


## Article

# MultiRPN-DIDNet: Multiple RPNs and Distance-IoU Discriminative Network for Real-Time UAV Target Tracking

Li Zhuo <sup>1,2</sup>, Bin Liu <sup>1</sup>, Hui Zhang <sup>1,2,\*</sup> , Shiyu Zhang <sup>1</sup> and Jiafeng Li <sup>1,2</sup>

<sup>1</sup> Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China; zhuoli@bjut.edu.cn (L.Z.); lbet@emails.bjut.edu.cn (B.L.); S201739017@emails.bjut.edu.cn (S.Z.); lijiafeng@bjut.edu.cn (J.L.)

<sup>2</sup> Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing University of Technology, Beijing 100124, China

\* Correspondence: huizhang@bjut.edu.cn

**Abstract:** Target tracking in low-altitude Unmanned Aerial Vehicle (UAV) videos faces many technical challenges due to the relatively small sizes, various orientation changes of the objects and diverse scenes. As a result, the tracking performance is still not satisfactory. In this paper, we propose a real-time single-target tracking method with multiple Region Proposal Networks (RPNs) and Distance-Intersection-over-Union (Distance-IoU) Discriminative Network (DIDNet), namely MultiRPN-DIDNet, in which ResNet50 is used as the backbone network for feature extraction. Firstly, an instance-based RPN suitable for the target tracking task is constructed under the framework of Simas Neural Network. RPN is to perform bounding box regression and classification, in which channel attention mechanism is integrated to improve the representative capability of the deep features. The RPNs built on the Block 2, Block 3 and Block 4 of ResNet50 output their own Regression (Reg) coefficients and Classification scores (Cls) respectively, which are weighted and then fused to determine the high-quality region proposals. Secondly, a DIDNet is designed to correct the candidate target's bounding box finely through the fusion of multi-layer features, which is trained with the Distance-IoU loss. Experimental results on the public datasets of UAV20L and DTB70 show that, compared with the state-of-the-art UAV trackers, the proposed MultiRPN-DIDNet can obtain better tracking performance with fewer region proposals and correction iterations. As a result, the tracking speed has reached 33.9 frames per second (FPS), which can meet the requirements of real-time tracking tasks.

**Keywords:** visual object tracking; channel attention mechanism; region proposal network; DIoU discriminative network; unmanned aerial vehicle (UAV) videos



**Citation:** Zhuo, L.; Liu, B.; Zhang, H.; Zhang, S.; Li, J. MultiRPN-DIDNet: Multiple RPNs and Distance-IoU Discriminative Network for Real-Time UAV Target Tracking. *Remote Sens.* **2021**, *13*, 2772. <https://doi.org/10.3390/rs13142772>

Academic Editor: Andrzej Staczny

Received: 13 June 2021

Accepted: 12 July 2021

Published: 14 July 2021

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



**Copyright:** © 2021 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

In recent years, due to their many outstanding advantages in performance and cost, unmanned aerial vehicles (UAVs) have increasingly been deployed in many fields, such as security monitoring, disaster relief, agriculture, military equipment, sports and entertainments, etc. Correspondingly, a huge amount of visual data has been produced, and the demand for intelligent processing of UAV videos has increased significantly.

Due to the release of new benchmark datasets and the improved methodologies, single-target tracking has become a research hotspot, and the related work has made considerable advances. From the perspective of technical means, the current mainstream single-target trackers can be divided into two categories: trackers based on Discriminative Correlation Filter (DCF) and trackers based on deep learning. Minimum Output Sum of Squared Error (MOSSE) is one of the most representative trackers based on DCF [1]. These kind of trackers have fast tracking speed and are easy to transplant to the embedded hardware platform for real-time processing, but the tracking accuracy is relatively low. Therefore, it is difficult for them to meet the high-accuracy tracking requirements. Afterwards, the researchers

proposed various improved DCF-based trackers through optimizing in many aspects, such as Circulant Structure of tracking-by-detection with Kernels (CSK) tracker [2], Kernelized Correlation Filters (KCF) tracker [3], and Spatially Regularized Discriminative Correlation Filter (SRDCF) tracker [4], etc. These trackers achieve a significant improvement in tracking accuracy, but at the same time, the tracking speed is significantly reduced.

With the rapid development of deep learning, many trackers based on Convolution Neural Networks (CNN) have emerged. Compared to the previous trackers, they can yield higher tracking accuracy [5–13]. However, for UAV target tracking scenarios, due to numerous challenges, such as relatively small object sizes and various orientation changes, the above trackers show degraded performance to different degrees. An accurate and efficient tracker is still needed to perform the target tracking task in UAV videos.

In this paper, we propose a real-time target tracking method based on multiple Region Proposal Networks (RPNs) and Distance-IoU Discriminative Network (DIDNet), namely MultiRPN-DIDNet. First, the channel attention mechanism is incorporated into RPN [14], constructing a network suitable for the target tracking task. Multiple RPNs are combined together to determine the high-quality region proposals. Secondly, a discriminative network is proposed under the framework of Siamese Neural Network (SNN) to finely correct the position of the region proposals, in which multi-layer convolution feature fusion strategy is adopted, and Distance-IoU (DIOU) loss is used to train the network to finely model the spatial relationship of the image patches.

The main contributions of this work are summarized as follows:

- The SNN-based multiple RPNs with channel attention mechanism for the target tracking tasks is constructed, which can select the high-quality region proposals.
- A DIOU discriminative network with multi-layer feature fusion is designed, which is trained with DIOU loss to finely model the spatial relationship of image patches.
- Extensive experiments on the standard UAV tracking benchmark datasets indicate that the proposed tracker can achieve a good balance between the tracking accuracy and speed. At the same time, the tracking speed can reach 33.9 frames per second (FPS), which can meet the requirements of real-time tracking on a general GPU platform.

For the rest of the paper, a brief overview of the related work is given in Section 2. Section 3 introduces our Multiple RPNs and Distance-IoU Discriminative Network. In Section 4, we evaluate the performance of our proposed tracker in several UAV tracking benchmarks. Finally, the paper is concluded in Section 5.

## 2. Related Work

Single target tracking is one of the important tasks in intelligent processing of UAV videos. In this paper, we focus on the study of single target tracking. Visual tracking has been widely studied. In this section, only some representative work is sampled and the work closely related to our work is discussed.

### 2.1. SNN Based Trackers

SNN-based trackers have attracted significant attention for their good tradeoff between the tracking accuracy and efficiency. These trackers formulate visual tracking as a cross correlation problem and obtain a good performance through end-to-end learning. A Y-shaped CNN that combines two network branches, one for the object template and the other for the search region, is constructed to learn a similarity map through cross-correlation.

Siamese Region Proposal Network (SiamRPN) [15] is one of the most representative SNN based trackers, which converts the tracking problem into a local area detection problem. It yields an outstanding tracking performance. Afterwards, many improved variants have been proposed continuously.

Zhu et al. [10] proposed a data enhancement method based on SiamRPN and constructed a new tracker, namely Distractor-aware SiamRPN (DaSiamRPN), which effectively improves the tracking accuracy. Ren et al. [16] designed an SNN model that can be trained online based on SiamRPN from the perspective that SNN only uses the reference frames

without updating model parameters. The Squeeze and Excitation (SE) [17] module is added to enhance the effective feature channels for the tracking task and suppress the ineffective feature channels. Thereby, the representative capability of the features can be improved, and thus, achieving a higher performance.

Zhang et al. [18] studied the factors that affect the tracking accuracy of SNN. They found that the filling operation in the convolution process would have a negative impact on the tracking performance. Therefore, they proposed a Cropping-Inside Residual (CIR) unit, which successfully trains the SiameseFC and SiamRPN. Li et al. [12] proposed the SiamRPN++ tracker, in which a simple yet effective spatial-aware sampling strategy is proposed to train the ResNet network [19]. Moreover, they proposed a novel model architecture to perform layer-wise and depth-wise aggregations, which not only further improves the accuracy but also reduces the model size.

Siamese Fully Convolutional (SiameseFC) [8] uses AlexNet as the backbone network for feature extraction and adopts a fully convolutional network to obtain a good balance between the computational efficiency and tracking performance. Residual Attentional Siamese Network (RASNet) [20] incorporates spatial attention, channel attention, and residual attention mechanisms into SiameseFC to further optimize the tracking accuracy and robustness.

In general, the SNN-based trackers can achieve relatively robust tracking performance when dealing with the variations of target scale, aspect ratio, and rotation in UAV videos.

## 2.2. Classification CNN Based Trackers

The basic idea of this type of trackers [21–24] is to divide the video frame into the background and the target area, so the target tracking problem is transformed into a classification problem. Multi-domain Network (MDNet) tracker [21] adopts a shallow CNN consisting of three convolutional layers and three fully connected (FC) layers, by using a multi-domain learning strategy to improve the tracking accuracy. But due to the high computational complexity, it is difficult to realize real-time processing.

MobileNet-based tracking by detection (MBMD) tracker [23] combines the SiamRPN and MDNet together to generate a large number of region proposals through RPN, which are then sent to the verification network for classification and scoring to obtain the final tracking results. When the tracking confidence is very low, the target would be found in the whole image through the sliding window.

MS-Faster R-CNN tracker [24] integrates multi-stream (MS) into Faster-R-CNN and combines it with the Simple Online and Real-time Tracking with a Deep Association Metric (Deep SORT) algorithm to achieve real-time tracking capabilities on UAV images.

In summary, with the rapid development of deep learning, the CNN network structure designed for the target tracking task is gradually showing diversification. The current deep-learning based trackers overcome the limitation of only using shallow networks, and also migrate the network models used in the target detection task, and thus, improving the tracking performance. However, in order to meet the real-time processing requirements in practical UAV applications, it is usually necessary to design the specific strategy to reduce the computational complexity of the trackers and cater for the UAV video processing platform.

