remote sensing

MDPI

Article

GCL-YOLO: A GhostConv-Based Lightweight YOLO Network for UAV Small Object Detection

Jinshan Cao [1], Wenshu Bao [1,*], Haixing Shang [2], Ming Yuan [1] and Qian Cheng [1]

[1] School of Computer Science, Hubei University of Technology, Wuhan 430068, China; caojs@hbut.edu.cn (J.C.); 2110300925@hbut.edu.cn (M.Y.); 2110300928@hbut.edu.cn (Q.C.)
[2] Northwest Engineering Corporation Limited, Power China Group, Xi'an 710064, China; shang_hx@nwh.cn
* Correspondence: 102111128@hbut.edu.cn

Abstract: Precise object detection for unmanned aerial vehicle (UAV) images is a prerequisite for many UAV image applications. Compared with natural scene images, UAV images often have many small objects with few image pixels. These small objects are often obscured, densely distributed, or in complex scenes, which causes great interference to object detection. Aiming to solve this problem, a GhostConv-based lightweight YOLO network (GCL-YOLO) is proposed. In the proposed network, a GhostConv-based backbone network with a few parameters was firstly built. Then, a new prediction head for UAV small objects was designed, and the original prediction head for large natural scene objects was removed. Finally, the focal-efficient intersection over union (Focal-EIOU) loss was used as the localization loss. The experimental results of the VisDrone-DET2021 dataset and the UAVDT dataset showed that, compared with the YOLOv5-S network, the mean average precision at IOU = 0.5 achieved by the proposed GCL-YOLO-S network was improved by 6.9% and 1.8%, respectively, while the parameter amount and the calculation amount were reduced by 76.7% and 32.3%, respectively. Compared with some excellent lightweight networks, the proposed network achieved the highest and second-highest detection accuracy on the two datasets with the smallest parameter amount and a medium calculation amount, respectively.

Keywords: unmanned aerial vehicle (UAV); small object detection; lightweight network; efficient network; YOLO; GhostConv

Citation: Cao, J.; Bao, W.; Shang, H.; Yuan, M.; Cheng, Q. GCL-YOLO: A GhostConv-Based Lightweight YOLO Network for UAV Small Object Detection. *Remote Sens.* **2023**, *15*, 4932. https://doi.org/10.3390/rs15204932

Academic Editors: Pedro Melo-Pinto and Sander Oude Elberink

Received: 19 August 2023
Revised: 30 September 2023
Accepted: 9 October 2023
Published: 12 October 2023

1. Introduction

Unmanned aerial vehicles (UAVs) equipped with optical remote sensing cameras have the characteristics of simple operation, low cost, and few restrictions. They can easily collect low-altitude remote sensing images, and they have been widely used in many fields, such as smart cities [1], post-disaster rescue [2], agricultural monitoring [3], and traffic management [4]. At present, with the rapid development of UAVs and optical cameras, object details captured in UAV images are becoming more and more abundant, which places higher requirements on the understanding and interpretation of UAV images. As a key technology, the object detection of UAV images is a prerequisite for many subsequent vision tasks, such as object tracking and object positioning. Therefore, there exists an important research significance and application value in studying object detection from UAV images.

At present, most UAV object detection methods are migrated from computer vision, such as the famous you only look once (YOLO) network. The version of the YOLO network has been updated from 1.0 to 8.0, and many impressive results have been obtained using the YOLO network. However, the YOLO network was mainly proposed and used for object detection in natural scene images rather than UAV images. For natural scene images, such as the MS COCO [5] and Pascal VOC datasets [6], the scene contents are generally simple; thus, the objects can be easily detected. Compared with natural scene images, UAV images,

such as the VisDrone-DET2021 [7] dataset and the UAVDT [8] dataset, often have more complex scene contents, and object detection becomes more difficult. It is not appropriate to directly apply the YOLO network to detect objects in UAV images.

Generally, UAV platforms often have limited real-time computing and storage resources. Therefore, a lightweight small object detection network that can achieve a good detection accuracy for UAV images is often pursued in actual applications. Additionally, the application of the YOLO network to UAV object detection faces the following problems: First, as is shown in Figure 1a, due to variations in UAV flight altitudes and attitudes, objects may appear at different viewpoints and scales, resulting in many small objects and different features being displayed. Second, as is shown in Figure 1b, differences in imaging angles and image coverage areas, as well as the presence of obstacles, increase the complexity of UAV image scenes. Third, UAVs may be affected by wind, vibrations, exposure, and other factors during flight, resulting in blurred or distorted images, as shown in Figure 1c. These problems make UAV object detection very difficult. In practice, the difficulty in detecting UAV small objects comes mainly from the small number of pixels occupied by these objects. As a result, small object detection frame bias has a much larger impact on the detection results than large object detection frame bias.

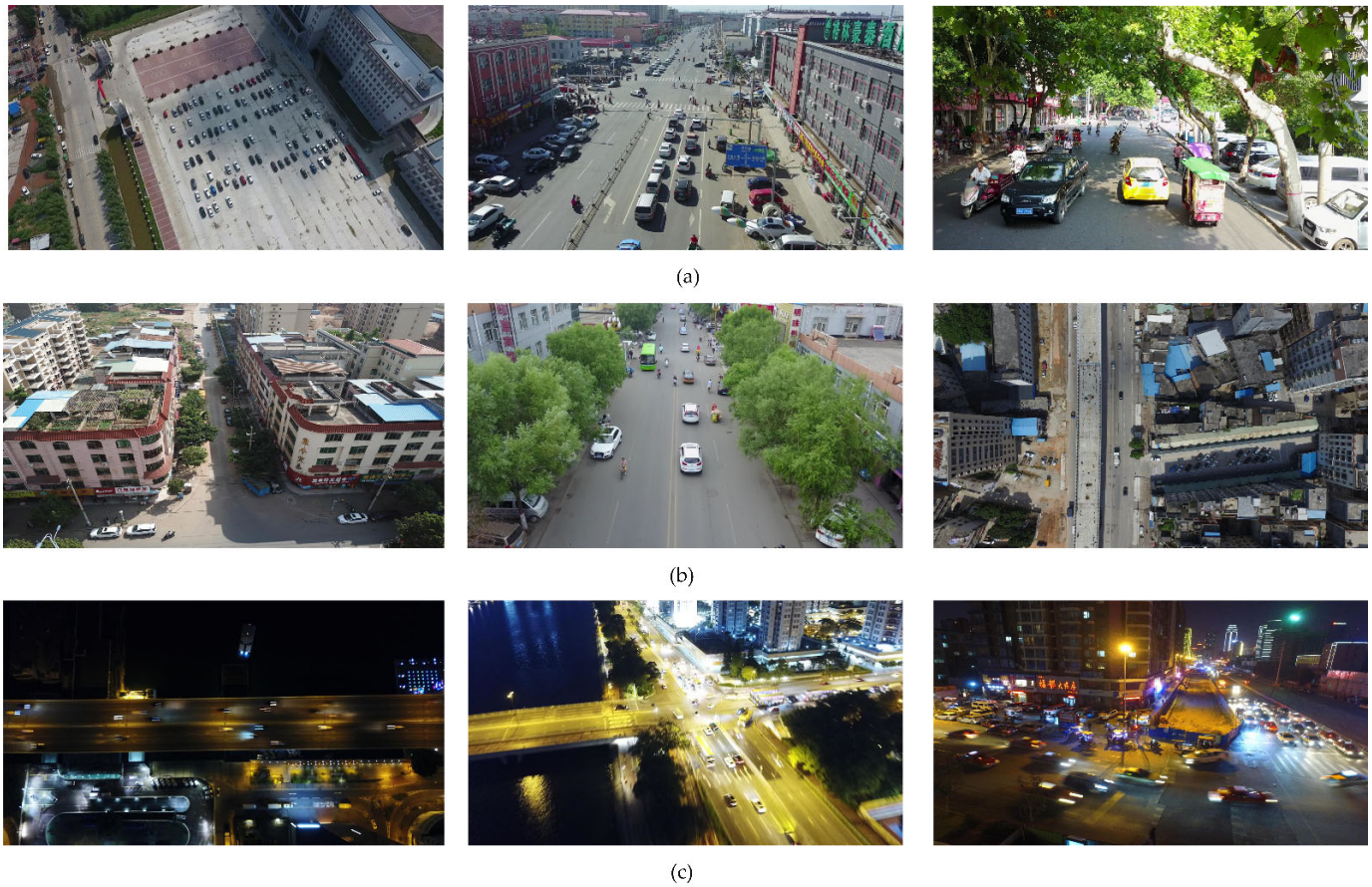


Figure 1. Examples for (**a**) object scale variations, (**b**) complex scenes, and (**c**) blurred images in UAV images.

Aiming at solving the above problems, a GhostConv-based lightweight YOLO (GCL-YOLO) network for small object detection in UAV images is proposed in this study. The proposed GCL-YOLO network is derived from the YOLOv5 network, and the structure of each part in the YOLOv5 network is analyzed and redesigned. As in the case of the YOLOv5 network, the proposed GCL-YOLO network introduced a width coefficient and a depth coefficient, and it could also be categorized into four networks (i.e., GCL-YOLO-N,

GCL-YOLO-S, GCL-YOLO-M, and GCL-YOLO-L) to meet the requirements of UAV object detection in different applications. The experimental results on the VisDrone-DET2021 dataset show that, compared with other excellent lightweight networks, the proposed GCL-YOLO-S network achieved the highest detection accuracy with the smallest parameter amount and a medium calculation amount, as shown in Figure 2. Additionally, the proposed network achieved the second-highest detection accuracy on the UAVDT dataset.

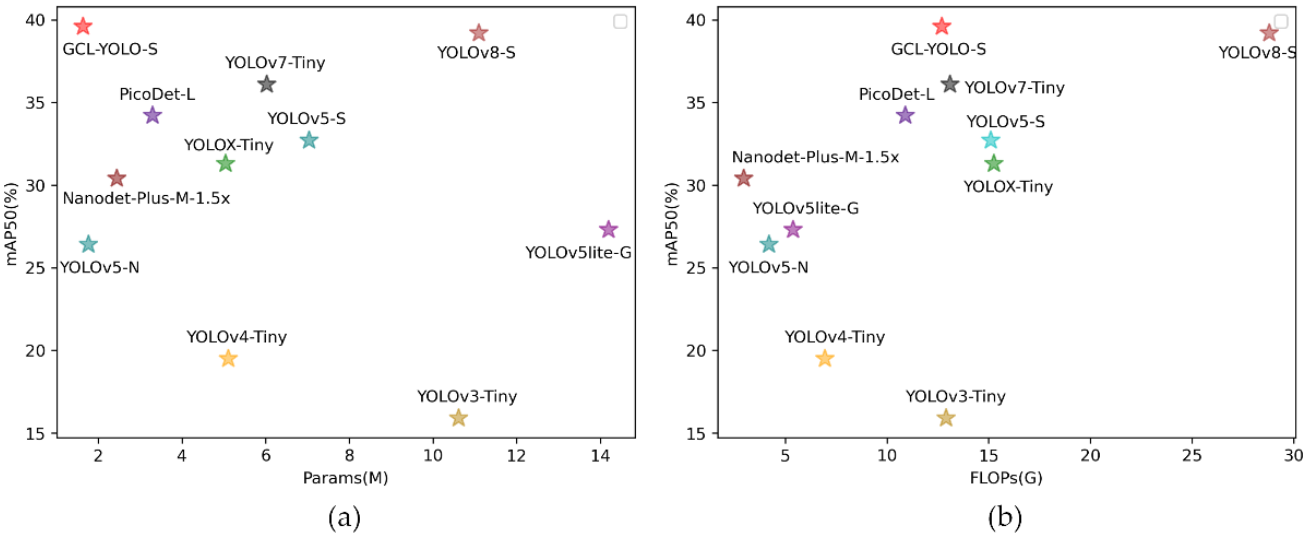


Figure 2. (**a**) Object detection accuracy versus parameter amount and (**b**) object detection accuracy versus calculation amount in different networks.

