

## Article

# Helmet Wearing Detection of Motorcycle Drivers Using Deep Learning Network with Residual Transformer-Spatial Attention

Shuai Chen <sup>1,2,†</sup>, Jinhui Lan <sup>1,2,†</sup>, Haoting Liu <sup>1,2,\*</sup>, Chengkai Chen <sup>1,2</sup> and Xiaohan Wang <sup>1</sup>

<sup>1</sup> Shunde Innovation School, University of Science and Technology Beijing, Foshan 528399, China

<sup>2</sup> Beijing Engineering Research Center of Industrial Spectrum Imaging, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China

\* Correspondence: liuhaoting@ustb.edu.cn

† These authors contributed equally to this work.

**Abstract:** Aiming at the existing problem of unmanned aerial vehicle (UAV) aerial photography for riders' helmet wearing detection, a novel aerial remote sensing detection paradigm is proposed by combining super-resolution reconstruction, residual transformer-spatial attention, and you only look once version 5 (YOLOv5) image classifier. Due to its small target size, significant size change, and strong motion blur in UAV aerial images, the helmet detection model for riders has weak generalization ability and low accuracy. First, a ladder-type multi-attention network (LMNet) for target detection is designed to conquer these difficulties. The LMNet enables information interaction and fusion at each stage, fully extracts image features, and minimizes information loss. Second, the Residual Transformer 3D-spatial Attention Module (RT3DsAM) is proposed in this work, which digests information from global data that is important for feature representation and final classification detection. It also builds self-attention and enhances correlation between information. Third, the rider images detected by LMNet are cropped out and reconstructed by the enhanced super-resolution generative adversarial networks (ESRGAN) to restore more realistic texture information and sharp edges. Finally, the reconstructed images of riders are classified by the YOLOv5 classifier. The results of the experiment show that, when compared with the existing methods, our method improves the detection accuracy of riders' helmets in aerial photography scenes, with the target detection mean average precision (mAP) evaluation indicator reaching 91.67%, and the image classification top1 accuracy (TOP1 ACC) gaining 94.23%.

**Keywords:** helmet wearing detection; LMNet; UAV; residual transformer-spatial attention; super-resolution reconstruction



**Citation:** Chen, S.; Lan, J.; Liu, H.; Chen, C.; Wang, X. Helmet Wearing Detection of Motorcycle Drivers Using Deep Learning Network with Residual Transformer-Spatial Attention. *Drones* **2022**, *6*, 415. <https://doi.org/10.3390/drones6120415>

Academic Editor: Diego González-Aguilera

Received: 7 November 2022

Accepted: 30 November 2022

Published: 15 December 2022

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



**Copyright:** © 2022 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

Motorcycles and electric vehicles are still the primary means of transportation in developing countries; however, in road traffic, they are “vulnerable road users” and are more likely to cause injuries or deaths in the event of an accident. According to the World Health Organization’s 2018 global road safety status report [1], riders, cyclists, and pedestrians accounted for half of all global road traffic deaths. China’s road traffic safety situation is also dismal. According to the China-2021 statistical yearbook [2], 75,758 traffic accidents involving motorcycles and electric vehicles occurred in China in 2020, resulting in 102,054 fatalities. In response to a large number of illegal acts committed without the use of safety helmets, China’s Ministry of Public Security launched the “One Helmet One Belt” security protection action nationwide in June 2020, investing a large number of traffic police to conduct artificial supervision. Artificial supervision, on the other hand, is typically time-consuming and labor-intensive, with limited coverage and difficulty achieving the desired effect. Furthermore, long-term work can easily exhaust a police officer, resulting in a relaxation of supervision. As a result, reducing artificial monitoring while ensuring that riders wear safety helmets consciously has become an urgent issue.

At the moment, computer vision algorithms are a relatively mature technology. Helmet detection in road scenes has the potential to reduce artificial monitoring and force riders to wear helmets consciously in order to protect their lives [3]. Unmanned aerial vehicle (UAV) aerial photography detection [4] is a typical platform monitoring use case that has become a frontier subject in various fields such as image processing, computer vision, pattern recognition, and automatic control. UAVs have advantages such as small size, low cost, ease of use, good mobility, strong environmental adaptability, and so on. It can perform some tasks more efficiently than humans and has a wide range of applications in civil fields such as disaster prevention and relief [5], animal and plant research, and environmental protection. As a consequence, using UAV to detect drivers' helmet wearing states in urban scenes is extremely practical.

Recently, scholars have proposed various neural networks to detect the use of security helmets. Chen et al. [6] proposed an improved faster regions with a convolutional neural network (Faster RCNN) [7] to check worker helmet wearing state, as well as the K-means++ algorithm to accommodate small-size helmets. Similarly, Li et al. [8] proposed an object detection framework that combined online hard example mining (OHEM) and multi-part combination and used an improved Faster RCNN [7] to detect workers' helmets and their parts. However, when the target size change was large, the detection effect of this method could not meet the requirements, and its detection speed was also slow. Li et al. [9] proposed a target detection algorithm based on single shot multibox detector (SSD)-MobileNet for the real-time detection of safety helmets in construction sites, but it had some disadvantages, including low accuracy and poor robustness when detecting helmets with relatively small pixels in the image. Han et al. [10] used a similar method and proposed an SSD-based object detection algorithm; their method introduced a feature pyramid, multi-scale sensing module, and attention mechanism to improve the robustness of detecting object scale changes. However, its multi-scene generalization ability was poor, as was its small object detection accuracy. Cheng et al. [11] constructed a shallow sandglass residual module based on deep separable convolution and channel attention mechanism and proposed a you only look once version 3 (YOLOv3)-tiny-based target detection algorithm for safety helmet recognition of roadside close shooting scenes. Zhou et al. [12] proposed a helmet detection algorithm based on the attention mechanism YOLO (AT-YOLO), which was used to identify workers' safety helmets at close range when shooting scenes. Chen et al. [13] proposed a modified YOLOv4 model for detecting helmet wear of site workers in aerial scenes. Jia et al. [14] proposed a modified YOLOv5 model for detecting the helmets of motorcycles and riders in a surveillance scenario. These methods above are all designed for specific scenarios; unfortunately, either their detectors are not highly accurate in detecting small targets, or they have poor scene generalization ability and low robustness can be observed when used across scenarios.

Since being proposed, the attention mechanism has been widely used in the fields of natural language processing and computer vision such as machine translation, object detection, and intrusion detection [15]. In terms of deep learning visual attention mechanism, it has also been extensively studied in recent years to improve neural network performance, and its significance has also been widely acknowledged in various models [16–20]. The attention mechanism can be thought of as a dynamic selection process of important information input to an image, which is realized by the adaptive weight of features. In the study of channel attention, Hu et al. [21] proposed a channel attention module, which corrected channel characteristics by modeling the relationship between channels, and improved the representation ability of neural networks. Woo et al. [22] proposed the convolutional block attention module (CBAM) by combining channel and space attention, which improved performance in a variety of computer vision tasks. In the study of location pixel attention, Wang et al. [23] proposed a non-local network, which effectively captured the dependencies between various elements in sequences and considered the global characteristics of the image; however, the module had a large number of parameters. To take the location information into account, Hou et al. [24] proposed a new attention mechanism for the

neural network, namely “coordinate attention”, by embedding the location information into the channel attention, which had a good performance in target detection and semantic segmentation. The mentioned methods above have some drawbacks. On one hand, the above methods consider the attention mechanism of channel or spatial information, which makes modeling remote dependency impossible and lacks global consideration of overall information. On the other hand, the attention mechanism based on location or neuron information ignores the modeling of correlation between position pixels, limiting algorithm performance improvement.

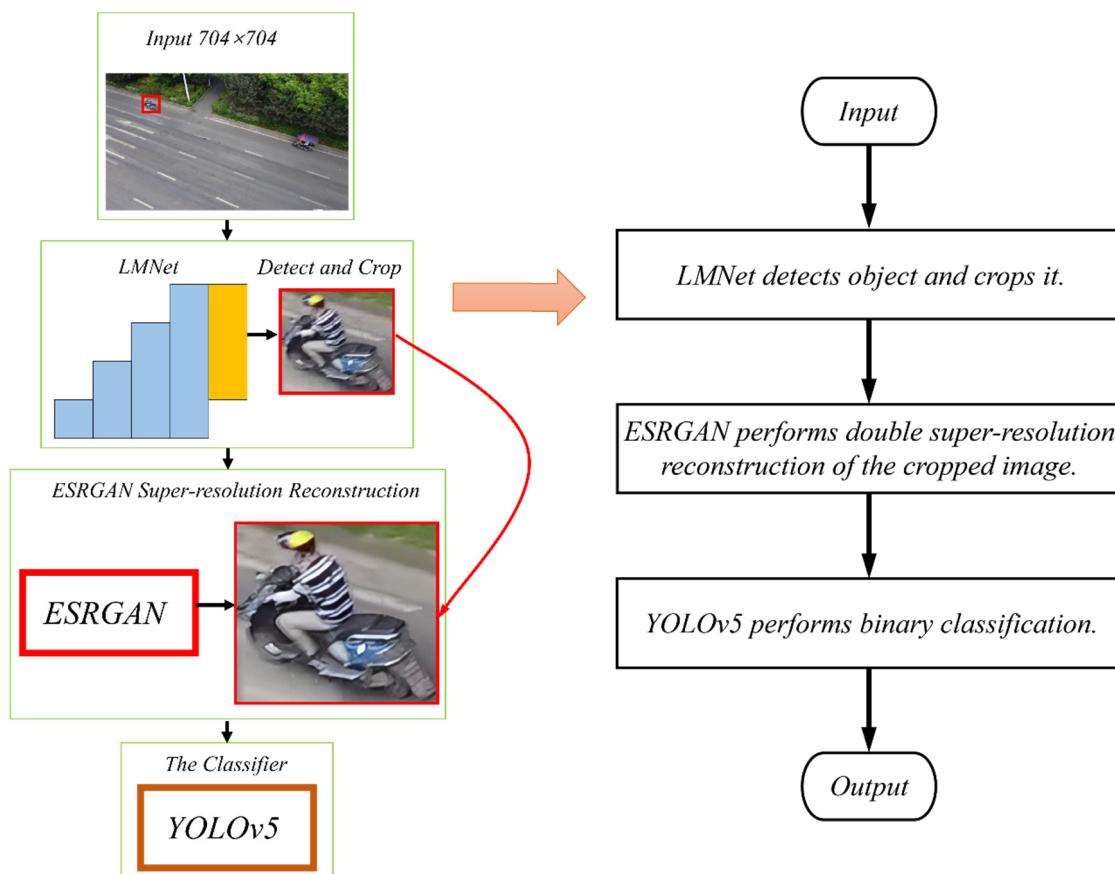
To solve the above problems, we propose an advanced method, which uses the YOLOv5 target detector, attention module, super-resolution reconstruction network, and classifier to detect the helmet wearing problem of riders. A ladder-type multi-attention network (LMNet) target detection algorithm is designed. First, a target detection algorithm based on LMNet, which uses high-resolution information for full interactive fusion to enhance feature extraction, is proposed. Second, a new attention module is developed, which makes full use of channel information, location information, and spatial information to enhance feature representation. Finally, a new paradigm of safety helmet wearing detection is also designed. The main contributions of this paper are as follows:

- We propose an LMNet target detection network and combine the method of super-resolution reconstruction to improve the network classification accuracy, bringing in a novel solution for small target recognition in aerial situations.
- We develop a unique plug-and-play residual transformer 3D-spatial attention module (RT3DsAM) that can significantly increase the detection accuracy of small targets in aerial photographic settings. Furthermore, when numerous modules are employed, the detection accuracy can be raised incrementally without the addition of too many parameters.

The following is how the article is structured: The second section discusses the proposed computational flow chart; the third section illustrates the main computational model details; the fourth section provides experimental analysis and comprehensive discussion to validate the superiority of our method; and the last section summarizes the study.