## 2.3. Trackers for UAV Videos

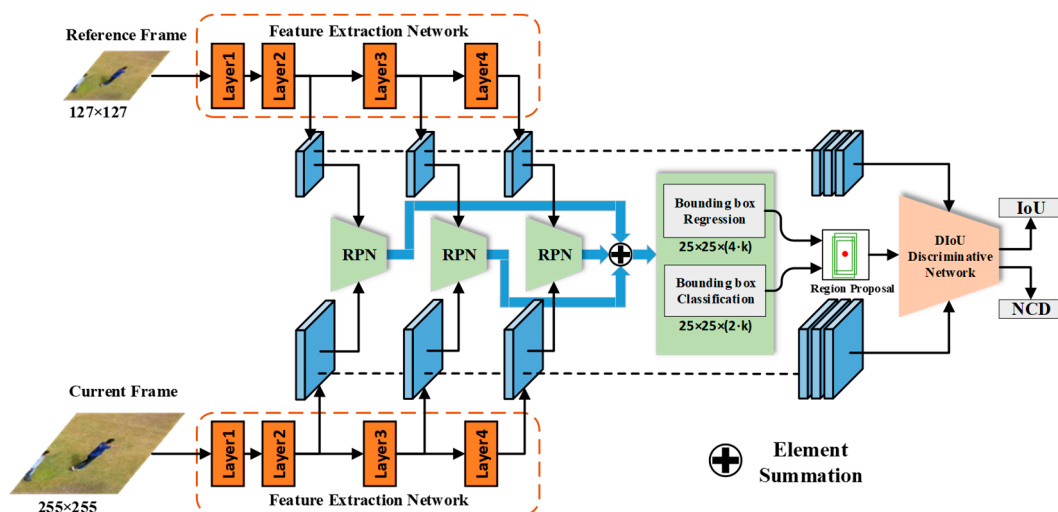
Considering the high processing accuracy and computational efficiency of DCF, most of the existing UAV trackers are implemented based on the DCF algorithm [25–28], so as to meet the real-time requirements of UAV tracking. ARCF [25] designs a cropping matrix and a regularization term to enlarge search region and aberrance repression, effectively improves the robustness and accuracy of the tracker. AutoTrack [26] introduces the spatially local response map variation as spatial regularization, proposed a novel approach to online automatically and adaptively learn spatial regularization term. Ye et al. [27] propose a tracking algorithm based on a multi-regularized correlation filter. The tracker enables

smooth response variations and adaptive channel weight distributions simultaneously, leading to favorable adaption to object appearance variations and enhancement of discriminability. BASCF [28] presents a tracking method specifically designed for UAV, which uses aberrances response suppression mechanism to resist background interference and introduces a log-polar coordinate system to obtain more accurate target state information.

### 3. Proposed Single Object Tracking Method

#### 3.1. Overall Architecture

As shown in Figure 1, the single object tracking method proposed in this paper consists of multiple RPNs and a DIoU discriminative network, in which ResNet-50 [19] is used as the backbone network for feature extraction. RPN is constructed under the framework of SNN to perform bounding box regression and classification. The RPNs built on the Block 2, Block 3 and Block 4 of ResNet50 output their own Reg coefficients and Cls scores respectively. They are weighted and then fused through a set of offline learning weight coefficients, obtaining the final Reg coefficients and Cls scores. The foreground with a higher Cls score is selected as the anchor, and the corresponding region proposal is determined by combining the Reg coefficients of the anchor. The convolutional features from multiple layers of ResNet50 are fused. The fused features and the information of the candidate area are input into the DIoU discriminative network, and the region proposal with the best DIoU value is finally determined as the tracking result.

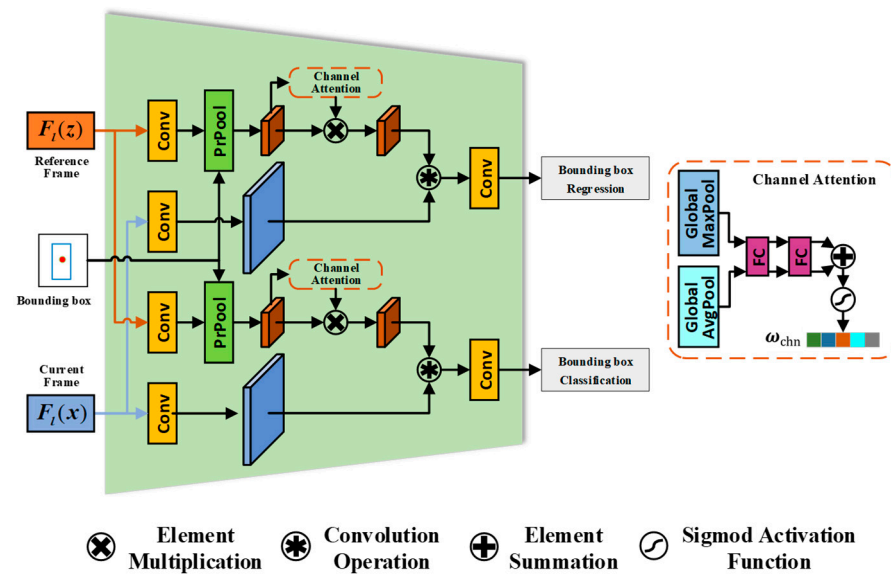


**Figure 1.** Target tracking framework combining multiple RPNs and DIoU discriminative network. The multiple RPNs are used to determine high-quality candidate regions, and the DIoU discriminative network performs the correction of the candidate regions, and then outputs the final result.

In the following section, we will introduce the RPN and DIoU discriminative network in details.

#### 3.2. SNN-Based RPN with Channel Attention Mechanism

RPN takes the CNN features as input and predicts the regression coefficient Reg and classification score Cls for each pre-set anchor frame with different areas and proportions. The Cls score is used to select the classified anchor frame as the foreground. Then the corresponding Reg coefficient and the coordinates of the anchor frame are used to determine the region proposals. However, for the visual target tracking task, more attention is paid to distinguish the target and background defined by the first frame, rather than the target category in target detection task. Therefore, it is necessary to build an instance-based RPN suitable for the target tracking task. The RPN structure based on the SNN framework proposed in this paper is shown in Figure 2.



**Figure 2.** SNN-based RPN structure with channel attention mechanism. The structure includes the branches of bounding box regression and classification. The input is the features of the  $l$ -th layer of the reference frame and the current frame. The bounding box regression branch obtains the regression output of the bounding box coordinates, while the bounding box classification branch obtains the classification scores of the target and the background.

The RPN structure is constructed under the framework of SNN. It contains two branches: bounding box regression and bounding box classification. The CNN features  $F_l(z)$  and  $F_l(x)$  of the reference frame and the current frame are respectively sent to the above two branches. In order to preserve the target information as much as possible and suppress the interference of the background, the features of the reference frame have also undergone Precise Region-Of-Interest (ROI) Pooling (PrPool) operation and channel attention operation. PrPool can perform the pooling operation on the features according to the ROI in the two-dimensional continuous space of each channel, which is specifically expressed as:

$$PrPool(\beta, B) = \frac{\int_{y_1}^{y_2} \int_{x_1}^{x_2} g(x, y) dx dy}{(x_2 - x_1) \times (y_2 - y_1)} \quad (1)$$

where  $(x_1, y_1)$  and  $(x_2, y_2)$  are the coordinates of the upper left and lower right corners of the target bounding box  $B$  respectively, and  $g(\cdot)$  represents the feature quantity operation on feature  $\beta$  after bilinear interpolation operation. The features after channel attention operation are shared by two fully connected layers. The results are element-wise summed. After the Sigmoid activation function, the final channel attention weight  $\omega_{chn}$  can be obtained. Furthermore, a convolution operation is performed between the features of the reference frame and those of the current frame. And the result is adjusted by a convolutional layer.

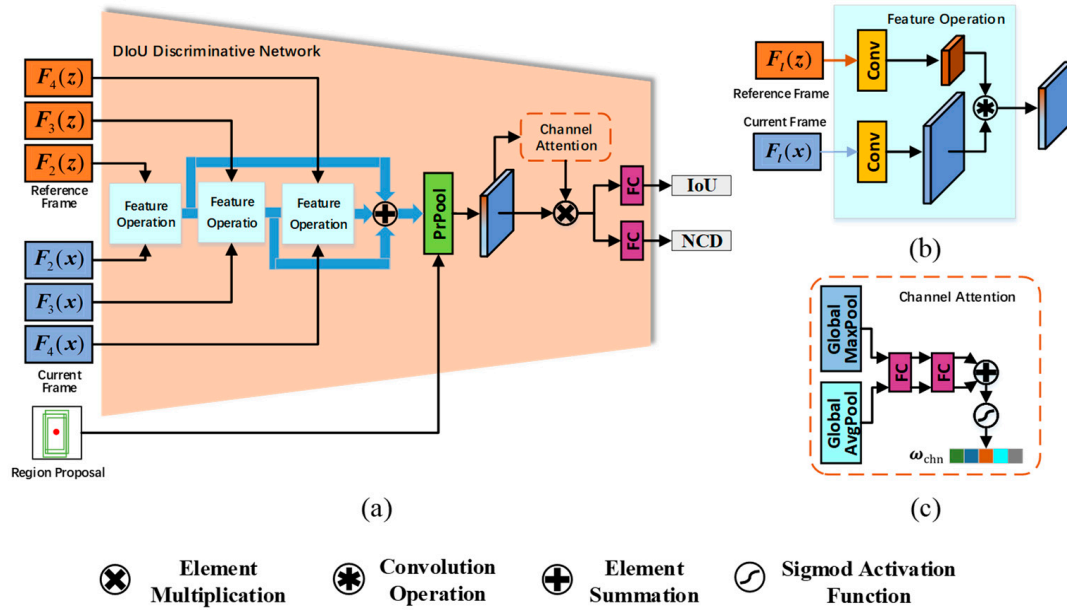
In this paper, the reference frame size is set to  $127 \times 127$ , and the current frame image size is set to  $255 \times 255$  respectively. Corresponding to  $k$  initial anchor boxes, the bounding box regression branch generates the regression output of  $4k$  coordinates, and the bounding box classification branch outputs  $2k$  categories as the classification scores of the target and the background.

