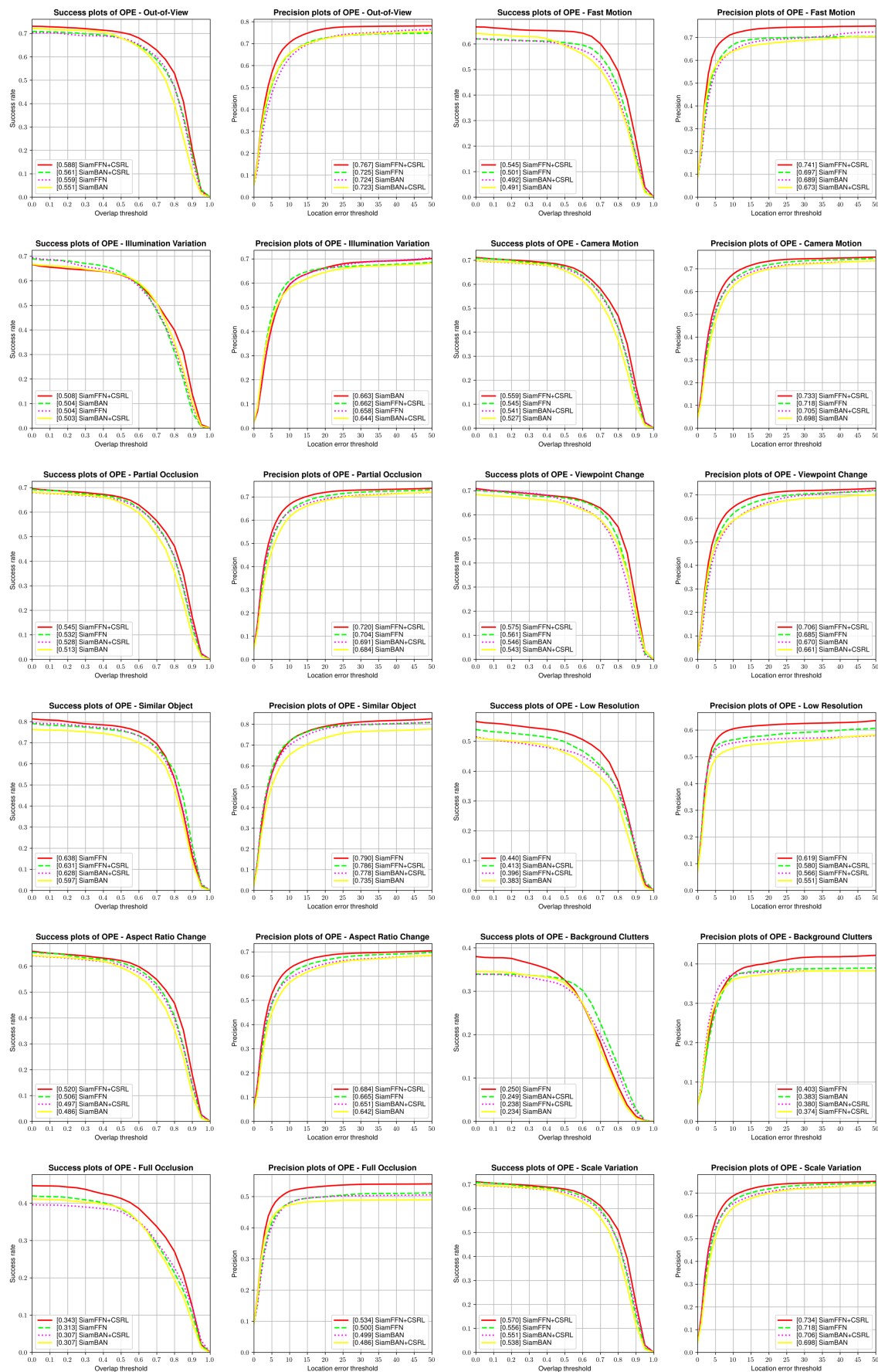


Figure 8. Tracking output of attribute analysis on UAV20L.

Table 3. Ablation study on UAV20L.