## 2. Proposed Computational Flow Chart

The traffic circumstances on an urban traffic road are complex, and vehicles such as electric cars, bicycles, motorcycles, and tricycles are frequently mixed together, challenging helmet recognition. It is difficult for UAV to reliably determine whether motorcycle and electric vehicle drivers wear helmets in such complex traffic conditions. In this section, we propose a target detection network framework with combined super-resolution reconstruction, YOLOv5 classifier, and LMNet based on the proposed joint detection framework (shown in Figure 1). The video stream from the UAV aerial photography is first collected, and then a ladder-type target detection network based on a multi-attention mechanism is employed to recognize as many motorbikes or electric vehicles and their drivers as possible in the video. Second, we use the target box detected in the first phase as an input and utilize enhanced super-resolution generic adversary networks (ESRGAN) [25] to reconstruct it in super-resolution, supplementing the details lost owing to aerial photography issues, and improving the classification outcome. Finally, we construct an RT3DsAM to compensate for the shortcomings of the visual object detection network, eliminate false detection, and reduce missing detection.



**Figure 1.** Computational flow chart of the proposed method.

### 3. Key Computational Models

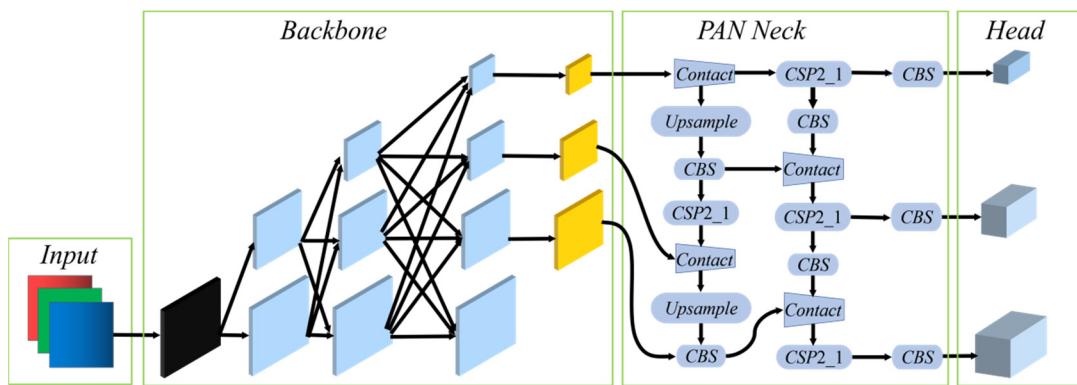
#### 3.1. LMNet

A hybrid model, LMNet, based on YOLOv5 and an RT3DsAM, is proposed in this study. It can extract features from small aerial targets using high-resolution representation. The residual transformer 3D-spatial attention can establish a global long-distance dependence and take the representation of context information into account to improve the accuracy of small target recognition. The overall structure of LMNet is made up of the ladder backbone network, the path aggregation network (PAN) [26] neck, and the detecting head. To obtain multi-scale and high-resolution fusion features, we first employ the ladder-type backbone network as the YOLOv5 backbone. Then, we design a ladder-type backbone network based on the high-resolution encoder of the high-resolution network (HRNet) [27], connect an RT3DsAM to the fourth stage's 1/8, 1/16, and 1/32 resolution feature map, and feed them into the PAN neck. Finally, the soft non-maximum suppression (Soft-NMS) [28] algorithm is utilized to process the detection results in order to lower the missing detection rate of numerous tiny targets overlapping samples. The LMNet structure is shown in Figure 2. Each black rectangle represents a bottleneck, each blue rectangle represents a basic block, and each yellow rectangle represents a residential transformer 3D-spatial attention module.

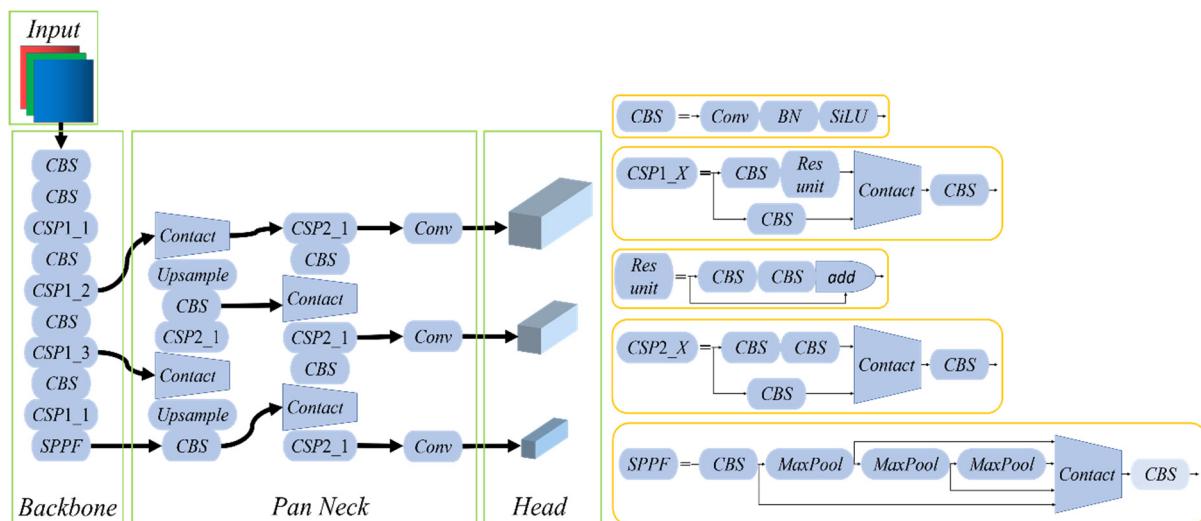
- YOLOv5 target detection network

YOLO [29–32] is a classical single-stage target detection algorithm. The YOLOv5 algorithm is developed based on YOLOv4 [32] and YOLOv3 [31] and turns the detection problem into a regression problem. Unlike the two-stage detection network, it does not extract the region of interest, but directly generates the bounding box coordinates and probability of each class through regression, which is faster than Faster RCNN. YOLOv5 has four versions: YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. YOLOv5s is the

lightest version, with the least amount of parameters and the fastest detection speed. The network structure is shown in Figure 3. The CBS block contains convolution layers, batch normalization, and SiLU functions. The CSP1\_x block contains the CBS block and x residual connection units. The CSP2\_x block contains x CBS blocks. And the SPPF mainly includes three MaxPool layers.



**Figure 2.** Structure of proposed LMNet.



**Figure 3.** Structure of YOLOv5. CBS is Conv BN SiLU; BN refers to batch normalization; SiLU denotes sigmoid-weighted linear units; CSP means cross-stage partial; SPPF is spatial pyramid pooling fast.

The YOLOv5 model is composed of four parts: input, backbone, neck, and head. First, YOLOv5's input adopts mosaic data enhancement, which enriches the data set via random scaling, random clipping, and random layout. Random scaling, in particular, adds many small targets, making the network more robust. Second, as can be observed in Figure 3, the backbone network has a relatively simple structure. In terms of feature information extraction and fusion interaction, it is not as well integrated as HRNet's backbone network and lacks the fusion between high and low-resolution features. Third, inspired by the path aggregation network (PANet) [26], the neck structure of the feature pyramid network (FPN) + PAN is designed. Finally, three prediction branches are designed. The prediction information includes the target coordinate, category, and confidence. The post-processing method of detecting the target object adopts weighted non-maximum suppression.

- Ladder-type Backbone Network

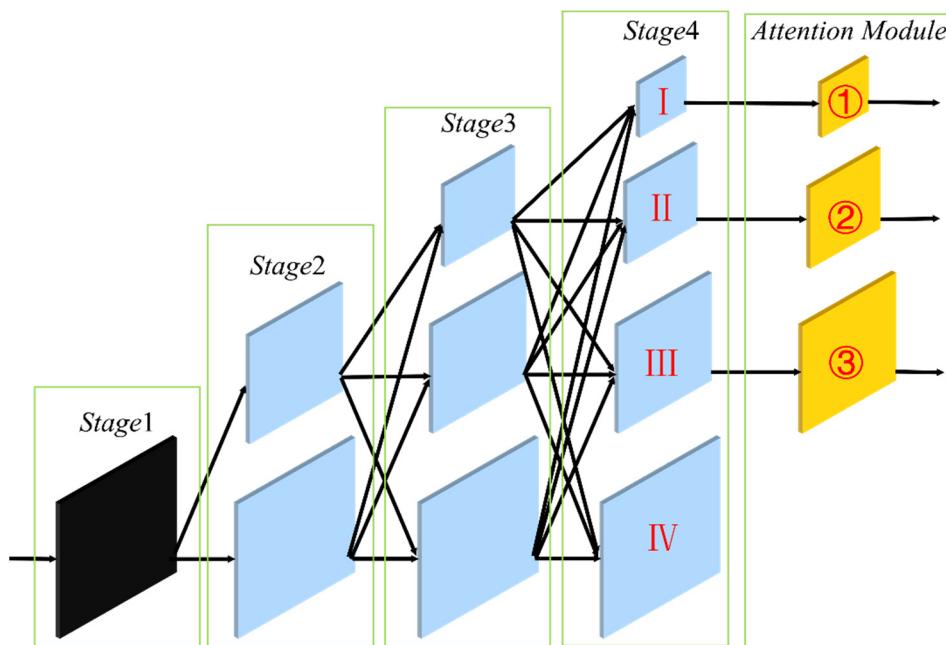
U-Net [33], SegNet [34], DeconvNet [35], Hourglass [36], and other mainstream backbone networks typically use the method of reducing resolution first and then increasing resolution. They typically encode the image as a low-resolution representation, convolu-

tionally connect the high and low-resolution representations, and then restore the high-resolution representation from the low-resolution representation. The HRNet, on the other hand, connects high and low-resolution subnetworks in parallel and performs iterative multi-scale fusion to obtain spatially accurate results. The proposed model in this paper makes use of the aforementioned high-resolution network. Some researchers in the field of small target detection have combined HRNet with small target detection and achieved good results. Wang et al. [37] presented a small object detection method for remote sensing images based on candidate region feature alignment. To some extent, the problem of small targets in UAV optical remote sensing images had been addressed. When detecting traffic signs in bad weather, Zhou et al. [38] proposed a parallel fusion attention network in conjunction with HRNet. More information could be obtained to improve accuracy through repeated multi-scale fusion of high and low-resolution representations.

In order to solve the problem of small-driver helmet detection in aerial photography, we need to extract more information from the limited resolution. As a necessary consequence, we design an LMNet: during the feature extraction stage, a ladder-type backbone network is designed, which can better retain high-resolution image information, make high and low resolution information fully interactive fusion, and is more sensitive to object location information. Figure 4 depicts a ladder-type backbone network. To reduce redundant parameters, we only use a bottleneck for the first stage of the network and a basic block for the remaining stages, which significantly reduces the number of parameters and speeds up model reasoning. In ladder-type backbone networks, multi-scale fusion is dependent on step size convolution, up sampling, down sampling, and summation operations. We use the resolution fusion in Stage3 to demonstrate the network's feature fusion. Since Stage2 outputs three different resolution representations of  $\{H_r^i, r = 1, 2, 3\}$ , while Stage3 outputs three corresponding resolution representations of  $\{H_r^o, r = 1, 2, 3\}$ ; the formula is as follows (1).

$$H_r^o = g_{1r}(H_1^i) + g_{2r}(H_2^i) + g_{3r}(H_3^i) \quad (1)$$

where  $r$  represents the indicator of resolution and  $g_{xr}(\bullet)$  represents transformation function.



**Figure 4.** Proposed ladder-type backbone network. I, II, III, and IV stand for Branch I, Branch II, Branch III and Branch IV. ①, ②, and ③ denote the attention modules behind Branch I, Branch II, and Branch III, respectively.

When the cross-phase fusion is performed, such as from Stage3 to Stage4, another calculated output is shown in the following Formula (2).

$$H_4^o = g_{14}(H_1^i) + g_{24}(H_2^i) + g_{34}(H_3^i) \quad (2)$$

Regarding Formulas (1) and (2),  $x$  represents the input resolution size in  $g_{xr}(\bullet)$ , and  $i$  is the output resolution size. If  $x = r$ ,  $g_{xr}(H) = H$ . If  $x > r$ ,  $g_{xr}(H)$  upsamples the input  $H$  and adjusts the number of channels by a convolution of  $1 \times 1$ . If  $x < r$ ,  $g_{xr}(H)$  executes  $(r-s)$  step convolution of input  $H$  to subsample it. By performing feature fusion between different resolution branches, we finally include the  $1/8$ ,  $1/16$ , and  $1/32$  resolution feature maps as the output of a ladder-type backbone network, and introduce the RT3DsAM in each branch, effectively improving the detection accuracy of small targets in aerial photography.

