electronics

MDPI

Article

Object Detection Algorithm of UAV Aerial Photography Image Based on Anchor-Free Algorithms

Qi Hu [1], Lin Li [2], Jin Duan [2,*], Meiling Gao [2], Gaotian Liu [2], Zhiyuan Wang [2] and Dandan Huang [2]

[1] College of Artificial Intelligence, Chang Chun University of Science and Technology, Changchun 130022, China
[2] College of Electronic Information Engineering, Chang Chun University of Science and Technology, Changchun 130022, China
* Correspondence: duanjin@vip.sina.com

Abstract: Aiming at the problems of the difficult extraction of small target feature information, complex background, and variable target scale in unmanned aerial vehicle (UAV) aerial photography images. In this paper, an anchor-free target detection algorithm based on fully convolutional one-stage object detection (FCOS) for UAV aerial photography images is proposed. For the problem of complex backgrounds, the global context module is introduced in the ResNet50 network, which is combined with feature pyramid networks (FPN) as the backbone feature extraction network to enhance the feature representation of targets in complex backgrounds. To address the problem of the difficult detection of small targets, an adaptive feature balancing sub-network is designed to filter the invalid information generated at all levels of feature fusion, strengthen multi-layer features, and improve the recognition capability of the model for small targets. To address the problem of variable target scales, complete intersection over union (CIOU) Loss is used to optimize the regression loss and strengthen the model's ability to locate multi-scale targets. The algorithm of this paper is compared quantitatively and qualitatively on the VisDrone dataset. The experiments show that the proposed algorithm improves 4.96% on average precision (AP) compared with the baseline algorithm FCOS, and the detection speed is 35 frames per second (FPS), confirming that the algorithm has satisfactory detection performance, real-time inference speed, and has effectively improved the problem of missed detection and false detection of targets in UAV aerial images.

Keywords: object detection; drone aerial photography; global context block; multi-scale feature fusion; adaptive equalization network

Citation: Hu, Q.; Li, L.; Duan, J.; Gao, M.; Liu, G.; Wang, Z.; Huang, D. Object Detection Algorithm of UAV Aerial Photography Image Based on Anchor-Free Algorithms. *Electronics* **2023**, *12*, 1339. https://doi.org/10.3390/electronics12061339

Academic Editor: Donghyeon Cho

Received: 18 January 2023
Revised: 14 February 2023
Accepted: 26 February 2023
Published: 11 March 2023

1. Introduction

In recent years, Unmanned aerial vehicles (UAVs) have been widely used in traffic monitoring, sea area search and rescue, aerial photography, and other fields due to their small size, convenient operation, and high imaging resolution. UAV object detection is one of the important branches of computer vision tasks, and the target instances in the images can be captured efficiently by processing the images captured by UAVs.

The design of traditional object detection algorithms is mainly based on artificially constructed features, such as scale invariant feature transform (SIFT) [1], Haar-like (Haar) [2], Deformable Part Model (DPM) [3], etc.

However, its limitations are that the manually designed features require a large amount of prior knowledge, fail to make full use of deep semantic information, and have weak generalization ability. In recent years, with the rise and development of deep learning technology, the use of Convolutional Neural Networks (CNNs) has been applied to object detection tasks.

CNN-based object detection algorithms are generally divided into two categories, namely, two-stage algorithms and single-stage algorithms. The two-stage algorithm is to

first generate a series of candidate frames as samples by the algorithm, and then classify the samples through CNNs. The single-stage object detection algorithm does not need to generate a candidate frame, but directly predicts the bounding box and target type of the object. Typical representatives of two-stage algorithms include region-CNN (R-CNN) [4], Faster R-CNN [5], Mask R-CNN [6], etc. Typical representatives of single-stage algorithms include You Only Look Once (YOLO) [7], single shot multi-box detector (SSD) [8], RetinaNet [9], etc. Aiming at the problems of object detection in UAV aerial images, many scholars have carried out a series of studies. Liu et al. [10] designed and added a multi-branch parallel feature pyramid network (MPFPN) on the Faster R-CNN and introduced a supervised spatial attention module (SSAM) to effectively improve the detection performance of UAV image targets in complex backgrounds, but the detection of small targets still needs to be improved. Liang et al. [11] proposed a spatial context analysis method for object re-inference based on the SSD algorithm, which greatly improves the detection accuracy of small targets, but there are false detection cases for targets in complex contexts. Zhou et al. [12] designed a metric-based object classification method to solve the classification problem of untrained subclass objects and modified the localization loss function to improve the localization performance of small objects.

As for the object detection algorithm, it can be divided into anchor-based algorithm and anchor-free algorithm according to the setting of anchor frame or not. The anchor-based method needs to pre-set a certain number of anchors at each position in the feature map of the image, and then classify and regress each anchor. The anchor-free method does not need to pre-set the anchor and directly detects the object on the image. The main difference between the two methods is whether to use anchor to generate proposal. Compared with the anchor-based algorithm, the anchor-free algorithm can greatly reduce the amount of additional parameters and reduce the memory occupied by the calculation. Many anchor-free networks that have emerged in recent years are also suitable for object detection of UAV aerial images. For example, CornerNet [13] proposed for the first time to predict the target as a pair of key points through a single neural network, using box-to-corner prediction instead of anchor for localization and target detection. CenterNet [14] models the detection object as a single center point of the bounding box and uses the heat map generated by the convolutional network to predict and classify the single centroid. Zhang et al. [15] improved on the basis of YOLOX network and proposed the skip scale feature enhancement module BiNet, which effectively improved the detection accuracy of small targets. Inspired by FoveaBox, Liu et al. [16] reset the target detection layer and proposed a HollowBox algorithm for multi-size features, which effectively reduces the false detection probability of drone detection. Hou et al. [17] applied the fully convolutional one-stage object detection (FCOS) algorithm to ship detection to further improve the detection performance of ship targets. Mao et al. [18] proposed ResSARNet based on the improvement of FCOS to obtain powerful detection performance by compressing the model parameters. The above anchor-free frame algorithm, in which FCOS performs detection by pixel-by-pixel point-wise regression, not only gets rid of the anchor frame but also outperforms most target detection algorithms in terms of performance. However, it still has limitations. Although the algorithm uses feature pyramid network (FPN) for multi-level prediction, the detection effect is still unsatisfactory for targets with large scale changes and cases where different targets overlap each other.

