drones

MDPI

Article

A Lightweight Uav Swarm Detection Method Integrated Attention Mechanism

Chuanyun Wang [1], Linlin Meng [1], Qian Gao [1,*], Jingjing Wang [2], Tian Wang [3], Xiaona Liu [4], Furui Du [5], Linlin Wang [1] and Ershen Wang [6]

[1] College of Artificial Intelligence, Shenyang Aerospace University, Shenyang 110136, China
[2] China Academic of Electronics and Information Technology, Beijing 100041, China
[3] Institute of Artificial Intelligence, Beihang University, Beijing 100191, China
[4] Yantai Science and Technology Innovation Promotion Center, Yantai 264003, China
[5] State Key Laboratory of Automatic Control Technology for Mining and Metallurgy Process, Beijing 102628, China
[6] School of Electronic and Information Engineering, Shenyang Aerospace University, Shenyang 110136, China
* Correspondence: gaoqian@buaa.edu.cn

Abstract: Aiming at the problems of low detection accuracy and large computing resource consumption of existing Unmanned Aerial Vehicle (UAV) detection algorithms for anti-UAV, this paper proposes a lightweight UAV swarm detection method based on You Only Look Once Version X (YOLOX). This method uses depthwise separable convolution to simplify and optimize the network, and greatly simplifies the total parameters, while the accuracy is only partially reduced. Meanwhile, a Squeeze-and-Extraction (SE) module is introduced into the backbone to improve the model's ability to extract features; the introduction of a Convolutional Block Attention Module (CBAM) in the feature fusion network makes the network pay more attention to important features and suppress unnecessary features. Furthermore, Distance-IoU (DIoU) is used to replace Intersection over Union (IoU) to calculate the regression loss for model optimization, and data augmentation technology is used to expand the dataset to achieve a better detection effect. The experimental results show that the mean Average Precision (mAP) of the proposed method reaches 82.32%, approximately 2% higher than the baseline model, while the number of parameters is only about 1/10th of that of YOLOX-S, with the size of 3.85 MB. The proposed approach is, thus, a lightweight model with high detection accuracy and suitable for various edge computing devices.

Keywords: object detection; Unmanned Aerial Vehicle (UAV) swarm; lightweight model; attention mechanism; data augment

check for updates

Citation: Wang, C.; Meng, L.; Gao, Q.; Wang, J.; Wang, T.; Liu, X.; Du, F.; Wang, L.; Wang, E. A Lightweight Uav Swarm Detection Method Integrated Attention Mechanism. *Drones* **2023**, *7*, 13. https://doi.org/10.3390/drones7010013

Academic Editor: Carlos Tavares Calafate

Received: 27 November 2022
Revised: 22 December 2022
Accepted: 23 December 2022
Published: 25 December 2022

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In recent years, with the rapid development and wide application of Unmanned Aerial Vehicle (UAV) technology in civil and military fields, there has been a tremendous escalation in the development of applications using UAV swarms. Currently, the main research effort in this context is directed toward developing unmanned aerial systems for UAV cooperation, multi-UAV autonomous navigation, and UAV pursuit-evasion problems [1].

In modern wars, where UAVs are widely used, the technical requirements for anti-UAV technologies are becoming increasingly significant [2]. However, existing anti-UAV technologies are not enough to effectively deal with the suppression of UAV swarms [3]. For a small number of UAVs, countermeasures such as physical capture, navigation deception, seizing control and physical destruction can be used. But it is difficult to cope with a large number of UAVs once they gather together to form a UAV swarm. It is imperative to be able to detect the incoming UAV swarm from a long distance in time and then carry out scale estimation, target tracking, and other operations.

Therefore, the development of UAV swarm target detection and tracking, etc., is the premise and key to achieving comprehensive awareness, scientific decision-making, and active response in battlefield situations. Because anti-UAV swarm systems have high requirements for the accuracy and speed of object location and tracking methods and radar detection, passive location, and other methods experience significant interference from other signal clutter, resulting in false detections or missing detection problems. Therefore, using computer vision technology to detect and track the UAV swarm has significant research value.

This paper focuses on UAV target detection for anti-UAV systems. Specifically, under our proposed method, once the UAV swarm is detected, detectors are rapidly deployed on the ground to obtain video, and quickly and accurately detect the target to facilitate subsequent countermeasures.

Most of the existing object detection algorithms consider the object scale to be of medium size, while a low-flying UAV accounts for a very small proportion of the image, and there is little available texture information. It is difficult to extract useful features, especially against a complex background, and, thus, it is easy to mistakenly detect or miss the UAV target. Therefore, in order to improve the capability of UAV swarm detection in different scales and complex scenes, and meet the application requirements in resource-constrained situations, such as in terms of computing power and storage space, this paper proposes a lightweight UAV swarm detection method that integrates an attention mechanism. Data augmentation technology is applied to expand the dataset to improve the diversity of the training set. In addition, depthwise separable convolution [4] is used to compress the main structure of the network, with the aim of building a model that meets the accuracy requirements and takes up as little computing resources as possible.

We train and test based on the UAVSwarm dataset [5], and the experimental results show that the mAP value of the proposed method reaches 82.32%, while the number of parameters is only about 1/10th of that of the YOLOX-S model and the model size is only 3.85 Mb. Under the same experimental conditions, compared with other YOLO series lightweight models, the detection accuracy of the proposed method is 15.59%, 15.41%, 1.78%, 0.58%, and 1.82% higher than MobileNetv3-yolov4, GhostNet-Yolov4, YOLOv4-Tiny, YOLOX-Tiny, and YOLOX-Nano models, respectively. At the same time, the total network parameters and model size are excellent.

The main innovations of this paper are as follows:

(1) The depthwise separable convolution method is used to compress the model, and a nano network is constructed to achieve the lightweight UAV swarm detection network.