The main contributions of this paper can be summarized as follows:

1. A GhostConv-based backbone network is proposed. Compared with the CSP-Darknet53 used by the YOLOv5 network, the proposed network can reduce the amount of network parameters by half without losing the object detection accuracy.
2. The feature fusion network in the YOLOv5 network is reconstructed. A shallow feature fusion layer and a new object prediction head are added to precisely predict the location of high-density small objects. Meanwhile, the prediction head for large objects is removed to further reduce the amount of network parameters.
3. A lightweight network for UAV small object detection is proposed. Compared with some lightweight networks, the proposed GCL-YOLO-S network can achieve a better object detection accuracy with the smallest parameter amount and a medium calculation amount.

The remainder of this paper is organized as follows: Section 2 reviews related work on the YOLO network, detection methods for UAV small objects, and lightweight neural networks. Section 3 details the proposed GCL-YOLO network. Section 4 describes the use of the VisDrone-DET2021 dataset to analyze the feasibility and effectiveness of the proposed GCL-YOLO network. Section 5 describes the conclusions.

2. Related Work

At present, deep-learning-based detection methods have become the most popular object detection methods for optical images. With the rapid development of deep learning, the object detection methods based on convolutional neural networks (CNNs) have achieved many impressive results. As a representative one-stage object detection method, the YOLO network has attracted more and more attention. In order to apply the YOLO network for UAV small object detection successfully, many researchers have conducted a considerable amount of research. Related studies are summarized in the following.

2.1. YOLO Networks

In its early days, object detection research was mainly focused on two-stage networks. Such networks searched regions of interest through a region proposal network (RPN); then, an object classification and an anchor box prediction were performed. Although such two-stage networks could achieve good object detection results, they were often bloated and inefficient due to the use of RPNs.

In 2016, Joseph et al. [9] proposed a full convolutional neural network, YOLOv1. The YOLOv1 network transformed the object detection task into a regression problem, and it could achieve a high detection accuracy with an ultra-fast detection speed. The structure of the YOLOv1 network consisted of three parts: backbone network, neck, and prediction head. Compared with the conventional two-stage networks, the YOLOv1 network was an end-to-end and one-stage network. Since the YOLOv1 network was proposed, the one-stage object detection methods have attracted more and more attention. Many researchers tried to optimize the YOLOv1 network and proposed several versions of the YOLO network [10–14]. Generally, these proposed YOLO networks achieved great breakthroughs in terms of both detection accuracy and detection speed.

At present, the YOLO network has become a very popular object detection network in both industry and academia because of its real-time performance and high detection accuracy. Benefitting from the researchers' constant attention, the version of the YOLO network was updated from 1.0 to 8.0. Considering the detection accuracy, the detection speed, and the network size comprehensively, the YOLOv5 network was more representative and more popular compared with other versions. In the YOLOv5 network, CSPDarknet53 was taken as the backbone network and a path aggregation network (PAN) was taken as the feature fusion module. Optimization modules (e.g., SPPF) and data augmentation modules (e.g., mosaic augmentation) were used, and the detection performance of the YOLOv5 network was thereby greatly improved. Meanwhile, a scale factor was designed to adjust the YOLOv5 network into several networks of different network sizes; thus, that the YOLOv5 network could be better used in different applications.

2.2. UAV Small Object Detection

In the field of UAV small object detection, many researchers introduced deep-learning-based detection methods to improve the detection effect of UAV small objects.

UAV images often have the problem of an imbalanced distribution of small objects, such as densely presented small objects (e.g., crowds). There may be a large number of small object instances in some images, while there may be only a small number or even none in other images, which is the reason for the reduction in the diversity of small object locations. To address these problems, Kisantal et al. [15] proposed a data augmentation method for small object detection. In the proposed method, small objects were copied and pasted several times to increase the diversity of small object locations. The imbalanced distribution of small objects could therefore be eliminated, and an effective accuracy improvement in small object detection was achieved. However, such data augmentation required segmentation annotations and was not suitable for all datasets. Bosquet et al. [16] proposed another data augmentation method called the downsampling generative adversarial network (DS-GAN). The proposed DS-GAN combined a GAN-based object generator with the techniques of object segmentation, image inpainting, and image blending to produce high-quality synthetic data. Due to the introduction of a vast number of computations and parameters, the inference speed of the DS-GAN was seriously decreased. Aiming to solve the problem of the features of small objects being easy to lose and difficult to be expressed, Ma et al. [17] proposed an end-to-end and scale-aware feature split–merge–enhancement network (SME-Net). The SME-Net eliminated salient information and suppressed background noise to highlight the features of small objects in low-level feature maps.

In order to effectively improve the detection effect of small objects in UAV images, Zhu et al. [18] integrated the transformer prediction heads (TPHs) and the convolutional block attention model into the YOLOv5 network and proposed a TPH-YOLOv5 network.

Based on the YOLOv3 network, Sun et al. [19] proposed a new real-time small object detection (RSOD) method in UAV-based traffic monitoring. Li et al. [20] investigated the image cropping strategy and proposed a density-map-guided object detection network (DMNet). The DMNet fully utilized the spatial and contextual information between objects to improve the detection performance. Based on the YOLOv4 network, Liu et al. [21] used the MobileNet as the backbone network and proposed an improved lightweight MYOLO-lite network. In the proposed network, the number of network parameters and the calculation complexity were effectively reduced in order to meet the speed requirements of UAV object detection in practical applications. Zhang et al. [22] developed a UAV object detection network called SlimYOLOv3 by pruning the YOLOv3 network. The detection speed of the SlimYOLOv3 network was twice as fast as that of the YOLOv3 network.

Generally, the above detection methods could achieve good results in terms of the detection accuracy, the detection speed, or the network size. However, in order to better meet the application requirements of UAV object detection, we should further improve the detection accuracy, increase the detection speed, and optimize the network size.

2.3. Lightweight Neural Networks

To achieve better feature extraction results, the existing neural networks often require a large number of parameters and calculations. Accordingly, the structures of these large networks are very complicated. Due to the limitation of computational resources, it is difficult to use such large and complicated networks on UAV platforms. Therefore, a series of previous studies have been devoted to designing lightweight networks to achieve a better trade-off between speed and accuracy.

MobileNet, ShuffleNet, and GhostNet are three series of representative lightweight neural networks. In the MobileNet series [23–25], a deep separable convolution (DSConv) module was proposed to decompose the standard convolution into a depthwise convolution and a pointwise convolution. Consequently, the number of network parameters was significantly reduced. In the ShuffleNet series [26,27], a DSConv -based group convolution module and channel shuffling module were added, and four principles for designing lightweight networks were proposed. Additionally, a channel split structure was introduced, in which the add operation was replaced with a join operation. The number of network parameters was then reduced. Aiming to improve upon the large number of redundant features, a GhostConv module was proposed in the GhostNet series [28,29]. The GhostConv module only performed depth-separable convolution on half of the channels and then stitched original feature channels to improve the utilization of redundant features. An attractive advantage of the GhostNet was that an excellent feature extraction result could be achieved, while the number of parameters could be greatly reduced. Benefiting from such an advantage, GhostNet attracted more and more attention.

3. Proposed Network

3.1. Overview of the Proposed Network

The GCL-YOLO network is a lightweight and high-performance detection network that was built based on the GhostConv. The network structure is shown in Figure 3. As in the case of the YOLOv5 network, the GCL-YOLO network can also be differentiated into GCL-YOLO-N, GCL-YOLO-S, GCL-YOLO-M, and GCL-YOLO-L networks by changing the channel width coefficient and the depth coefficient, as shown in Table 1; thus, the detection requirements of different UAV applications can be met.

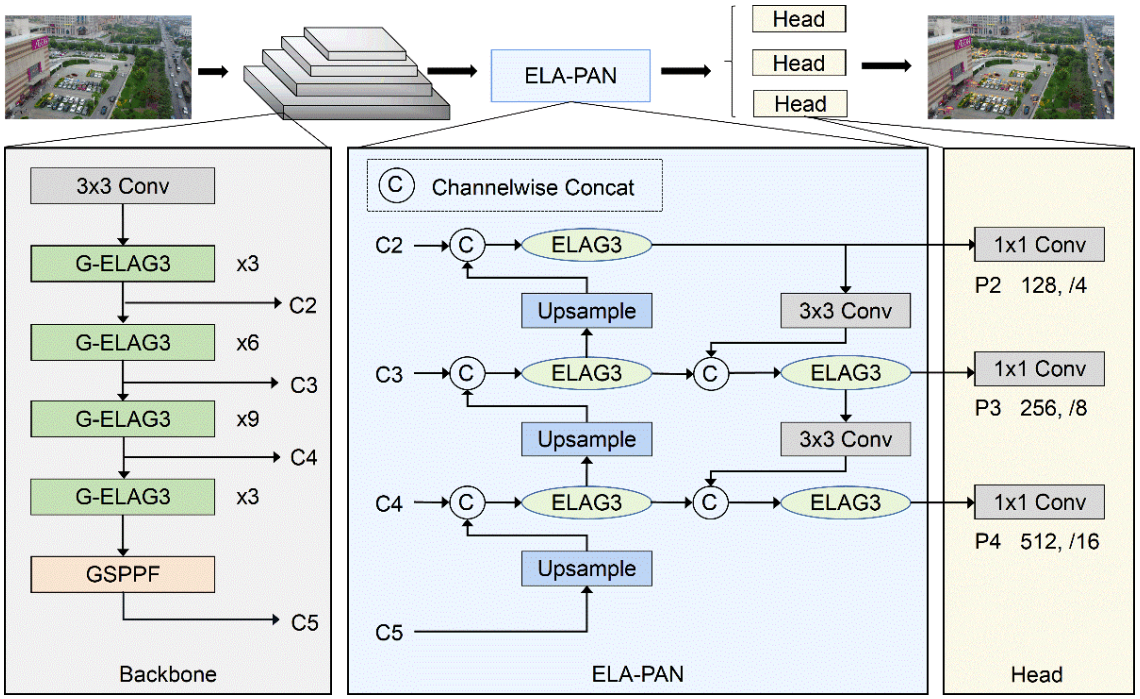


Figure 3. Structure of the GCL-YOLO network. The G-ELAG3 module represents a combined module of the G-Bneck2 module and the ELAG3 module. The backbone network outputs feature maps C2, C3, C4, and C5, whose number of channels are 128, 256, 512, and 512, respectively. The neck is an efficient layer aggregation and path aggregation network (ELA-PAN), whose inputs are four feature maps and outputs are three feature maps. The number of channels of three feature maps is 128, 256, and 512, respectively, and the downsampling rate is 4, 8, and 16, respectively. Note that the channel number of C5 is adjusted to 512 and is consistent with that of C4.

Table 1. Specifications of the different GCL-YOLO networks.

Network	α	β	Number of Feature Channels
GCL-YOLO-N	0.33	0.25	(32, 64, 128)
GCL-YOLO-S	0.33	0.50	(64, 128, 256)
GCL-YOLO-M	0.67	0.75	(96, 192, 384)
GCL-YOLO-L	1.00	1.00	(128, 256, 512)

Note: The network size is adjusted according to width multiplier α and depth multiplier β. Number of feature channels denotes the number of feature channels of prediction heads P2, P3, and P4.

In terms of the GCL-YOLO network structure, large changes were made to the backbone network, the neck, and the prediction head of the YOLOv5 network in order to improve the detection results of UAV small objects. Firstly, a GhostConv-based backbone network with fewer parameters was built. Secondly, a lightweight neck was designed. Finally, the Focal-EIOU loss was introduced into the prediction head.