### 3.3. DIoU Discriminative Network with Multi-Layer Feature Fusion

IoU can describe the relationship between two bounding boxes and is often used to evaluate the performance of target tracking and target detection tasks. However, the usage of IoU as a loss function in the target tracking task still has some shortcomings. For example, when the target bounding box and the region proposal do not overlap, the IoU value cannot reflect the distance between the two. Moreover, when the target bounding



box and the candidate regions are overlapped and the IoU value is the same, IoU cannot reflect the overlap mode and relative position of the two. To alleviate the shortcomings of IoU, we propose a DIoU discriminative network to describe the positional relationship between the region proposal and the true value of the target bounding box. The DIoU discriminative network can generate different correction results for the region proposal with the same overlapped area with the target bounding box, which is used to finely correct the target position and improve the tracking accuracy. The proposed DIoU discriminative network structure is shown in Figure 3.



**Figure 3.** DIoU discriminative network with Multi-layer feature fusion. DIoU discriminative network fuses the extracted convolutional features of multiple layers, and finally obtains the DIoU score of the region proposal through PrPool operation and channel attention operation.

The procedure of the DIoU discriminative network is described as follows. The features of the reference frame and the current frame are first extracted from Block 2 to Block 4 of ResNet50 and fed into the Feature Operation module, in which they are weighted and fused. The fused features are fed into PrPool module together with the region proposal generated by RPN to obtain the features of the region proposal, which are further weighted by the Channel Attention module to enhance their representative capability. And the output features are fed to different prediction branches of FC to obtain IoU and Normalized Center Distance (NCD) [29].

The structure of the Feature Operation module is shown in Figure 3b, which mainly consists of two convolutional layers and a convolutional operation. The two convolutional layers is to adjust the number of the different channels to the same value, which facilitates the weighted fusion of the feature maps.

The convolutional features extracted from the Block 2 to Block 4 of ResNet50 are fused with the following equation:

$$Fusion(F, \gamma) = \sum_{i,l} \gamma_i F_l(z) * F_l(x) \quad (2)$$

where  $z$  and  $x$  represent the reference frame and current frame respectively.  $\gamma_i$  represents the weight coefficient of the  $i$ -th layer,  $\gamma = (0.3507, 0.5035, 0.1457)$ .  $F_l(\cdot)$  represents the feature extraction process of the  $l$ -th layer. To solve the problem of varIoUs sizes of the feature maps, we introduce the dilated convolution into ResNet-50. By filling 0 elements between each spatial position of the convolution kernel, the convolution kernel is expanded, so that the spatial resolution of convolutional features is the same, while enlarging the

receptive field. The network parameters of the modified ResNet-50 are shown in Table 1. For each image with an input width and height of  $255 \times 255$ , the spatial resolution of the features output by each Block is listed in Table 1.

**Table 1.** Comparison of the spatial resolution of the output feature map of ResNet50 before and after modification.

	Conv 1	Block 1	Block 2	Block 3	Block 4
Original ResNet-50	$128 \times 128$	$64 \times 64$	$32 \times 32$	$16 \times 16$	$8 \times 8$
Modified ResNet-50	$128 \times 128$	$64 \times 64$	$32 \times 32$	$32 \times 32$	$32 \times 32$

The structure of the Channel Attention module is shown in Figure 3c. The module uses both average pooling and max pooling to achieve global pooling, and the pooled features are input to the Multilayer Perceptron (MLP) consisting of two FCs. The two sets of features share the MLP. The output features of the FCs are element-wise summed. The weights of each channel  $\omega_{chn}$  are obtained by the Sigmoid function.

The outputs of the DIoU discriminative network are the IoU value and NCD value of the region proposal, respectively. For the region proposal  $B_t$ , we use Equations (3) and (4) to calculate the IoU score and NCD value respectively:

$$IoU(B_t) = P_{IoU}(\omega_{chn} \cdot PrPool(Fusion(F, \gamma), B_t)) \quad (3)$$

$$NCD(B_t) = P_{NCD}(\omega_{chn} \cdot PrPool(Fusion(F, \gamma), B_t)) \quad (4)$$

where  $Fusion(F, \gamma)$  represents the fused features,  $\omega_{chn}$  represents the output result of the channel attention module,  $P_{IoU}(\cdot)$  and  $P_{NCD}(\cdot)$  represent the calculations of the IoU score and NCD value respectively.

Finally, the DIoU score of the region proposal  $B_t$  is calculated by the following equation:

$$DIoU(B_t) = IoU(B_t) - NCD(B_t) \quad (5)$$

With Equation (5), when the target bounding box and the region proposal do not overlap, DIoU can reflect the distance between the two. When the target bounding box and the region proposal are overlapped and the IoU value is the same, DIoU can better reflect the overlapping mode and relative position of the two. In the tracking process, we calculate the DIoU scores of three region proposals on Block 2, Block 3 and Block 4 of ResNet50 respectively, and select the region proposal with the largest DIoU value as the final tracking result.

### 3.4. Implementation Details

A large number of image samples are required for training the network. In this paper, the reference frame image size is set to  $127 \times 127$ , the current frame image size is set to  $255 \times 255$ , and there is a 10% probability that the two frames do not come from the same video. The experimental results in [10] have proved that this strategy can enhance the discriminative ability of the network.

For the data required for training the RPN, we first generate 5 initial anchor boxes with a size of  $64 \times 64$  and an aspect ratio of 0.33, 0.5, 1, 2, and 3 respectively. And then slide on the feature maps to obtain  $25 \times 25 \times 5$  anchor boxes. For each anchor box, calculate the IoU score between it and the true value of the labeled target bounding box. And then mark the anchor box according to Equation (6):

$$Label(IoU) = \begin{cases} 0 & IoU < 0.3 \\ -1 & 0.3 \leq IoU \leq 0.6 \\ 1 & IoU > 0.6 \end{cases} \quad (6)$$

where 1 represents the foreground, 0 represents the background, and  $-1$  represents not caring about and not participating in the error calculation.

For the anchor box marked as the foreground, we use Equation (7) to calculate the true value of the bounding box regression coefficient  $T^{gt} = (T_x^{gt}, T_y^{gt}, T_w^{gt}, T_h^{gt})$ :

$$\begin{cases} T_x^{gt} = (G_x - P_x) / P_w \\ T_y^{gt} = (G_y - P_y) / P_h \\ T_w^{gt} = \ln(G_w / P_w) \\ T_h^{gt} = \ln(G_h / P_h) \end{cases} \quad (7)$$

where  $P = (P_x, P_y, P_w, P_h)$  represents the region proposal and  $G = (G_x, G_y, G_w, G_h)$  represents the true value of the corresponding target bounding box, respectively.

The loss function of RPN includes regression loss and classification loss, which can be formulated as:

$$L_{RPN} = L_{Cls} + \lambda L_{Reg} \quad (8)$$

where  $\lambda = 1.2$  is the weight value.  $L_{Cls}$  and  $L_{Reg}$  are formalized by equations (9) and (10) respectively:

$$L_{Cls} = -\frac{1}{N_{pos}} \sum_j (Cls^{gt} \cdot \log(Cls^p) + (1 - Cls^{gt}) \cdot \log(1 - Cls^p)) \quad (9)$$

$$L_{Reg} = \sum_{i \in x, y, w, h} Smooth_{L_1}(T_i^{gt} - T_i^p) \quad (10)$$

where  $L_{Cls}$  is the binary cross entropy loss,  $N_{pos}$  represents the number of positive samples,  $j$  represents the subscript of the training sample,  $Cls^{gt}$  and  $Cls^p$  represent the true value and predicted value of the bounding box classification respectively,  $T^{gt}$  and  $T^p$  represents the true value and predicted value of the bounding box regression coefficient respectively, where  $Smooth_{L_1}(\cdot)$  represents the improved mean square error function, and its calculation equation is:

$$Smooth_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (11)$$

For the DIoU discriminative network, we do not use the region proposals generated by RPNs in the training process, but perturbed the input box by adding Gaussian noise with standard deviation  $\sigma = 0.1$  to the coordinates of the target bounding box, generating 16 region proposals. At the same time, the minimum IoU score between region proposals and the ground truth of the target bounding box is required to be 0.1. For each region proposal, calculate the true value of  $IoU_{gt}$  and the normalized center distance  $NCD_{gt}$ , and obtain  $DIoU_{gt}$ . The image pair, target bounding box, and a group of region proposals are sent to the DIoU discriminative network together. The predicted values  $IoU_p$  and  $NCD_p$  are then obtained. The loss function of the DIoU discriminative network is:

$$\begin{aligned} L_{DIoU} = & \lambda_1 Smooth_{L_1}(DIoU_{gt} - (IoU_p - NCD_p)) \\ & + \lambda_2 Smooth_{L_1}(IoU_{gt} - IoU_p) \\ & + \lambda_3 Smooth_{L_1}(NCD_{gt} - NCD_p) \end{aligned} \quad (12)$$

where  $\lambda_1$ ,  $\lambda_2$  and  $\lambda_3$  are the weight coefficients, and the empirical value 0.4, 0.3 and 0.3 are used in this paper. The parameters of the entire network are obtained by optimizing the loss function in Equation (13):

$$L = \beta L_{RPN} + \alpha L_{DIoU} \quad (13)$$

where  $\alpha = 1.1$  and  $\beta = 1.3$  are two learnable hyper parameters, which represent the loss weight of the DIoU discriminative network and the RPN, respectively.



In order to enable the network to adapt to the targets with varIoUs scales, we use the video target detection dataset VID [30], the image target detection dataset COCO [31], and the target tracking dataset LaSOT [32] to pre-train the network, and then use the UAV video dataset VisDrone2018 [33] for fine-tuning, obtaining the optimal network parameters. The target position in the reference frame is not always in the center, and the target motion degree is simulated by random disturbance.