Therefore, this paper uses the single-stage target detection algorithm FCOS without anchor frames as the benchmark algorithm to improve it. The main contributions of the article are as follows: (1) To improve the backbone network, introduce the Global Context Block (GC-Block) into the residual block of the ResNet50 network, and improve the network's capture of UAV targets in complex backgrounds ability. (2) Propose the Adaptive Feature Balancing Subnet (AFBS) structure, which can effectively balance the low-level and high-level features from the multi-level feature map, avoiding the dilution of its information flow when passing across layers, thus effectively improving the detection accuracy of small targets. (3) Use complete intersection over union (CIOU) Loss to optimize the regression

loss, thus giving the model regression process scale sensitivity and strengthening the algorithm's ability to detect multi-scale targets.

2. Materials and Methods

2.1. Baseline

FCOS is a single-stage anchor-free object detection algorithm based on FCN proposed by Tian Z et al. [19], which detects by means of pixel-by-pixel regression. The specific method is that FCOS performs a regression operation on each feature point on the feature map to predict four values (l,r,t,d), which, respectively, represent the distance from the feature point to the upper, lower, left, and right sides of the target boundary frame. As shown in Figure 1, the network consists of three parts: the backbone network (Backbone), the feature pyramid (Feature Pyramid Network, FPN) [20], and the output section Detection head, which includes Classification, Regression, and Center-ness branches.

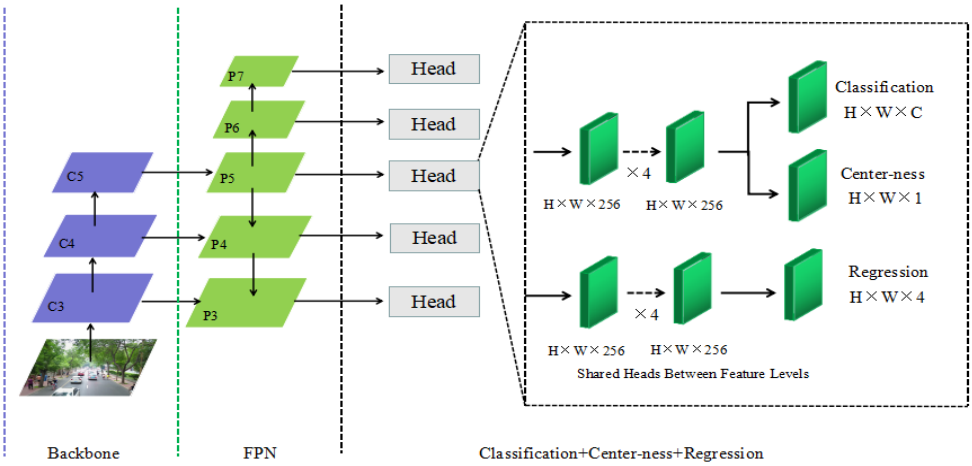


Figure 1. FCOS network architecture.

FCOS mainly has the following advantages: (1) By getting rid of the anchor box, it avoids the complex intersection over union (IOU) calculation and reduces the training memory footprint. (2) It can be used as a Region Proposal Network (RPN) for two-stage detectors, and its performance is significantly better than anchor-based RPN. (3) Strong universality, the improved model can be applied to other visual tasks. In summary, this paper chooses the FCOS algorithm as the benchmark algorithm.

2.2. Algorithm of This Paper

The algorithmic network architecture of this paper is shown in Figure 2.

The model uses the ResNet50 network for feature extraction of the input image to obtain the initial features, selects the obtained C3, C4, and C5 features to send to FPN for feature fusion, and then uses the outputs P3, P4, and P5 as the input feature map of adaptively spatial feature fusion (ASFF) [21]. Firstly, ASFF adjusts and integrates the features of other levels to the same resolution and then multiplies and, finally, sums the fusion with the corresponding weights of the feature maps at each level, and the features of different levels are adaptively fused to achieve the purpose of filtering conflicting information. The output feature maps from this network are M3, M4, M5, and M5 are down-sampled twice to obtain M6 and M7, respectively. The five-level features of M3, M4, M5, M6, and M7 are used as the input of balanced feature pyramid (BFP) [22], which first integrates the five-level features to generate more balanced semantic features and then refines to obtain the more differentiated feature maps N3, N4, N5, N6, and N7. Finally, the identity (layer-by-layer addition) operation is executed to add M3~M7 to N3~N7, correspondingly, to enhance the original features. The detection head located at the end of the network detects the enhanced 5-layer features, which enter the detection head first

through 4 H×W×256 convolutional layers for feature enhancement and then upstream in parallel through H×W×C and H×W×1 convolution to obtain two branches of classification and center-ness. The center-ness reflects the distance of a point on the feature map from the target center. By multiplying the predicted category probability with the corresponding center-ness, the bounding boxes with high scores are kept in order according to their scores, so that low-quality bounding boxes are filtered out in the non-maximum suppression (NMS) process, and the regression detection results are obtained by H×W×4 convolution in the downstream.