3.2. GhostConv-Based Backbone Network

In the YOLOv5 network, the CSP-Darknet53 was used as the backbone. Although the cross-stage partial (CSP) module can reduce the number of network parameters to a certain extent, the network size is still too large for the application of UAV object detection.

For the compression and pruning of deep learning networks, many studies have showed that the feature maps generated by many mainstream CNNs often contain considerable redundant features, and some of these features are similar to each other. Based on this situation, GhostNet assumes that redundant features in feature maps are the key to the good detection results achieved by some neural networks; then, a GhostConv module

was proposed, as shown in Figure 4. Compared with conventional convolution modules, the GhostConv module uses half of the feature channels for 5 × 5 DSConv to expand the receptive field at one time. Then, the other half of the feature channels are spliced to reuse the original features. Benefiting from these cheap operations, more redundant feature maps are generated and better detection results can be achieved by GhostNet.

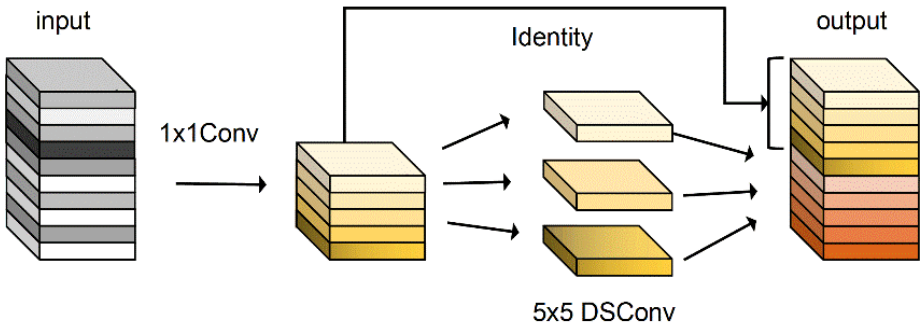


Figure 4. Structure of the GhostConv module.

Inspired by this advantage, the GhostConv module was introduced into the YOLOv5 network to reduce the parameter and calculation amounts in this study. As it is shown in Figure 4, the GhostConv module uses half of the features for convolution and then connects them with the original features, which aims to create redundant features and reduce the parameter and calculation amounts. However, the main purpose of the backbone network is to extract richer features. Directly replacing the convolution modules in the YOLOv5 network with the GhostConv modules inevitably leads to a performance degradation, due to only half of the features being used in the GhostConv modules. In order to solve this problem, a ghost bottleneck with stride = 1 (G-Bneck1) and a ghost bottleneck with stride = 2 (G-Bneck2) were designed. Then, an efficient layer aggregation ghost bottleneck with a 3 convolutions (ELAG3) module was proposed to replace the CSP-Bottleneck module in the YOLOv5 network, in order to reduce the parameter and calculation amounts with little performance loss. Additionally, a GhostConv-based fast spatial pyramid pooling (GSPPF) module was proposed to reduce the number of network parameters. With the help of the redesigned ghost bottleneck, the ELAG3 module, and the GSPPF module, a GhostConv-based backbone network with fewer parameters was built.

Ghost Bottleneck. In order to achieve an expansion and squeeze effect in the GCL-YOLO network, two different ghost bottleneck structures were designed according to the locations where the ghost bottleneck is used. The G-Bneck1 was used as a feature extraction layer, as shown in Figure 5a; the bottleneck structure was used and the number of intermediate channels was halved to reduce the number of parameters, just as in the case of the bottleneck structure in the YOLOv5 network. In Figure 5a, the first GhostConv was used to compress the number of channels, reduce the parameter amount, and reduce the influence of high-frequency noises. The second GhostConv was used to restore the number of channels. The original features were then added through a residual connection operation to supplement the information loss caused by the channel compression. Generally, such a bottleneck structure is conducive to reducing the parameter amount with little information loss. The G-Bneck2 was used as a downsampling layer, as shown in Figure 5b; the fusiform structure was used to double the number of intermediate channels. Therefore, the DSConv acting as the sampling module can have more channels and more powerful feature extraction capabilities. Additionally, an efficient channel attention (ECA) [30] module was inserted into the ghost bottleneck module in order to suppress noisy information.

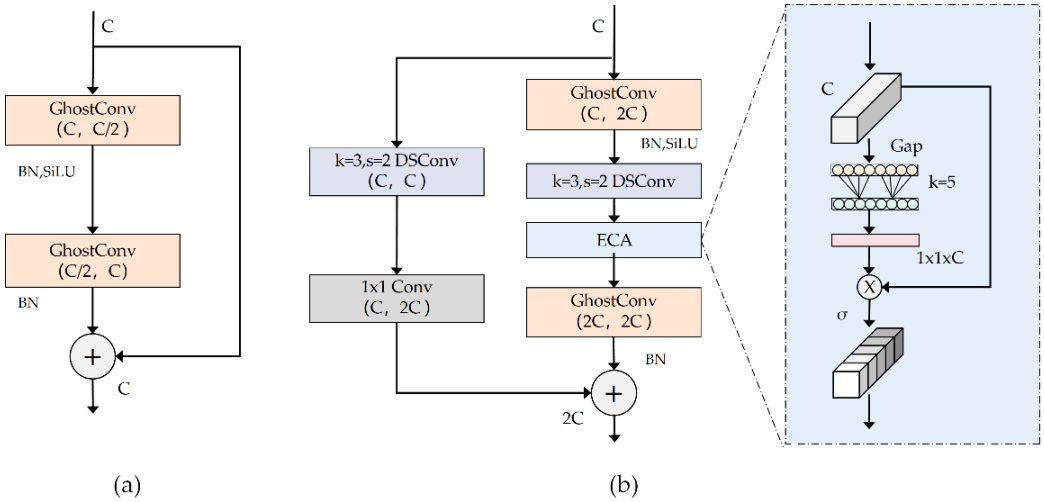


Figure 5. Structure of the ghost bottleneck. (**a**) The G-Bneck1 with stride = 1 was used as a feature extraction layer and (**b**) the G-Bneck2 with stride = 2 was used as a downsampling layer. C denotes the number of channels; SiLU denotes the activation function; BN denotes the normalization; k denotes the convolution kernel size; Gap is the global maximum pooling; and σ is the sigmoid.

ELAG3. In the YOLOv5 network, the CSP [31] module used can reduce the repeated gradient calculations, but at the same time, it misses the transmission of intermediate gradient information. In our feature extraction module, the proposed ELAG3 module was used to replace the CSP-Bottleneck. As it is shown in Figure 6a, the ELAG3 module was proposed based on the ELAN [14] and the redesigned G-Bneck1. In the backbone network, the feature extraction module usually has multiple bottlenecks. The ELAN can splice and aggregate the intermediate information, and the G-Bneck1 can further reduce the number of parameters. Compared with the CSP module, the ELAN can achieve better detection results, although a few parameters and calculations were added. Therefore, all of the CSP-Bottleneck modules were replaced with an ELAG3 module in the GCL-YOLO network.

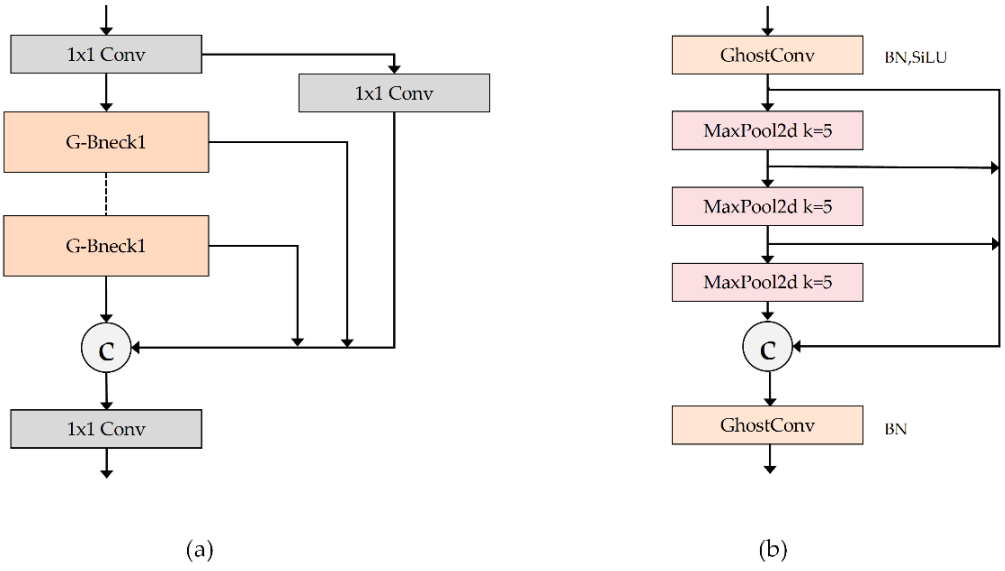


Figure 6. Structures of (**a**) the ELAG3 and (**b**) the GSPPF.

GSPPF. At the end of the backbone network, the SPPF module has vast calculations due to the considerable number of channels. Therefore, two pointwise convolutions were used to reduce the calculation amount and the parameter amount in the YOLOv5 network.

In the proposed GCL-YOLO network, a GSPPF module was introduced, as shown in Figure 6b, that is, two GhostConvs were used to replace the two pointwise convolutions to further compress the network. Meanwhile, the activation function in the second GhostConv was not used to maintain the linear relationship of the feature information.

3.3. Lightweight Neck

In the process of object detection, semantic information is needed for object classification, and location information is needed for object localization. With the continuous sampling in the backbone network, rich semantic information can be captured, but a vast quantity of location information is lost at the same time, especially for small objects. For this reason, researchers proposed a multi-scale feature fusion method, which fully integrates high-level semantic information and low-level location information in the feature extraction layer to improve the detection effect.

In the YOLOv5 network, the PAN module is used to fuse three feature maps of different resolutions in the deep part of the backbone network. Three prediction heads (i.e., P5 for large objects, P4 for medium objects, and P3 for small objects) were designed, and then three feature maps were used to detect objects of different sizes. However, the YOLOv5 network is generally used to detect objects in natural scene images. In comparison, objects in UAV images are much smaller. When the size of an object in UAV images is only several or a dozen pixels, it is very difficult to capture the feature information of such a small object in feature maps. As a result, it will be unable to detect the object.

For UAV small object detection, the location information of small objects in low-level feature maps is theoretically required. Figure 7 shows the feature maps and the heat maps of a UAV example image obtained with different sampling rates. It can be seen from Figure 7c,f that, when the sampling rate reaches 32, the receptive field of the feature map is too large. It is very difficult to see the texture and outline of small objects. In contrast, when the sampling rate is 4 in Figure 7b,e, the receptive field of the feature map is smaller, and more detailed location information of small objects is captured. According to such a characteristic of small objects in UAV images, an ELA-PAN was designed in the proposed GCL-YOLO network, as shown in Figure 3. Compared with the PAN module, low-level features were fused and a new prediction head P2 for UAV small objects was added in the ELA-PAN. By fusing low-level features, the multi-level features of small objects in UAV images can be learned better, and the detection effect of small objects can be improved.