(2) A Squeeze-and-Extraction (SE) module [6] is introduced into the backbone to improve the network's ability to extract object features. The introduction of a Convolutional Block Attention Module (CBAM) [7] in the feature fusion network makes the network pay more attention to important features and suppress unnecessary features.

(3) During the training process, Distance-IoU (DIoU) [8] is used instead of Intersection over Union (IoU) to calculate the regression loss, which is beneficial for model optimization. At the same time, Mosaic [9] and Mixup [10] data augmentation technologies are used to expand the dataset to achieve a better detection effect.

2. Related Work

Swarm intelligence algorithms play an extremely important role in multiple UAV collaborations such as collision avoidance, task assignment, path planning, and formation reconfiguration. Object detection is an important computer vision task. Traditional object detection methods are mostly based on manual feature construction [11], which has weak generalization ability and takes up large computing resources. In recent years, with the vigorous development and wide application of deep learning technology in various fields, algorithms based on deep learning have been widely studied by researchers.

2.1. Related Work for UAV Swarm

Currently, lots of researchers pay attention to the development of UAV systems for UAV cooperation, multi-UAV autonomous navigation, and UAV pursuit–evasion problems. For successful communication among collaborating UAVs in a swarm, Cheriguene, Y. et al. [12] proposed COCOMA, an energy-efficient multicast routing protocol for UAV swarms. This method builds a multicast tree that can convey data from a single source to the swarm's UAVs in order to pick the shortest distance between UAVs, optimize total network energy consumption, and extend the network lifetime. Tzoumas, G. et al. [13] newly developed a control algorithm called dynamic space partition (DSP) for a swarm system consisting of high payload UAVs to monitor large areas for firefighting operations. Sastre, C. et al. [14] proposed and validated different algorithms to optimize the take-off time of drones belonging to a swarm, and the experiments proved that the proposed algorithms provide a robust solution within a reasonable time frame. Sastre, C. et al. [15] proposed a collision-free take-off strategy for UAV swarms. Experimental results show that the proposed method can significantly improve time efficiency and keep the risk of collision at zero.

2.2. Related Work for UAV Detection

Object detection algorithms based on deep learning are mainly divided into two categories: two-stage and one-stage detectors. In the former approach, first, a region proposal network is used to estimate a candidate object bounding box. Then, in the second stage, the network extracts features from each candidate box and performs classification and bounding box regression. In this manner, several methods such as R-CNN [16], Fast R-CNN [17], and Faster R-CNN [18] have been proposed. The latter object detector uses a single deep neural network with a regression strategy to directly classify and detect objects. It should be noted that, in this approach, the process of region proposal is avoided. In this manner, several methods such as the You Only Look Once (YOLO) series, Single Shot Detector (SSD) [19], and RetinaNet [20] have been proposed.

With the rapid development of computer vision technology, researchers have carried out a lot of research on image-based UAV detection algorithms. Hu Y. et al. [21] introduced an algorithm based on YOLOv3 into UAV object detection for the first time. In the prediction process, the last four scale feature maps are adopted to conduct multi-scale prediction to enrich the texture and contour information. At the same time, the size of the UAV in four scales feature maps is calculated according to input data, and then the number of anchor boxes is also adjusted. This approach improves the accuracy of small object detection while ensuring speed. Sun H. et al. [22] proposed a UAV detection network named TIB-Net, integrating a structure called cyclic pathway into the existing efficient method Extremely Tiny Face Detector (EXTD) to enhance the capability of the model to extract effective features of small objects. Furthermore, they integrated a spatial attention module into the backbone network to better locate small-size UAVs and further improve detection performance. Ma J. et al. [23] integrated the attention mechanism module into the PP-YOLO detection algorithm and introduced the Mish activation function to eliminate the gradient disappearance problem in the back-propagation process, which significantly improved the detection accuracy. Yavariabdi A. et al. [24] proposed a multi-UAV detection network named FastUAV-NET based on YOLOv3-tiny that can be used for embedded platforms. By increasing the depth and width of the backbone network, local and global features are extracted from the input video stream, providing higher detection accuracy and saving computing time. Liu B. et al. [25] replaced the backbone with a lightweight network Efficient-lite based on YOLOv5s to reduce the number of parameters of the model, introduced adaptive spatial feature fusion technology to balance the loss of accuracy caused by simplifying the network model, and, finally, introduced a constraint of angle into the original regression loss function to avoid the mismatch between the prediction frame and the real frame orientation during the training process in order to improve the speed of network convergence. Wang C. et al. [26] used the Se-ResNet as a feature extraction

network by introducing the SeNet attention mechanism into the backbone to improve the correlation between feature channels and enhance the features of the target to solve the problem of UAVs in low-altitude airspace being submerged in complex background clutter. The differences between the proposed method and the existing UAV detection frameworks are shown in Table 1.

2.3. Related Work for Lightweight Network

Although R-CNN, YOLO, and SSD series algorithms have excellent performance in object detection, they generally have high computational complexity and large model volume, which makes them unable to fully meet the application requirements in resource-constrained situations such as limited computing power, storage space, and/or power consumption [27]. With the development of intelligent mobile devices toward marginalization and mobility, various lightweight object detection algorithms have been developed successively. The goal is to keep good detection performance on devices with low hardware conditions to adapt to the development trend of intelligent devices. The basic idea of MobileNet [4,28,29], based on depthwise separable convolution, is to use depthwise convolution to replace the filter in traditional convolution for feature extraction, and use point convolution instead of filter to combine features while reducing the number of parameters and amount of computation. ShuffleNet [30,31] was published by Zhang X. et al. in 2018, the core of which is to reduce the computation of a large number of point convolutions in MobileNet by using the strategy of combining group convolution and channel shuffle. Tan M. et al. [32] proposed MnasNet in 2019. Its core innovation lies in the proposed multi-objective optimization function and the decomposed hierarchical search space, which correspond to the optimization accuracy and reasoning delay, and improve the diversity between different layers. Han K. et al. [33] proposed GhostNet in 2020. First, they used less convolution to check the input for conventional convolution, obtaining the output features with fewer channels as the internal feature map. Then, they linearly transformed each channel of the internal feature map to obtain its corresponding Ghost feature map. Finally, they connected the internal feature map with the Ghost feature map to obtain the final GhostNet convolution output feature. Xiong Y. et al. [34] proposed a lightweight object detection network called MobileDets in 2021, based on a Neural Architecture Search (NAS) network architecture for object detection tasks, and achieved the state-of-the-art in mobile accelerators.