Tracker	Success Rate	Precision Rate
SiamBAN	0.543	0.713
SiamFFN	0.561	0.732
SiamBAN + CSRL	0.557	0.720
SiamFFN + CSRL	0.575	0.747

Improvement of Classification Supervised Regression loss (CSRL): CSRL optimizes both the classification and regression branches together, resulting in more consistent classification scores and regression bounding boxes. We use CSRL to train SiamBAN, and the tracking results are improved, reaching 0.557/0.720 on success plot and precision plot. As shown in Table 4, CSRL helps to improve the success rate of 0.014 and the precision rate of 0.007. We also apply it to the training process of multiple trackers. As shown in Table 4, training SiamFFN and SiamRPN++ with CSRL can further improve their performance. The success rate of SiamFFN improved by 0.014, and the precision rate improved by 0.015. The success rate of SiamRPN++ improved by 0.013, and the precision rate improved by 0.038.

Table 4. Ablation study of CSRL.

Tracker	Success Rate	Precision Rate
SiamBAN	0.543	0.713
SiamBAN + CSRL	0.557	0.720
SiamFFN	0.561	0.732
SiamFFN + CSRL	0.575	0.747
SiamRPN++	0.528	0.696
SiamRPN++ + CSRL	0.541	0.734

5. Conclusions

In this paper, we present a Siamese network framework for efficient object tracking. Specifically, we introduce a feature fusion head that fully takes into account multi-level semantic features and merges them based on their significance. Furthermore, we use a Classification Supervised Regression Loss to optimize both classification and regression branches. Experimental results on three tracking benchmarks shows that our proposed Siamese Feature Fusion Network (SiamFFN) achieves state-of-the-art performance, running at 40 fps on a Nvidia RTX 3090, confirming its effectiveness and efficiency.

Author Contributions: Conceptualization, M.Y., K.C., Y.Y. and J.B.; investigation and analysis, M.Y., K.C. and Y.Y.; software, J.B.; validation, J.B.; writing—original draft preparation, J.B. and M.Y.; writing—review and editing, J.B., K.C. and Y.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

SiamFFN	Three Siamese Feature Fusion Network
DW-Xcorr	Depth-wise crosscorrelation
CSRL	Classification Supervised Regression Loss
IoU	Intersection over Union
CARL	Classification-Aware Regression Loss
EAO	Expected Average Overlap

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. Guo, S.; Rigall, E.; Ju, Y.; Dong, J. 3d hand pose estimation from monocular rgb with feature interaction module. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 5293–5306. [\[CrossRef\]](#)
3. Gao, M.; Jin, L.; Jiang, Y.; Guo, B. Manifold siamese network: A novel visual tracking ConvNet for autonomous vehicles. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 1612–1623. [\[CrossRef\]](#)
4. 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, 11–14 October 2016; pp. 850–865.
5. 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.
6. Li, B.; Wu, W.; Wang, Q.; Zhang, F.; Xing, J.; Yan, J. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4282–4291.
7. Chen, Z.; Zhong, B.; Li, G.; Zhang, S.; Ji, R. Siamese box adaptive network for visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6668–6677.
8. Bromley, J.; Guyon, I.; LeCun, Y.; Säckinger, E.; Shah, R. Signature verification using a “siamese” time delay neural network. *Adv. Neural Inf. Process. Syst.* **1993**, *6*, 737–739. [\[CrossRef\]](#)
9. 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.
10. Wang, Q.; Zhang, M.; Xing, J.; Gao, J.; Hu, W.; Maybank, S.J. Do not lose the details: Reinforced representation learning for high performance visual tracking. In Proceedings of the 27th International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018.
11. 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.
12. 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*, 1–14. [\[CrossRef\]](#) [\[PubMed\]](#)
13. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–15.
14. 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, Nashville, TN, USA, 20–25 June 2021; pp. 8126–8135.
15. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
16. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1–9. [\[CrossRef\]](#)
17. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
18. Fan, H.; Ling, H. Siamese cascaded region proposal networks for real-time visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7952–7961.
19. Xu, Y.; Wang, Z.; Li, Z.; Yuan, Y.; Yu, G. Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12549–12556.
20. Zhang, Z.; Peng, H.; Fu, J.; Li, B.; Hu, W. Ocean: Object-aware anchor-free tracking. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 771–787.
21. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
22. Gao, Z.; Xie, J.; Wang, Q.; Li, P. Global second-order pooling convolutional networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3024–3033.

23. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020.
24. Liu, Y.; Ju, Y.; Jian, M.; Gao, F.; Rao, Y.; Hu, Y.; Dong, J. A deep-shallow and global-local multi-feature fusion network for photometric stereo. *Image Vis. Comput.* **2022**, *118*, 104368. [[CrossRef](#)]
25. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
26. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
27. 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.
28. Cao, Y.; Chen, K.; Loy, C.C.; Lin, D. Prime sample attention in object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11583–11591.
29. Wang, K.; Zhang, L. Reconcile Prediction Consistency for Balanced Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 10–17 October 2021; pp. 3631–3640.
30. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9627–9636.
31. Peng, J.; Jiang, Z.; Gu, Y.; Wu, Y.; Wang, Y.; Tai, Y.; Wang, C.; Lin, W. Siamrcr: Reciprocal classification and regression for visual object tracking. *arXiv* **2021**, arXiv:2105.11237.
32. Bao, J.; Chen, K.; Sun, X.; Zhao, L.; Diao, W.; Yan, M. SiamTHN: Siamese Target Highlight Network for Visual Tracking. *arXiv* **2023**, arXiv:2303.12304.
33. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [[CrossRef](#)]
34. Yu, J.; Jiang, Y.; Wang, Z.; Cao, Z.; Huang, T. Unitbox: An advanced object detection network. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 516–520.
35. Wu, Y.; Lim, J.; Yang, M.H. Object Tracking Benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1834–1848. [[CrossRef](#)]
36. Kristan, M.; Matas, J.; Leonardis, A.; Vojř, T.; Pflugfelder, R.; Fernandez, G.; Nebehay, G.; Porikli, F.; Čehovin, L. A novel performance evaluation methodology for single-target trackers. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 2137–2155. [[CrossRef](#)]
37. Mueller, M.; Smith, N.; Ghanem, B. A benchmark and simulator for uav tracking. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 445–461.
38. 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](#)]
39. 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.
40. 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](#)]
41. Cao, Z.; Huang, Z.; Pan, L.; Zhang, S.; Liu, Z.; Fu, C. TCTrack: Temporal Contexts for Aerial Tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 14798–14808.
42. 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, Montreal, BC, Canada, 10–17 October 2021; pp. 10448–10457.
43. Zhang, Z.; Peng, H. Deeper and wider siamese networks for real-time visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4591–4600.
44. 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.
45. Danelljan, M.; Hager, G.; Shahbaz Khan, F.; Felsberg, M. Convolutional features for correlation filter based visual tracking. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Santiago, Chile, 7–13 December 2015; pp. 58–66.
46. 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.
47. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. Atom: Accurate tracking by overlap maximization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4660–4669.
48. Voigtlaender, P.; Luiten, J.; Torr, P.H.; Leibe, B. Siam r-cnn: Visual tracking by re-detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6578–6588.
49. Danelljan, M.; Gool, L.V.; Timofte, R. Probabilistic regression for visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 7183–7192.
50. Wang, G.; Luo, C.; Xiong, Z.; Zeng, W. Spm-tracker: Series-parallel matching for real-time visual object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3643–3652.

51. Cao, Z.; Fu, C.; Ye, J.; Li, B.; Li, Y. HiFT: Hierarchical Feature Transformer for Aerial Tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 10–17 October 2021; pp. 15457–15466.
52. Cao, Z.; Fu, C.; Ye, J.; Li, B.; Li, Y. SiamAPN++: Siamese attentional aggregation network for real-time uav tracking. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 3086–3092.
53. Fu, C.; Cao, Z.; Li, Y.; Ye, J.; Feng, C. Siamese anchor proposal network for high-speed aerial tracking. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 510–516.
54. Kiani Galoogahi, H.; Fagg, A.; Lucey, S. Learning background-aware correlation filters for visual tracking. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1135–1143.
55. Li, F.; Tian, C.; Zuo, W.; Zhang, L.; Yang, M.H. Learning spatial-temporal regularized correlation filters for visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4904–4913.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.