- RT3DsAM

In the motorcycle driver helmet detection stage, we use the motorcycle and driver as a whole target to overcome the problem of pedestrian misunderstanding. The motorcycle is classified as an electric motorcycle, a fuel motorcycle, and an electric bicycle; their appearances, sizes, and postures differ greatly between cars; additionally, during the cycling state, the appearances and postures of the tricycle, bicycle, and motorcycle are similar, resulting in a small difference between classes. As a result, large intra-class gaps and small inter-class gaps can lead to a large number of false positives and low detection accuracy. To address these sample flaws, we build an RT3DsAM that uses channel global self-attention to capture long-distance dependencies.

In general, the attention mechanism is divided into two types: responsive attention and soft attention. The hard attention mechanism's goal is to select the most useful part of the input features, whereas the soft attention mechanism learns a weighting vector to weight all of these features. Soft attention is commonly used in image classification and object detection. For example, Wang et al. [39] used scale attention to weight the output of convolutions with different filter sizes. A lightweight channel attention was proposed by squeezing and stimulating channel features. Furthermore, Woo et al. [22] created the CBAM, which could serially generate attention feature maps in two dimensions of channel and space, and then multiplied two feature maps to produce the final feature map, which improved object detection and image classification performance. In this paper, we design an RT3DsAM, taking into account not only the adaptive recalibration of the input feature maps, but also the missing correlations between the deep abstract positional pixel information, and focusing on the adaptive selection of high-level semantic information and the refinement of learned small target features.

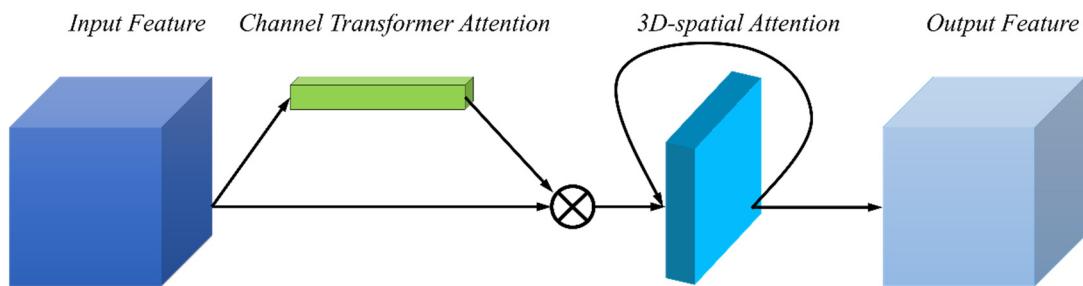
Given the input feature  $I \in R^{H \times W \times C}$ , we generate a one-dimensional channel residual transformer attention  $M_{ct} \in R^{1 \times 1 \times C}$  and a three-dimensional spatial attention  $M_{sa} \in R^{H \times W \times C}$ . As shown in Figure 5, where  $H$ ,  $W$ , and  $C$  indicate height, width, and channel, respectively. The above two attention modules are used for global long-distance self-attention modeling and self-selection of spatial information in each layer, respectively. The calculation process of overall attention can be summarized as Formulas (3)–(5).

$$I' = M_{ct}(I) \otimes I \quad (3)$$

$$I'' = M_{sa}(I') \otimes I_{sa} \quad (4)$$

$$I_{sa} = Conv_{1 \times 1}(Cat(Avgpool(I'), Maxpool(I'))) \quad (5)$$

where  $\otimes$  represents element multiplication,  $Conv(*)$  is the concatenate on the channel dimension, and  $Conv$  is a 2D convolution. Details of the attention calculation are given in the next two paragraphs.



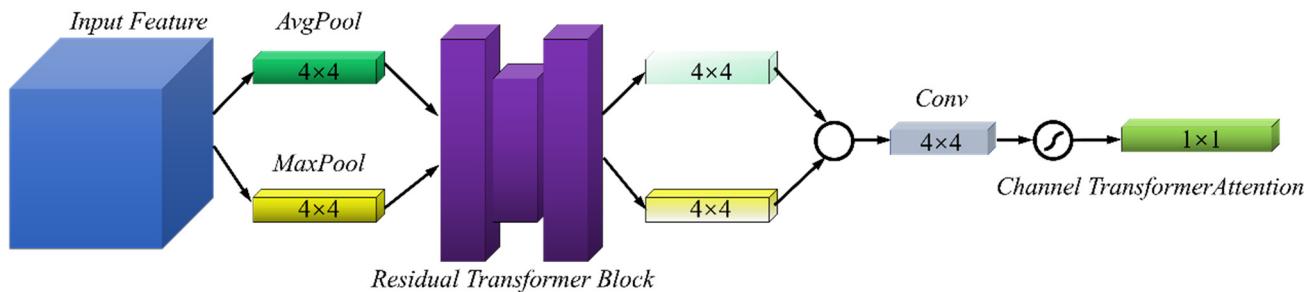
**Figure 5.** Residual transformer 3D-spatial attention module. Two attention modules are ordered sequentially. Given input or intermediate features, the channel global attention for each target class is calculated and used as input to subsequent spatial attention. Finally, the RT3DsAM provides output features.

- Residual Channel Transformer Attention Module (RCTAM)

The RCTAM aims to emphasize the significance of extracting information from the global image that is useful for feature representation and final classification detection, as well as establishing self-attention between them. To accomplish the aforementioned goal, we must create a channel global information driver function to map the input features to the target weight vector, which means that the target will consider not only the helmet but also the riders and the riding type of vehicle. This can reduce the intra-class gap while increasing the inter-class gap. The RCTAM is shown in Figure 6. To generate the summary statistics for the target-wide  $x^{avg} \in R^{4 \times 4 \times C}$ , the adaptive global average pooling operates on each feature of the spatial dimension  $H \times W$ . The feature matrix with the  $n$ th dimensional output height and width ( $H \times W$ ) of  $x^{avg}$  is calculated as shown in Formula (6).

$$x_{n,H \times W}^{avg} = AdaptiveAvgPool(p_n^{H \times W}) \quad (6)$$

where the  $p_n^{H \times W}$  is the input feature matrix of the  $n$ th channel and *AdaptiveAvgPool* denotes an adaptive global average pooling operation.



**Figure 6.** Residual channel transformer attention module. The RCTAM utilizes both max-pooling outputs and average-pooling outputs with a residual transformer network.

According to reference [22], which was studied in CBAM, global max pooling played an important role as supplementary global information for global average pooling. It is also used in this paper. The feature matrices of  $x^{max}$  with the  $n$ th dimension output height and width are calculated as shown in Formula (7).

$$x_{n,H \times W}^{max} = AdaptiveMaxPool(p_n^{H \times W}) \quad (7)$$

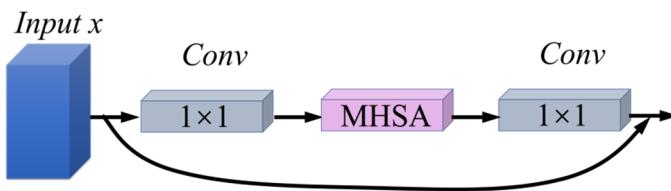
where the *AdaptiveMaxPool* denotes an adaptive global max pooling operation.

To fully capture the interaction between global high-dimensional information and establish a correlation between cross-channel position pixels and cross-object position

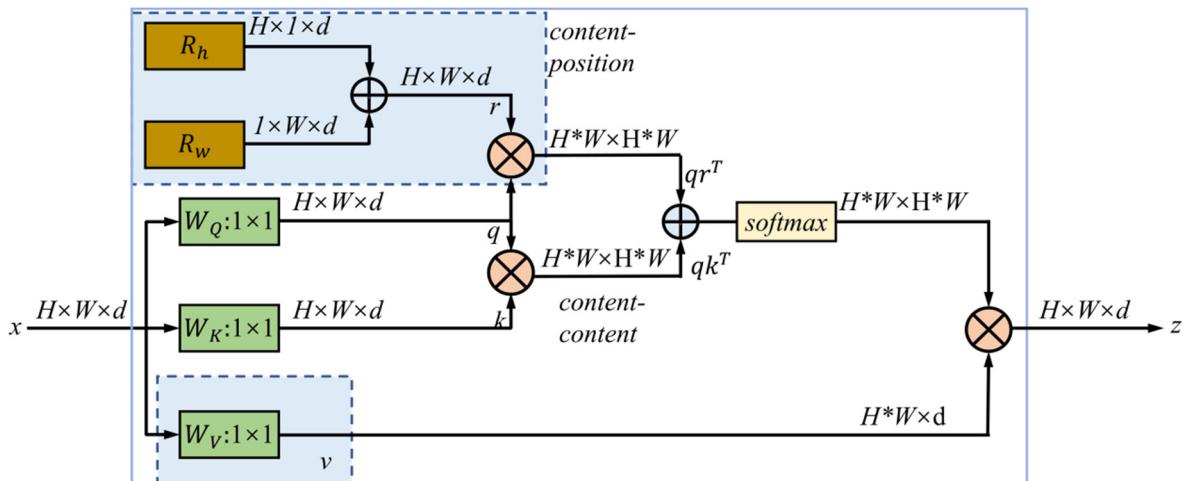
awareness, respectively, a residual transformer block (RTB) with two convolution layers around the nonlinearity and multi-head self-attention is operated on  $x$ . The RTB is shown in Figure 7 and the multi-head self-attention (MHSA) is shown in Figure 8. The first convolution layer is a dimensionality-reduction layer parameterized by  $W_1$  with a reduction ratio  $r$  and a rectified linear unit (ReLU). The second is a multi-head self-attention layer parameterized by  $\hat{T}$ . The third is a multi-head dimensionality-increasing layer parameterized by  $W_2$ .

$$O_{ct} = \delta(g(x, W) + x) = \delta(W_2(\hat{T}(\delta(W_1 x))) + x) \quad (8)$$

where  $\delta$  refers to the ReLU [40] function,  $W_1 \in R^{(C/r) \times C}$ ,  $\hat{T} \in R^{(C/r) \times (C/r)}$  and  $W_2 \in R^{C \times (C/r)}$ , reduction ratio  $r$  set to 1 in our experiment.



**Figure 7.** Residual Transformer Block.



**Figure 8.** Multi-head self-attention module.

To build the RTB-based channel global self-attention and reduce the loss of information, we share the parameters  $\{W_1, \hat{T}, W_2\}$  of RTB for the output of both global adaptive average pooling and global adaptive max pooling. Then, the outputs of RTB  $O_{ct}^{avg}$  and  $O_{ct}^{max}$  are concatenated together on the channel dimension calculated by a 2D convolution:

$$M_{ct} = \sigma(Conv_{4 \times 4}(O_{ct}^{avg} \oplus O_{ct}^{max})) \quad (9)$$

where  $\sigma$  refers to the sigmoid function and  $\oplus$  denotes a concatenation operator.  $Conv_{4 \times 4}$  is a manipulation with a convolution of step 1, padding 0, and kernel  $4 \times 4$ .

The final output of RCTAM is obtained by rescaling the input features with the output activation  $M_{ct}$ :

$$I'_c = f_{scale}(I_c, M_{ct}) = M_{ct} \cdot I_c \quad (10)$$

where  $I' = I'_1, I'_2, \dots, I'_c$  and  $f_{scale}(I_c, M_{ct})$  refer to pixel-wise multiplication between the feature map  $I_c \in R^{H \times W \times C}$  and the scalar  $M_{ct} \in R^{1 \times 1 \times C}$ .

In Figure 8,  $\oplus$  and  $\otimes$  represent the element-wise sum and matrix multiplication, respectively; while  $1 \times 1$  is a pointwise convolution. According to the reference [41], we use four heads in multi-head self-attention in this paper. The highlighted blue boxes represent position encodings and the value projection, in addition to the use of multiple

heads. Furthermore, the feature map after global adaptive pooling is sent to all2all attention for execution, and  $R_h$  and  $R_w$  are encoded for the height and width on the input feature map, respectively. Attention logits is the  $qk^T + qr^T$ , where  $q$ ,  $k$ , and  $r$  represent the query, key, value, and position encoding, respectively. Similarly, here we use the relative distance position for encoding [42–44]. According to the studies [43–45], relative-distance-aware position encodings are better suited for vision tasks. This is due to attention not only taking into account the content information but also relative distances between features at different locations, thereby being able to effectively associate information across objects with positional awareness [41].