Table 1. A comparative overview of UAV detection methods.

Method	Detection Strategy	Backbone	Dataset
Hu Y. et al. [14]	YOLOv3-based	DarkNet-53	self-built
Sun H. et al. [15]	TIB-Net	EXTD	self-built
Ma J. et al. [16]	PP-YOLO	ResNet50-vd	Drone-vs-Bird, TIB-Net, and self-built
Yavariabdi A. et al. [17]	FastUAV-NET	Inception module	self-built
Liu B. et al. [18]	YOLOv5-based	Efficientlite	self-built
Wang C. et al. [19]	Se-ResNet	ResNet-18	Drone-vs-Bird and LaSOT
Proposed Method	YOLOX-based	CSPDarkNet	UAVSwarm [33]

3. Materials and Methods

3.1. Overview

The object detection model can generally be abstracted into backbone, neck, and head networks, as shown in Figure 1. The backbone network performs feature extraction, the

neck network performs multi-scale feature fusion on the feature layer obtained by the backbone network, and the head network performs classification and regression analysis.

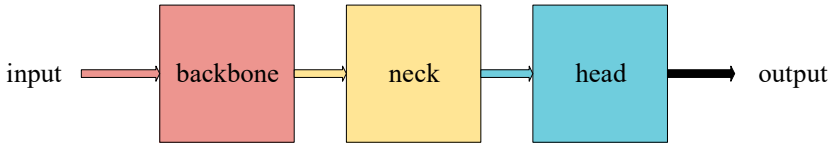


Figure 1. Overall structure of object detection model.

3.2. Backbone Network

CSPDarkNet [35] is used as the backbone of our UAV swarm detection model, consisting mainly of convolution layers and a CSP structure, as shown in Table 2. First, a 640 × 640 RGB three-channel image is input into the network, and the image size and the number of channels are adjusted through Focus. Then, four stacked Resblock body modules are used for feature extraction. In the last Resblock body module, the image is processed through the SPP module; that is, the max pooling operation with different kernel sizes is used for feature extraction to improve the receptive field of the network. The final output of CSPDarkNet is the feature maps of the 2nd, 3rd, and 4th Resblock body modules, with the shapes of 80 × 80 × 256, 40 × 40 × 512, and 20 × 20 × 1024, respectively.

Table 2. The structure of CSPDarkNet.

Module	Structure	Output	Valid Output
Inputs	Inputs	640 × 640 × 3	
Focus	Focus	320 × 320 × 12	
Conv2D-BN-SiLU	Conv2D-BN-SiLU	320 × 320 × 64	
Resblock body1	Conv2D-BN-SiLU	160 × 160 × 128	
	CSPLayer	160 × 160 × 128	
Resblock body2	Conv2D-BN-SiLU	80 × 80 × 256	Output 1
	CSPLayer	80 × 80 × 256	
Resblock body3	Conv2D-BN-SiLU	40 × 40 × 512	Output 2
	CSPLayer	40 × 40 × 512	
Resblock body4	Conv2D-BN-SiLU	20 × 20 × 1024	Output 3
	SPPBottleneck	20 × 20 × 1024	
	CSPLayer	20 × 20 × 1024	

In the Resblock body module, the CSPLayer is similar to the residual structure. The input first passes through convolutional layers with a kernel size of 1×1 and 3×3 for n times, then the result and the original input are concatenated as output, as shown in Figure 2a. In order to further improve the detection of UAV swarm targets, referring to the MobileNet V3 model [29], a Squeeze-and-Extraction (SE) module [6] is introduced into the CSPLayer structure, as shown in Figure 2b. The UAV detection model uses SiLU as the activation function. SiLU has the characteristics of no upper bound, with a lower bound and smooth and non-monotone functions. It can converge faster during training and its formula is as follows:

$$SiLU(x) = x \cdot sigmoid(x), \tag{1}$$

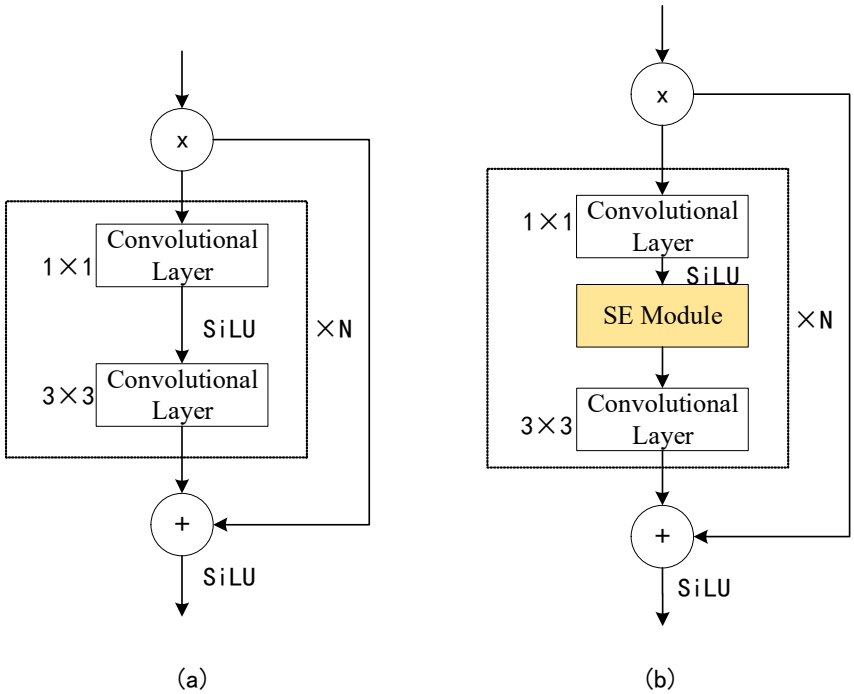


Figure 2. (**a**) Original CSPLayer structure (**b**) CSPLayer structure with SE module.