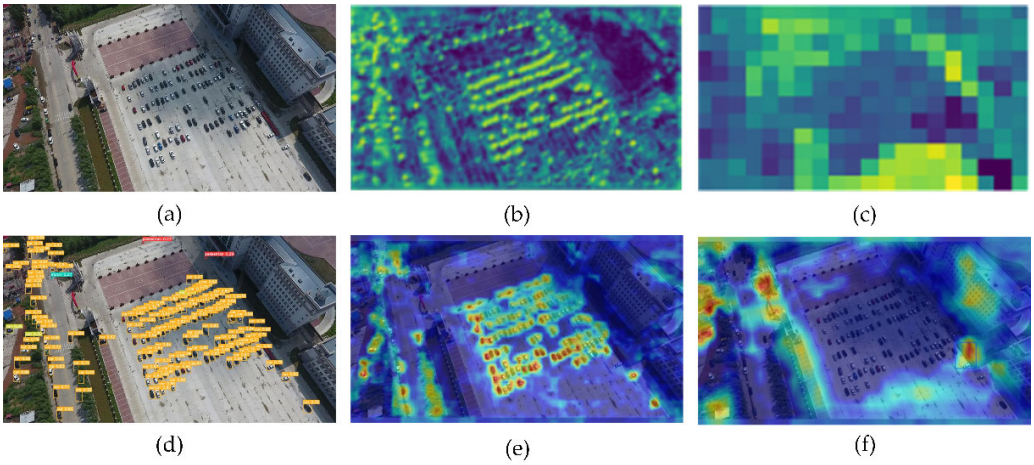


Figure 7. Feature maps and heat maps of a UAV example image obtained with different sampling rates: (**a**) original image, (**b**) fine-grained feature map obtained with the sampling rate of 4, (**c**) fine-grained feature map obtained with the sampling rate of 32, (**d**) object detection result, (**e**) gradient-weighted class activation mapping (Grad-CAM) heat map obtained with the sampling rate of 4, and (**f**) Grad-CAM heat map obtained with the sampling rate of 32.

In addition, high-level feature fusion in the PAN module needs a large number of parameters and calculations. As is shown in Table 2, approximately 75% of the parameters were used by the prediction head P5. In other words, high-level feature fusion was mainly used to improve the detection effect of large objects. There are often few large objects in UAV images; thus, it is unnecessary to pay such a high price to improve the detection effect of large objects. In order to balance the detection accuracy and speed, the prediction head P5 for large objects was abandoned. Meanwhile, three convolutions used to adjust the number of channels in the upsampling process were deleted in the ELA-PAN. Therefore, the number of channels at the end of the backbone network can be reduced and the number of network parameters can be significantly reduced.

Table 2. Parameter comparisons of different prediction heads.

Prediction Head	Downsampling Rate	Feature Map Size	Number of Channels	Parameters (MB)
P2, P3, P4, and P5	4×/8×/16×/32×	160/80/40/20	128/256/512/1024	13.26
P5	32×	20	1024	9.97
P2	4×	160	128	0.17

Note: The size of input images is 640 × 640 pixels.

3.4. Focal-EIOU Loss

The loss function of the YOLOv5 network consists of a classification loss (L_{cls}), an object confidence loss (L_{obj}), and a localization loss (L_{loc}). The localization loss often uses the intersection over union (IOU) to calculate the similarity between a predicted box and the corresponding ground truth, as shown in Figure 8. In the latest version of the YOLOv5 network (i.e., version 7.0), a complete IOU (CIOU) is used as its bounding box loss [32]. Unlike the IOU, the CIOU considers not only the overlap between the two bounding boxes but also their sizes and location relationship. The expression of the CIOU loss is as follows:

$$L_{CIOU} = 1 - IOU + \frac{\rho^2(b,\ b^{gt})}{c^2} + \alpha\upsilon \tag{1}$$

where b and b^{gt} are the center points of the predicted box and the ground truth, respectively; $\rho(\cdot)$ represents the Euclidean distance; and c represents the diagonal length of the smallest enclosing box covering the two boxes. The IOU, the positive trade-off parameter α, and the consistence parameter υ of the aspect ratio are calculated as follows:

$$IOU = \frac{\left|B \cap B^{gt}\right|}{\left|B \cup B^{gt}\right|} \tag{2}$$

$$\upsilon = \frac{4}{\pi^2}\left(\arctan\frac{\omega^{gt}}{h^{gt}} - \arctan\frac{\omega}{h}\right)^2 \tag{3}$$

$$\alpha = \frac{\upsilon}{1 - IOU + \upsilon} \tag{4}$$

where B and B^{gt} are the predicted box and the ground truth, respectively; ω^{gt} and h^{gt} are the width and height of the ground truth, respectively; and ω and h are the width and height of the predicted box, respectively.

We can see from Equation (1) that the aspect ratio of a predicted box was used as an influencing factor in the CIOU loss. When the aspect ratios of two predicted boxes of different widths and heights are the same and both center points of these two boxes are coincident with the center point of the ground truth, the two CIOU losses may be the same, while these two predicted boxes may be greatly biased. Therefore, using the CIOU loss will inevitably result in uncertainties, especially for UAV small object detection. As the aspect ratios of UAV objects (e.g., the VisDrone-DET2021 dataset in Section 4.1) differ significantly, the effect of the CIOU loss will be unsatisfactory. For this reason, the efficient

IOU (EIOU) [33] loss is commonly used as the localization loss. The EIOU loss uses the real width and height of a predicted box rather than the aspect ratio for regression, which can eliminate the negative influence of the uncertainty of the aspect ratio. Compared with the CIOU loss, the EIOU loss is unfriendly to tiny objects in small objects. For relatively large objects in small objects, using the EIOU loss often can achieve a better detection accuracy. Hence, the overall accuracy difference between the CIOU loss and the EIOU loss is not significant. However, the EIOU loss is more beneficial to network optimization.

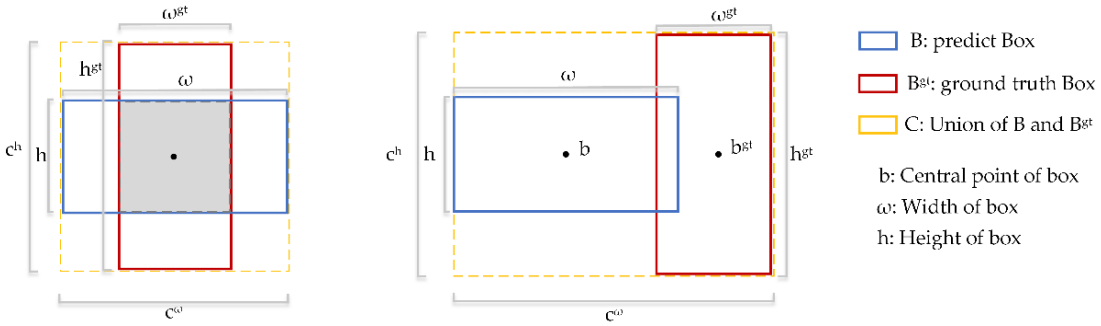


Figure 8. Overlap between the predicted boxes and ground truth boxes.

The EIOU loss is calculated as follows:

$$L_{EIOU} = 1 - IOU + \frac{\rho^2(b, b^{gt})}{c^2} + \frac{\rho^2(\omega, \omega^{gt})}{c_\omega^2} + \frac{\rho^2(h, h^{gt})}{c_h^2} \quad (5)$$

where c_ω and c_h are the width and height of the smallest box covering the two predicted boxes, respectively.

However, the existing UAV object datasets often have a large deviation in the object number of different object categories. The EIOU loss is generally not suitable for such a dataset. Aiming to solve this problem, the Focal-EIOU loss proposed by Zhang et al. [33] was introduced in the proposed GCL-YOLO network. Based on the EIOU loss, the Focal-EIOU loss uses a focal loss to solve the problem of imbalanced object labels, and it can be calculated as follows:

$$L_{Focal-EIOU} = IOU^\gamma \cdot L_{EIOU} \quad (6)$$

where γ is a parameter to control the degree of inhibition of outliers.

4. Experimental Results

In this section, the implementation details and experimental results of the proposed GCL-YOLO network are presented. The proposed network was compared with other lightweight networks, UAV object detection networks, and YOLO networks using the VisDrone-DET2021 dataset and the UAVDT dataset. Additionally, ablation experiments were performed to verify the effectiveness of the proposed network.

4.1. Experimental Datasets

In this study, the VisDrone-DET2021 dataset and the UAVDT dataset were used as the experimental datasets. These datasets are two of the most representative datasets for UAV small object detection.

The VisDrone-DET2021 dataset has 6471 training images, 548 validation images, and 3190 test images. The dataset consists of ten object categories: pedestrian, people, bicycle, car, van, truck, tricycle, awning tricycle, bus, and motor. The label number of different object categories differs significantly, as shown in Figure 9a, that is, the number of object labels are imbalanced. Figure 9b shows the number of object labels of different sizes. Generally, objects smaller than 32 × 32 pixels are considered as small objects, while objects

larger than 96 × 96 pixels are considered as large objects. Objects whose size ranges from 32 × 32 to 96 × 96 pixels are considered as medium objects. It can be seen from Figure 9b that approximately 60% of the objects in this dataset were small objects, while large objects accounted for only 5.5%.

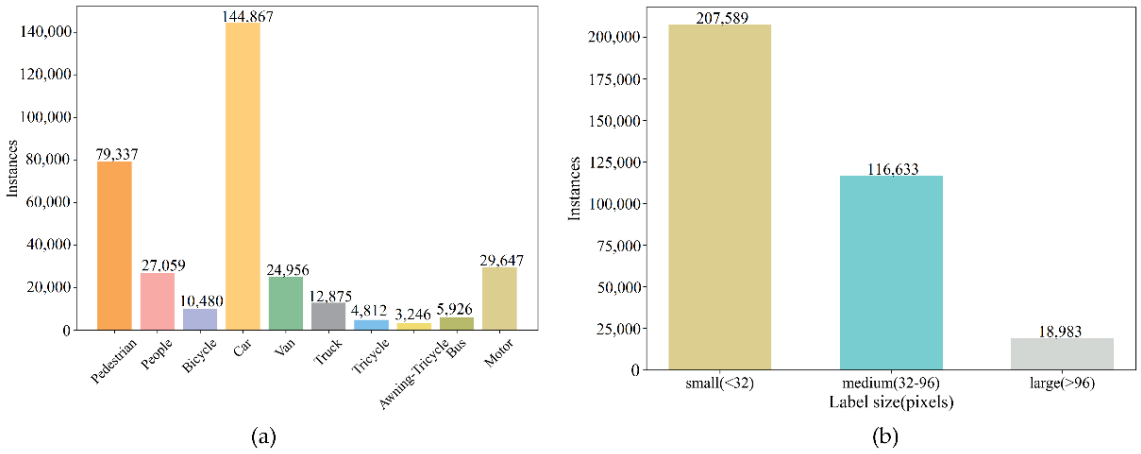


Figure 9. (**a**) The label number of each object category and (**b**) the number of object labels of different sizes in the VisDrone-DET2021 dataset.

The UAVDT dataset consists of 50 videos with a total of 40,376 images and has three object categories: car, truck, and bus. Since the UAVDT dataset is taken from videos, image sequences in the same video contain a number of similar contents. This leads to many redundancies and is not conducive to the performance comparison of different networks. Therefore, only 20% of image sequences in each video were extracted to construct a subset. Finally, the number of training, testing, and evaluation images was 4756, 1568, and 1841, respectively.

4.2. Evaluation Metrics

In this study, the mean average precision at IOU = 0.5 (mAP50) and the mean average precision at IOU = 0.5:0.05:0.95 (mAP50-95) were used to evaluate the detection accuracy of all object categories. The indicator of frames per second (FPS) was used to evaluate the detection speed of the networks. The indicator of giga floating-point operations per second (GFLOPs) was used to evaluate the calculated amount of the networks, and the parameter indicator was used to evaluate the parameter amount of the networks.