For the setting of network parameters, since the size of the feature maps of each layer is the same, the parameter settings of all RPNs are basically the same. So we only provide the parameters of RPN1, as shown in Table 2. The parameters of DIoU discriminative network are shown in Table 3.

**Table 2.** Parameter settings of RPN1.

Layers	Kernel Size	Stride	Output Size
Conv-1_1r	$3 \times 3$	1	$32 \times 32 \times 256$
PrPool-1r	$7 \times 7$	-	$7 \times 7 \times 256$
MaxPool-1r	$7 \times 7$	1	$1 \times 1 \times 256$
AvgPool-1r	$7 \times 7$	1	$1 \times 1 \times 256$
FC-1_1r	$256 \times 16$	-	$1 \times 16$
FC-1_2r	$16 \times 256$	-	$1 \times 256$
Conv-1_1t	$3 \times 3$	1	$32 \times 32 \times 256$
Conv-Reg	$1 \times 1$	1	$25 \times 25 \times (4 \times 5)$
Conv-2_1r	$3 \times 3$	1	$32 \times 32 \times 256$
PrPool-2r	$7 \times 7$	-	$7 \times 7 \times 256$
MaxPool-2r	$7 \times 7$	1	$1 \times 1 \times 256$
AvgPool-2r	$7 \times 7$	1	$1 \times 1 \times 256$
FC-2_1r	$256 \times 16$	-	$1 \times 16$
FC-2_2r	$16 \times 256$	-	$1 \times 256$
Conv-1_1t	$3 \times 3$	1	$32 \times 32 \times 256$
Conv-Cls	$1 \times 1$	1	$25 \times 25 \times (2 \times 5)$

**Table 3.** Parameter settings of DIoU discriminative network.

Layers	Kernel Size	Stride	Output Size
Conv-1_1r	$3 \times 3$	1	$32 \times 32 \times 256$
Conv-1_2r	$3 \times 3$	1	$32 \times 32 \times 256$
Conv-1_3r	$3 \times 3$	1	$32 \times 32 \times 256$
Conv-1_1r	$3 \times 3$	1	$32 \times 32 \times 256$
Conv-1_2r	$3 \times 3$	1	$32 \times 32 \times 256$
Conv-1_3r	$3 \times 3$	1	$32 \times 32 \times 256$
PrPool	$5 \times 5$	-	$5 \times 5 \times 256$
MaxPool	$5 \times 5$	1	$1 \times 1 \times 256$
AvgPool	$5 \times 5$	1	$1 \times 1 \times 256$
FC-1_1	$256 \times 16$	-	$1 \times 16$
FC-1_2	$16 \times 256$	-	$1 \times 256$
FC-IoU	$6400 \times 1$	-	$1 \times 1$
FC-NCD	$6400 \times 1$	-	$1 \times 1$

## 4. Experimental Results and Analysis

### 4.1. Datasets and Evaluation

#### 4.1.1. Datasets

We use UAV20L [34] and DTB70 [35] datasets to conduct a comprehensive performance evaluation on the target tracking method proposed in this paper.

The UAV20L dataset is a single target long-term tracking dataset released by Mueller et al. in 2016. It contains 20 fully annotated and challenging sequences. The sequence length varies from 1717 to 5527 frames. These sequences are labeled with 12 different attributes, namely Scale Variation (SV), Aspect Ratio Change (ARC), Camera Motion (CM),

Full Occlusion (FOC), Illumination Variation (IV), Fast Motion (FM), Low Resolution (LR), Similar Object (SOB), Out-of-View (OV), Partial Occlusion (POC), Background Clutter (BC), and Viewpoint Change (VC).

DTB70 is also a single target tracking dataset. The dataset contains 70 video sequences with a total number of about 16,000 frames, which is suitable for short-term tracking. Each sequence has 11 video attribute annotations, including Similar Objects Around (SOA), Fast Camera Motion (FCM), Occlusion (OCC), Scale Variation (SV), Deformation (DEF), In-Plane Rotation (IPR), Out-of-Plane Rotation (OPR), Aspect Ratio Variation (ARV), Out-of-View (OV), Background Cluttered (BC), and Motion Blur (MB).

#### 4.1.2. Evaluation Metrics

In this paper, we use the metric proposed in OTB2013 [36] to evaluate the tracking performance of UAV videos. One-Pass Evaluation (OPE) metric is used for each sequence, that is, the tracking method is initialized from the starting frame. Until the last frame, if the target is lost in the middle, the tracking method will not be reinitialized. The evaluation metric of tracking results usually adopts the precision plot and the success rate plot.

##### Precision Plot

First, calculate the Euclidean distance  $d^t$  between the central point coordinates  $(x_p^t, y_p^t)$  of the tracking result of the  $t$ -th frame and the true central point coordinates  $(x_{gt}^t, y_{gt}^t)$ , namely location error. And then, given an arbitrary positioning error threshold  $T_{location\_error}$ , the precision is defined as the ratio of all frames with  $d^t > T_{location\_error}$  to the total number of frames in the video sequence. In general, the precision is used to sort the trackers, when  $T_{location\_error}$  is 20 pixels.

##### Success Rate Plot

First, calculate the overlap score  $S^t$  of the tracking result bounding box  $B_p^t$  and the ground truth bounding box  $B_{gt}^t$  of the  $t$ -th frame. The definition of the overlap score is as follows:

$$S^t = \frac{|B_p^t \cap B_{gt}^t|}{|B_p^t \cup B_{gt}^t|} \quad (14)$$

where  $\cap$  and  $\cup$  represent the intersection and union of two bounding boxes respectively, and  $|\cdot|$  represents the number of pixels in the area. Given the overlap score threshold  $T_{overlap\_score}$ ,  $0 \leq T_{overlap\_score} \leq 1$ , the success rate plot represents the percentage of the frames, which meet  $S^t > T_{overlap\_score}$ , to the total number of frames in the video sequence. In general, the Area Under Curve (AUC) of the success rate plot is used to sort the trackers.

#### 4.2. Parameter Settings

We first train the channel attention based RPN and the DIoU discriminative network. After 9 epochs, the convolutional features of ResNet-50 [19] are added, and then 16 epochs are jointly trained. The ResNet-50 is pre-trained on ImageNet [37], which has proven to be a very good initialization for the target tracking tasks. Throughout the training process, we use the Stochastic Gradient Descent (SGD) optimizer for iterative optimization. The learning rate is 0.005, the momentum is 0.9, the weight decay is 0.0001, and the mini-batch size is 20. We also use image inversion and color dithering for data enhancement to increase the scale of training data. The experimental platform uses Ubuntu 16.04 with an Intel Xeon(R) E5-2602 v4 CPU, 16 G memory, and an Nvidia RTX 2080Ti GPU.

#### 4.3. Evaluation Results

##### 4.3.1. Evaluation on UAV20L Benchmark

##### Overall Evaluation

We compared the proposed tracking method with 13 state-of-the-art trackers on the UAV20L dataset. The 13 trackers are MS-Faster R-CNN [24], KCC [38], SRDCF [4], Auto-

Track [26], CSRDCF [39], ECO\_HC [40], ARCF [25], ARCF\_H [25], STRCF [41], BACF [42], MDNet [21], SiamFCpp\_googlenet [43] and SiamRPN++ [12]. The comparison results by using different trackers are shown in Table 4. The center location error threshold is set as 20 pixels.

**Table 4.** Comparison results of the AUC score and DP score by using the proposed method and eleven state-of-the-art trackers on the UAV20L dataset.

Tracker	Venue	AUC	DP
Ours	-	<b>54.1</b>	<b>72.8</b>
MS-Faster R-CNN [24]	Remote sens.2021	40.3	58.7
SiamFCpp_googlenet [43]	AAAI2020	53.5	71.5
AutoTrack [26]	CVPR2020	34.9	51.2
SiamRPN++ [12]	CVPR2019	53.0	69.5
ARCF [25]	ICCV2019	38.1	54.4
ARCF_H [25]	ICCV2019	38.6	55.7
STRCF [41]	CVPR2018	41.0	57.5
KCC [38]	AAAI2018	32.4	48.3
ECO_HC [40]	CVPR2017	37.7	49.8
CSRDCF [39]	CVPR2017	35.0	50.1
BACF [42]	ICCV2017	41.5	58.4
MDNet [21]	CVPR2016	45.2	60.1
SRDCF [4]	ICCV2015	34.3	50.7

It can be seen from Table 4 that the Area Under Curve (AUC) score and Distance Precision (DP) score obtained by the proposed method are 54.1% and 72.8%, respectively, which exceeds the second SiamFCpp\_googlenet tracker by 0.6% and 1.3%. In addition, the AUC score and DP score of the proposed method are 1.1% and 3.3% higher than the SiamRPN++ tracker that also incorporates multiple RPNs. This is because the proposed method uses multiple channel attention based RPNs to generate higher quality region proposals, and the DIOU discriminative network with multi-layer feature fusion to further optimize the region proposals.

#### Attribute-Based Evaluation

According to different video attributes on the UAV20L dataset, we further evaluate and analyze the proposed method and the other trackers, as shown in Tables 5 and 6.