Specifically, the SE module generates different weight coefficients for each channel by using the correlation between feature channels, multiplies them with the previous features, and adds them to the original features to enhance the features. As shown in Figure 3, the detailed process of the SE attention mechanism is as follows: First, the extracted feature $X \in \mathbb{R}^{H' \times W' \times C'}$ is mapped to $U \in \mathbb{R}^{H \times W \times C}$ through the conversion function F_{tr}. Then, the global information of each channel is represented with a channel characteristic description value through global average pooling $F_{sq}(\cdot)$; and then the channel characteristic description value is adaptively calibrated by $F_{ex}(\cdot, W)$ to make the weight value more accurate. Finally, the enhanced feature is obtained by multiplying the weight value and the original feature through $F_{scale}(\cdot, \cdot)$.

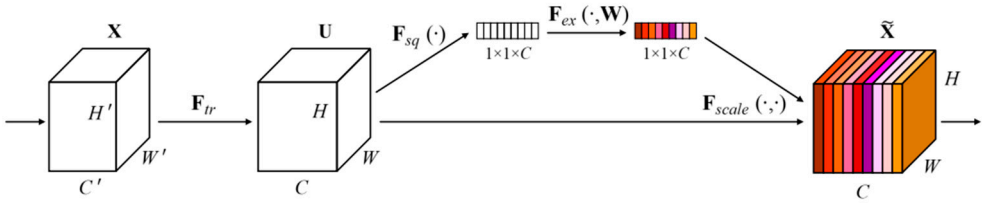


Figure 3. Diagrammatic sketch of the SE module.

3.3. Neck Network

The three feature layers obtained by CSPDarkNet are sent to the neck network for enhanced feature extraction and feature fusion. The neck network of the UAV swarm detection model is constructed based on the Path Aggregation Network (PANet) [36]. The input feature map is resized through a convolution layer and then fused through up- and down-sampling operations. The specific network structure is shown in Figure 4.

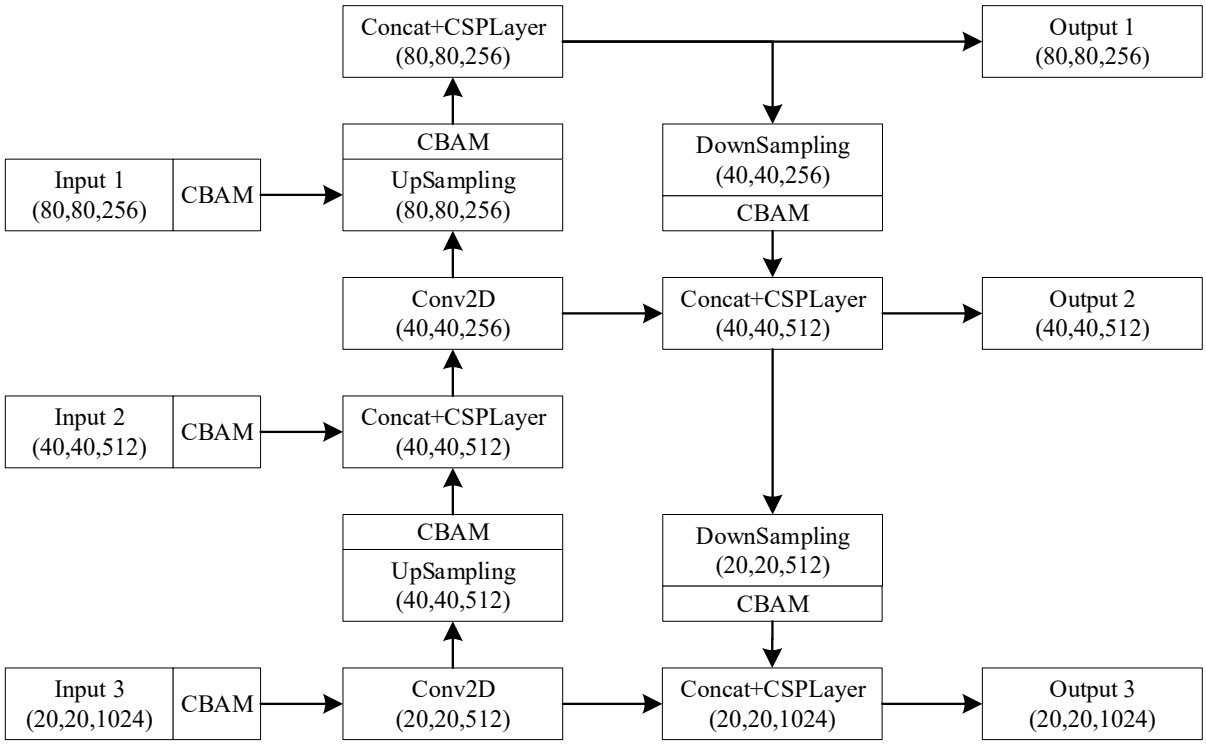


Figure 4. The structure of the neck network.