- 3D-spatial Attention Module (3DsAM)

3DsAM aims to further mine the spatial correlation between max and average pooling information, enhance the spatial information of pixels with labels of the same category in the neighborhood, and suppress pixels with different classes of labels. Therefore, the ideal output of 3DsAM should be a feature matrix with the same height and width as the input feature through 3D spatial attention, and which carries the information of adaptive selection. It first obtains detailed spatial information about the intra and inter-class objects from two channels, then establishes a spatial attention map for each input channel, and finally forms a 3D spatial attention that adaptively adjusts the weights layer-by-layer. Figure 9 shows the 3DsAM. As in CBAM [22], we apply the global max pooling and global average pooling on the input across channels.

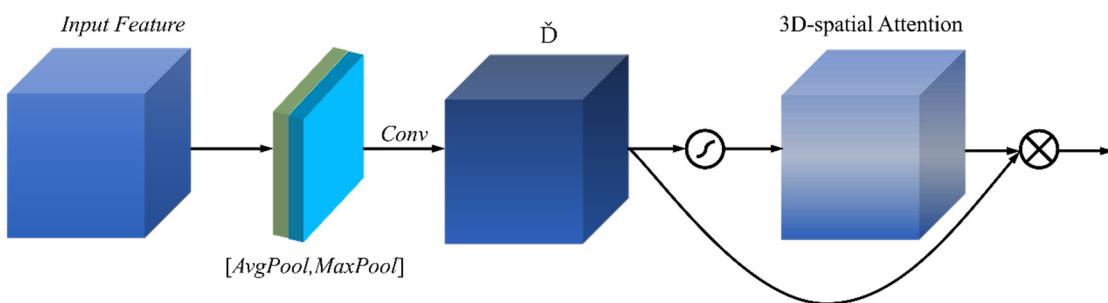
$$F_{i,j}^{avg} = \frac{1}{c} \sum_{C=1}^C I'_C(i,j) \quad (11)$$

$$F_{i,j}^{max} = \max(I'_C) \quad (12)$$

where  $I'_C(i,j)$  is the value at position  $(i,j)$  of the  $C$ th channel. Then, two outputs are concatenated horizontally as the input of a new convolutional layer followed by a sigmoid activation function:

$$M_{sa} = \sigma(Conv_{1 \times 1}(F^{avg} OF^{max})) \quad (13)$$

where  $Conv_{1 \times 1}$  is a convolution with step 1, padding 0, and stride  $1 \times 1$ ;  $M_{sa}$  is 3D attention map by activating features through  $Conv_{1 \times 1}$  operation with the sigmoid function.



**Figure 9.** 3D-spatial attention module. Symbol  $\check{D}$  denotes a 3D feature vector.

The final output of 3DsAM is obtained by rescaling the input features  $I_{sa}$  with the output activation  $M_{sa}$ :

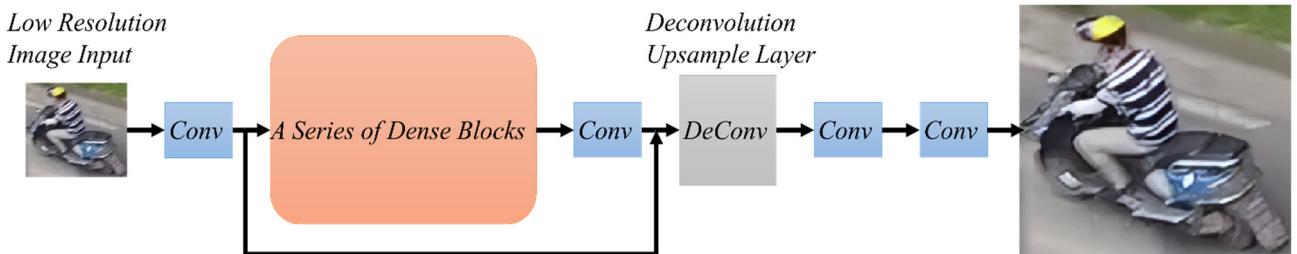
$$I'' = M_{sa} \circledast I_{sa} \quad (14)$$

where  $\circledast$  is the spatial-wise of each channel multiplication operation between the feature map  $I_{sa} \in R^{H \times W \times C}$  and 3D spatial attention map  $M_{sa} \in R^{H \times W \times C}$ .

### 3.2. ESRGAN

The ESRGAN [25] model is improved based on the image super-resolution generative adversarial network (SRGAN) [46]. Based on SRGAN, the generator neural network

of SRGAN, the discriminator identification object, and the loss function are adjusted and optimized, respectively. Thus, the SRGAN algorithm's performance is significantly improved. The generator neural network structure of ESRGAN is shown in Figure 10.



**Figure 10.** Generator neural network of ESRGAN.

Unlike SRGAN, ESRGAN uses dense connection blocks to replace residual modules. Each dense connection block is an improved residual module. It is distinguished by the use of a multi-layer residual (residual-in-residual) structure, which employs a deeper convolutional neural network to improve the depth learning algorithm's performance. To address the increased computational amount in this process, the ESRGAN algorithm employs a similar strategy to that of the face super-resolution generative adversarial network (FSRCNN) [47]. Simultaneously, ESRGAN points out that BN operation is easy to generate artifacts for deep layer GAN network training, so the backbone network abandons the use of BN operation.

ESRGAN also improves the discriminator's mode. Although the reconstructed image is judged to be false at times in the original SRGAN algorithm, this does not imply that the reconstructed image is not realistic enough, but the discriminator may not be able to correctly identify the image of the content. On the other hand, if the output value of a reconstructed image is high but the corresponding value of the original image is higher, it indicates that the reconstruction result needs to be improved. Thus, in the SRGAN algorithm, the method of passing the reconstructed image and original image through the discriminator and subtracting the output is used to replace the method of identifying and evaluating only the reconstructed image and original image. As a result, when the output value of an image in the generator is very low but the output value of its corresponding original image is lower, the resistance loss value is greatly reduced when compared to SRGAN, which has little effect on the generator. On the other hand, if the output value of the reconstructed image in the discriminator is high but the output value of the original image is higher, the loss will still increase so that the generator can generate a more realistic reconstructed image. These steps can be represented as follows:

$$F(M^{SR}, M^{HR}) = \sigma(F(M^{SR}) - F(M^{HR})) \quad (15)$$

$$F(M^{HR}, M^{SR}) = \sigma(F(M^{HR}) - F(M^{SR})) \quad (16)$$

$$L_g^{G-ESRGAN} = E[-\log(1 - F(M^{HR}, M^{SR}))] - E[\log(F(M^{SR}, M^{HR}))] \quad (17)$$

where the reconstructed image  $M^{SR}$  and the original image  $F(M^{HR})$  are input into the discriminator network in a random order,  $M^{SR}$  and  $F(M^{HR})$  are the probabilities (the number between 0 and 1 is output) for the discriminator to judge them as true. Function  $\sigma(\bullet)$  is a simple impulse function that maps the output to a region of 0 to 1, and  $E[\bullet]$  is the process of finding expectations. In this process, the discriminator does not know the input order of the image, and only calculates the probability that two images are true, and then subtracts them. Function  $F(M^{SR}, M^{HR})$  represents the pair of the input image, the reconstructed image is in the front, the real image is in the back, and the output of this value is close to 1 under the ideal state (the reconstructed image of generator makes the discriminator unable to distinguish between true and false images).  $F(M^{HR}, M^{SR})$  represents that the real image is in the front and the reconstructed image is in the back. By

subtracting two probabilities, in the ideal state, the value is close to 0 through a simple impulse function. The  $L_g^{G-ESRGAN}$  of Formula (18) is the mathematical expression of the ESRGAN generator's resistance loss function. The total loss of the ESRGAN discriminator can be expressed as:

$$L_d^{D-ESRGAN} = E[-\log(1 - F(M^{SR}, M^{HR}))] - E[\log(F(M^{HR}, M^{SR}))] \quad (18)$$

ESRGAN also includes the L-1 loss function to further improve the reconstruction accuracy. To reconstruct more accurate underlying details, the total loss function of the ESRGAN is defined as follows.

$$L_{GAN} = L_c^{G-ESRGAN} + \theta_1 L_g^{G-ESRGAN} + \theta_2 L_1 \quad (19)$$

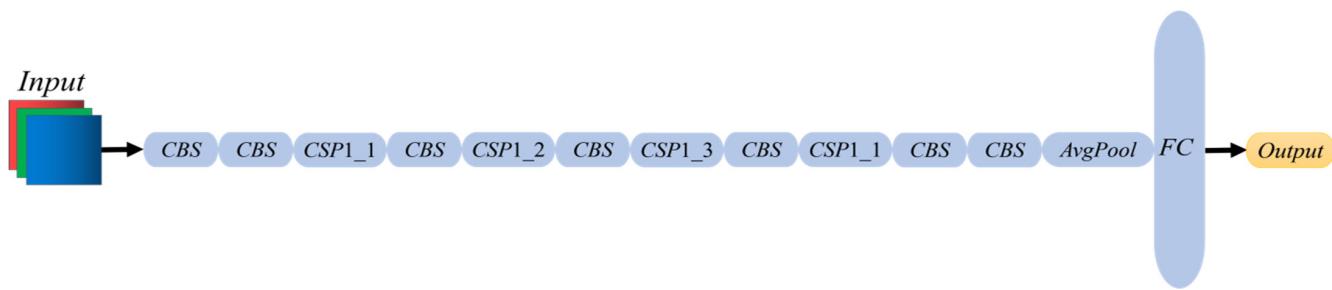
The  $L_c^{G-ESRGAN}$  is the structural loss of ESRGAN, which is roughly the same as that of the SRGAN. The difference is that the loss function in the visual geometry group network [48] feature extractor removes the output activation function. In this paper, we use the weights generated by various 2K and flick 2k (DF2K) and outdoorscene (OST) training sets to reconstruct the original image twice as well. Figure 11 shows that the rider and helmet have more realistic textures and sharp edges. It can add more features for future network learning.



**Figure 11.** The difference between with and without ESRGAN. (a) shows the original image as well as the enlarged helmet area. (b) is the reconstructed image with the helmet area enlarged.

### 3.3. YOLOv5 Classifier

In this paper, we crop out the detected riders and divide them into two types of data sets with helmets and without helmets, and then use ESRGAN to reconstruct them to get a clearer and more realistic target image. Finally, we adopt the YOLOv5 as the helmet classifier. When computing, first, YOLOv5 reads two types of data as input, then uses the backbone network to extract features, and finally, outputs the results by the softmax function. The backbone network of the YOLOv5 classifier is the same as that of the YOLOv5s detector, which has the least number of parameters, fast reasoning speed, and high precision classification performance. It is very suitable for practical engineering use. Figure 12 shows its network structure. In addition, we do not use the data enhancement method in YOLOv5, because random cropping will lead to the loss of some features of the target image that we have cropped, making the classification training effect worse.