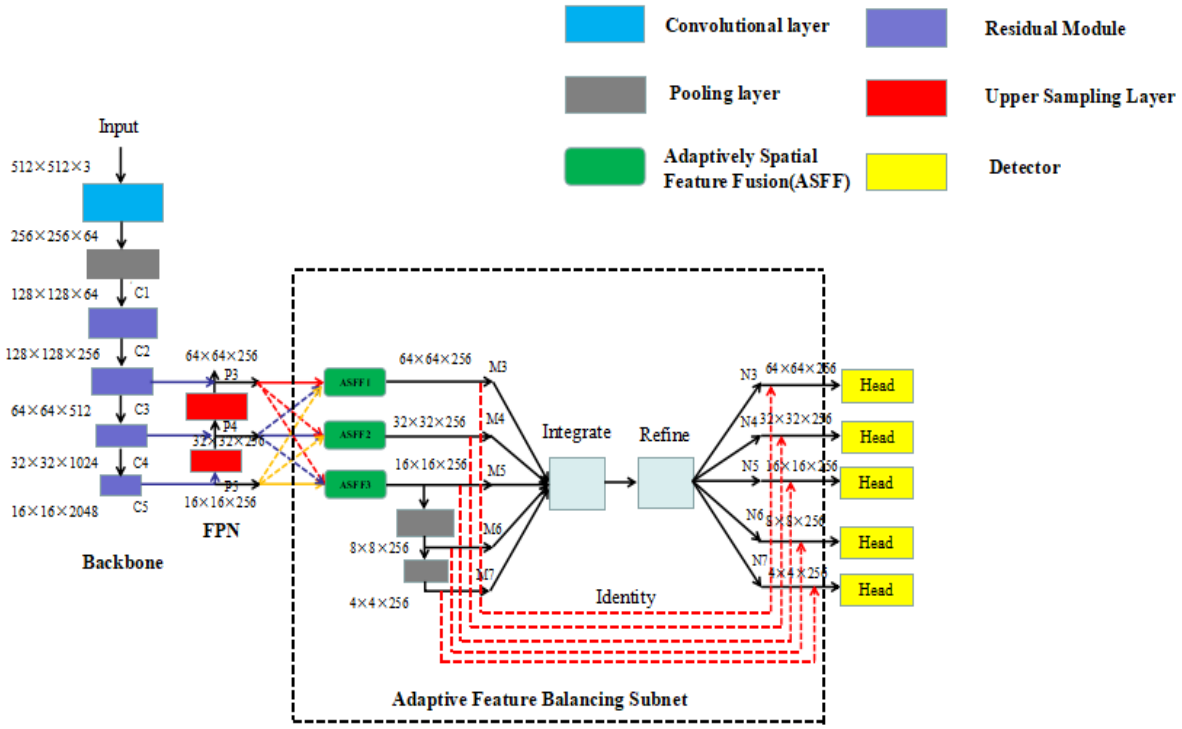


Figure 2. The algorithm network architecture of this paper.

2.2.1. Improved Backbone Network

The general target detection model uses convolution operation to extract image features, but, since the convolution kernel only acts on the local receptive field, only the depth stacking of the convolution layer can associate all the regional information of the image. Multiple convolution stacking will increase the difficulty of training, and the network learning efficiency will be low, which will greatly reduce the positioning accuracy of the model for UAV image targets. In order to solve the above problems, this paper introduces the global context block (GC-Block) [23] to improve the residual block of ResNet50, strengthens the ability of ResNet50 to capture long-distance dependencies, and uses the self-attention mechanism in the module to model the dependencies between long-distance pixels on the image. The improved backbone network is shown in Figure 3.

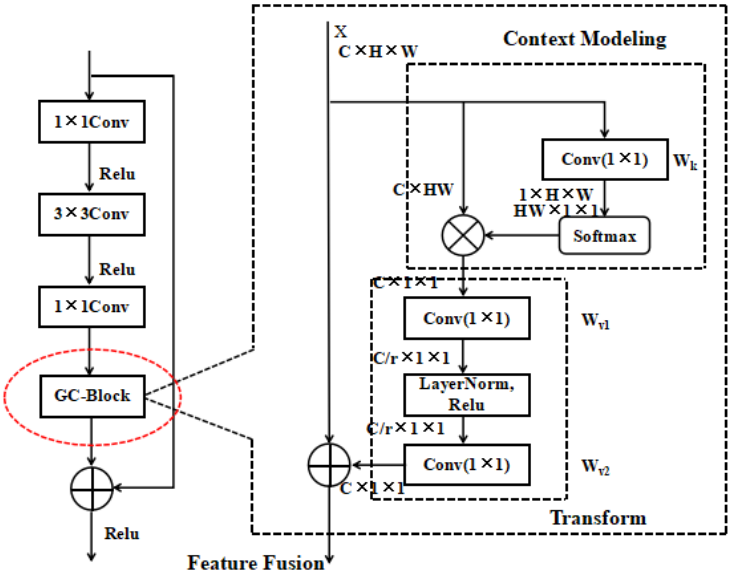


Figure 3. Improved backbone network structure.

2.2.2. Adaptive Feature Equalization Subnetwork

Adaptive Feature Balancing Subnet (AFBS) consists of two parts: ASFF and BFP. The sub-network can not only adaptively learn the spatial weight of the multi-scale feature map, but also use the deeply integrated balanced semantic features to balance and strengthen the multi-level feature information, thus the information of small objects can be completely displayed. The network structure is shown in Figure 4.