In order to improve the detection performance of the model for UAVs, a Convolutional Block Attention Module (CBAM) [7] is firstly applied to the three feature maps obtained by the backbone network, and then sent to the neck network for feature fusion. The CBAM is also applied after each up-sampling and down-sampling operation in the PANet. CBAM is a simple and effective attention module for feedforward convolutional neural networks. It combines the two dimensions of channel and spatial features. When the feature map is input, it first goes through the Channel Attention Module (CAM) and then Spatial Attention Module (SAM). The calculation formula is (2):

$$\begin{cases} F_c = M_c(F) \otimes F \\ F_a = M_s(F_c) \otimes F_c' \end{cases} \tag{2}$$

As shown in Figure 5, in the CAM, for the input feature map $F \in \mathbb{R}^{C \times H \times W}$, first, Global Average Pooling (GAP) and Global Maximum Pooling (GMP) operations are performed based on the width and height of the input feature map, and then they are processed by a shared neural network Multilayer Perceptron (MLP), respectively. The two processed results are added together, and a one-dimensional channel attention vector $M_c \in \mathbb{R}^{C \times 1 \times 1}$ is obtained through the Sigmoid function. Finally, the feature map F_c with channel weights is generated by multiplying the channel attention vector M_c and the feature map F. In the SAM module, for the feature map F_c obtained by CAM, a pooling operation is performed, and then the 7×7 convolution and Sigmoid function to obtain a two-dimensional spatial attention vector $M_s \in \mathbb{R}^{1 \times H \times W}$. Finally, F_c and M_s are multiplied to obtain the final feature map F_a.

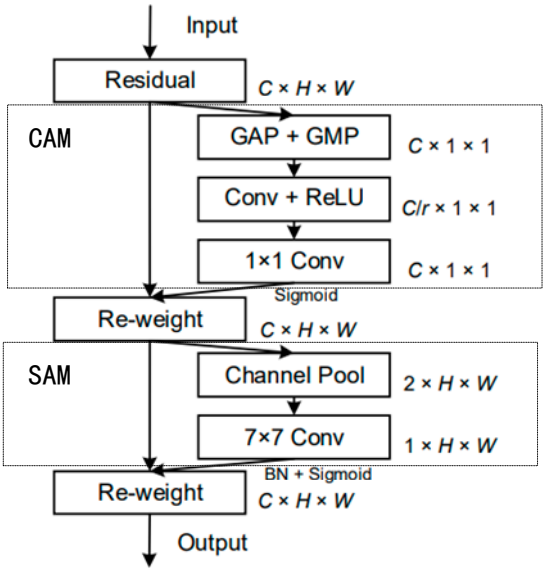


Figure 5. Diagrammatic sketch of the CBAM.

3.4. Head Network

After feature fusion and enhanced feature extraction are completed, the head network conducts classification and regression analysis on the three feature layers of different scales, and finally outputs the recognition results. Its network structure is shown in Figure 6. The head network of the UAV detection model proposed in this paper has two convolution branches [35], one of which is used to achieve object classification and output object categories. The other branch is used to judge whether the object in the feature point exists and regress the coordinates of the bounding box. Thus, for each feature layer, three prediction results can be obtained:

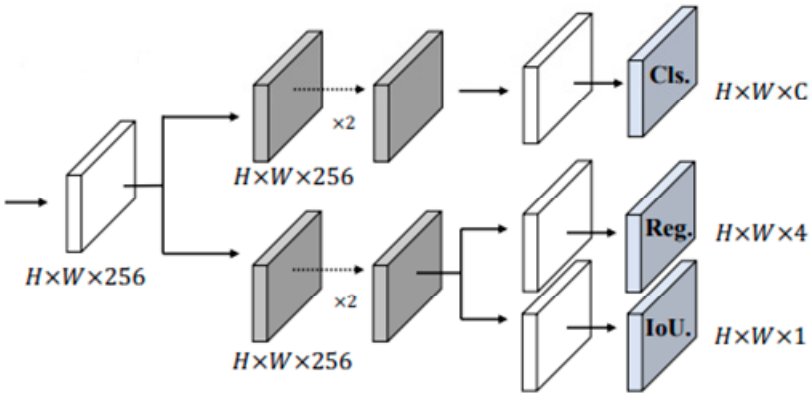


Figure 6. The structure of the head network.

(1) Reg (h, w, 4): The position information of the target is predicted. The four parameters are x, y, w, and h, where x and y are the coordinates of the center point of the prediction box, and w and h are the width and height.

(2) Obj (h, w, 1): This is used to judge whether the prediction box is a foreground or a background. After being processed by the Sigmoid function, it provides the confidence of the object contained in each prediction box. The closer the confidence is to 1, the greater the probability of the existence of a target.

(3) Cls (h, w, num_classes): Determine what type of object, each object is, give each type of object a score, and obtain the confidence level after the sigmoid function processing.

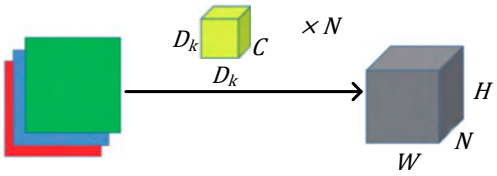
The above three prediction results are stacked, and the prediction result of each feature layer is (h, w, 4+1+num_classes). The first four parameters of the last dimension

are regression parameters of each feature point and the fifth parameter is used to judge whether each feature point contains an object, and the last num_classes parameter is used to judge the category of the object contained in each feature point.

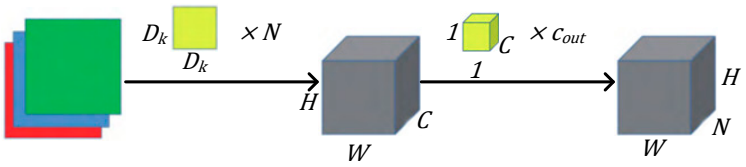
3.5. Lightweight Model

The essence of a lightweight model is to solve the limitations of storage space and energy consumption on the performance of traditional neural networks on equipment with low-performance hardware. Aiming at the problem that the traditional deep convolution neural network consumes a large amount of computing resources, this paper pays more attention to how to reduce the complexity of the model and the amount of computation, while improving the accuracy of object detection. Considering that the depthwise separable convolution method [4] can effectively compress the model size while retaining the ability of feature extraction, this paper uses it to simplify and optimize the UAV swarm detection network.