**Table 5.** AUC comparison results of the state-of-the-art trackers on the UAV20L dataset.

Tracker	SV	ARC	CM	FOC	IV	FM	LR	SOB	OV	POC	BC	VC
ours	<b>53.6</b>	<b>49.0</b>	<b>52.7</b>	<b>29.8</b>	<b>52.2</b>	<b>50.2</b>	<b>36.5</b>	<b>57.1</b>	<b>53.1</b>	<b>51.4</b>	<b>28.9</b>	<b>51.0</b>
SiamFCpp_googlenet [43]	52.8	48.1	52.0	<b>33.7</b>	49.5	46.9	34.9	54.3	<b>57.9</b>	50.7	26.0	<b>55.8</b>
AutoTrack [26]	33.0	27.7	32.9	19.8	32.1	23.4	23.8	35.3	32.5	31.9	21.9	30.3
ARCF [25]	36.6	32.0	37.2	20.5	38.0	24.3	27.3	41.5	36.2	36.5	22.9	33.9
ARCF_H [25]	36.8	31.9	37.8	19.9	38.5	20.1	24.0	44.1	37.7	37.3	21.0	33.4
STRCF [41]	39.3	33.1	39.2	21.7	34.4	24.3	29.3	43.9	37.5	40.1	22.7	33.2
KCC [38]	30.8	24.9	30.8	15.9	29.4	19.5	21.9	35.1	30.6	29.8	16.0	27.9
ECO_HC [40]	36.0	28.9	35.7	16.3	33.3	21.9	26.0	44.5	37.3	34.9	13.3	35.3
CSRDCF [39]	33.4	29.4	33.6	21.1	35.8	19.3	23.4	38.4	31.0	32.8	22.7	30.9
BACF [42]	39.9	34.5	40.4	20.0	41.0	23.4	27.0	46.6	40.2	39.9	20.9	37.3
MDNet [21]	43.8	37.6	43.5	24.1	43.9	28.2	31.5	47.5	44.9	43.6	26.3	41.9
SRDCF [4]	33.2	27.0	32.7	17.0	29.5	19.7	22.8	39.7	32.9	32.0	15.6	30.3

**Table 6.** Precision comparison results of the state-of-the-art trackers on the UAV20L dataset.

Tracker	SV	ARC	CM	FOC	IV	FM	LR	SOB	OV	POC	BC	VC
ours	<b>71.4</b>	<b>66.1</b>	<b>71.4</b>	<b>49.4</b>	<b>70.3</b>	<b>71.4</b>	<b>54.6</b>	<b>73.4</b>	<b>71.1</b>	<b>70.0</b>	<b>45.9</b>	<b>64.4</b>
SiamFCpp_googleNet [43]	70.1	64.5	70.0	<b>52.1</b>	65.6	66.1	50.9	69.5	<b>77.4</b>	68.7	39.5	<b>69.9</b>
AutoTrack [26]	48.7	41.8	48.7	40.3	44.3	41.9	42.5	44.9	50.6	49.0	37.4	42.0
ARCF [25]	52.2	47.6	54.4	40.1	54.2	43.9	48.1	54.3	53.1	54.2	37.1	45.7
ARCF_H [25]	53.4	45.4	53.4	37.8	48.8	35.4	44.0	55.8	53.8	54.3	32.9	46.5
STRCF [41]	55.3	47.2	55.3	40.6	42.9	50.6	51.3	54.7	52.5	56.4	33.0	44.1
KCC [38]	45.6	35.6	45.6	32.6	33.8	35.3	42.4	46.6	45.9	45.8	24.7	37.5
ECO_HC [40]	47.1	37.5	47.1	33.9	39.1	35.9	43.6	51.8	49.5	47.6	23.5	42.3
CSRDCF [39]	48.6	43.0	47.5	40.3	51.4	37.7	41.5	49.5	43.5	46.3	37.7	43.5
BACF [42]	56.2	48.2	56.2	37.8	52.4	40.8	46.4	58.1	56.8	56.6	32.9	50.0
MDNet [21]	58.0	50.8	58.0	43.0	54.8	48.1	52.5	56.5	58.8	58.7	41.1	52.7
SRDCF [4]	48.1	38.9	48.2	33.1	41.1	32.7	42.9	52.2	52.5	49.1	25.2	41.4

It can be seen that the AUC scores of the proposed method rank first, including SV (53.6%), SOB (57.1%), POC (51.4%), LR (36.5%), IV (52.2%), FM (50.2%), CM (52.7%), BC (28.9%), ARC (49.0%). Only OV (53.1%) and VC (55.8%) rank second. Specifically, when there is a rapid target movement in the UAV videos, the AUC score of SiamFCpp\_googleNet is only 46.9%, while the proposed method can reach 50.2%, which is 3.3% higher than that of SiamFCpp\_googleNet. Regarding the tracking precision, our method can obtain higher Precision than SiamFCpp\_googleNet in the case of target rotation, illumination changes, partial occlusion, and small targets. For each attribute, compared with the trackers ranked second, our method has a significant improvement in three attributes of IV, FM, and BC, increasing by 4.7%, 5.3%, and 4.7%, respectively. It can be seen that the quality of the region proposal has an important impact on the tracking precision.

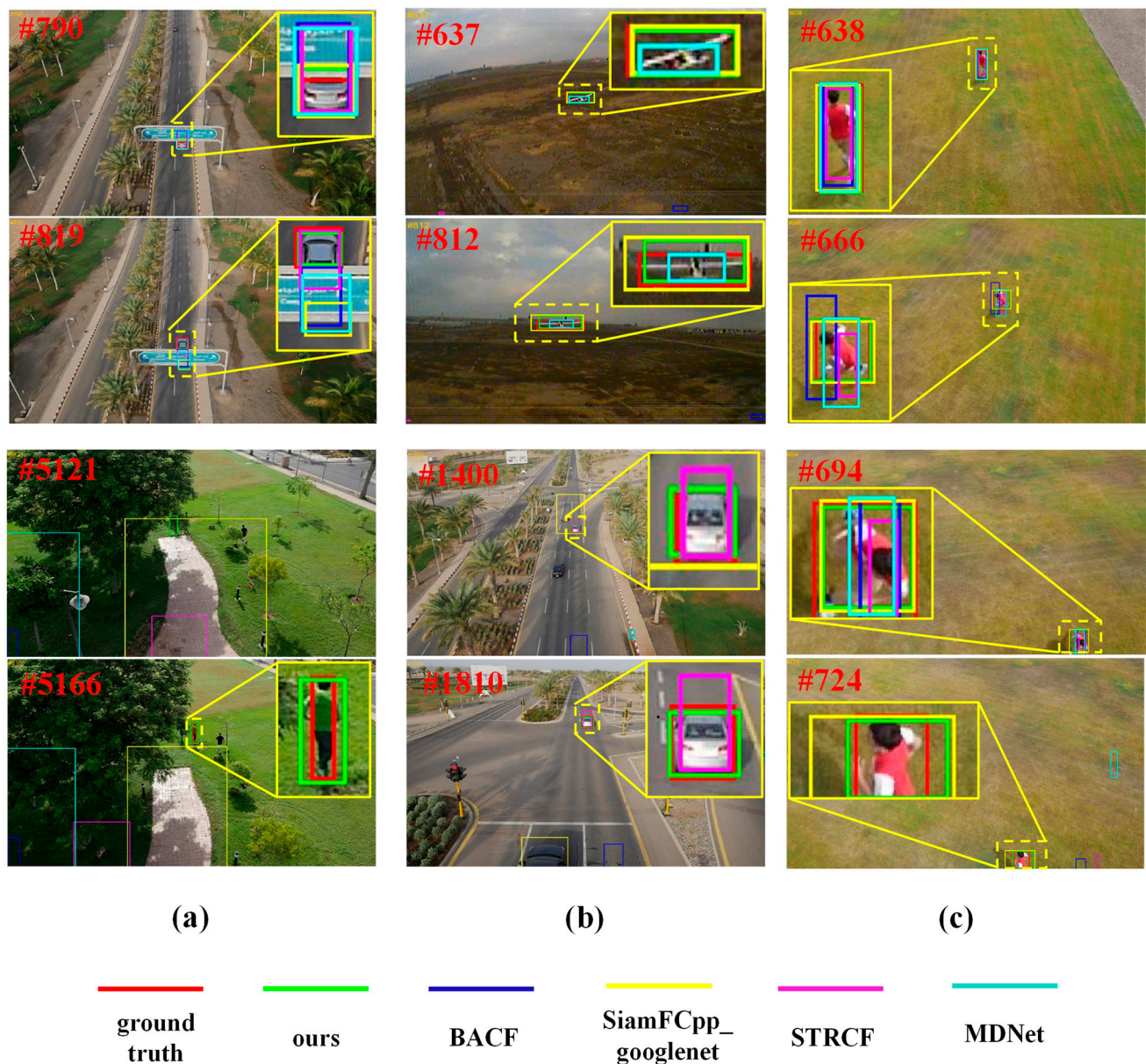
#### Tracking Results Analysis

In order to verify the performance of the proposed method more comprehensively, we display the tracking results obtained by the top five trackers with reference to the AUC scores, as shown in Figure 4.

Figure 4a shows the tracking results on *car9* and *group3* sequences, which involve full occlusion (FOC) and partial occlusion (POC). The tracking results demonstrate that our method can handle occlusions in these sequences well. Although the target is temporarily lost due to the target being completely occluded during tracking, our method can quickly retrieve the target when the target reappears. For example, in the sequence of *car9*, BACF [42], STRCF [41], and MDNet [21] are blocked by road signs at the 790th and 819th frames and cannot accurately determine the location of the target, while the proposed method produces better tracking results. For the sequence of *group3*, trees occluded due to the movement of the target and the camera at 5121th and 5166th frames. Other trackers cannot handle the occlusions well, resulting in drift. The proposed method can further optimize the region proposals generated by the channel attention based RPNs, yielding better tracking results.

Figure 4b shows the tracking results on the sequences of *uav1* and *car9*, which involve fast motion (FM) and scale variation (SV). We observe that in the sequence of *uav1*, SiamFCpp\_googleNet, MDNet and our method can locate the target very well, such as 637th and 812th frames. By comparison, our method can more accurately locate the position of the target. However, when the scale of the target in *car9* changes, such as the 1400th and 1810th frames, SiamFCpp\_googleNet and MDNet cannot adapt to the target size change well, but our method can adapt accordingly, showing its good tracking performance.





**Figure 4.** Tracking results of the proposed tracker and other state-of-art trackers on the UAV20L dataset. From top to down and left to right are the screenshots of the tracking results on the videos of *car9*, *group3*, *uav1*, *car9*, and *person7* respectively. The video sequences in (a), (b) and (c) mainly involve FOC and POC, FM and SV, FM and OV, respectively.