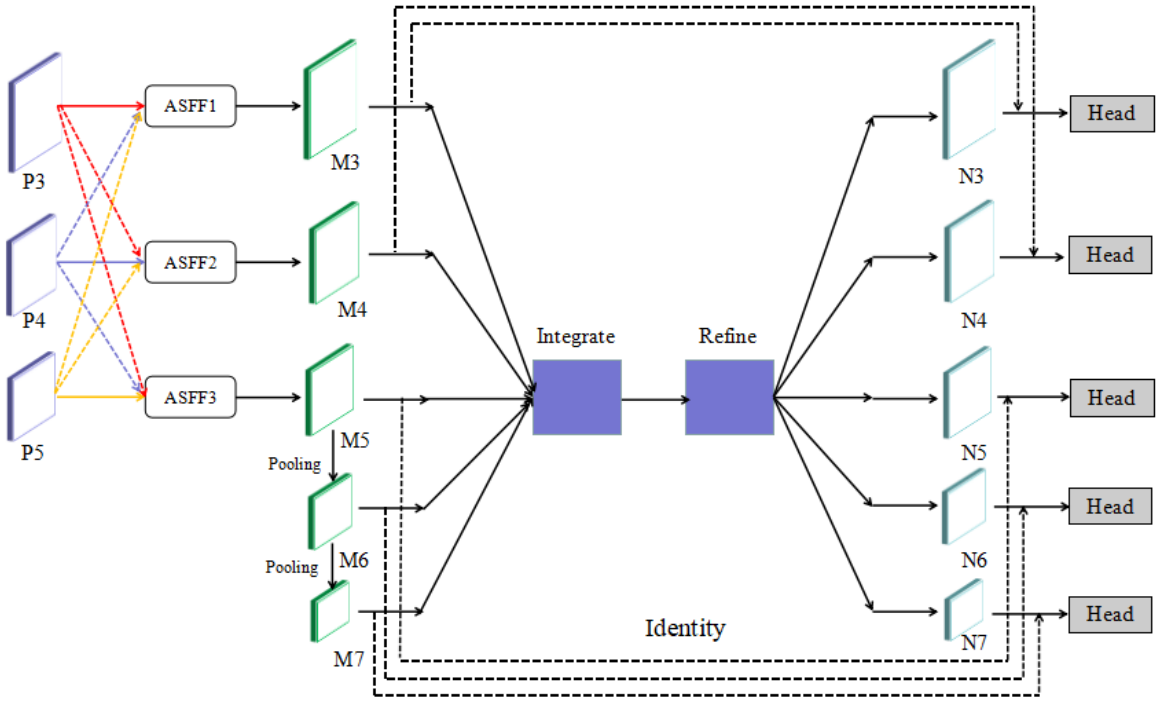


Figure 4. Architecture diagram of Adaptive Feature Equalization Subnetwork.

Adaptive Spatial Feature Fusion Module

The key idea of adaptive spatial feature fusion is to learn the fused spatial weights of features at different scales. multiply the learned parameters of each weight with the input to filter conflicting information and retain useful information to solve the problem of

conflicting information when multi-layer features are fused. The specific implementation steps of this method are as follows:

(1) Feature input. Input the feature maps of different scales in the backbone network.
(2) Feature scaling. Scaling is to keep the channel of feature fusion the same. For the feature layer that needs to be upsampled, first use 1×1 convolution to adjust the number of channels to be consistent with the target layer, and then use interpolation to increase the resolution and adjust the size. For the $1/2$ scale downsampling layer, a convolution of size 3×3 with stride 2 is used. For the $1/4$ scale downsampling layer, it is necessary to add a maximum pooling layer with a stride of 2 to the convolution with a size of 3×3 and a stride of 2.
(3) Feature Fusion. Assuming that the target layer is l, $x_{i,j}^{n \to l}$ represents the feature vector adjusted from layer n to layer l at feature map (i, j), and α_{ij}^{l}, β_{ij}^{l}, and γ_{ij}^{l} are the spatial weight parameters of features $x^{1 \to l}$, $x^{2 \to l}$, and $x^{3 \to l}$ fused to layer (i, j) at l, respectively. The feature vectors of different feature maps at (i, j) are multiplied with their respective weights and then summed. l layer fusion outputs the following equation:

$$F_{ij}^{l} = a_{ij}^{l} \cdot x_{ij}^{1 \to l} + \beta_{ij}^{l} \cdot x_{ij}^{2 \to l} + \gamma_{ij}^{l} \cdot x_{ij}^{3 \to l} \tag{1}$$

where the weights α, β, γ represent the spatial importance of the features at different levels, ranging from [0, 1] and summing to 1, generated using the Softmax function and with $\lambda_{\alpha_{ij}}^{l}$,$\lambda_{\beta_{ij}}^{l}$,$\lambda_{\gamma_{ij}}^{l}$ as control parameters, calculated as follows:

$$a_{ij}^{l} = \frac{e^{\lambda_{aij}^{l}}}{e^{\lambda_{aij}^{l}} + e^{\lambda_{\beta ij}^{l}} + e^{\lambda_{\gamma ij}^{l}}} \tag{2}$$

Balanced Feature Pyramid

The balanced feature pyramid fully fuses the multi-dimensional features of different depth feature maps; thus, the fused features take into account both powerful semantic information and rich geometric information. The work process is divided into four steps:

(1) Feature size adjustment

The five features M3, M4, M5, M6, and M7 participating in feature fusion are adjusted to the same resolution through interpolation and maximum pooling operations. Because choosing a larger resolution will increase the network computing burden, a smaller resolution will be detrimental to small target detection. Therefore, this paper uniformly adjusts the same size as M5, and this process can avoid the input of additional parameters.

(2) Feature fusion

Feature fusion is to integrate features of different sizes and resolutions to remove redundant information, as to obtain better feature expression. The fusion is performed as follows to obtain balanced semantic features:

$$C = \frac{1}{L} \sum_{\min}^{\max} C_l \tag{3}$$

Among them, C_l represent the l layer feature, $l_{\min}$ and $l_{\max}$ denote the highest and lowest layer features, respectively.

(3) Feature refinement

The Gaussian non-local module [24] is used to refine the fused features. This module can refine the fused semantic features to make them more distinguishable, thereby further improving the performance of object detection in the UAV scene.

(4) Feature enhancement

The idea of strengthening comes from the design concept of the residual structure. M3~M7 are added correspondingly to the optimized features through cross-connection and finally output N3~N7.

2.2.3. Loss Function

The loss function, as the basis for the deep neural network to judge the false detection samples, largely influences the model's convergence effect, while providing optimization direction for the training of object detection network. The loss function of the algorithm in this paper contains three main components: Focal Loss is used as the classification loss function, Binary Cross Entropy (BCE) is used as the loss function of center-ness branch, and CIOU [25] is used as the regression loss function. The total loss L is defined as follows:

$$L = L_{cls} + L_{center} + L_{reg} \tag{4}$$

L_{cls} is the classification loss, L_{center} is the loss of center-ness branch, and L_{reg} is the regression loss.

(1) Classification loss function

Focal Loss is a loss function used to deal with unbalanced sample classification. When there are too many negative samples, the classification accuracy will be reduced. By reducing the weight of easily classified samples, Focal loss enables the model to learn difficult classified samples in a centralized manner, as to prevent a large number of easily classified negative samples from dominating model training in the training process. The formula is as follows:

$$L_{Focal} = \begin{cases} -(1-\alpha)y*^{\gamma}\log(1-y*), y = 0 \\ -\alpha(1-y*)^{\gamma}\log y*, y = 1 \end{cases} \tag{5}$$

Among them, y is the real value, y^* is the predicted value, which α is a balance factor to balance the importance of positive and negative samples, and the value range is [0, 1], which γ is an adjustable focal length parameter.

(2) Binary Cross Entropy loss function.