4.3. Implementation Details

The experiments in this study were implemented on an Ubuntu 20.04 system. The experimental environment was Python 3.7, PyTorch 1.8, and CUDA 11.1. All the networks were trained on two NVIDIA GeForce RTX3060 GPUs using the dual-card-distributed hybrid training method, and the epochs of all the networks were set to 300. In the training process, the stochastic gradient descent (SGD) optimizer was used. The initial learning rate was set to 0.01, and the final learning rate was set to twelve percent of the initial learning rate. The sizes of all the input images were adjusted to 640 × 640 pixels. For a fair comparison, all the networks in the experiments were trained from scratch without using the official pre-training weights.

4.4. Comparisons with the YOLOv5 Network

In this section, a comprehensive comparison is made between the GCL-YOLO network proposed in this paper and the YOLOv5 network on VisDrone-DET2021 dataset in terms of detection accuracy, detection speed, and network size. The experimental results are shown in Table 3.

Table 3. Comparison between the GCL-YOLO network and the YOLOv5 network.

Network	Params(M)	GFLOPs	FPS	mAP50 (%)
YOLOv5-N	1.77	4.2	**68**	26.4
GCL-YOLO-N	**0.43**	**3.2**	58	**31.7**
YOLOv5-S	7.04	15.8	**60**	32.7
GCL-YOLO-S	**1.64**	**10.7**	53	**39.6**
YOLOv5-M	20.89	48.0	**57**	36.4
GCL-YOLO-M	**4.30**	**25.9**	48	**43.2**
YOLOv5-L	46.15	107.8	**53**	38.7
GCL-YOLO-L	**8.76**	**50.7**	42	**45.7**

It can be seen from Table 3 that, compared with the YOLOv5 network, the proposed GCL-YOLO network achieved a better detection accuracy with fewer parameters and calculations. For example, compared with the YOLOv5-S network, the mAP50 achieved by the GCL-YOLO-S network was improved by 6.9%, while the parameter amount was reduced from 7.04 MB to 1.64 MB. In addition, the calculated amount was reduced from 15.8 GFLOPs to 10.7 GFLOPs, and the detection efficiency was slowed down, but it still maintained a detection speed of 53 FPS. Overall, compared to the improved detection accuracy, the reduced parameter amount, and the reduced calculation amount, the reduced detection speed was acceptable.

4.5. Ablation Experiments

In order to demonstrate the effectiveness of the proposed GCL-YOLO network, we performed ablation experiments based on the most popular YOLOv5-S network. The experiments were conducted using the VisDrone-DET2021 dataset in a single comparison and stack-by-stack fashion. The ablation experiments were designed as follows:

(1) Ablation A1: Each convolution module was replaced by a GhostConv module;
(2) Ablation A2: The G-Bneck2 was used as the downsampling layer;
(3) Ablation A3: Each CSP-Bottleneck module was replaced by an ELAG3 module in the feature extraction layer;
(4) Ablation A4: The GSPPF module combined with a GhostConv was used to reduce the parameter amount;
(5) Ablation A5: The prediction head P2 for UAV small objects was added;
(6) Ablation A6: The prediction head P5 for large natural objects was removed, and the number of channels at the end of the backbone network was reduced;
(7) Ablation A7: The CIOU loss was replaced by the Focal-EIOU loss.

For a fair comparison, the training configurations from ablation A1 to ablation A7 were set to be the same. The experimental results are shown in Table 4.

Table 4. Comparisons between the GCL-YOLO-S network and the YOLOv5-S network.

Network	A1	A2	A3	A4	A5	A6	A7	Params (M)	GFLOPs	FPS	mAP50 (%)	mAP50-95 (%)
YOLOv5-S								7.04	15.8	60	32.7	17.2
GCL-YOLO-S	√							3.70	7.8	64	28.8	15.4
		√						6.24	16.2	59	34.5	18.6
		√	√					4.18	11.1	55	32.9	17.4
		√	√	√				3.87	10.9	54	32.7	17.2
		√	√	√	√			3.97	12.7	50	37.2	20.2
		√	√	√	√	√		1.64	10.7	53	38.3	20.9
		√	√	√	√	√	√	1.64	10.7	53	39.6	21.5

From the results in Table 4, several conclusions were drawn, as follows:

(1) Directly replacing the convolution modules with the GhostConv modules in ablation A1 resulted in a detection accuracy degradation, due to only half of the features being used in the GhostConv modules in order to reduce the parameter and calculation amounts, as described in Section 3.2.
(2) Using the G-Bneck1 as the downsampling layer reduced the parameter amount and improved the detection accuracy. The reason for this was that the GhostConv used only half the number of channels, while the fusiform structure provided more intermediate channel numbers to extract features. Unfortunately, the calculation amount was increased due to the continuous use of GhostConv in the ghost bottleneck.
(3) When all the CSP-Bottleneck modules were replaced with the ELAG3 module, the parameter amount was reduced by approximately 40.6% and the calculation amount was reduced by approximately 29.7%. With regard to the accuracy decrease, the main reason was that the bottleneck in the CSP module was replaced by the G-Bneck1. Only half of the features were used in the GhostConv modules in the G-Bneck1. To reduce such a negative influence, the ELAG3 module was therefore designed based on the ELAN rather than the CSP module in order to effectively splice and aggregate the intermediate information.
(4) When the SPPF module was replaced by a GSPPF module, the parameter amount and the calculation amount could be further reduced without a detection accuracy loss.
(5) When ablation A4 and ablation A5 were successively performed, the parameter amount was reduced from 7.04 M to 1.64 M and the calculation amount was reduced from 15.8 GFLOPs to 10.7 GFLOPs. Meanwhile, the mAP50 and the mAP50-95 were improved by 5.6% and 3.7%, respectively. This demonstrated that the prediction head P5 for natural large objects was indeed unnecessary for UAV object detection in the test dataset. The added new prediction head P2 for UAV small objects effectively fused the low-level feature information and improved the detection accuracy. The main reason was that the anchor box sizes were generated by adaptive clustering. After the prediction head P5 was removed, the feature map sizes of prediction heads P2, P3, and P4 were changed from 80 × 80 pixels, 40 × 40 pixels, and 20 × 20 pixels to 160 × 160 pixels, 80 × 80 pixels, and 40 × 40 pixels, respectively. Therefore, more location information could be observed, and the anchor box generated by the adaptive clustering was closer to the prediction box.
(6) When the CIOU loss was replaced by the Focal-EIOU loss, the negative influence of the unstable aspect ratios of prediction boxes and the imbalances of object labels was reduced. Accordingly, the detection accuracy was slightly improved without increasing the parameter amount and the calculation amount.

4.6. Comparisons with Lightweight Object Detection Networks

In this section, in order to evaluate the advancement of the proposed network, a comparison experiment between the GCL-YOLO-S network and some representative lightweight networks was firstly performed on the VisDrone-DET2021 dataset. The experimental results are shown in Table 5.

We can see from Table 5 that the proposed GCL-YOLO-S network achieved the best detection accuracy with the smallest parameter amount, especially for pedestrian, bicycle, car, and motor. The main reason for the relatively poor detection effect of other lightweight networks is that these networks were mainly designed to detect objects in natural scenes. In order to achieve an optimal detection effect, high-level features were mainly used. However, when these lightweight networks were used for UAV small object detection, fewer features of small objects were left in the high-level layers, as shown in Figure 7, resulting in a poor detection effect.

Table 5. Comparisons between the GCL-YOLO-S network and some lightweight networks on the VisDrone-DET2021 dataset.

Network	Params (M)	GFLOPs	FPS	mAP50 (%)										
				All	Pedestrian	People	Bicycle	Car	Van	Truck	Tricycle	Awning Tricycle	Bus	Motor
YOLOv3-Tiny [10]	8.68	12.9	**86**	15.9	19.4	18.4	3.2	49.9	12.7	9.7	8.2	4.0	14.6	18.9
YOLOv4-Tiny [11]	5.89	7.0	78	19.5	21.1	25.1	4.3	57.5	15.1	15.0	12.0	5.3	23.7	16.4
YOLOv5-N	1.77	4.2	68	26.4	33.1	27.1	6.9	67.8	23.5	18.4	12.9	9.1	32.0	31.3
YOLOv5-Lite-G	5.39	15.2	56	27.3	34.6	26.6	7.7	69.3	28.4	24.0	13.8	6.7	28.3	33.4
Nanodet-Plus-M-1.5x	2.44	**3.0**	78	30.4	27.9	24.1	7.4	73	35.1	27.8	17.9	8.4	49.3	33.3
YOLOX-Tiny [12]	5.04	15.3	70	31.3	35.8	21.9	9.6	73.3	34.7	28.1	18.1	10.2	46.3	34.9
YOLOv5-S	7.04	15.8	60	32.7	38.9	31.6	11.3	72.4	34.8	28.5	19.3	9.5	43.4	37.0
PP-PicoDet-L [34]	3.30	8.9	67	34.2	40.2	35.3	12.8	75.6	35.4	29.3	21.1	12.1	44.3	36.3
YOLOv7-Tiny [14]	6.03	13.1	51	36.8	41.5	**38.3**	11.9	77.3	38.9	29.1	23.4	11.7	48.6	47.1
YOLOv8-S	11.10	28.8	56	39.2	42.0	32.5	13.5	79.6	**45.2**	**35.6**	**28.3**	**15.0**	**55.8**	44.6
GCL-YOLO-S	**1.64**	10.7	53	**39.6**	**48.0**	37.6	**15.7**	**81.1**	42.5	32.7	26.4	12.4	53.8	**46.0**

In the compared lightweight networks, the YOLOv8-S network was second only to the GCL-YOLO-S network. The outstanding detection effects achieved were mainly due to the feature extraction module used in the networks. Specifically, the feature information under different gradients was aggregated in the feature extraction module. Compared with gradient accumulations, gradient aggregations obtained a better return.

In order to fully demonstrate the superiority of the proposed GCL-YOLO-S network for UAV small object detection, a comparison of detection effects under a typical complex scene is shown in Figure 10. The scene contains a number of small objects to be detected. These objects are densely distributed and some of them are heavily occluded by the adjacent objects. In addition, different objects have different image sizes due to the influence of the UAV flight altitude and camera imaging angles. It can be seen from Figure 10 that, compared with other excellent lightweight networks, the proposed network achieved a higher detection rate, a lower missed detection rate, and a lower false detection rate. Our prediction boxes had fewer overlaps when the objects were densely distributed. The detection was relatively more accurate, when the objects were heavily occluded.

In general, compared with other excellent lightweight networks, the proposed network achieved better detection results with lower parameter and calculation amounts. Hence, it is more suitable for the actual applications of UAV object detection.