**Figure 12.** The YOLOv5 classifier network structure.

#### 4. Experiments and Discussions

##### 4.1. Data Source and Evaluation Indicators

- Dataset

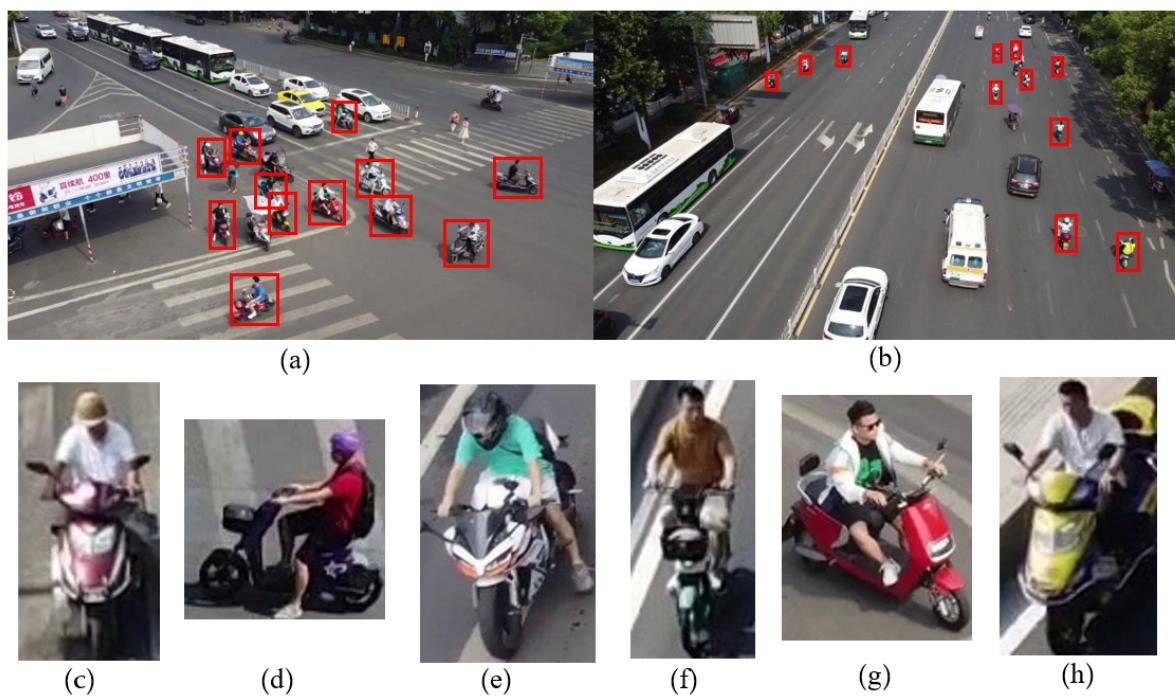
We capture a series of aerial photographic images. The dataset is obtained through aerial photography from DJI drones, which captures 3010 images with a resolution of  $1920 \times 1080$ , covering the case of small target, medium target motion blur and complex lighting. Electric bikes, electric motorcycles, gasoline-powered motorcycles, and their riders are among the ground targets. Cropping label targets based on detection data yields the image classification dataset. There are 4555 target images with helmets and 3164 target images without helmets. The data set is also randomly divided into training, validation, and test sets in the following proportions: 6:2:2.

Figure 13 depicts data samples from our dataset. There are two groups of drone aerial images: (a) represents the medium target case in the aerial image, and (b) represents the small target case. Because the aerial height is between 10 and 15 m, it is higher than the traffic camera monitoring support pole. Thus, there is no large target in the scene. Figure 13c–e show helmeted riders on electric motorcycles, electric bikes, and gas-powered motorcycles, respectively. (f)–(h) depict riders without helmets operating electric bicycles, electric motorcycles, and gasoline motorcycles, respectively. Small targets accounts for about 1.0~3.0% of the total pixel area, while medium targets account for 3.0%~6.0% of the total image. Furthermore, the aerial image quality is poor due to the small target in the aerial image, which is accompanied by various factors such as motion blur and complex light conditions (as shown in Figure 14). Missing and false detections will be severe if the target is directly detected from the aerial image.

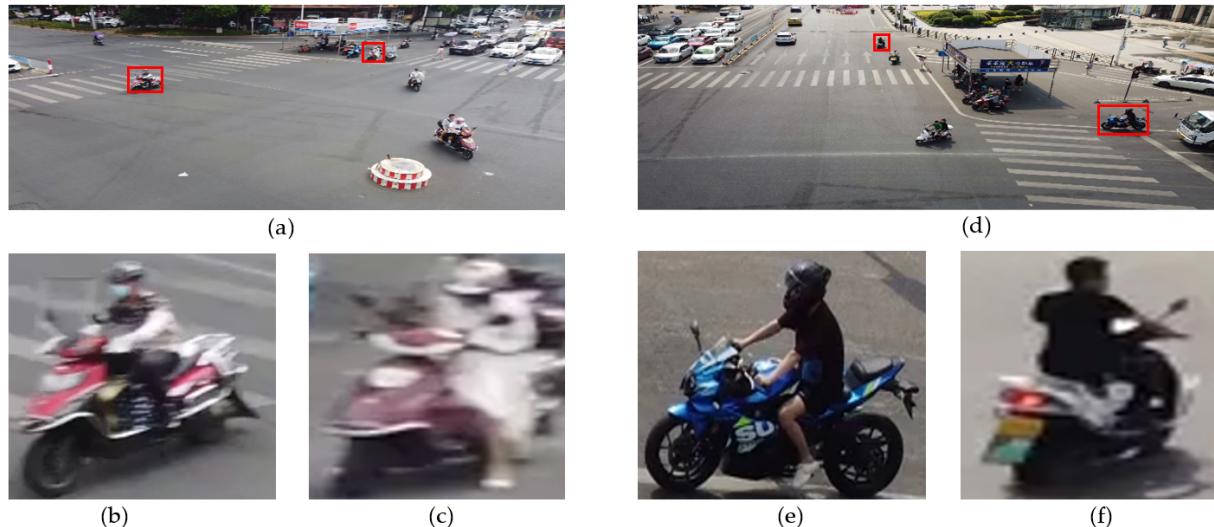
- Evaluation Indicators

The primary problem in this paper is detecting riders and finding as many objects in the aerial image as possible. The second challenge is to classify the detected objects. In Pascal VOC [49] target detection evaluation indicators use the standard evaluation indicator, namely the average precision (AP), the recall rate (Recall), and the mean average precision (mAP). Top1 accuracy (Top1 ACC) is used as the classifier's result evaluation indicator.

More narrowly, the recall is defined by  $\frac{TP}{TP+FN}$ , precision is defined by  $\frac{TP}{TP+FP}$ , where TP (True Positive) indicates the number of correctly detected targets, FP (False Positive) indicates the number of incorrection, and FN (False Negative) indicates the number of missed targets (the targets that should be detected are not detected).  $AP = \sum_{i=1}^n precision(i) \times \Delta recall(i)$ , where  $n$  is the total number of images in the dataset, precision( $i$ ) is the precision at a cut-off point of  $i$  images,  $\Delta recall(i)$  is the difference in recall between cut-off point  $i-1$  and the cut-off point  $i$ . The MAP is obtained by averaging the AP of each class again. When the IoU of the detection box and the label true box is 0.5, the sample is considered positive, and otherwise, a negative sample occurs. The TOP1 ACC is defined by  $\frac{TP+TN}{TP+TN+FP+FN}$ .



**Figure 13.** Image samples from the annotated dataset. (a) is a medium target image taken at close range, while (b) is a small target image taken at long range. (c–e) denote helmet-wearing riders, while those without helmets are denoted as (f–h).



**Figure 14.** Image samples of low quality from the annotated dataset. (a) is a motion blur image, and (b), (c) are local enlargements of it (a). (d) is shot in the backlight, and (e), (f) are hazy images of cyclists (d).

To evaluate the superiority of our method, the experiments are carried out on a computer with NVIDIA 3090 GPU, I7 10700K CPU, and 32GB memory. The software environment is based on Windows10 Professional Edition operating system and Pytorch deep learning framework. The initial learning rate of LMNet is set to 0.01, the optimizer uses the stochastic gradient descent (SGD) algorithm, the momentum parameter is set to 0.937, the decay coefficient is 0.0005, and the batch size is 16. A total of 300 iterations are conducted for the detector model training. In addition, we set the input resolution of all network models to  $704 \times 704$ . Except for the variables mentioned, other variables are consistent in all studies.

#### 4.2. Evaluation Experiments

- Performance Comparison of Detectors

To improve the fusion of high-resolution feature maps and transformer spatial attention, LMNet combines the respective advantages of YOLOv5 and HRNet, which enhances the deep spatial information representation through information interaction. The deep semantic information is fully utilized by transformer spatial attention, which improves the model's accuracy and generalization ability. To validate LMNet's superior performance, Table 1 shows the results of the YOLOv5s, YOLOv5m, YOLOv6s [50], RetinaNet [51], Faster RCNN [7], and LMNet evaluation on the test set. The test results show that cyclists and their vehicles can be directly detected on the original captured images. In Figure 13, we combine (c–h) into a single class for detection, which is a one-class problem. In the two-class problem, (c–e) are classified as having a helmet, whereas (f–h) are classified as not having a helmet, and detectors distinguish between them. According to Table 1, all target detection networks detect only one-class target.

**Table 1.** Evaluation results of different detectors on the one-class problem.

| Item         | mAP   | Precision | Recall |
|--------------|-------|-----------|--------|
| YOLOv5s      | 89.41 | 81.84     | 84.85  |
| YOLOv5m      | 90.79 | 84.17     | 82.73  |
| YOLOv6s      | 89.08 | 86.94     | 84.11  |
| RetinaNet    | 71.21 | 82.95     | 54.28  |
| Faster RCNN  | 78.88 | 57.10     | 88.67  |
| LMNet (Ours) | 91.67 | 84.53     | 87.97  |

As shown in Table 1, our network achieves state-of-the-art (SOTA) detection performance, demonstrating that our network is reasonable and effective in the detection of small aerial targets. YOLOv5s' poor performance is due to its shallow network, which results in insufficient feature information extraction. When compared to YOLOv5s, YOLOv5m performs better because its network is deeper, but its backbone network lacks information interaction, resulting in poor feature information fusion. LMNet is between YOLOv5s and YOLOv5m in terms of parameter number, so YOLOv5x and YOLOv5l are not selected for comparison (the number of network parameters of YOLOv5x and YOLOv5l is larger than that of the LMNet). YOLOv6s has higher precision than our model, but its deeper network and larger parameters result in lower mAP and Recall. Although RetinaNet has effectively addressed the issue of sample imbalance, its backbone network continues to employ the traditional residual structure, resulting in relatively poor detection accuracy for small targets. The reason for Faster RCNN's poor detection effect is that its network's feature map lacks multi-scale feature fusion, and the final feature resolution is usually small, which is difficult for detecting small targets and objects with large-scale changes. As a result, the LMNet ladder backbone network solves the problem of multi-scale feature interaction, and the RT3DsAM resolves the difficulties of information adaptive selection, resulting in good detection performance.

- Comparison of Detectors Performance with and without ESRGAN

As mentioned above, our detection target only accounts for 1.0~6.0% of the pixel area in the entire aerial image. Sliding window detection is one method for improving the accuracy of small aerial photography targets, but it is extremely time-consuming and not suitable for practical engineering applications. Another option is to enhance the image, which has extremely high requirements for outdoor applications. The enhanced image must not only retain the original details and texture, but it must also be supplemented with more realistic details. Four experiments are designed to demonstrate the importance of cropping the detection target box before super-resolution reconstruction and classification. In the first experiments, ESRGAN is used to reconstruct the entire original aerial image, and LMNet is used to detect the one class and two classes of targets. Two classes of targets refer to the simultaneous detection of riders with and without helmets. In the second experiment,

ESRGAN is not used, and LMNet detects one class and two classes, respectively. The corresponding results are shown in Table 2. In the third experiment, ESRGAN reconstructs the entire original aerial image. The target image is then cropped from it. The cropped images are divided into two groups: those with helmets and those without. They are finally classified by YOLOv5. The fourth experiment crops the target image from the original aerial images, then reconstructs the cropped images with ESRGAN, and finally classifies the reconstructed cropped images using the YOLOv5. Table 3 displays the corresponding experimental results.