FCOS uses the center-ness branch to suppress low-quality detection frames in UAV image samples. The regression object's center-ness of a certain position in the sample is defined as follows:

$$Centerness* = \sqrt{\frac{\min(l^*, r^*)}{\max(l^*, r^*)} \times \frac{\min(t^*, b^*)}{\max(t^*, b^*)}} \tag{6}$$

Among them, the l^*, r^*, t^*, b^* represent vertical distances from the point to the upper, lower, left, and right boundaries of the ground truth box, respectively.

(3) Improved regression loss function

The regression loss is mainly used to train the ability of the model to accurately locate the small target of the UAV. The benchmark algorithm uses IOU Loss as the regression loss. The value of IOU is 0 when the two boundary frames do not overlap. It is effective only when the two boundary frames overlap, the actual distance between the predicted frame and the real frame cannot be judged.

Therefore, this paper adopts CIOU Loss instead of IOU Loss. CIOU not only considers the overlap area and center point distance but also the aspect ratio in the process of bounding box regression, CIOU Loss can overcome its own defects while making full use of the advantages of IOU Loss and is sensitive to the transformation of the target's

bounding box shape, which is more conducive to the detection of UAV multi-scale targets. The expressions of IOU and CIOU are as follows:

$$IOU = \frac{B \cap B^{gt}}{B \cup B^{gt}} \tag{7}$$

$$L_{CIOU} = 1 - IOU + \frac{\rho^2(B, B^{gt})}{C^2} + \beta v \tag{8}$$

Among them:

$$\beta = \frac{v}{(1 - IOU) + v} \tag{9}$$

$$v = \frac{4}{\pi^2}(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h})^2 \tag{10}$$

β is a positive trade-off parameter, and v is used to measure the consistency of the aspect ratio. B is the predicted frame, B_{gt} is the ground truth, and C is the minimum frame diagonal length containing two frames.

2.3. Experimental Conditions

2.3.1. Dataset

The data used in this paper comes from the VisDrone [26] image target detection public dataset. The dataset includes 10 categories: pedestrians (people with walking or standing posture), people (people with other posture), cars, vans, buses, trucks, motorcycles, bicycles, awning tricycles, and tricycles. The VisDrone dataset is composed of 288 video clips, providing a total of 10,209 static images captured by drones of different heights, including 6471 images for training, 548 images for validation, and 3190 images for testing, totaling 2.6 million target instance samples.

2.3.2. Experiment Settings

The experimental platform in this paper used the Ubuntu 18.04 operating system. The GPU was an RTX A4000 16 G, and the CPU was an Intel(R) Xeon(R) Gold 5320 CPU @ 2.20 GHz. The deep learning framework chosen was PyTorch, and the input image size was 512 × 512. When building the network, the batch size was 8, the training was 100 epochs, the initial learning rate was set to 0.001, and the Adam optimizer was used.

2.4. Evaluation Metrics

In order to verify the effectiveness of the algorithm in this paper, evaluation was performed from both qualitative and quantitative aspects. Qualitative analysis was mainly evaluated from a subjective perspective, and quantitative analysis was mainly evaluated from objective evaluation indexes as a reference.

In this paper, comprehensive average precision AP (Average Precision), AP_S, AP_M, AP_L, FPS (Frame Per Second), Params (Parameters), and FLOPs (Floating Point Operations) indicators are used to evaluate the performance of the model. *AP* means that the IOU is within the range of [0.50, 0.95], with a step of 0.05. A total of 10 thresholds are used to change the comprehensive average precision. The higher the *AP* value, the better the detection effect of the algorithm. The formula is shown in (11).

$$AP = \frac{1}{classses}\sum_{c}(\frac{1}{|thresholds|}\sum_{t}\frac{TP(t)}{TP(t) + FP(t)}) \tag{11}$$

In the formula, classes and thresholds represent the number of target categories and the IOU threshold, respectively. c is the element in classes, and t represents the value in the threshold interval. *TP* is True Positives, representing positive samples that are correctly classified. *FP* stands for False Positives, which represent positive samples that have been misclassified. FPS is used to evaluate the real-time performance of the model, and the

higher the value the better the real-time performance of the algorithm. According to the COCO evaluation system, AP_S, AP_M, and AP_L, respectively, represent the absolute pixel area of the object under small (area less than 32^2), medium (area greater than 32^2, less than 96^2), and large (area greater than 96^2) average precision.

Params is the total number of parameters in the network layer including parameters, which measures the space resource occupation of the model, the formula is shown in (12).

$$Params = \sum_{l=1}^{D} K_l^2 \times N_{l-1} \times N_l \tag{12}$$

Among them, D represents the total number of layers of the network, K_l, N_{l-1}, and N_l are the convolution kernel size, the number of input and output channels, respectively.

FLOPs measure the number of floating-point operations of the model, reflecting the computational complexity of the model. The formula is shown in (13).

$$FLOPs = \sum_{l=1}^{D} H_l \times W_l \times K_l^2 \times N_{l-1} \times N_l \tag{13}$$

In the formula, D represents the total number of layers of the network, H_l, W_l represent the height and width of the output feature map of the layer, and K_l, N_{l-1}, and N_l are the convolution kernel size and the number of input and output channels, respectively.

3. Results

3.1. Module Ablation Experiment

Baseline is FCOS algorithm, M1 is FCOS + GC-Block, M2 is FCOS + GC-Block + AFBS, M3 is FCOS + GC-Block + AFBS + CIOU, which is the algorithm in this paper. All experiments are tested on the VisDrone dataset, using AP, FLOPs, Params as metrics. The final performance comparison results are shown in Table 1.

Table 1. Comparison of ablation experiments.

Model	Baseline	GC-Block	AFBS	CIOU	AP (%)	FLOPs (G)	Params (M)
FCOS	√				18.86	77.79	32.02
M1	√	√			19.95	77.83	34.12
M2	√	√	√		23.43	82.73	39.32
M3	√	√	√	√	23.82	82.73	39.32

According to the experimental results in Table 1, compared with the baseline algorithm, it can be seen that, the AP of M1 has increased by 1.09%, and the Params increased by 2.1 M, the FOLPs have only increased by 0.04 G, which shows that the introduction of GC-Block increased the detection accuracy while generating negligible computational overhead. Compared with the baseline algorithm, M2 has increased AP by 4.57%, FLOPs increased by 4.94 G, and Params increased by 7.3 M, which shows that although AFBS improves the detection accuracy of the model through a stronger ability to adaptively fuse different feature information, the complex network structure increases the computational complexity of the model. M3 is the algorithm proposed in this paper, and the overall performance of the network reached the highest gain. Compared with the baseline algorithm, it increased AP by 4.96%.The values of the two evaluation indicators PLOPs and Params are basically the same as those in M2, which also shows that changing the loss function does not affect the calculation amount of the model.