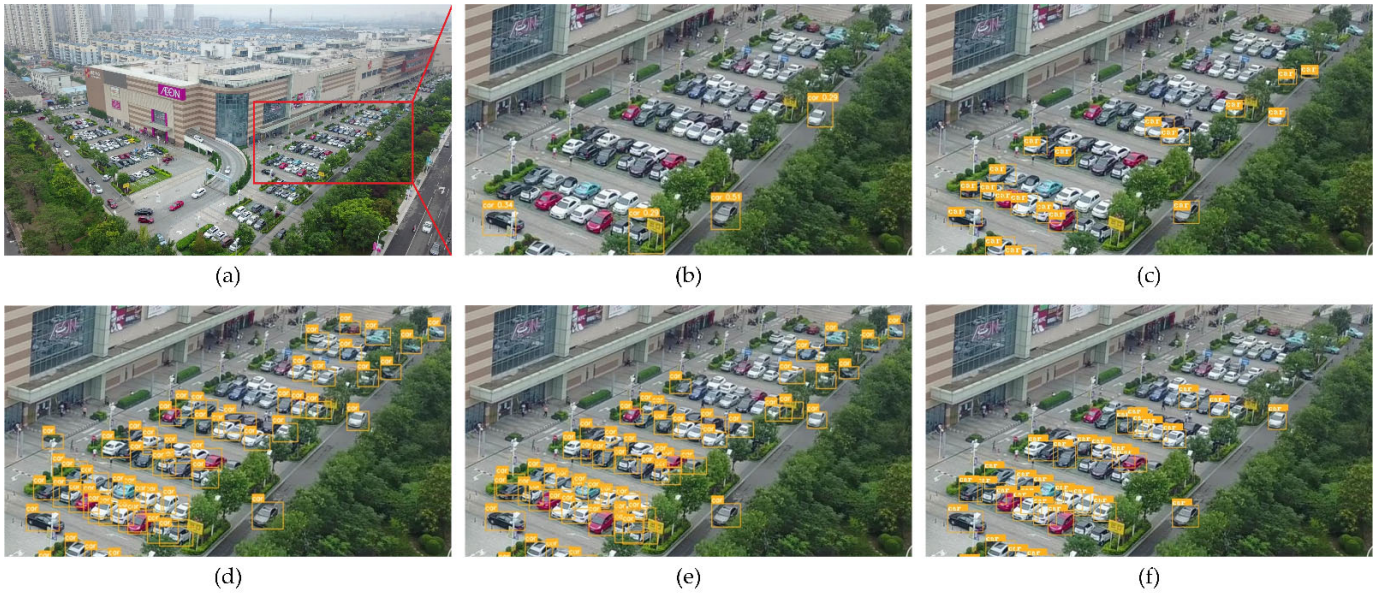


Figure 10. *Cont.*

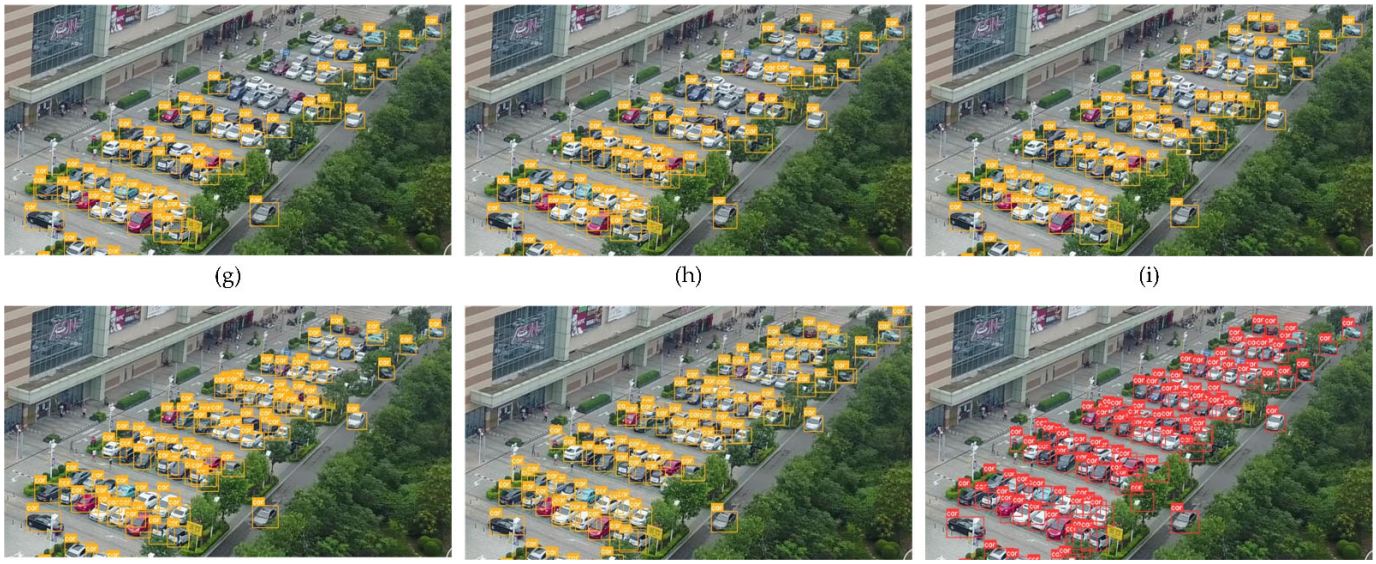


Figure 10. (**a**) Original image and local object detection results achieved by (**b**) YOLOv3-Tiny, (**c**) YOLOv4-Tiny, (**d**) YOLOv5-N (7.0), (**e**) YOLOv5-Lite-G, (**f**) Nanodet-Plus-M-1.5x, (**g**) YOLOX-Tiny, (**h**) YOLOv5-S (7.0), (**i**) PP-PicoDet-L, (**j**) YOLOv7-Tiny, (**k**) YOLOv8-S, and (**l**) GCL-YOLO-S.

The experimental results obtained from the UAVDT dataset are shown in Table 6 and Figure 11. We can see that, compared with the YOLOv5-S network, the mAP50 achieved by the proposed GCL-YOLO-S network was improved by 1.8%, while the parameter and calculation amounts were reduced by 76.7% and 32.3%, respectively. Additionally, the proposed network achieved almost the same detection accuracy as that achieved by the YOLOv8-S network. However, the parameter amount was reduced by 85.2%, and the calculation amount was reduced by 62.8%. This demonstrated the competitive advantage of the proposed network.

Table 6. Comparisons between the GCL-YOLO-S network and some lightweight networks on the UAVDT dataset.

Network	Params (M)	GFLOPs	FPS	mAP50 (%)			
				All	Car	Truck	Bus
YOLOv3-Tiny [10]	8.68	12.9	**87**	26.9	61.4	12.9	6.1
YOLOv4-Tiny [11]	5.89	7.0	77	27.7	63.5	13.4	6.4
YOLOv5-N	1.77	4.2	68	29.5	66.7	14.7	7.1
YOLOv5-Lite-G	5.30	15.1	58	27.6	65.2	12.7	4.8
Nanodet-Plus-M-1.5x	2.44	**3.0**	78	27.9	67.8	7.8	**8.0**
YOLOX-Tiny [12]	5.04	15.3	70	29.1	68.4	13.8	5.3
YOLOv5-S	7.04	15.8	62	29.8	70.1	14.4	5.0
PP-PicoDet-L [34]	3.30	8.9	67	31.1	71.2	16.5	5.8
YOLOv7-Tiny [14]	6.03	13.1	50	31.2	70.4	16.8	6.8
YOLOv8-S	11.10	28.8	57	**31.9**	71.3	**17.4**	7.1
GCL-YOLO-S	**1.64**	10.7	54	31.6	**72.2**	16.2	6.4

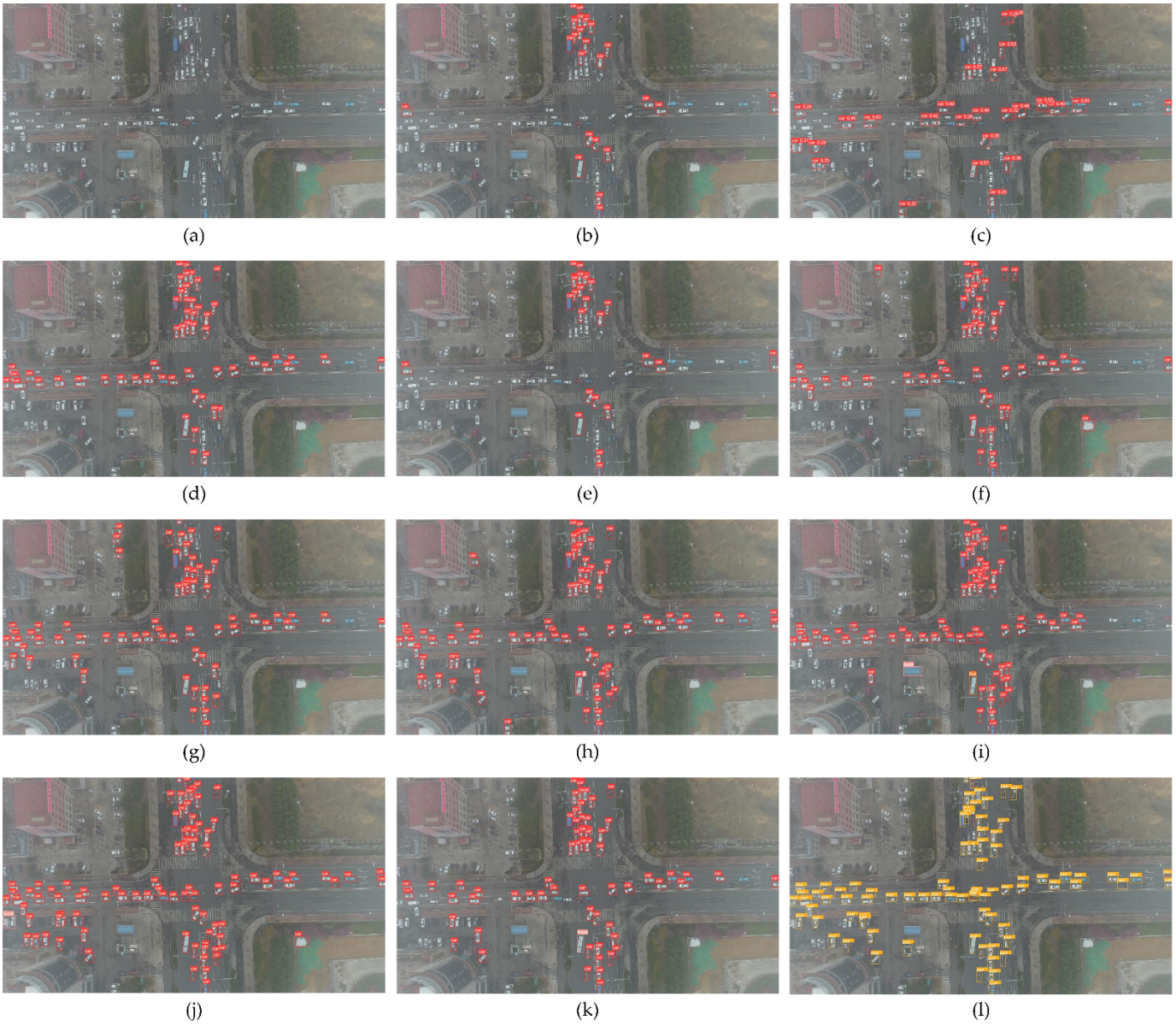


Figure 11. (**a**) Original image and local object detection results achieved by (**b**) YOLOv3-Tiny, (**c**) YOLOv4-Tiny, (**d**) YOLOv5-N (7.0), (**e**) YOLOv5-Lite-G, (**f**) Nanodet-Plus-M-1.5x, (**g**) YOLOX-Tiny, (**h**) YOLOv5-S (7.0), (**i**) PP-PicoDet-L, (**j**) YOLOv7-Tiny, (**k**) YOLOv8-S, and (**l**) GCL-YOLO-S.

4.7. Extended Experiments

4.7.1. Comparisons with Large-Scale Object Detection Networks

In order to demonstrate the competitiveness of the proposed GCL-YOLO network compared with other object detection networks, a comparison experiment was further performed on the VisDrone-DET2021 dataset. The compared networks included RSOD, SlimYOLOv3, and DMNet designed for UAV object detection, along with the state-of-the-art TOOD, VFNet, and TridentNet. The experimental results are shown in Table 7.

It can be seen from Table 7 that, compared with these advanced networks, the proposed GCL-YOLO-L network still had a strong competitiveness. Among all of these networks, the proposed network achieved the second-highest detection accuracy with the fewest parameters and calculations. Such an excellent detection effect achieved by the proposed network mainly benefited from the lightweight design of the GhostConv-based network and the feature fusion structure that was more suitable for small object detection.

Table 7. Comparisons between the GCL-YOLO-L network and some large-scale networks on the VisDrone-DET2021 dataset.