A standard convolution both filters and combines inputs into a new set of outputs in one step. Depthwise separable convolution splits this into two layers, for filtering and combining. While minimizing the loss of accuracy, this approach can greatly simplify the network parameters and reduce the amount of calculation. The depth-separable convolution operation divides the traditional convolution operation into two steps: depthwise convolution and pointwise convolution. The depthwise convolution applies a single filter to each input channel. The pointwise convolution then applies a 1×1 convolution to combine the outputs of the depthwise convolution. The standard convolution operation and the depthwise separable convolution operation are shown in Figure 7a,b, respectively. The depthwise separable convolution is used to replace the traditional convolution in the UAV detection network while reducing the network parameters and computation.



(a) Traditional Convolution



(b) Depthwise Separable Convolution

Figure 7. Comparison of the two convolution methods.

The total number of convolution kernel parameters and the total amount of convolution operations are analyzed to determine the amount of internal product operations. If we assume that N groups of convolutions, having the same kernel size, $D_k \times D_k \times C$, are taken to check the input image for convolution and that the required feature map size is $W \times H \times N$, then the quantity of parameters and operation required by the two methods are shown in Table 3. The depthwise separable convolution method can compress the network size and reduce the amount of computation. In the process of obtaining a fixed-size feature map using convolution kernels of the same width and height, by expressing convolution as a two-step process of filtering and combining, we get a reduction in necessary computations of $1/N + 1/{D_k}^2$, which is the key to achieving light weight.

Table 3. Comparison of the two convolution methods.

Method	Parameters	Computational Amount	Ratio
standard convolution	$D_k \times D_k \times C \times N$	$D_k \times D_k \times C \times N \times W \times H$	1
depthwise separable convolution	$D_k \times D_k \times C + C \times N$	$D_k \times D_k \times C \times W \times H + C \times N \times W \times H$	$\frac{1}{N} + \frac{1}{D_k^2}$

3.6. Model Training

In the training process of the UAV swarm detection model, Distance-IoU (DIoU) [8] is used instead of Intersection over Union (IoU) to calculate regression loss, which is beneficial for model optimization. In addition, Mosaic [9] and Mixup [10] data augmentation technologies are used to expand the dataset to achieve better UAV detection.

3.6.1. Data Augmentation

Data augmentation is a means of expanding the dataset in computer vision. The approach enhances the image data to compensate for the problem of insufficient training dataset images and achieve the purpose of expanding the training data. As the UAV swarm dataset UAVSwarm used for the experiment has few training samples and repetitive scenes, the Mosaic and Mixup algorithms are used in the image data preprocessing process in order to increase the diversity of training samples and enrich the background of the target, to avoid, as far as possible, the network falling into overfitting during the training process and improve the recognition accuracy and generalization ability of the network model.

The enhancement effect of the Mosaic algorithm is shown in Figure 8. First, the four images are randomly cut, scaled, and rotated, and then they are spliced into a new image as the input image for model training. It should be noted that the image during processing contains the coordinate information of the bounding box of the target, so the new image obtained also contains the coordinate information of the bounding box of the UAV. The advantage of this is that, on the one hand, the size of the object in the picture is reduced to meet the requirements for small object detection accuracy, and, on the other hand, the complexity of the background is increased, so that the UAV swarm detection model has better robustness toward complex backgrounds.



Figure 8. The enhancement effect of Mosaic.

The Mixup algorithm was originally used for image classification tasks. The core idea is to randomly select two images from each batch and mix them up to generate new images in a certain proportion. The Mixup algorithm is more lightweight than Mosaic, requiring

only minimal computational overhead, and can significantly improve the operation speed of the model. Its mathematical expression is as follows:

$$\begin{cases} \widetilde{x} = \lambda x_i + (1-\lambda)x_j \\ \widetilde{y} = \lambda y_i + (1-\lambda)y_j \\ \lambda \epsilon [0,\ 1] \end{cases}, \tag{3}$$

where (x_i, y_i) and (x_j, y_j) are two randomly selected samples and their corresponding labels, $(\widetilde{x}, \widetilde{y})$ are the newly generated samples and their corresponding labels that will be used to train the neural network model, and λ is a fusion coefficient. It can be seen from Formula (3) that Mixup essentially fuses two samples through a fusion coefficient. The enhancement effect of the Mixup algorithm is shown in Figure 9.



Figure 9. The enhancement effect of Mixup.

3.6.2. Loss Function

The goal of network training is to reduce the loss function and make the prediction box close to the ground truth box to obtain a more robust model. The loss function of object detection needs to indicate the proximity between the prediction box and the ground truth, whether the prediction box contains the target to be detected, and whether the object category in the prediction box is true. As predicted by the head network, the loss function consists of three parts, which are given by Formula (4):

$$Loss = Loss_{Reg} + Loss_{Obj} + Loss_{Cls}\ , \tag{4}$$

(1) Regression loss ($Loss_{Reg}$) is the loss of position error between the prediction box and the ground truth. The x, y, w, and h parameters predicted by the model can locate the position of the prediction box, and the loss is calculated based on the DIoU of the ground truth and the prediction box. Figure 10 shows the principle for DIoU to calculate regression loss, and the corresponding calculation Formula is (5):

$$DIoU = IoU - \frac{\rho^2(b, b^{gt})}{c^2} = IoU - \frac{d^2}{c^2}\ , \tag{5}$$

where b represents the parameter of the center coordinate of the prediction box and b^{gt} represents the parameter of the center coordinate of the ground truth; d is the distance between the center point of the prediction box and the ground truth; and c represents the diagonal length of the maximum bounding rectangle of the union of the prediction box and ground truth. IoU measures the intersection ratio between the prediction box and the ground truth. However, if there is no intersection between them, the result of IoU will always be 0. When one of the two boxes is inside the other, if the size of the box remains unchanged, the calculated IoU value will not change, which will make the model difficult

to optimize. If DIoU is used to calculate regression loss, this problem can be effectively solved and a good measurement effect can be obtained.