Figure 4c shows the tracking results on the *person7* sequence, which involves fast motion (FM), out-of-view (OV), and aspect ratio change (ARC). In the 638th frame, the aspect ratio changes due to the movement of the target. Although BACF, STRCF, and MDNet can roughly locate the target, they cannot adapt to the change of the target aspect ratio. Our method and SiamFCpp\_googlenet perform well. In the 694th and 724th frames, the target reappears. Our method can quickly retrieve it, but other trackers cannot handle it well. The target disappears and drifts to the background.

#### 4.3.2. Evaluation on DTB70 Benchmark

##### Overall Evaluation

We compared the performance of the proposed method with eight representative trackers on the DTB70 dataset, as shown in Table 7. The eight trackers are AutoTrack [26],

ARCF [25], ARCF\_H [25], STRCF [41], ECO\_HC [40], BACF [42], MDNet [21], and Staple\_CA [44].

**Table 7.** Comparison results of the AUC score and the DP score between the proposed method and eight state-of-the-art trackers on the DTB70 dataset.

Tracker	Venue	AUC	DP
ours	-	<b>59.5</b>	<b>78.2</b>
AutoTrack [26]	CVPR2020	47.8	71.6
ARCF [25]	ICCV2019	47.2	69.4
ARCF_H [25]	ICCV2019	41.6	60.7
STRCF [41]	CVPR2018	43.7	64.9
ECO_HC [40]	CVPR2017	44.8	63.5
Staple_CA [44]	CVPR2017	35.1	50.4
BACF [42]	ICCV2017	39.8	58.1
MDNet [21]	CVPR2016	45.6	69.0

As can be seen from Table 7, overall, our method yields significantly better performance than the other trackers. The AUC score (59.5%) and precision (78.2%) both achieve the highest precision. Our method is much better than the second-placed AutoTrack, and improves by 11.7% and 6.6% in the AUC score and DP score, respectively, and there is no obvious difference in tracking speed. In addition, compared to MDNet, our method improves the AUC score and DP score by 13.9% and 9.2%, respectively. This is because our method uses channel attention-based RPNs to generate higher quality region proposals, and DIoU discriminative network to further optimize the position of the region proposals.

#### Attribute-Based Evaluation

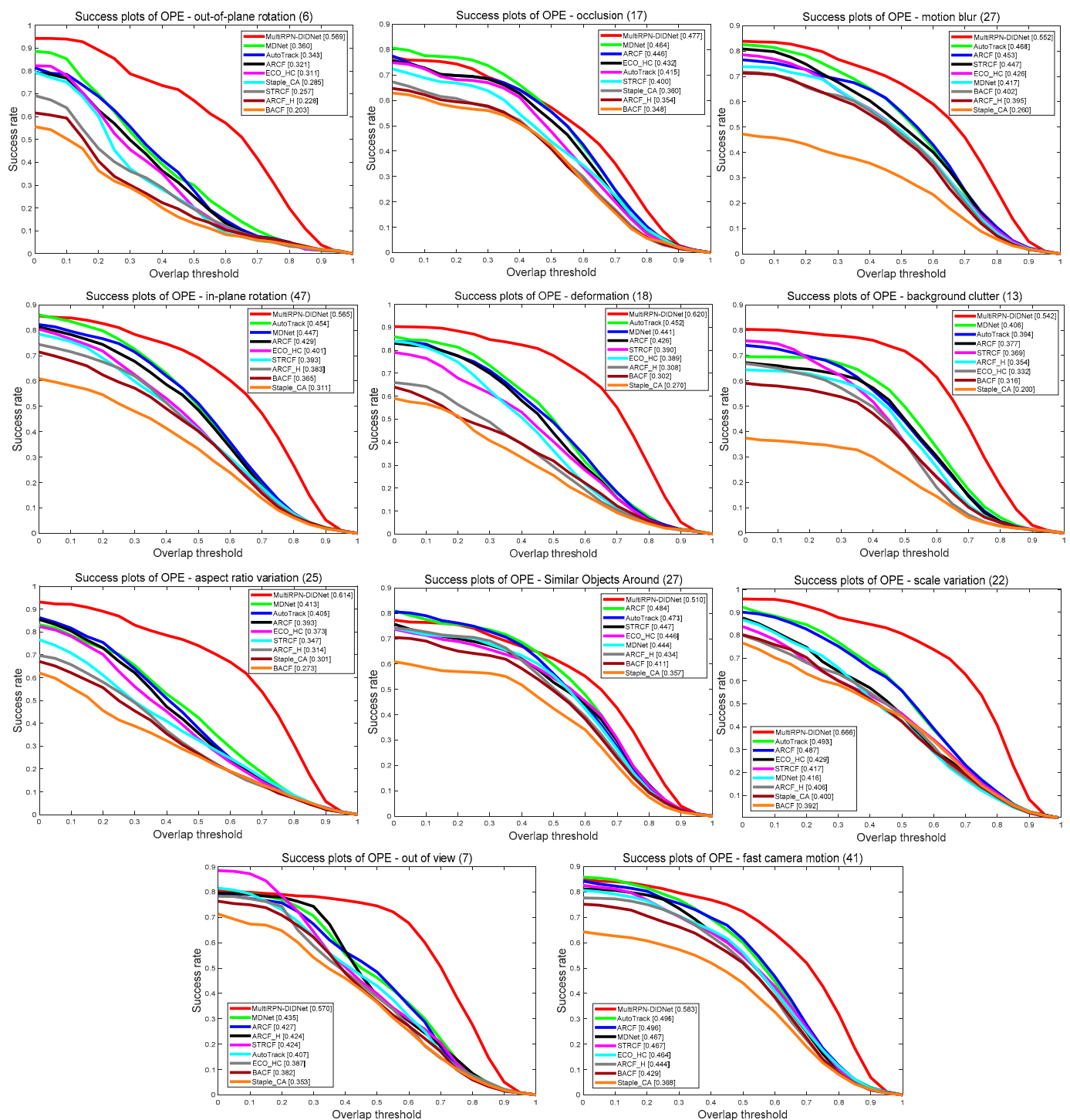
According to different video attributes in the DTB70, we further evaluated the performance of eleven attributes, as shown in Figures 5 and 6.

For the AUC score, our method yields the best performance, including ARV (61.4%), BC (54.2%), DEF (62.0%), FCM (58.3%), IPR (56.5%), MB (55.2%), OCC (47.7%), OV (57.0%), OPR (56.9%), SV (66.6%), and SOA (51.0%). Compared with the trackers ranking second, our method has a significant improvement in the four attributes of ARV, DEF, OPR, and SV, increasing by 20.1%, 16.8%, 20.9%, and 17.3% respectively. Compared with AutoTrack, our method improves the performance of OCC and OV attributes by 6.2% and 16.3%, respectively. For precision, our method also achieves the best performance, including ARV (74.7%), BC (76.4%), DEF (77.6%), FCM (78.9%), IPR (74.1%), MB (75.1%), OV (76.6%), OPR (61.0%), and SV (80.7%). Compared with the trackers ranking second, our method has a significant improvement in the five attributes of ARV, BC, DEF, OPR, and SV, increasing by 13.9%, 12.2%, 10.6%, 13.5%, and 10%, respectively. Besides, compared with AutoTrack, our method still performs well on its underperforming AVR, BC, OPR, and SV attributes, which are 14.2%, 12.9%, 17.1%, and 11.9%, respectively, showing better robustness.

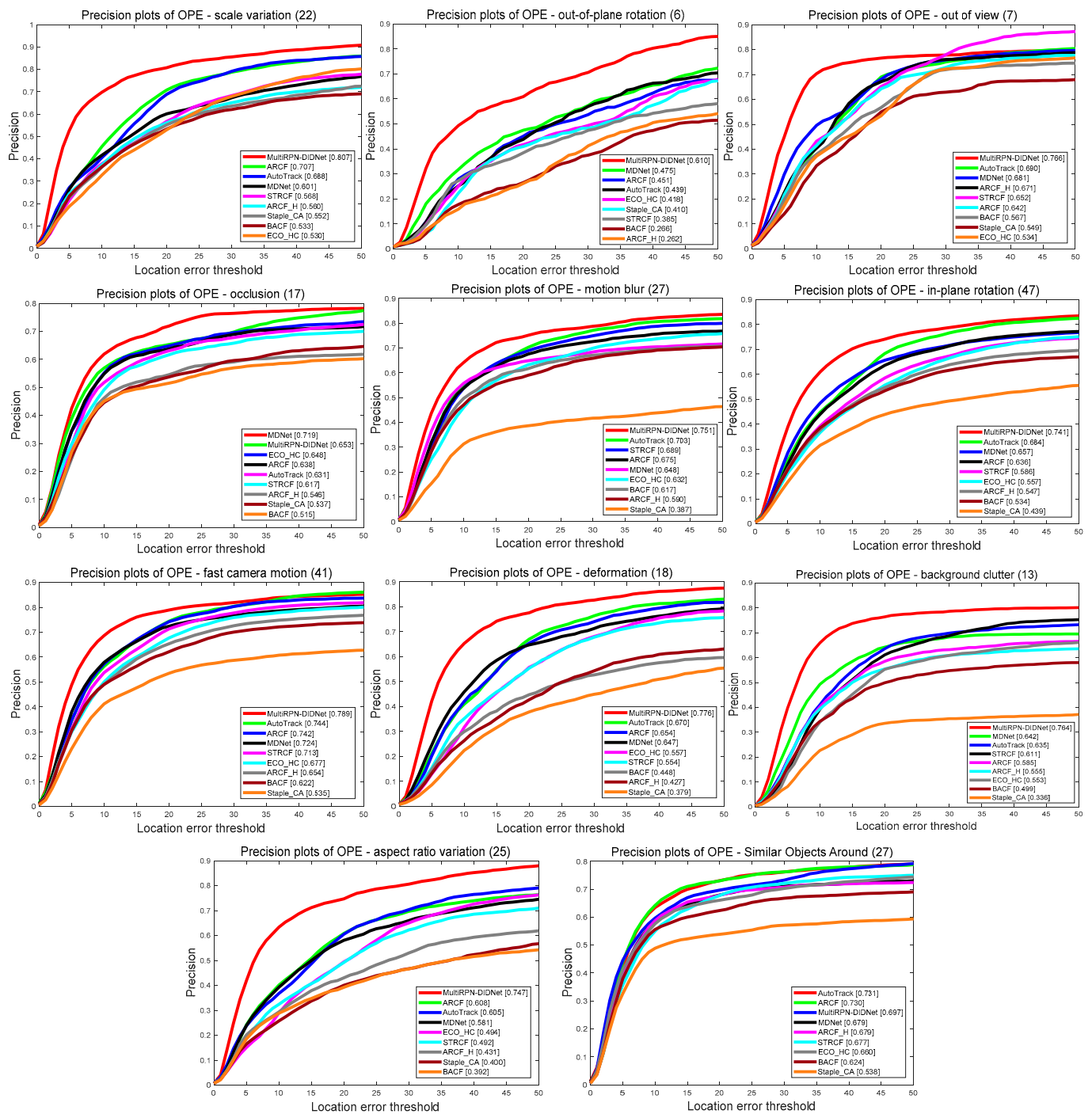
#### Tracking Results Analysis

In order to verify the performance of the proposed method more comprehensively, we also display the tracking results obtained by the top five trackers in terms of the AUC score, as shown in Figure 7.