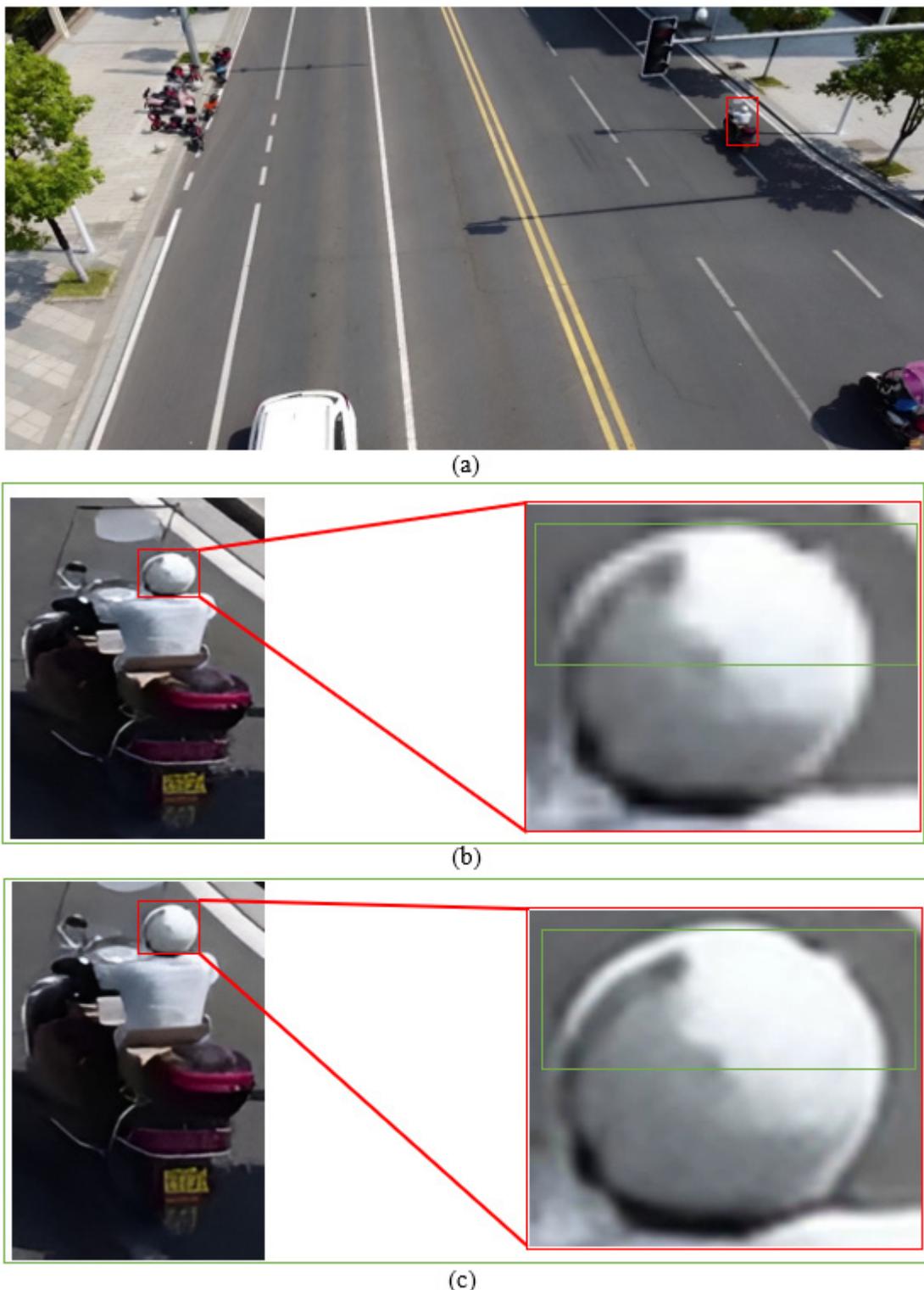
**Table 2.** LMNet performance comparison with and without ESRGAN.

| Item                 | Test Dataset Type | mAP   | Precision | Recall |
|----------------------|-------------------|-------|-----------|--------|
| LMNet with ESRGAN    | One class         | 89.14 | 85.95     | 83.66  |
|                      | Two classes       | 87.12 | 78.66     | 82.95  |
| LMNet without ESRGAN | One class         | 91.67 | 84.53     | 87.97  |
|                      | Two classes       | 87.64 | 80.54     | 83.85  |

**Table 3.** YOLOv5 classifier performance comparison with and without ESRGAN.

| Item                              | Top1 ACC |
|-----------------------------------|----------|
| YOLOv5 with ESRGAN after cropping | 94.23    |
| YOLOv5 without ESRGAN             | 91.88    |

Table 2 shows that in all experiments, the mAP and Recall of LMNet with ESRGAN are almost worse than those without it, indicating that LMNet has a better detection effect when ESRGAN is not used for the original image. The detection effect degrades when ESRGAN is used. We believe that the network model's input resolution is fixed, so more details will be lost when the reconstructed image is compressed. Furthermore, not only does the super-resolution reconstruction of large images take longer than the cropped small target image, but the processing of large-resolution images by models results in a significant increase in time consumption, which has a negative impact on detection speed. In other words, we do not need to use ESRGAN for the original image of aerial photography, just use it for the detected target box image. As a result, our proposed approach has practical implications. When ESRGAN is not used, LMNet detects two classes with lower mAP, Precision, and Recall than one class because riders' structures change in two classes, resulting in less data richness and uneven positive and negative samples (see Table 2). This demonstrates that it is perfectly reasonable for us to combine them into a single class for detection. The experimental results in Table 3 show that the Top1 ACC result of using ESRGAN for classification after cropping is 2.35% higher than that of not using ESRGAN after cropping, indicating an important feature of super-resolution reconstruction networks such as ESRGAN: the target in the image restored from the original image has less detail than the target image cropped from the original image and then reconstructed. We do not assess the strategy of first reconstructing the original aerial image with ESRGAN and then cropping it. As previously stated, one reason is that reconstructing high-resolution photos takes longer than tiny ones. Another reason is that after cropping, the rebuilt image may contain more information. Figure 15 shows that the helmet in (c) has more features and sharper edges than the helmet in (a,b). Consequently, before we apply ESRGAN, we must crop the target box, which has a significant impact on image classification. In terms of model reasoning speed, the LMNet is about 42.7ms per image. The target box cropping and ESRGAN reconstruction take approximately 33.22ms per image, and the classifier can even reach 2ms per image. Because the helmet violation detection is allowed to be delayed to some extent, our method meets the requirements of real-time detection.



**Figure 15.** Image detail comparison. (a) is an original aerial photograph. (b) is the target image extracted from (a) without using ESRGAN and the amplification image of the helmet. (c) is the figure using ESRGAN after cropping and the amplification image of the helmet.

To make our proposed method and networks more convincing, we design two classes of detection experiments without ESRGAN and present their results in Table 4. Table 4 demonstrates that even when the ESRGAN is not used, the mAP value of LMNet remains the highest and has no low Precision and Recall values on two classes of detection, in-

dicating that it has a greater generalization ability. When the experimental results of all detectors are compared to the results in Table 1, we can see that the strategy of not directly detecting two classes is appropriate. As a logical consequence, four stages are required: identifying all targets, cropping their images, reconstructing them with ESRGAN, and finally classifying them.

**Table 4.** Evaluation results of different detectors on the two-class problem.

| Item         | mAP   | Precision | Recall |
|--------------|-------|-----------|--------|
| YOLOv5s      | 86.83 | 79.69     | 82.92  |
| YOLOv5m      | 87.48 | 83.31     | 80.14  |
| YOLOv6s      | 86.12 | 84.87     | 83.27  |
| RetinaNet    | 60.06 | 75.67     | 43.33  |
| Faster RCNN  | 71.24 | 40.97     | 84.59  |
| HRNet-32     | 86.75 | 84.55     | 79.36  |
| LMNet (Ours) | 87.64 | 83.84     | 80.54  |

- Comparison of Attention Module

In this segment, we compare the proposed attention module to existing ones to validate our RT3DsAM’s superiority in target detection. Squeeze-and-excitation networks (SENet) [21], CBAM [22], efficient channel attention network (ECA-Net) [52], global context network (GCNet) [53], simple parameter-free attention module (Simam) [54], non-local neural networks (Non-Local) [30], and Shuffle attention for deep convolutional neural networks (Sa-Net) [55] are among the most advanced visual attention modules. In the ladder-type backbone network, as shown in Figure 4, all attention modules are placed behind Stage4. Table 5 displays the detection results.

**Table 5.** Comparison results of attention modules.

| Item      | mAP   | Precision | Recall |
|-----------|-------|-----------|--------|
| SENet     | 91.20 | 81.55     | 90.13  |
| CBAM      | 90.91 | 84.53     | 74.43  |
| ECA-Net   | 90.80 | 83.04     | 88.76  |
| GCNet     | 91.37 | 82.95     | 89.39  |
| Simam     | 90.98 | 84.52     | 87.30  |
| Non-Local | 90.76 | 83.50     | 89.77  |
| Sa-Net    | 91.33 | 82.05     | 89.59  |
| Ours      | 91.67 | 84.53     | 87.97  |

Table 5 shows that our module has the highest mAP and Precision values, and the Recall value is not low in comparison to other attention modules. We attribute the improved detection effect to two modules: RTB and 3D-spatial attention. The RTB can not only establish the long-distance dependence of cross-channel information, but it can also model the correlation of position pixel relationship in high-level semantic information, which aids in the elimination of deep features for redundant information and the refinement of feature representation of small targets. Meanwhile, unlike the spatial attention in the reference [22], which uses a single 2D attention map to adjust the weights of all layers adaptively, 3D-spatial attention can make adaptive weight adjustments layer-by-layer. Otherwise, each layer’s weight tends to be homogenized. What’s more, layer-by-layer feature adaptive weight adjustment will refine the information learned by each layer of features, allowing for the retention of more differentiated information. The above two modules can effectively increase the inter-class gap while decreasing the intra-class gap, improving target detection accuracy.

- Ablation Study

In this section, we analyze the effects of the proposed algorithm’s various components further. The remaining parameters in the ablation experiments, such as the input resolution

and related hyperparameters, are all the same. In this experiment, we look into the attention module as well as the role of HRNet in LMNet, and the effects of various components in the attention module. First, we consider the distinction between LMNet when one, two, and three attention modules are removed; second, we consider the difference between LMNet when RT3DsAM is placed at different stages; third, we consider the difference between HRNet-32 as the LMNet backbone and the ladder-type backbone network; finally, we consider the impact of each parameter in RT3DsAM on the performance of the attention mechanism. Figure 4 depicts modules ①, ②, ③ as well as branches I, II, III, IV.

As shown in the results in Table 6, the detection effect is the worst when the proposed attention model is not used. As shown in the results in Table 6, the detection effect is the worst when the proposed attention model is not used. The detection effect improves when only one RT3DsAM is used. When two RT3DsAMs are used, the detection effect improves even more than before. We discovered that when the RT3DsAM at branch III is included, the detection effect is superior to those without it. This is because branch III has fewer downsampling multiples and less information loss. It also demonstrates that our attention module can fully utilize and filter the input information. However, using three RT3DsAMs at the same time yields the best LMNet detection effect. HRNet-32 performs worse than LMNet when used as the LMNet backbone without the attention module. Furthermore, when three RT3DsAMs are added to the final output of HRNet-32, the performance improves slightly but remains inferior to ours. Because the HRNet-32 network is too deep and the dataset is insufficient, the extracted information is too redundant, resulting in poor detection accuracy. As shown in the last two lines of Table 6, including RT3DsAM can effectively remove redundant information while also improving detection accuracy. As a result, appropriate deep networks and attention modules are the best solutions for specific applications.

**Table 6.** Ablation analysis results using different attention and backbone models.

| Item  | ① | ② | ③ | HRNet-32 | mAP   | Precision | Recall |
|-------|---|---|---|----------|-------|-----------|--------|
| LMNet | x | x | x | x        | 90.56 | 85.42     | 86.84  |
|       |   |   | x | x        | 90.65 | 85.96     | 84.88  |
|       | x |   | x | x        | 90.73 | 86.20     | 85.29  |
|       | x | x |   | x        | 90.80 | 87.63     | 84.75  |
|       |   |   | x | x        | 91.08 | 86.45     | 85.43  |
|       |   | x |   | x        | 91.22 | 86.01     | 85.33  |
|       |   | x | x | x        | 91.40 | 85.37     | 87.47  |
|       | x | x | x |          | 91.67 | 84.53     | 87.97  |
|       |   |   |   |          | 91.01 | 83.24     | 88.02  |
|       |   |   |   |          | 91.42 | 86.00     | 85.64  |

As shown in Figure 4, we continue to add the RT3DsAM to the output of various stages for study. We add three, three, and two RT3DsAMs to the Stage4 (Branch I, II, III), Stage3, and Stage2 outputs, respectively. Finally, an RT3DsAM is added to Stage1's input and output separately. Table 7 shows the ablation results of using RT3DsAM at each stage.