In order to further evaluate the detection effect of the improved algorithm proposed in this paper in real special scenes, UAV aerial images with dense distribution of small targets, multi-scale targets and complex backgrounds are selected in the VisDrone dataset, and the FCOS algorithm and the algorithm in this paper are tested. The effect comparison is shown in Figure 5.

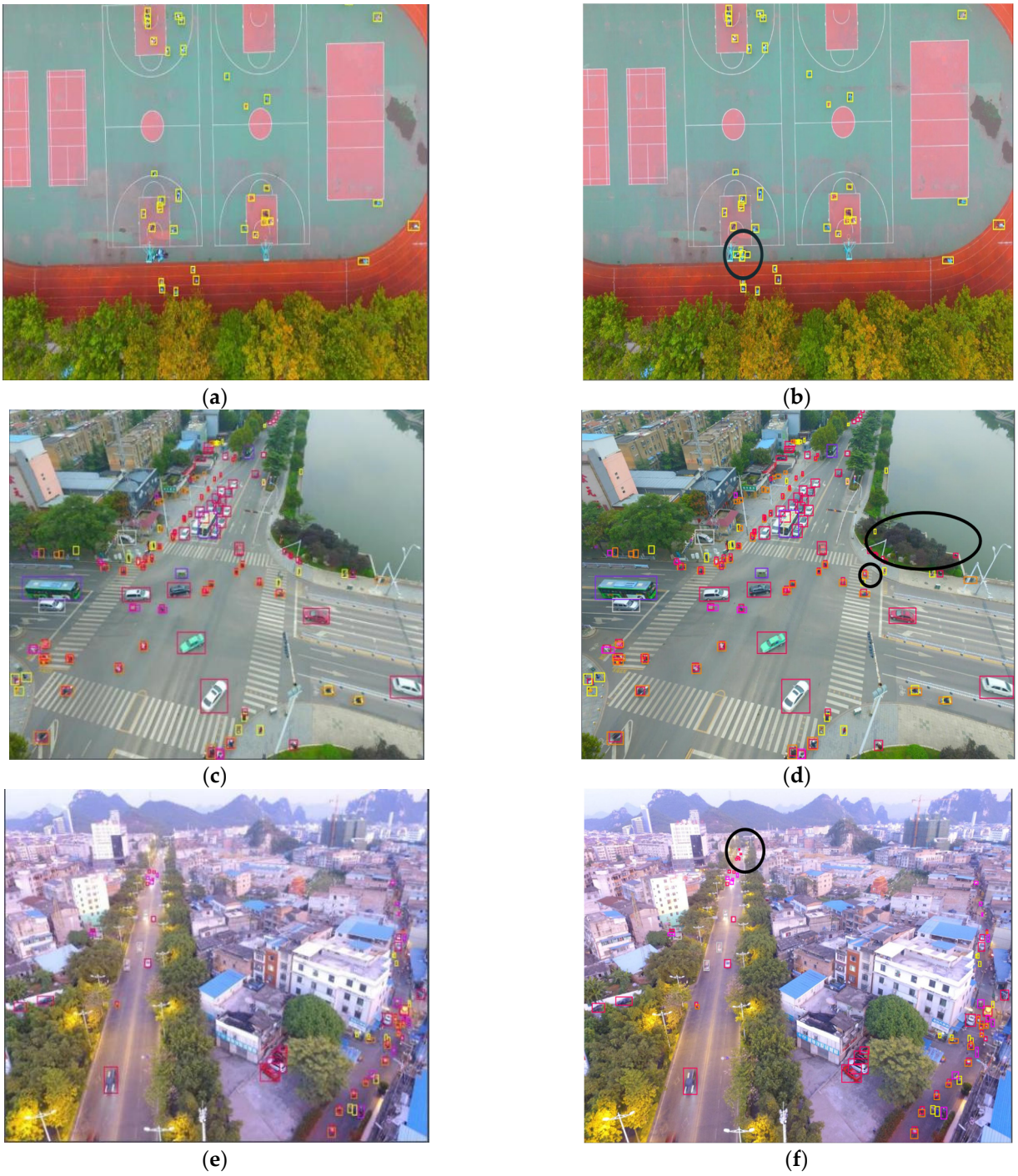


Figure 5. Visual comparison of detection effect between the FCOS algorithm and the improved algorithm in this paper. (**a**,**c**,**e**) are the detection results of FCOS ; (**b**,**d**,**f**) are the detection results of the algorithm in this paper.

Comparing Figure 5a,b, in the case of dense distribution of small targets, the FCOS algorithm mistakenly recognizes the school uniforms stacked next to the basketball poles as people, while the algorithm in this paper does not have this error. Comparing Figure 5c,d, there are a large number of targets of different scales in the figure. The FCOS algorithm did not recognize the cars on the river bank, the people in the grass, and the tricycle driving on the sidewalk on the right, and missed detection. The algorithm in this paper can better adapt to the change in the target size and thus accurately identify it. Comparing Figure 5e,f, in the case of complex background environments, the algorithm in this paper

can still identify vehicles farther away on the road, and it can also detect overlapping targets normally, while FCOS misses detection. According to the comparison, it can be seen that the algorithm in this paper can better combine the superior information in high-level features and low-level features by adaptively fusing multi-layer features and has stronger identification and positioning capabilities for small targets and multi-scale targets.

3.2. Comparative Experiment

In order to verify the effectiveness of the algorithm in this paper, the model in this paper is compared with the current classic model. All experiments are trained on the VisDrone dataset and tested under the same hardware conditions. The experimental results are shown in Table 2.

Table 2. Performance comparison of each algorithm.