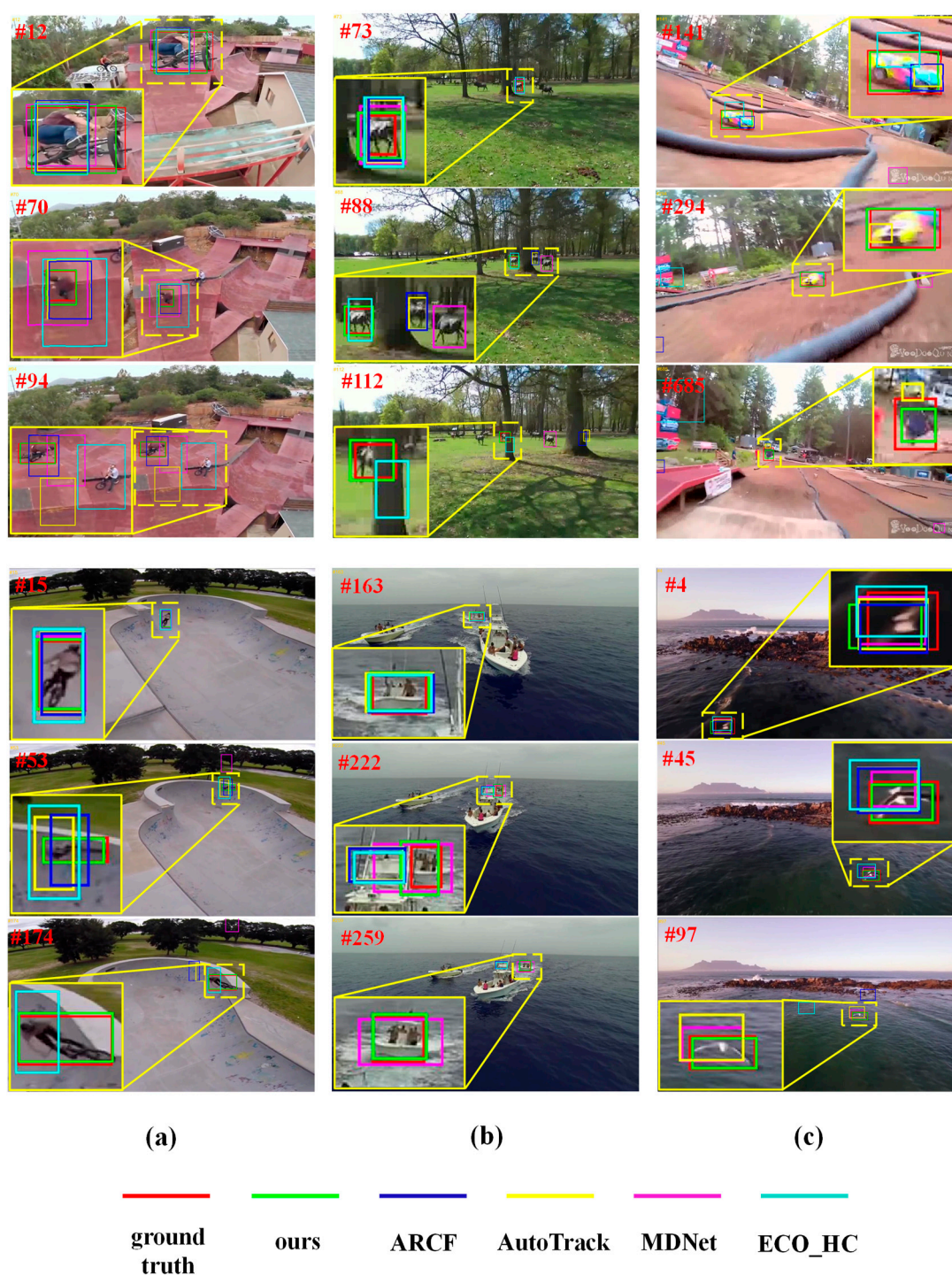




**Figure 5.** AUC comparison results of several state-of-the-art trackers on the DTB70 dataset. From left to right and top to down are the success rate plots under OPR, OCC, MB, IPR, DEF, BC, ARV, SOA, SV, OV and FCM video attributes.



**Figure 6.** Precision comparison results of several state-of-the-art trackers on the DTB70 dataset. From left to right and top to down are the precision plots of SV, OPR, OV, OCC, MB, IPR, FCM, DEF, BC, ARV, and SOA video attributes.



**Figure 7.** Tracking results of the proposed tracker and other state-of-art trackers on the DTB70 dataset. From top to down and left to right are the screenshots of the tracking results on the videos of *BMX3*, *BMX5*, *Hours1*, *Yatch4*, *RcCar4*, and *GULL1*. The video sequences in (a), (b) and (c) mainly involve SV, ARV and IPR, OCC and SOA, MB, BA and FCM, respectively.

Figure 7a shows the tracking results on the sequences of *BMX3* and *BMX5*. These sequences mainly involve the target's scale variation (SV), aspect ratio variation (ARV), and in-plane rotation (IPR). We have observed that for the *BMX3* sequence challenge, our method can locate the target well, while the other methods cannot adapt to the target scale change and aspect ratio change well, and even produce positioning drift, such as the 94th frame. For the *BMX5* sequence challenge, the target scale and aspect ratio change dramatically due to the rotation of the target in the plane. At the 174th frame, our method

can accurately locate the target. ECO\_HC can basically locate the target, but it can't adapt to the scale variation. Other trackers are unable to deal with the target scale changes well, or drifting in the background.

Figure 7b shows the tracking results on the sequences of *Hours1* and *Yatch4*. These sequences mainly involve occlusion (OCC) and similar objects around (SOA). For the sequence of *Hours1*, with two occlusions caused by trees, only our method can still accurately locate the position of the target at the 112th frame. For the sequence of *Yatch4*, MDNet and our method can locate the target well, such as the 259th frame. By comparison, because our method can further optimize the region proposals, it can locate relatively small targets more accurately.

Figure 7c shows the tracking results on the sequences of *RcCar4* and *GULL1*. These sequences mainly involve motion blur (MB), background cluttered (BC), and fast camera motion (FCM). For the sequence of *RcCar4*, due to the rapid movement of the camera, the motion blur and background disturbance occur, which are great challenges for object tracking. Although AutoTrack can basically find the target, it cannot adapt to the target's motion blur, so the target cannot be accurately selected. ARCF, ECO\_HC, and MDNet cannot handle out-of-view (OV), and thus, drifting in the background. Our method performs well, such as at the 294th and 685th frames. For the sequence *GULL1*, at the 97th frame, AutoTrack and our method can locate the target very well. But due to the design of DIOU discriminative network, our method can produce better tracking results.

#### 4.3.3. Tracking Speed Analysis

The UAV video trackers are often required to meet the strict requirements of real-time tracking in practical applications. Next, we will test the real-time performance of the trackers. Tables 8 and 9 show the comparison results of the proposed method on the UAV20L and DTB70 datasets in terms of Frames Per Second (FPS). It can be seen that the proposed method can reach 33.9 FPS and 33.0 FPS on the UAV20L and DTB70 respectively, which can meet the requirements of real-time tracking. This is because the method in this paper uses fewer region proposals and correction iterations.

**Table 8.** Comparison of processing speed between the proposed method and other state-of-art object tracking methods on the UAV20L dataset.

	Ours	KCC	SRDCF	AutoTrack	CSRDCF	ECO_HC	ARCF_H	STRCF	BACF
FPS	33.9	29.2	7.5	44.8	9.53	51.83	31.8	17.4	32.0

**Table 9.** Comparison of processing speed between the proposed method and other state-of-art object tracking methods on the DTB70 dataset.

	Ours	STRCF	AutoTrack	ECO_HC	ARCF_H	BACF	Staple_CA
FPS	33.0	21.9	48.6	51.9	37.1	37.7	50.66

#### 4.3.4. Ablation Studies

To verify the effectiveness of the method in this paper, we tested and compared the variant of the method in this paper on the UAVDT [45] dataset, as shown in Table 10. Single RPN structure indicates that only a single SNN-based RPN with channel attention mechanism is used to obtain the region proposal while the multiple RPNs structure denotes the method proposed in this paper.



**Table 10.** Inner module ablation study by comparing the proposed method and the variants of our method on the UAVDT dataset.