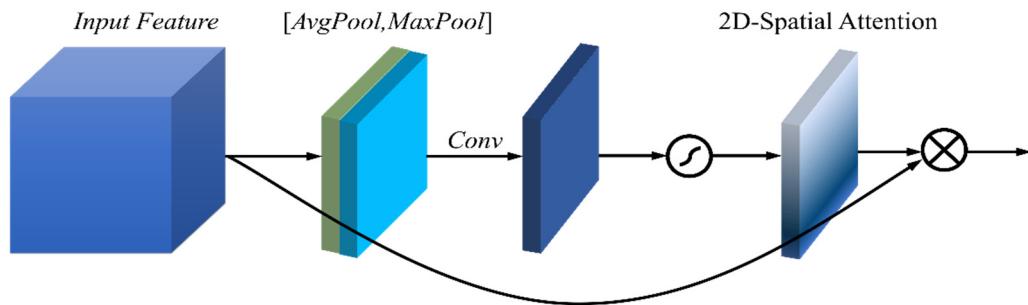
**Table 7.** Ablation analysis of attention on different stages in Figure 4.

| Item          | mAP   | Precision | Recall |
|---------------|-------|-----------|--------|
| Before Stage1 | 89.64 | 82.36     | 85.21  |
| Stage1        | 89.88 | 83.65     | 86.07  |
| Stage2        | 90.96 | 84.09     | 87.27  |
| Stage3        | 91.26 | 86.68     | 86.43  |
| Stage4 (Ours) | 91.67 | 84.53     | 87.97  |

The experimental results in Table 7 show that the location of RT3DsAM in LMNet is optimal, with the highest detection accuracy. Table 7 clearly shows that when the RT3DsAM is located in a shallower layer of the network, its detection accuracy suffers. Because the

shallow features are local, when RT3DsAM is placed in the shallow network, it can only strengthen the correlation between the local features, which leads to the weakening of the correlation between the global information of the deep features, making the network detection effect worse. The results also show that RT3DsAM is capable of high-dimensional global modeling.

In this paper, we also investigate the use of the RCTAM and 3D-spatial attention alone. We design an experiment using only the RCTAM and three additional sets of experiments using only the 3D-spatial attention. Figure 9 shows how the 3D-spatial attention expands the max pooling and average pooling into a 3D feature vector ( $\check{D}$ , as shown in Figure 9). A 3D attention map ( $M_{sa}$ ) is obtained through the sigmoid function. The final output of the 3DsAM consists of  $M_{sa} \times \check{D}$ . When  $\check{D}$  degenerates into a 2D feature, however, the 3DsAM also reduces to a 2D-spatial attention module (2DsAM, as shown in Figure 16) and  $M_{sa}$  becomes a 2D attention map ( $M_{sa}^{2D}$ ). Here we study the effects of  $M_{sa} \times I$  ( $I$  belongs to the input feature vector) and  $M_{sa} \times \check{D}$  in the 3DsAM. While degraded to the 2DsAM, the effects of  $M_{sa}^{2D} \times I$  are considered. It must be noted that the RCTAM remains removed during the study of three experiments. Table 8 displays the experimental results, which show that the effect of RCTAM alone is excellent, indicating that the RTB is highly capable of capturing global correlations. Likewise, in the spatial attention experiment, the 3DsAM outperforms the 2DsAM, and the output of  $M_{sa} \times \check{D}$  has a greater effect on improving the model's detection accuracy. Because the  $1 \times 1$  convolution expands the maximum pooling and average pooling spatial information, all of the input feature's important information is mapped onto a 3D feature vector. Finally, after the  $M_{sa}$  selects the spatial information layer-by-layer, the output feature contains more useful information and fully suppresses redundant information.



**Figure 16.** 2D-spatial attention module.

**Table 8.** Ablation analysis of RCTAM and 3DsAM.

| Model             | Item                             | mAP   | Precision | Recall |
|-------------------|----------------------------------|-------|-----------|--------|
| Spatial Attention | $M_{sa} \times I$                | 90.84 | 82.88     | 88.08  |
|                   | $M_{sa}^{2D} \times I$           | 90.49 | 86.02     | 85.21  |
|                   | $M_{sa} \times \check{D}$ (Ours) | 91.07 | 83.59     | 87.24  |
| RCTAM             | Ours                             | 91.41 | 84.55     | 88.08  |

To further confirm the powerful performance of 3DsAM, we also combine RCTAM to conduct comparative experiments. The first experiment calculates  $M_{sa} \times \check{D}$ , the second keeps all parameters in RCTAM unchanged and then calculates  $M_{sa} \times I$ , the third is  $M_{sa}^{2D} \times I$  while all parameters in RCTAM remain equally unchanged. It must be noted that RCTAM always exists in RT3DsAM during these experiments. The experimental results are shown in Table 9, and it can be observed that almost all the groups with 3DsAM have higher accuracy than the 2DsAM, indicating that  $M_{sa}$  in 3DsAM is very effective not only for the layer-by-layer adaptive adjustment of  $\check{D}$  but also for the adjustment of input feature  $I$ .  $M_{sa}^{2D} \times I$  has poor effect because it loses some important target information when 3DsAM degenerates to 2DsAM.

**Table 9.** Ablation analysis of 3D-spatial attention fusion.

| Item                             | mAP   | Precision | Recall |
|----------------------------------|-------|-----------|--------|
| $M_{sa} \times I$                | 91.18 | 84.34     | 87.79  |
| $M_{sa}^{2D} \times I$           | 91.15 | 82.91     | 88.59  |
| $M_{sa} \times \check{D}$ (Ours) | 91.67 | 84.53     | 87.97  |

#### 4.3. Discussions

The main challenges of helmet wearing and related aerial small target detection are the large intra-class difference in small targets, as well as the small inter-class difference, which causes a slew of issues. Small target detection will be hampered by factors such as drone motion, target motion, camera out-of-focus, and ambient light during aerial photography. There is currently no excellent algorithm for the helmet detection of UAV aerial photography riders due to hardware computing power limitations and the characteristics of small targets [56]. To address this difficult problem, we propose a novel helmet detection paradigm based on attention mechanisms, super-resolution reconstruction, and classification algorithms. Our new paradigm achieves excellent results with their assistance. Furthermore, drone aerial detection can allow people to collect relevant data without regard to location, reduce the need for a large labor force, and effectively force riders to wear safety helmets. Clearly, the above experimental results demonstrate that our proposed new paradigm is robust and has excellent scene generalization capability, allowing us to provide high-precision detection results for relevant UAV aerial photography applications.

Given the difficulty of directly improving the accuracy of multiple classes of small objects in UAV aerial images, we convert the detection method to increase the accuracy of one class of target detection as much as possible. As a result, we combine all classes into one class in order to include all targets. Figure 17 depicts the actual test results, which show that almost all of the riders can be detected with only a few missed targets. We can decrease the confidence if abandoning the pursuit of high detection accuracy and instead focusing on detecting all objects in the image to avoid missing detection. ESRGAN then crops out and reconstructs the detected target boxes. We know from previous experiments that there are more details in Figure 15, which is helpful for subsequent image classification. As shown in Table 3, the classification results of YOLOv5 have improved from 91.88% to 94.23%, indicating that the method of first cropping the target box, then using ESRGAN, and finally using the classifier is meaningful.



**Figure 17.** The LMNet detection results. (a) is the small target detection results; (b) is the medium-target detection results.

There is no doubt that the attention mechanism can improve the robustness of neural networks, but research and discussion on the impact of its internal parameters on model performance are lacking. As a necessary consequence, we first investigate the impact of

the MHSA input resolution in RCTAM on the performance of the attention model. Second, different reduction rates are one of the factors influencing RCTAM performance. Finally, the 3DsAM is discussed in greater detail. In RCTAM, feature maps of  $4 \times 4$  are generated by two global adaptive pooling. Channel attention is generated by RTB, where RTB needs to establish global positional pixel attention across channels through the input of a  $4 \times 4$  feature map, which can build a global long-distance dependence. In addition, input feature maps with different resolutions, such as  $5 \times 5$ ,  $6 \times 6$ , and  $7 \times 7$  can be chosen; however, the higher the resolution of the input feature map, the greater the computation of the model. RTB's minimum input resolution is only  $3 \times 3$ . Alternatively, by varying the reduction rates  $r$ , the RTB structure can form a bottleneck layer (as shown in Figure 7). RTB always sets  $r$  to 1 when selecting different input feature resolutions. As  $r$  is a variable, the resolution of the input feature is fixed at  $4 \times 4$ . In these two groups of experiments, we leave 3DsAM and its parameters alone and only discuss RCTAM's internal parameters. Table 10 displays the results of experiments with various input feature resolutions, while Table 11 displays the results of experiments with various reduction rates.

**Table 10.** Comparison of different input feature resolutions.

| Item      | $7 \times 7$ | $6 \times 6$ | $5 \times 5$ | $4 \times 4$ (Ours) | $3 \times 3$ |
|-----------|--------------|--------------|--------------|---------------------|--------------|
| mAP       | 91.13        | 91.15        | 91.23        | 91.67               | 91.34        |
| Precision | 85.45        | 85.19        | 83.54        | 84.53               | 85.87        |
| Recall    | 87.17        | 86.22        | 88.47        | 87.97               | 87.49        |

**Table 11.** Comparison of different reduction  $r$ .

| Item      | 16    | 8     | 4     | 1 (Ours) |
|-----------|-------|-------|-------|----------|
| mAP       | 90.67 | 90.79 | 90.98 | 91.67    |
| Precision | 85.96 | 84.78 | 86.29 | 84.53    |
| Recall    | 86.53 | 87.47 | 85.73 | 87.97    |

The results in Table 10 show that the RCTAM performance reaches a maximum value when using the input resolution of  $4 \times 4$ . The LMNet performance is improved regardless of the input resolution, as long as RT3DsAM is used. Because the number of data sets is small and the class is only one, increasing the resolution will not improve detection accuracy. The results of the tests in Table 11 show that the reduction rate  $r$  has a significant impact on RCTAM's performance. The model's performance decreases gradually as  $r$  increases, but it improves when compared to the situation in Table 6 when the attention module is not used.

In this section, we study the influence of various 3DsAM parameters on the effectiveness of the attention model. The RCTAM is left unchanged when the 3DsAM's parameters are altered. According to reference [22], the spatial attention module uses average and maximum pooling to gather crucial spatial information. The spatial information from the maximum pooling and mean pooling channels is then compressed into a 2D attention map using the  $7 \times 7$  convolution kernel. There are numerous options for the size of the convolution kernel during the dimension compression process, including  $1 \times 1$ ,  $3 \times 3$ ,  $5 \times 5$ , and  $7 \times 7$ . But in 3DsAM, we use the  $1 \times 1$  convolution kernel to expand the spatial information from the max pooling and average pooling channels to a 3D feature vector. Similar to dimension expansion, there are various options for the convolution kernel's size. Here, the kernels  $1 \times 1$ ,  $3 \times 3$ ,  $5 \times 5$ , and  $7 \times 7$  are chosen. Table 12 shows the experimental results using various convolution kernel parameters.

As can be seen from the experimental results in Table 12, the expansion process is generally superior to compression in the spatial attention module, demonstrating that the expansion process can notice more information and suppress more irrelevant information. The attention model is not significantly affected by different convolution kernel parameters, but it is clear that the model with a  $1 \times 1$  convolution kernel performs the best. The  $1 \times 1$  convolution kernel only changes the channel count, so it barely affects

the spatial information in the channel. Padding will be used by convolution kernels of other sizes to process channel features, potentially increasing the amount of unnecessary interference information.