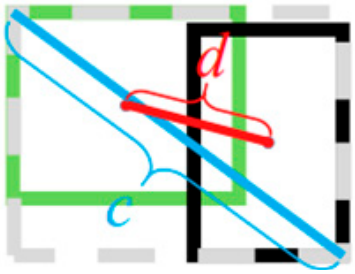


Figure 10. Schematic diagram of DIoU.

(2) Object loss ($Loss_{Obj}$) is to determine whether there is an object in the predicted box, which is a binary classification problem. According to the result predicted by the head network, whether the target is included can be known, while the feature points corresponding to all ground truths are positive samples, and the remaining feature points are negative samples. The Binary Cross-Entropy loss is calculated according to the prediction results of whether the positive and negative samples include the target.

(3) Classification loss ($Loss_{Cls}$) is applied to reflect the error in object classification. According to the feature points predicted by the model, the predicted category results of the feature points are extracted, and then the Binary Cross-Entropy loss is calculated according to the category of the ground truth and prediction results.

4. Experimental Results and Analysis

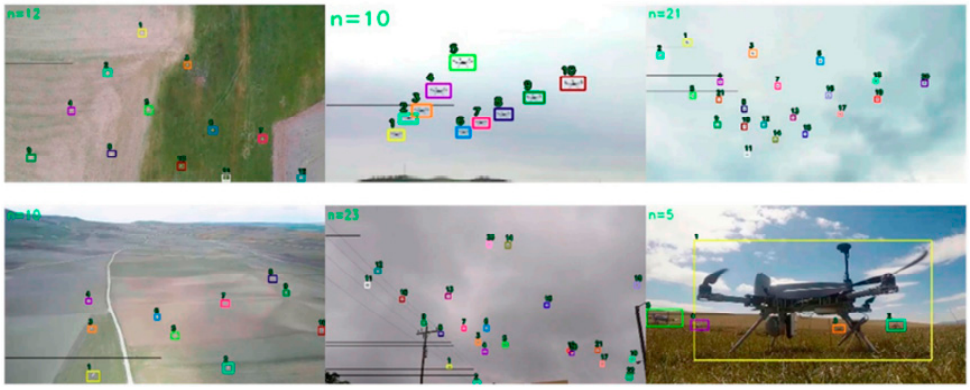
The proposed UAV swarm target detection model is constructed based on the deep learning framework Pytorch. The size of the input images needs to be adjusted to 640×640, and the number of input images in each batch is set to 12 during the training process; a total of 100 epochs are trained without using a pre-training weight. Furthermore, the Mosaic and Mixup data augment algorithms are used for the first 70 epochs and canceled for the last 30 epochs. The gradient descent optimization strategy adopts the SGD optimizer, and the initial learning rate is set to 0.01. In this experiment, the mean Average Precision (mAP), the number of parameters, model size, latency, and Frame Per Second (FPS) are used as measurement metrics of the experimental results. We train and test on a computer equipped with dual Intel Xeon E5 2.40GHz CPUs, a single NVIDIA GTX 1080TI GPU, and 32 GB RAM.

4.1. UAVSwarm Dataset

Wang C. et al. [5] collected 72 UAV image sequences and manually annotated them, creating a new UAV swarm dataset named UAVSwarm for UAV multi-object detection and tracking. This dataset contains 12,598 images in total, of which 23 are included in the images with the largest number of UAVs, 36 image sequences (6844 images) are included in the training set, and the remaining 36 sequences (5754 images) are included in the test set. The dataset we used largely excludes all objects (flocks of birds, etc.) except UAVs but some scenes are complex, leading to the UAVs being blocked. Figure 11 shows some images and annotation information of the dataset.



(a) Sample Images



(b) Sample Images with Labels

Figure 11. Samples of the UAVSwarm dataset.

4.2. Ablation Experiment

Firstly, the structure of YOLOX is simplified and optimized by depthwise separable convolution to build a nano network. In order to verify the effectiveness of the lightweight module, this paper uses the same training strategy to train three lightweight YOLOX models, namely, YOLOX-S, YOLOX-Tiny, and YOLOX-Nano, and tests them on the same test set to analyze their performance differences. It can be seen from Table 4 that the UAV detection accuracy of the three differently scaled YOLOX models toward the test set is more than 80%. As far as the network accuracy and scale of YOLOX of the same series are concerned, the results of this experiment are consistent with the general law of the object detection network—that is, the more layers and parameters of the convolutional neural network, the stronger its feature extraction and generalization ability, and the higher its recognition accuracy. When the DIoU threshold score is set to 0.5, the mAP scores of the three networks are largely the same, but the size of the model and the total number of parameters greatly differ Among them, the model size and the total number of parameters of nano network are about 1/10th of those of the version S network. This shows that under the same hardware conditions, the lightweight nano model can process more input images and reduce the equipment cost on the premise of meeting the accuracy requirement. Therefore, YOLOX-Nano is selected as the baseline model for research in this paper.

Table 4. Performance comparison of YOLOX models with different scales.

Model	mAP@0.5 (%)	Params (×106)	Model Size/Mb
YOLOX-S	82.12	8.94	34.30
YOLOX-Tiny	81.74	5.03	13.70
YOLOX-Nano	80.50	0.89	3.70

Table 5 shows that the mAP score of the proposed model on the UAVSwarm test set is 82.32%, which is about 2% higher than that of the baseline model, while the total number of parameters and model size are only about 40 Kb higher. It can also be found that the introduction of the SE and CBAM modules and the improvement of the loss functions have brought about an increase in mAP compared with the baseline model, which proves the effectiveness of the above three modules.

Table 5. Ablation experiment based on YOLOX-Nano.