	AUC	DP	BC	CR	OR	SO	IV	OB	SV	LO
Single RPN	38.5	69.4	35.2	38.6	34.3	38.7	38.0	38.5	35.5	37.5
Multiple RPNs	60.4	81.4	53.0	55.5	58.6	55.7	62.6	62.7	61.6	52.2

As shown in Table 10, for the AUC scores, the multiple RPNs structure improves the performance by 21.9% compared to the single RPN structure. In addition, the multiple RPNs structure also improves the precision value by 12% compared to the method using only the single RPN structure at the threshold of location error is set as 20 pixels. For sequences with scale variation (SV) attribute, the proposed method improves the performance by 26.1% compared with the single RPN structure. This is due to the fact that the features acquired by the single RPN structure are not sufficient and are easily disturbed by the scale variation, leading to tracking drift. In contrast, multiple RPNs structure can handle the different scales of the target based on the fused features and perform the instance detection of local regions independently for each frame, so that better tracking precision can be obtained. For the sequences with object blur (OB) attribute, the performance of multiple RPNs structure is improved by 24.2% compared with the single RPN structure. This proves that the multiple RPNs structure is quite effective in dealing with object blur in UAV scenarios. For the sequences with other attributes, the AUC scores of the method in this paper are all higher than those of the single RPN structure, which proves that the multiple RPN object tracking framework proposed in this paper is effective, and it can guide the DIOU discriminative network to obtain more accurate tracking results.

## 5. Conclusions

In this paper, we exploit a solution based on the combination of multiple channel attention-based RPNs and DIOU discriminative network for real-time target tracking in UAV videos. First of all, for the generation of target region proposals, we propose a channel attention-based multiple RPNs structure suitable for the target tracking task, which can generate high-quality region proposals. In addition, we propose a DIOU discriminative network with multi-layer feature fusion to further refine the region proposal, improving the tracking accuracy. The experimental results on the UAV20L and DTB70 datasets show that, compared with 14 state-of-the-art trackers, the proposed method can yield better tracking performance on both the long-term and short-term tracking tasks. Since a high-quality region proposal is obtained, it only needs three optimizations to get better tracking results. Because the gradient is only propagated back in the DIOU discriminative network, the proposed method can reach the tracking speed of 33.9 FPS, which can meet the requirements of real-time processing.

We will continue to study and improve the following aspects in the future: (I) Effectively mine and utilize the scene context information. The current work of this paper fails to fully consider the contextual information of the scene. Actually, various spatiotemporal contextual information of UAV videos should be fully excavated to improve the reliability of tracking. (II) Fuse multi-source and multi-modal information. The proposed method only uses the visual information of the target, but the UAV video collection is susceptible to the influence of complex weather factors, and it is necessary to use the multi-source and multi-modal information obtained by the multiple sensors on the UAV platform, such as GPS, altimeter, gyroscope, etc. By fusing this information, the tracking speed and accuracy can be improved. (III) In terms of experimental data, we use the traditional target tracking dataset and target detection dataset to train the network, and then use the UAV video dataset for fine-tuning. This is because the UAV video dataset usually only contains test set. Only VisDrone2018 and VisDrone2019 datasets contain training sets. Therefore, it is necessary to collect more UAV videos to train and test the network to further enhance the generalization and robustness of the deep model.

**Author Contributions:** Conceptualization, L.Z. and B.L.; methodology, L.Z., B.L., S.Z. and J.L.; validation, H.Z.; formal analysis, H.Z.; investigation, B.L. and S.Z.; resources, J.L.; data curation, L.Z.; writing—original draft preparation, B.L.; writing—review and editing, L.Z. and H.Z.; funding acquisition, H.Z. and J.L. 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 (No. 61602018) and the Science and Technology Development Program of Beijing Education Committee, China (No. KM 202110005027).

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

## References

1. Bolme, D.S.; Beveridge, J.R.; Draper, B.A.; Lui, Y.M. Visual object tracking using adaptive correlation filters. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; pp. 2544–2550.
2. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. Exploiting the Circulant Structure of Tracking-by-Detection with Kernels. In Proceedings of the European Conference on Computer Vision (ECCV), Florence, Italy, 7–13 October 2012; pp. 702–715.
3. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. High-Speed Tracking with Kernelized Correlation Filters. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 583–596. [[CrossRef](#)] [[PubMed](#)]
4. Danelljan, M.; Häger, G.; Khan, F.S.; Felsberg, M. Learning Spatially Regularized Correlation Filters for Visual Tracking. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 4310–4318.
5. Valmadre, J.; Bertinetto, L.; Henriques, J.; Vedaldi, A.; Torr, P.H.S. End-to-End Representation Learning for Correlation Filter Based Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5000–5008.
6. Wang, Q.; Gao, J.; Xing, J.; Zhang, M.; Hu, W. Dcfnet: Discriminant correlation filters network for visual tracking. *arXiv* **2017**, arXiv:1704.04057.
7. Held, D.; Thrun, S.; Savarese, S. Learning to Track at 100 FPS with Deep Regression Networks. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 October 2016; pp. 749–765.
8. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H.S. Fully-Convolutional Siamese Networks for Object Tracking. In Proceedings of the European Conference on Computer Vision Workshops (ECCVW), Amsterdam, The Netherlands, 8–16 October 2016; pp. 850–865.
9. Tao, R.; Gavves, E.; Smeulders, A.W.M. Siamese Instance Search for Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 1420–1429.
10. 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. 103–119.
11. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. ATOM: Accurate Tracking by Overlap Maximization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 4655–4664.
12. 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 Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 4277–4286.
13. Huang, L.; Zhao, X.; Huang, K. Bridging the gap between detection and tracking: A unified approach. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Korea, 29 October–1 November 2019; pp. 3999–4009.
14. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
15. Li, B.; Yan, J.; Wu, W.; Hu, X. High Performance Visual Tracking with Siamese Region Proposal Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 8971–8980.
16. Ren, J.M.; Gong, N.S.; Han, Z.Y. Improved Target Tracking Algorithm Based on Siamese Convolution Neural Network. *J. Chin. Comput. Syst.* **2019**, *40*, 2686–2690.
17. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
18. Zhang, Z.; Peng, H. Deeper and Wider Siamese Networks for Real-Time Visual Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 4586–4595.
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 (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
20. 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 (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4854–4863.
21. 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 (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4293–4302.



22. Nam, H.; Baek, M.; Han, B. Modeling and Propagating CNNs in a Tree Structure for Visual Tracking. *arXiv* **2016**, arXiv:1608.07242.
23. Zhang, Y.; Wang, D.; Wang, L.; Qi, J.; Lu, H. Learning regression and verification networks for long-term visual tracking. *arXiv* **2018**, arXiv:1809.04320.
24. Avola, D.; Cinque, L.; Diko, A.; Fagioli, A.; Foresti, G.L.; Mecca, A.; Pannone, D.; Piciarelli, C. MS-Faster R-CNN: Multi-Stream Backbone for Improved Faster R-CNN Object Detection and Aerial Tracking from UAV Images. *Remote Sens.* **2021**, *13*, 1670. [[CrossRef](#)]
25. Huang, Z.; Fu, C.; Li, Y.; Lin, F.; Lu, P. Learning aberrance repressed correlation filters for real-time UAV tracking. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Korea, 29 October–1 November 2019; pp. 2891–2900.
26. Li, Y.; Fu, C.; Ding, F.; Huang, Z.; Lu, G. AutoTrack: Towards High-Performance Visual Tracking for UAV with Automatic Spatio-Temporal Regularization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 11920–11929.
27. Ye, J.; Fu, C.; Lin, F.; Ding, F.; An, S.; Lu, G. Multi-Regularized Correlation Filter for UAV Tracking and Self-Localization. *IEEE Trans. Ind. Electron.* **2021**. [[CrossRef](#)]
28. Li, T.; Ding, F.; Yang, W. UAV object tracking by background cues and aberrances response suppression mechanism. *Neural Comput. Appl.* **2021**, *33*, 3347–3361. [[CrossRef](#)]
29. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. *arXiv* **2019**, arXiv:1911.08287. [[CrossRef](#)]
30. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [[CrossRef](#)]
31. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C. Microsoft COCO: Common Objects in Context. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
32. Fan, H.; Lin, L.; Yang, F.; Chu, P.; 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 Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 5369–5378.
33. Zhu, P.; Wen, L.; Bian, X.; Ling, H.; Hu, Q. Vision meets drones: A challenge. *arXiv* **2018**, arXiv:1804.07437.
34. Mueller, M.; Smith, N.; Ghanem, B. A Benchmark and Simulator for UAV Tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 October 2016; pp. 445–461.
35. Li, S.; Yeung, D.Y. Visual Object Tracking for Unmanned Aerial Vehicles: A Benchmark and New Motion Models. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 4140–4146.
36. Wu, Y.; Lim, J.; Yang, M. Online Object Tracking: A Benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 2411–2418.
37. Deng, J.; Dong, W.; Socher, R.; Li, L.; Kai, L.; Li, F.-F. ImageNet: A large-scale hierarchical image database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, 20–25 June 2009; pp. 248–255.
38. Wang, C.; Zhang, L.; Xie, L.; Yuan, J. Kernel Cross-Correlator. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
39. Lukezic, A.; Vojir, T.; Zajc, L.C.; Matas, J.; Kristan, M. Discriminative Correlation Filter with Channel and Spatial Reliability. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4847–4856.
40. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. ECO: Efficient Convolution Operators for Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6931–6939.
41. Li, F.; Tian, C.; Zuo, W.; Zhang, L.; Yang, M. Learning Spatial-Temporal Regularized Correlation Filters for Visual Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4904–4913.
42. Galoogahi, H.K.; Fagg, A.; Lucey, S. Learning Background-Aware Correlation Filters for Visual Tracking. In Proceedings of the International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 1144–1152.
43. 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.
44. Mueller, M.; Smith, N.; Ghanem, B. Context-Aware Correlation Filter Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1387–1395.
45. Du, D.; Qi, Y.; Yu, H.; Yang, Y.; Duan, K.; Li, G.; Zhang, W.; Huang, Q.; Tian, Q. The Unmanned Aerial Vehicle Benchmark: Object Detection and Tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 375–391.