Method	Backbone	AP (%)	AP_S (%)	AP_M (%)	AP_L (%)	FPS	FLOPs (G)	Params (M)
Faster R-CNN	ResNet50	16.49	7.25	25.32	37.73	16	79.21	41.18
SSD	VGG-16	12.03	5.75	20.12	35.04	40	37.60	26.47
RetinaNet	ResNet50	16.85	7.91	23.97	36.82	23	84.35	37.03
R-FCN	ResNet101	19.65	9.89	26.35	41.28	19	132.38	78.16
YOLOV3	CSPDarkNet	15.05	6.28	21.45	36.18	38	75.14	61.50
FCOS	ResNet50	18.86	8.65	25.01	36.32	25	77.79	32.02
Proposal	ResNet50	**23.82**	**14.11**	**27.25**	**41.85**	**35**	**82.73**	**39.32**

As can be seen from Table 2, the Params of the single-stage target detection algorithm SSD is 26.47 M, the FLOPs are 37.60 G, and the AP value is 12.03% lower than other algorithms, but this algorithm has a greater advantage in Params. It can also be seen that although the R-FCN algorithm has relatively high detection accuracy, its computational complexity is also the highest. Compared with several other classical algorithms, the proposed algorithm has achieved the best detection effect. Among them, the improvement of small target detection accuracy is the most evident. Compared with the suboptimal R-FCN algorithm, the AP has increased by 4.22%, and the inference speed is relatively high. The FPS value is 35, and the FLOPs and Params are 82.73 G and 39.32 M, respectively. To sum up, the proposed algorithm achieves better detection performance on the premise of maintaining a small computational overhead, and it has great advantages compared with other algorithms in processing UAV aerial photography image target detection tasks.

Figure 6 is a visual comparison between the algorithm in this paper and other mainstream algorithms, which more intuitively reflects the detection accuracy and speed of each algorithm.

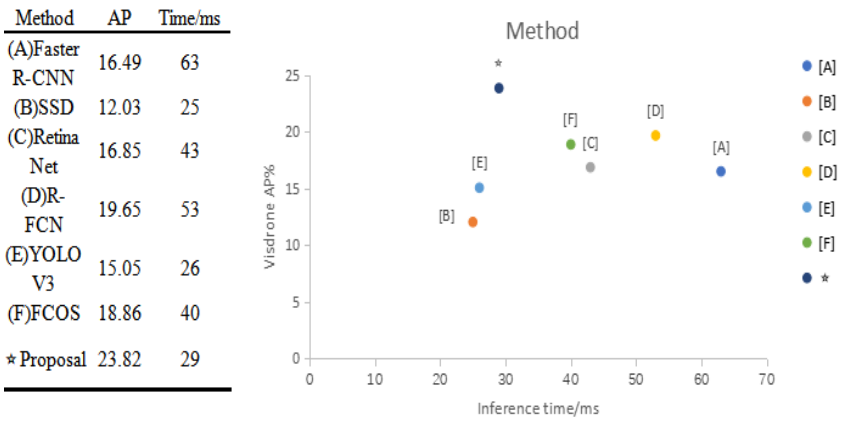
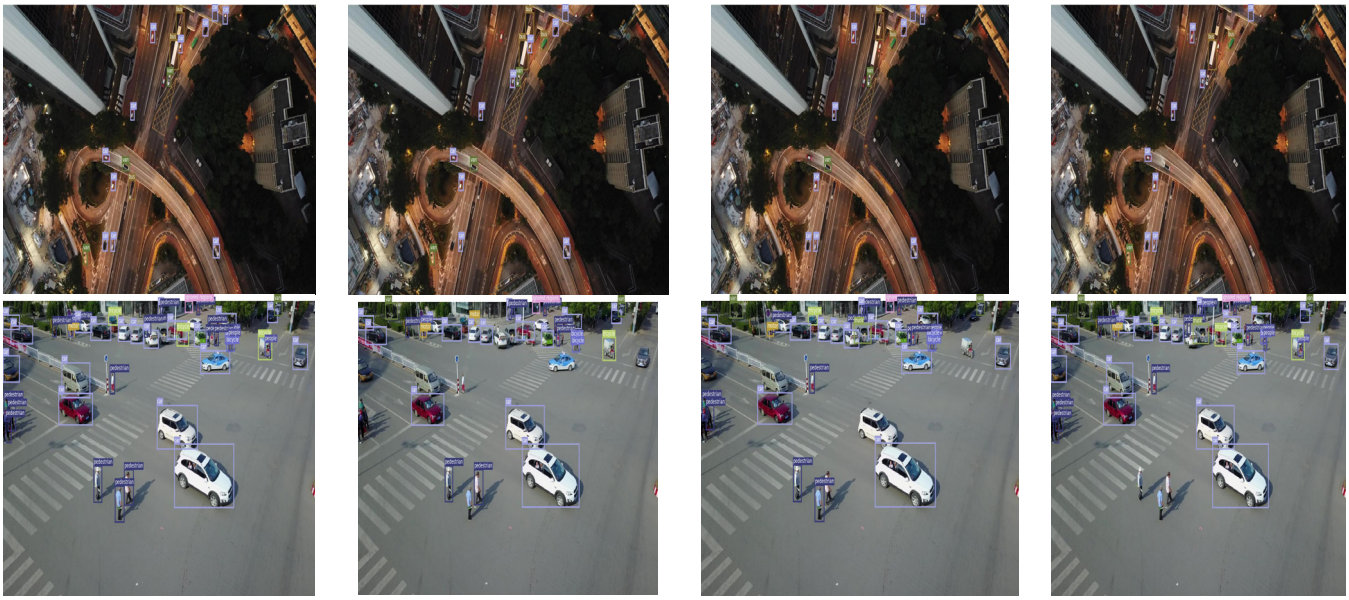


Figure 6. VisDrone test comparison visualization.

It can be seen from Figure 6 that the SSD algorithm has the highest inference speed, and the detection time of a single picture is only 50 ms. Faster R-CNN has the lowest detection efficiency, and the reasoning time for a single image takes 63 ms. Compared with several other algorithms, the reasoning efficiency of the algorithm in this paper is relatively high, and it has good real-time performance.

This paper also compares the three classic target detection algorithms selected on the VisDrone dataset, and the detection effect is shown in Figure 7:



(**a**) Proposal (**b**) RetinaNet (**c**) Faster R-CNN (**d**) YOLOV3

Figure 7. Comparison chart of the detection effect between the algorithm in this paper and some classic algorithms. (**a**) Proposal; (**b**) RetinaNet; (**c**) Faster R-CNN; (**d**) YOLOV3.