Network	Input size	Params (M)	GFLOPs	mAP50 (%)										
				All	Pedestrian	People	Bicycle	Car	Van	Truck	Tricycle	Awning Tricycle	Bus	Motor
TOOD [35]	1333 × 800	31.81	144.40	41.0	41.5	31.9	19.2	81.4	46.5	39.6	31.8	14.1	53.5	50.5
VFNet [36]	1333 × 800	33.50	140.10	41.3	41.8	25.4	20.0	80.4	47.4	**41.7**	**35.1**	15.5	57.0	48.8
Tridentnet [37]	1333 × 800	32.85	822.19	43.3	54.9	29.5	20.1	81.4	**47.7**	41.4	34.5	15.8	58.8	48.9
RSOD [19]	608 × 608	63.72	84.21	43.3	46.8	36.8	17.1	81.8	49.8	39.3	32.3	**19.3**	**61.2**	48.6
SlimYOLOv3 [22]	832 × 832	20.80	122.00	45.9	-	-	-	-	-	-	-	-	-	-
DMNet [20]	640 × 640	41.53	194.18	**47.6**	-	-	-	-	-	-	-	-	-	-
GCL-YOLO-L	640 × 640	**8.77**	**50.70**	45.7	**55.2**	**43.1**	**20.8**	**84.5**	50.0	40.9	33.3	15.7	60.3	**52.8**

Note: Limited by the code openness, the experimental results of the RSOD network, the SlimYOLOv3 network, and the DMNet network in their respective published papers were used here.

4.7.2. Comparisons with YOLO Series Networks

Considering the different requirements of UAV object detection in different applications, the derived GCL-YOLO-N, GCL-YOLO-S, GCL-YOLO-M, and GCL-YOLO-L networks were comprehensively compared with the excellent YOLO series networks on the VisDrone-DET2021 dataset. The experimental results are shown in Figure 12.

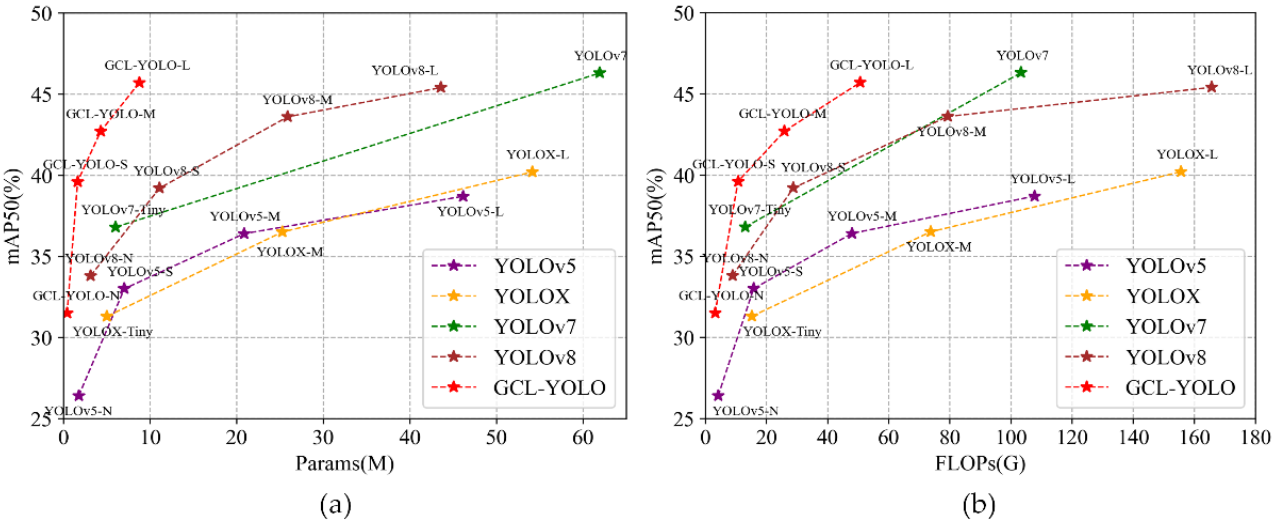


Figure 12. Comparisons between (**a**) the GCL-YOLO networks and (**b**) the YOLO networks.

In Figure 12, the proposed GCL-YOLO network achieved a detection accuracy that was second only to the YOLOv7 network on the VisDrone-DET2021 dataset, but the GCL-YOLO network had much fewer parameters and calculations. Overall, compared with the corresponding derived networks of the YOLO networks, the four derived networks of the GCL-YOLO network were lighter and more effective. Hence, it is expected that the GCL-YOLO network will have a wide application in the field of UAV object detection.

4.8. Qualitative Visualization of Detection Results

In order to visually demonstrate the performance of the proposed GCL-YOLO network, more detection results on the VisDrone-DET2021 dataset and the UAVDT dataset are shown in Figures 13 and 14. It can be seen that the proposed network performed well in terms of detecting small objects, dense objects, occluded objects, and blurred objects.

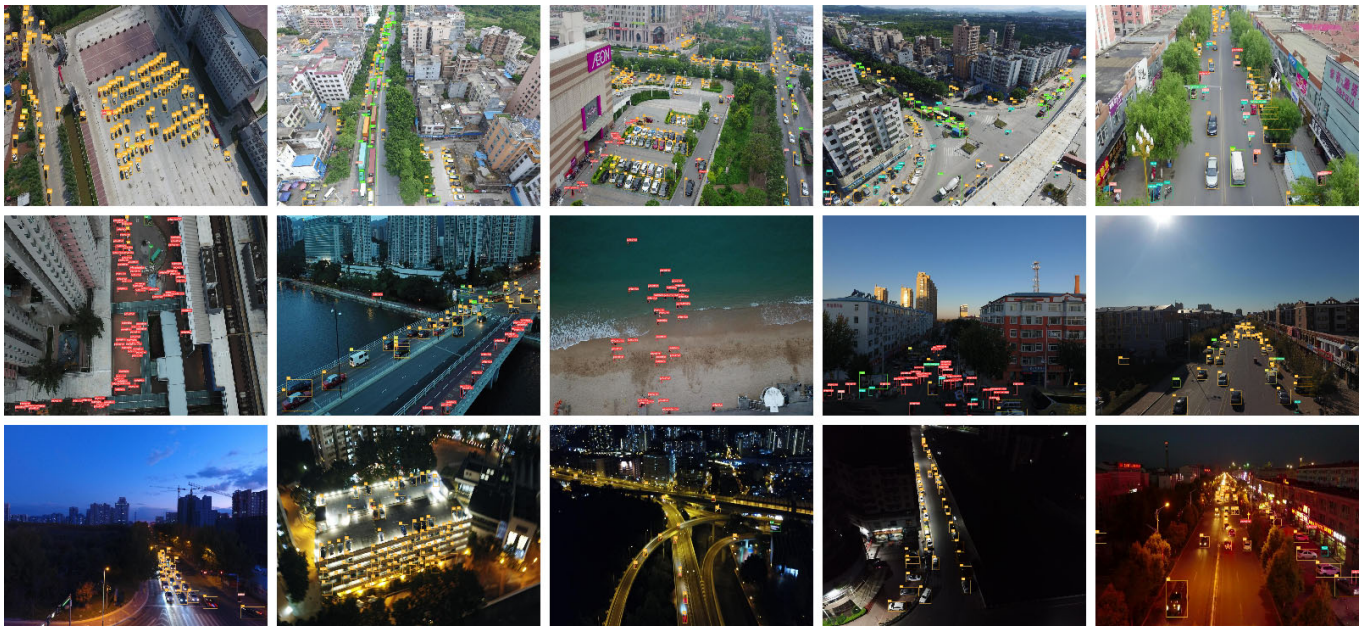


Figure 13. More detection results on the VisDrone2021 dataset. Different colored bounding boxes denote different object categories.

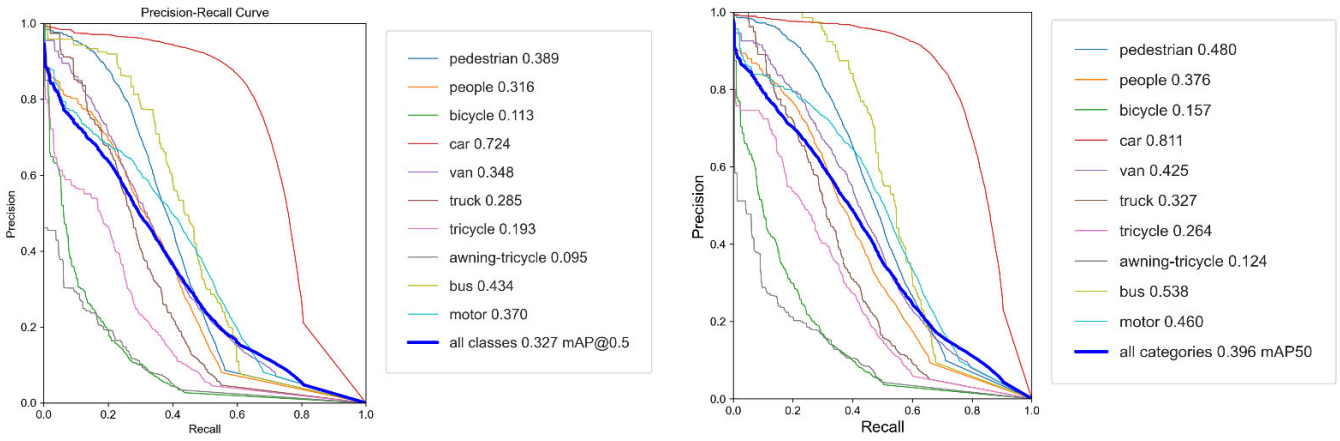
Figure 14. More detection results on the UAVDT dataset.

5. Discussion

Compared with the YOLOv5-S network, the mAP50 achieved by the GCL-YOLO-S network was improved by 6.9%, while the parameter amount was reduced by 76.7% and the calculation amount was reduced by 32.3%. Regrettably, the detection speed was reduced by 7 FPS. The detection accuracy improvement was mainly due to the fusion of the low-level feature information. Accordingly, the low-level feature information is too

large and requires a longer inference time; the detection speed was therefore reduced. The reduction in the parameter amount and calculation amount was mainly due to the use of the GhostConv module and the removal of the prediction head P5. Generally, such a reduced detection speed was acceptable compared to the improved detection accuracy and the reduced parameter amount and calculation amount.

In order to further demonstrate the feasibility and effectiveness of the proposed network, Figure 15 shows the precision–recall (P–R) curves obtained by the YOLOv5-S network and the GCL-YOLO-S network. In addition to the P–R curves of each object category in the VisDrone-DET2021 dataset, the corresponding mAP50 of each category is shown in the legend. It can be seen from Figure 15 that the detection accuracies of all object categories achieved by the proposed GCL-YOLO-S network were better than those achieved by the YOLOv5-S network. Especially for the four categories of pedestrian, van, bus, and motor, the mAP50 was improved by 9%–10%. Generally, pedestrian and motor in the test dataset had fewer image pixels. Such an accuracy improvement was mainly due to the fusion of object location information in the low-level feature maps and the processing of prediction boxes by the Focal-EIOU loss.



(a) (b)

Figure 15. P–R curves obtained by (**a**) the YOLOv5-S network and (**b**) the GCL-YOLO-S network.

Additionally, the confusion matrix obtained by the GCL-YOLO-S network is shown in Figure 16 to visualize the classification of each object category. In Figure 16, each row represents a predicted category and each column represents a true category. Each value on the diagonal represents a proportion of correct classifications in that category. It can be seen from Figure 16 that bicycle, tricycle, and awning tricycle had the highest rate of false negatives (FNs), which means that the predictions of these categories had a high rate of missed detection. Pedestrian and car had the highest rate of false positive (FPs), which means that the predictions of these categories also had a high rate of false detection. From the proportions of correct classifications on the diagonal, we can see that car had the best detection results, while awning tricycle had the worst detection results. The main reason for this was that the label number of different object categories used for training had a large deviation, as shown in Figure 9. The imbalance in the label number of different object categories had a great negative impact on the training of the network.