**Table 12.** Different spatial attention model comparison of kernel parameters.

| Item  | Parameter           | mAP   | Precision | Recall |
|-------|---------------------|-------|-----------|--------|
| 2DsAM | $1 \times 1$        | 91.17 | 82.91     | 88.59  |
|       | $3 \times 3$        | 90.76 | 86.87     | 86.19  |
|       | $5 \times 5$        | 90.84 | 81.97     | 89.73  |
|       | $7 \times 7$        | 90.51 | 85.25     | 85.87  |
| 3DsAM | $1 \times 1$ (Ours) | 91.67 | 84.53     | 87.97  |
|       | $3 \times 3$        | 90.75 | 85.58     | 86.19  |
|       | $5 \times 5$        | 91.17 | 86.75     | 86.54  |
|       | $7 \times 7$        | 91.56 | 83.34     | 88.26  |

The methods used in this paper have three or more benefits. Its detection stability is quite good, first and foremost. Since safety helmet wear detection accuracy is currently quite unstable and low, direct detection will frequently lead to missed and false detections. To enable the classifier to filter the target of false detection once more and super-resolution reconstruction to reduce its false detection, we simply need to ensure that there is no or very little missing detection in this task. Second, the proposed approach is easily accessible. Our network is smaller and has parameters that are only two-thirds of those of HRNet in this study, yet performance improves. The model has exceptional scene generalization ability due to its capacity for long-range modeling and adaptive information selection in the attention module. Third, the department is extremely scalable. The strategy of cropping the target box of observed cyclists can be advantageous for many intelligent transportation applications. Once it has been cropped, we can use the super-resolution reconstruction network to recover the facial information, enabling us to precisely identify illegal bikers. Alternatively, the character recognition network can be used to accurately identify the license plate number if the license plate can be super-reconstructed. There are some issues with our paradigm as well. For instance, during the super-resolution reconstruction phase, we reconstruct our dataset using the weights learned during DF2K and OST dataset training instead of training our super-resolution reconstruction model. If we do, the outcome of the image restoration might be better. In the future, a super-resolution reconstruction network for faces or license plates could be created using our paradigm in order to detect and precisely identify illegal riders.

## 5. Conclusions

We offer a novel paradigm for helmet detection in UAV aerial photography in this research. To begin, we employ a ladder-type backbone network to extract and fuse input information features. Second, the proposed RCTAM and 3DsAM implicitly realize global long-range modeling as well as adaptive layer-by-layer spatial information selection. Third, ESRGAN is used to reconstruct the cropped target box in order to recover the target image's more detailed textures and sharp edges. Finally, the classifier is applied in order to obtain results. A vast number of experimental findings demonstrate that our helmet detection paradigm is extremely valuable, particularly in the field of UAV aerial photography of small targets. Under the present hardware computing power limits, it is nearly the ideal option. In the future, we will combine the super-resolution reconstruction network into the classifier and use the classification results to determine the joint loss function of the super-resolution reconstruction network and the classifier, which can be used to train a hybrid model to improve classification accuracy even further.

**Author Contributions:** Conceptualization, S.C., J.L. and H.L.; Data curation, S.C. and X.W.; Formal analysis, S.C., H.L. and C.C.; Funding acquisition, S.C., H.L., C.C. and X.W.; Investigation, S.C., J.L. and C.C.; Methodology, S.C., H.L. and X.W.; Project administration, S.C., C.C. and X.W.; Resources, S.C., J.L., H.L. and C.C.; Software, X.W.; Supervision, S.C., J.L. and H.L.; Validation, H.L. and C.C.; Visualization, S.C., H.L. and X.W.; Writing—original draft, H.L. and X.W.; Writing—review & editing, S.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Scientific and Technological Innovation Foundation of Foshan, USTB under Grant BK20AF007, the National Natural Science Foundation of China under Grant 61975011, the Fund of State Key Laboratory of Intense Pulsed Radiation Simulation and Effect under Grant SKLIPR2024, and the Fundamental Research Fund for the China Central Universities of USTB under Grant FRF-BD-19-002A.

**Data Availability Statement:** Not applicable.

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

## References

1. World Health Organization (WHO). Global Status Report. Available online: <https://www.who.int/publications/i/item/9789241565684> (accessed on 13 July 2022).
2. National Bureau of Statistics of the People's Republic of China. Available online: <http://www.stats.gov.cn/tjsj/ndsj/2021/indexch.htm> (accessed on 13 July 2022).
3. Shine, L.; Jiji, C.V. Automated Detection of Helmet on Motorcyclists from Traffic Surveillance Videos: A Comparative Analysis Using Hand-crafted Features and CNN. *Multimed. Tools Appl.* **2020**, *79*, 14179. [CrossRef]
4. Li, Y.; Yuan, H.; Wang, Y.; Xiao, C. GGT-YOLO: A Novel Object Detection Algorithm for Drone-Based Maritime Cruising. *Drones* **2022**, *6*, 335. [CrossRef]
5. Mahmudnia, D.; Arashpour, M.; Bai, Y.; Feng, H. Drones and Blockchain Integration to Manage Forest Fires in Remote Regions. *Drones* **2022**, *6*, 331. [CrossRef]
6. Chen, S.; Tang, W.; Ji, T.; Zhu, H.; Ouyang, Y.; Wang, W. Detection of Safety Helmet Wearing Based on Improved Faster R-CNN. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–7.
7. 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.
8. Li, N.; Lyu, X.; Xu, S.K.; Wang, Y.R.; Wang, Y.S.; Gu, Y.W. Incorporate Online Hard Example Mining and Multi-part Combination into Automatic Safety Helmet Wearing Detection. *IEEE Access.* **2021**, *9*, 139536. [CrossRef]
9. Li, Y.; Wei, H.; Han, Z.; Huang, J.; Wang, W. Deep Learning-based Safety Helmet Detection in Engineering Management Based on Convolutional Neural Networks. *Adv. Civ. Eng.* **2020**, *2020*, 9703560. [CrossRef]
10. Han, G.; Zhu, M.; Zhao, X.; Gao, H. Method Based on The Cross-Layer Attention Mechanism and Multiscale Perception for Safety Helmet-Wearing Detection. *Comput. Electr. Eng.* **2021**, *95*, 107458. [CrossRef]
11. Cheng, R.; He, X.; Zheng, Z.; Wang, Z. Multi-Scale Safety Helmet Detection Based on SAS-YOLOv3-Tiny. *Appl. Sci.* **2021**, *11*, 3652. [CrossRef]
12. Zhou, Q.; Qin, J.; Xiang, X.; Tan, Y.; Xiong, N.N. Algorithm of Helmet Wearing Detection Based on AT-YOLO Deep Mode. *CMC Comput. Mater. Contin.* **2021**, *69*, 159. [CrossRef]
13. Chen, W.; Liu, M.; Zhou, X.; Pan, J.; Tan, H. Safety Helmet Wearing Detection in Aerial Images Using Improved YOLOv4. *Comput. Mater. Contin.* **2022**, *72*, 3159. [CrossRef]
14. Jia, W.; Xu, S.; Liang, Z.; Zhao, Y.; Min, H.; Li, S.; Yu, Y. Real-time Automatic Helmet Detection of Motorriders in Urban Traffic Using Improved YOLOv5 Detector. *IET Image Process.* **2021**, *15*, 3623. [CrossRef]
15. Kou, L.; Ding, S.; Wu, T.; Dong, W.; Yin, Y. An Intrusion Detection Model for Drone Communication Network in SDN Environment. *Drones* **2022**, *6*, 342. [CrossRef]
16. Mnih, V.; Heess, N.; Graves, A.; Kavukcuoglu, K. Recurrent Models of Visual Attention. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Montréal, QC, Canada, 8–13 December 2014; pp. 2204–2212.
17. Wang, C.; Shi, Z.; Meng, L.; Wang, J.; Wang, T.; Gao, Q.; Wang, E. Anti-Occlusion UAV Tracking Algorithm with a Low-Altitude Complex Background by Integrating Attention Mechanism. *Drones* **2022**, *6*, 149. [CrossRef]
18. Hu, Z.; Chen, L.; Luo, Y.; Zhou, J. EEG-Based Emotion Recognition Using Convolutional Recurrent Neural Network with Multi-Head Self-Attention. *Appl. Sci.* **2022**, *12*, 11255. [CrossRef]
19. Gregor, K.; Danihelka, I.; Graves, A.; Rezende, D.J.; Wierstra, D. DRAW: A Recurrent Neural Network for Image Generation. *Comput. Sci.* **2015**, *37*, 1462–1471.
20. Jaderberg, M.; Simonyan, K.; Zisserman, A.; Kavukcuoglu, K. Spatial Transformer Networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–10 December 2015; pp. 2017–2025.

21. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
22. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
23. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
24. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.
25. Wang, X.; Yu, K.; Wu, S.; Gu, J.; Liu, Y.; Dong, C.; Loy, C.C.; Qiao, Y.; Tang, X. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. In Proceedings of the European Conference on Computer Vision Workshops (ECCVW), Munich, Germany, 8–14 September 2018; pp. 63–79.
26. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
27. Sun, K.; Xiao, B.; Liu, D.; Wang, J. Deep High-resolution Representation Learning for Human Pose Estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 5693–5703.
28. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L.S. Improving Object Detection with One Line of Code. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 5562–5570.
29. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 779–788.
30. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525.
31. Redmon, J.; Farhadi, A. Yolov3: An incremental Improvement. *arXiv Preprint* **2018**, arXiv:1804.02767.
32. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. Scaled-YOLOv4: Scaling Cross Stage Partial Network. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 13024–13033.
33. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Proceedings of the 18th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), Munich, Germany, 5–9 October 2015; pp. 234–241.
34. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A Deep Convolutional Encoder-decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481. [[CrossRef](#)] [[PubMed](#)]
35. Noh, H.; Hong, S.; Han, B. Learning Deconvolution Network for Semantic Segmentation. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Las Condes, Chile, 11–18 December 2015; pp. 1520–1528.
36. Law, H.; Deng, J. CornerNet: Detecting Objects as Paired Keypoints. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 734–750.
37. Wang, J.; Shao, F.; He, X.; Lu, G. A Novel Method of Small Object Detection in UAV Remote Sensing Images Based on Feature Alignment of Candidate Regions. *Drones* **2022**, *6*, 292. [[CrossRef](#)]
38. Zhou, K.; Zhan, Y.; Fu, D. Learning Region-Based Attention Network for Traffic Sign Recognition. *Sensors* **2021**, *21*, 686. [[CrossRef](#)]
39. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual Attention Network for Image Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6450–6458.
40. Glorot, X.; Bordes, A.; Bengio, Y. Deep Sparse Rectifier Neural Networks. In Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS), Fort Lauderdale, FL, USA, 11–13 April 2011; pp. 315–323.
41. Srinivas, A.; Lin, T.Y.; Parmar, N.; Shlens, J.; Abbeel, P.; Vaswani, A. Bottleneck Transformers for Visual Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 16519–16529.
42. Shaw, P.; Uszkoreit, J.; Vaswani, A. Self-Attention with Relative Position Representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), New Orleans, LA, USA, 1–6 June 2018; pp. 464–468.
43. Bello, I.; Zoph, B.; Le, Q.; Vaswani, A.; Shlens, J. Attention Augmented Convolutional Networks. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3285–3294.
44. Prajit, R.; Parmar, N.; Vaswani, A.; Bello, I.; Levskaya, A.; Shlens, J. Stand-alone Self-attention in Vision Models. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Vancouver, BC, Canada, 8–14 December 2019; pp. 68–80.
45. Zhao, H.; Jia, J.; Koltun, V. Exploring Self-Attention for Image Recognition. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 10073–10082.
46. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-realistic Single Image Super-Resolution Using a Generative Adversarial Network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 105–114.

47. Dong, C.; Loy, C.C.; Tang, X. Accelerating the Super-resolution Convolutional Neural Network. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016; pp. 391–407.
48. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015; pp. 1150–1210.
49. Everingham, M.; Eslami, S.; Van Gool, L.; Williams, C.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes Challenge: A Retrospective. *Int. J. Comput. Vis.* **2014**, *111*, 98. [[CrossRef](#)]
50. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv Preprint* **2022**, arXiv:2209.02976.
51. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2980–2988.
52. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11531–11539.
53. Cao, Y.; Xu, J.; Lin, S.; Wei, F.; Hu, H. GCNet: Non-Local Networks Meet Squeeze-Excitation Networks and Beyond. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Republic of Korea, 27–28 October 2019; pp. 1971–1980.
54. Yang, L.; Zhang, R.Y.; Li, L.; Xie, X. Simam: A Simple, Parameter-Free Attention Module for Convolutional Neural Networks. In Proceedings of the 38th International Conference on Machine Learning (ICML), Virtual, 18–24 July 2021; pp. 11863–11874.
55. Zhang, Q.; Yang, Y. Sa-Net: Shuffle Attention for Deep Convolutional Neural Networks. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 2235–2239.
56. Zhou, H.; Ma, A.; Niu, Y.; Ma, Z. Small-Object Detection for UAV-Based Images Using a Distance Metric Method. *Drones* **2022**, *6*, 308. [[CrossRef](#)]