This paper extracts target sample instances during the day and night, respectively, and compares the detection results of the four algorithms. It can be seen that RetinaNet, Faster R-CNN, and YOLOV3 have different degrees of missing detection for small targets and targets with similar distances, while the algorithm feature learning in this paper is relatively sufficient. Compared with the other three algorithms, there were no missed or false detections. In summary, the detection accuracy of the proposed algorithm for all kinds of targets is higher than the other three, especially for small targets. This is because AFBS can better combine the superior information of high-level features and low-level features in the feature map through the adaptive fusion of multi-layer features and has stronger identification and localization ability for small targets and multi-scale targets. In the case of low illumination at night, the other three algorithms also have some missing detections. The algorithm in this paper weakens the background noise interference and strengthens the multi-scale features of interest in the network, showing strong anti-interference ability in the face of complex background information and effectively improves the missed alarm situation. In general, it has stronger recognition ability for small-scale, complex backgrounds and large scale transformation UAV image targets when processing UAV image target detection tasks, and it effectively avoids false alarms and missed alarms.

4. Conclusions

In this paper, we made improvements based on the FCOS algorithm to improve the effect of target detection for UAV aerial images. (1) Improvements were made to the backbone network by embedding the global context module in the backbone network and combining it with the FPN to enhance the algorithm's perception and understanding of the relevance of the environment in which the target is located and to improve the

detection accuracy of small UAV targets in complex backgrounds. (2) An adaptive feature balancing sub-network was designed to effectively balance the dominant information in multi-layer features and reduce the false detection probability of the algorithm for small targets. (3) Finally, CIOU Loss was used to improve the regression loss function to enhance the detection capability of the algorithm for targets with larger scale transformations. The results show that the algorithm in this paper has a better detection effect on different scale targets in different aerial photography scenes. Compared with the baseline algorithm, the algorithm in this paper improves the AP by 4.96%. Compared with other mainstream algorithms, the algorithm in this paper has strong competitiveness and reduces the cases of missing detection and false positives. It is an effective aerial image target detection algorithm. In addition, the proposed algorithm has good real-time performance, which is far better than Faster R-CNN, and the detection speed is comparable to that of YOLOV3.

Author Contributions: L.L. and M.G. conducted the algorithm design; G.L. and Z.W. made a Python implementation of the proposed algorithm and formulated the proposed algorithm. J.D., Q.H. and D.H. contributed to prepare and analyze the experimental data and the results. All authors were involved in modifying the article, the literature review, and the discussion of the results. All authors have read and agreed to the published version of the manuscript.

Funding: This research was Supported by the Jilin Provincial Science and Technology Department Development Project (20210203181SF).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors are grateful to the reviewers for enhancing the clarity and completeness of this article.

Conflicts of Interest: The authors have no conflicts of interest to declare that are relevant to the content of this article.

References

1. Lowe, D.G. Distinctive image features from scale invariant keypoint. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
2. Viola, P.; Jones, M.J. Robust real-time face detection. *Int. J. Comput. Vis.* **2004**, *57*, 137–154. [CrossRef]
3. Felzenszwalb, P.; McAllester, D.; Ramanan, D. A discriminatively trained, multiscale, deformable part mode. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; IEEE: Anchorage, AK, USA, 2008; pp. 1–8.
4. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014.
5. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transcations Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
6. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; IEEE Press: Venice, Italy, 2017; pp. 2980–2988.
7. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
8. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016.
9. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
10. Liu, Y.; Yang, F.; Hu, P. Small-Object Detection in UAV-Captured Images Multi-Branch Parallel Feature Pyramid Networks. *IEEE Access* **2020**, *8*, 145710–145750. [CrossRef]
11. Liang, X.; Zhang, J.; Zhuo, L.; Li, Y.; Tian, Q. Small object detection in unmanned aerial vehicle images using feature fusion and scaling-based single shot detector with spatial context analysis. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 1758–1770. [CrossRef]
12. 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]
13. 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.

14. Zhou, X.; Koltun, V.; Krähenbühl, P. Objects as points. *arXiv* **2019**, arXiv:1904.07850.
15. Zhang, Q.; Zhang, H.; Lu, X.; Han, X. Anchor-Free Small Object Detection Algorithm Based on Multi-scale Feature Fusion. In Proceedings of the 2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), Chengdu, China, 30 June 2022; pp. 370–374. [CrossRef]
16. Liu, S.; Qu, J.; Wu, R. HollowBox: An anchor-free UAV detection method. *LET Image Process* **2022**, *16*, 2922–2936. [CrossRef]
17. Hou, X.; Jin, G.; Tan, L. SAR Ship Target Detection Algorithm Based on Anchor—Free Frame Detection Network FCOS. In *National Security Geophysics Series (16) Big Data and Geophysics*; Xi'an Map Press: Xi'an, China, 2020; pp. 162–166.
18. Mao, Y.; Li, X.; Li, Z.; Li, M.; Chen, S. An Anchor-free SAR ship detector with only 1.17 M parameters. In Proceedings of the 2020 International Conference on Aviation Safety and Information Technology, Weihai, China, 14–16 October 2020; pp. 182–186.
19. Tian, Z.; Shen, C.; Chen, H.; He, T. FCOS: Fully convolutional one-stage object detection. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9627–9636.
20. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
21. Liu, S.; Huang, D.; Wang, Y. Learning spatial fusion for single-shot object detection. *arXiv* **2019**, arXiv:1911.09516.
22. Pang, J.; Chen, K.; Shi, J.; Feng, H.; Ouyang, W.; Lin, D. Libra R-CNN: Towards balanced learning for object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 821–830.
23. 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, Seoul, Republic of Korea, 27–28 October 2019; pp. 1971–1980.
24. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
25. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU Loss: Faster and better learning for bounding box regression. In Proceedings of the AAAI Conference on Artificial Intelligence, Hilton, NY, USA, 7–12 February 2020; Volume 34, pp. 12993–13000.
26. Zhu, P.; Wen, L.; Du, D.; Bian, X.; Hu, Q.; Ling, H. Vision meets drones: Past, present and future. *arXiv* **2020**, arXiv:2001.06303.