Overall, the proposed GCL-YOLO network is a lightweight UAV small object detection network. It is able to achieve a good detection accuracy with fewer computing and storage resources. Therefore, it is more suitable for real-time object detection on UAV platforms with limited computing resources and storage resources.

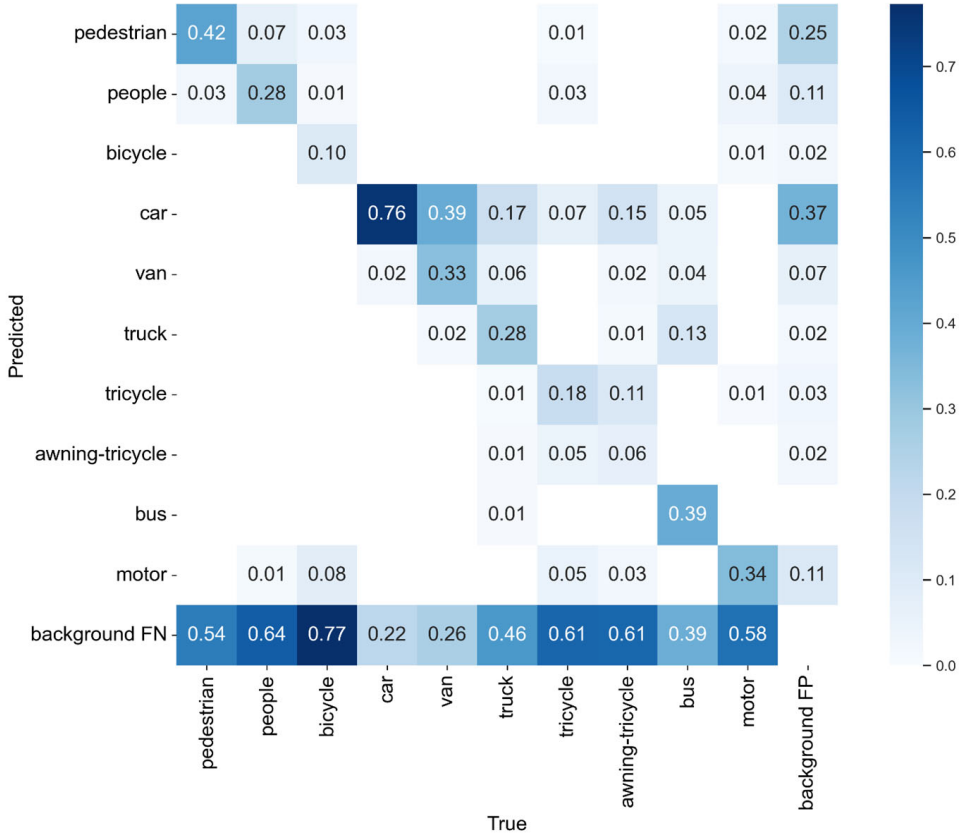


Figure 16. Confusion matrix obtained by the GCL-YOLO-S network with the IOU threshold of 0.45 and the confidence threshold of 0.25.

6. Conclusions

As the application requirement is that UAV small objects should be precisely detected with limited computing resources and storage resources, the structure of each part in the YOLOv5 network was analyzed and redesigned; then, a GCL-YOLO network for UAV small object detection was proposed in this study.

The proposed GCL-YOLO network was tested on the VisDrone-DET2021 dataset. The experimental results show that, compared with the YOLOv5 network, the proposed GCL-YOLO network could effectively improve the object detection accuracy and reduce the parameter and calculation amounts. Compared with some lightweight networks, the proposed GCL-YOLO-S network achieved the best detection accuracy with the smallest parameter amount and a medium calculation amount. Compared with some large-scale networks, the proposed GCL-YOLO-L network achieved the second-highest detection accuracy with the fewest parameters and calculations. As such, the experimental results demonstrate the effectiveness of the proposed network.

At present, limited by appropriate UAV platforms, the proposed GCL-YOLO network was only evaluated on the VisDrone-DET2021 dataset and the UAVDT dataset. Further studies on a UAV platform and more UAV datasets are necessary to evaluate the effectiveness of the proposed network in a real-world application.

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

Funding: This research was funded, in part, by the Scientific Research Foundation of Hubei University of Technology, grant number BSQD2020055, and partly funded by the Northwest Engineering Corporation Limited Major Science and Technology Projects, grant number XBY-ZDKJ-2020-08.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the anonymous reviewers and members of the editorial team for their comments and contributions.

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

Abbreviations

UAV	Unmanned Aerial Vehicle
YOLO	You Only Look Once
GCL-YOLO	GhostConv-Based Lightweight YOLO
CNN	Convolutional Neural Network
RPN	Region Proposal Network
PAN	Path Aggregation Network
DS-GAN	Downsampling Generative Adversarial Network
SME-Net	Scale-Aware Feature Split–Merge–Enhancement Network
TPH	Transformer Prediction Head
RSOD	Real-Time Small Object Detection
DMNet	Density-Map-Guided Object Detection Network
DSC	Deep Separable Convolution
ELA-PAN	Efficient Layer Aggregation and Path Aggregation Network
CSP	Cross-Stage Partial Bottleneck
ELAG3	Efficient Layer Aggregation Ghost Bottleneck with 3 convolutions
GSPPF	Fast Spatial Pyramid Pooling
G-Bneck1	Ghost Bottleneck with Stride=1
G-Bneck2	Ghost Bottleneck with Stride=2
ECA	Efficient Channel Attention
Grad-CAM	Gradient-Weighted Class Activation Mapping
IOU	Intersection over Union
CIOU	Complete IOU
EIOU	Efficient IOU
SGD	Stochastic Gradient Descent
P–R	Precision–Recall
FP	False Positive
FN	False Negative
mAP	Mean Average Precision
FPS	Frames Per Second
GFLOPs	Giga Floating-Point Operations Per Second

References

1. Audebert, N.; Le Saux, B.; Lefèvre, S. Beyond RGB: Very High Resolution Urban Remote Sensing with Multimodal Deep Networks. *ISPRS J. Photogram. Remote Sens.* **2018**, *140*, 20–32.
2. Kundid Vasić, M.; Papić, V. Improving the Model for Person Detection in Aerial Image Sequences Using the Displacement Vector: A Search and Rescue Scenario. *Drones* **2022**, *6*, 19.
3. Pizarro, S.; Pricope, N.G.; Figueroa, D.; Carbajal, C.; Quispe, M.; Vera, J.; Alejandro, L.; Achallma, L.; Gonzalez, I.; Salazar, W.; et al. Implementing Cloud Computing for the Digital Mapping of Agricultural Soil Properties from High Resolution UAV Multispectral Imagery. *Remote Sens.* **2023**, *15*, 3203.
4. Muhmad Kamarulzaman, A.M.; Wan Mohd Jaafar, W.S.; Mohd Said, M.N.; Saad, S.N.M.; Mohan, M. UAV Implementations in Urban Planning and Related Sectors of Rapidly Developing Nations: A Review and Future Perspectives for Malaysia. *Remote Sens.* **2023**, *15*, 2845.
5. Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Proceedings of the Computer Vision—ECCV 2014, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
6. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vision* **2010**, *88*, 303–338.
7. Cao, Y.; He, Z.; Wang, L.; Wang, W.; Yuan, Y.; Zhang, D.; Zhang, J.; Zhu, P.; Van Gool, L.; Han, J.; et al. VisDrone-DET2021: The Vision Meets Drone Object Detection Challenge Results. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision Workshops, Montreal, QC, Canada, 11–17 October 2021; pp. 2847–2854.

8. Yu, H.; Li, G.; Zhang, W.; Huang, Q.; Du, D.; Tian, Q.; Sebe, N. The Unmanned Aerial Vehicle Benchmark: Object Detection, Tracking and Baseline. *Int. J. Comput. Vision* **2020**, *128*, 1141–1159.
9. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
10. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
11. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
12. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
13. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976.
14. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 7464–7475.
15. Kisantal, M.; Wojna, Z.; Murawski, J.; Naruniec, J.; Cho, K. Augmentation for Small Object Detection. *arXiv* **2019**, arXiv:1902.07296.
16. Bosquet, B.; Cores, D.; Seidenari, L.; Brea, V.M.; Mucientes, M.; Del Bimbo, A. A Full Data Augmentation Pipeline for Small Object Detection Based on Generative Adversarial Networks. *Pattern Recognit.* **2023**, *133*, 108998.
17. Ma, W.; Li, N.; Zhu, H.; Jiao, L.; Tang, X.; Guo, Y.; Hou, B. Feature Split–Merge–Enhancement Network for Remote Sensing Object Detection. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–17.
18. Zhu, X.; Lyu, S.; Wang, X.; Zhao, Q. TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-Captured Scenarios. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision Workshops, Montreal, QC, Canada, 11–17 October 2021; pp. 2778–2788.
19. Sun, W.; Dai, L.; Zhang, X.; Chang, P.; He, X. RSOD: Real-Time Small Object Detection Algorithm in UAV-Based Traffic Monitoring. *Appl. Intell.* **2022**, *52*, 8448–8463.
20. Li, C.; Yang, T.; Zhu, S.; Chen, C.; Guan, S. Density Map Guided Object Detection in Aerial Images. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 190–191.
21. Liu, Y.; Gao, M.; Zong, H.; Wang, X.; Li, J. Real-Time Object Detection for the Running Train Based on the Improved YOLO V4 Neural Network. *J. Adv. Transp.* **2022**, *2022*, 1–17.
22. Zhang, P.; Zhong, Y.; Li, X. SlimYOLOv3: Narrower, Faster and Better for Real-Time UAV Applications. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop, Seoul, Republic of Korea, 27–28 October 2019.
23. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. Mobilenetv2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
24. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
25. Howard, A.; Sandler, M.; Chu, G.; Chen, L.-C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V. Searching for Mobilenetv3. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.
26. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6848–6856.
27. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. *arXiv* **2018**, arXiv:1807.11164.
28. Tang, Y.; Han, K.; Guo, J.; Xu, C.; Xu, C.; Wang, Y. GhostNetv2: Enhance Cheap Operation with Long-Range Attention. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 9969–9982.
29. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. GhostNet: More Features from Cheap Operations. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1580–1589.
30. 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 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11534–11542.
31. Wang, C.-Y.; Liao, H.-Y.M.; Wu, Y.-H.; Chen, P.-Y.; Hsieh, J.-W.; Yeh, I.-H. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 390–391.
32. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. *arXiv* **2019**, arXiv:1911.08287.
33. Zhang, Y.; Ren, W.; Zhang, Z.; Jia, Z.; Wang, L.; Tan, T. Focal and Efficient IOU Loss for Accurate Bounding Box Regression. *arXiv* **2021**, arXiv:2101.08158. [CrossRef]
34. Yu, G.; Chang, Q.; Lv, W.; Xu, C.; Cui, C.; Ji, W.; Dang, Q.; Deng, K.; Wang, G.; Du, Y. PP-PicoDet: A Better Real-Time Object Detector on Mobile Devices. *arXiv* **2021**, arXiv:2111.00902.
35. Feng, C.; Zhong, Y.; Gao, Y.; Scott, M.R.; Huang, W. Tood: Task-Aligned One-Stage Object Detection. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 3490–3499.

36. Zhang, H.; Wang, Y.; Dayoub, F.; Sunderhauf, N. VarifocalNet: An IoU-Aware Dense Object Detector. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 8514–8523.
37. Li, Y.; Chen, Y.; Wang, N.; Zhang, Z. Scale-Aware Trident Networks for Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October 2019–2 November 2019; pp. 6054–6063.