SE	CBAM	Loss	mAP@0.5 (%)	Params ($\times 10^6$)	Model Size/Mb
-	-	-	80.50	0.89	3.70
✓	-	-	81.25	0.92	3.78
-	✓	-	80.62	0.92	3.84
-	-	✓	82.14	0.89	3.70
✓	✓	✓	82.32	0.93	3.85

The Loss curves of the different network models during the training process are shown in Figure 12. The abscissa and ordinate are the Epoch and Loss values, respectively. It can be seen that in the training process, the convergence of different models of the YOLOX series is similar. The loss of the training set decreases rapidly in the early stage. With the increase in epoch, the loss value gradually decreases and tends to be stable. Finally, the loss value of the proposed network model is the lowest, which proves that the training strategy and parameter settings are reasonable and effective for improving the model detection accuracy.

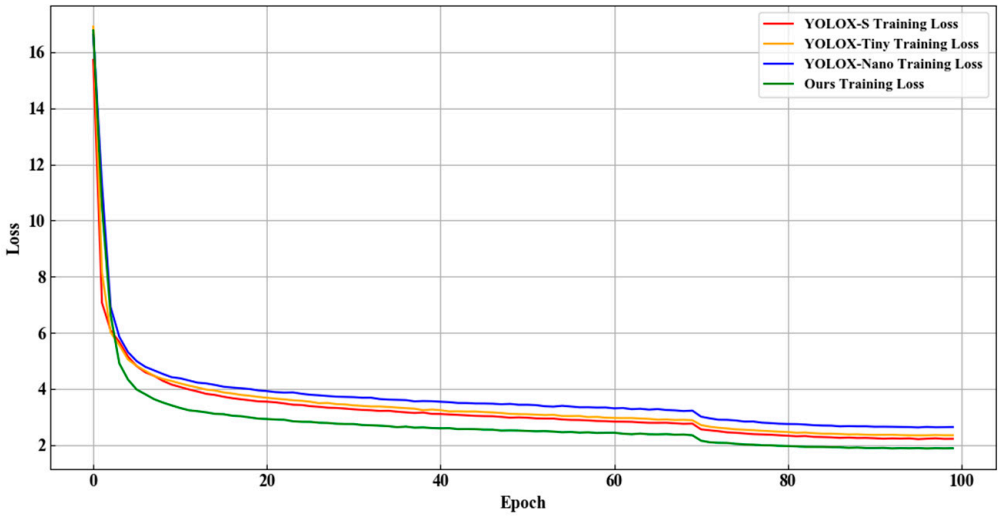
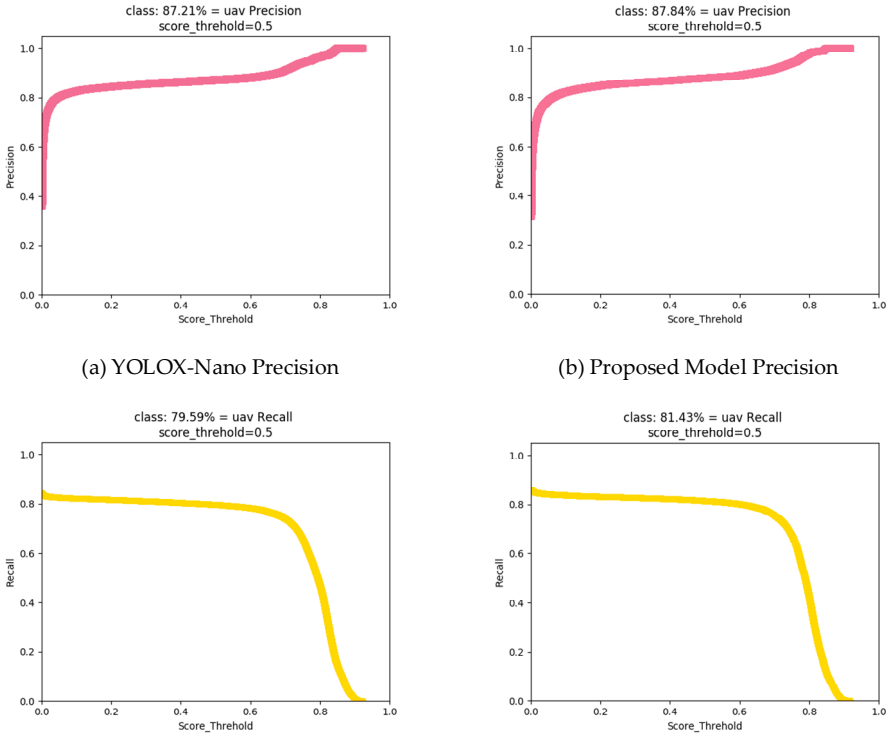


Figure 12. Loss curves of the different networks.

For the UAVSwarm dataset, as a typical small object, the network model proposed in this paper has improved the Precision and Recall indicators of the baseline model by 0.63% and 1.84%, respectively, when the threshold scores are both 0.5, as shown in Figure 13. The improvement of Recall shows that the optimization strategy we used can effectively increase the learning effect of the model on the target of positive foreground samples.



(a) YOLOX-Nano Precision (b) Proposed Model Precision

(c) YOLOX-Nano Recall (d) Proposed Model Recall

Figure 13. Precision and Recall of UAV detection.

4.3. Comparison Experiment

In order to objectively reflect the performance of the UAV swarm object detection network proposed in this paper, this study also uses the same settings to train other lightweight YOLO models and conducts a comparative analysis. The comparison results are shown in Table 6. It can be seen that under the same test set, the mAP value of the proposed UAV detection model has reached 82.32%, which is 15.59%, 15.41%, 1.78%, 0.58%, and 1.82% higher than MobileNetv3-yolov4, GhostNet-YOLOv4, YOLOv4-Tiny, YOLOX-Tiny, and YOLOX-Nano, respectively. At the same time, the total network parameters and model size are optimal.

Table 6. Comparison experiment of different networks.

Method	mAP@0.5 (%)	Params ($\times 10^6$)	Model Size/Mb	Latency/ms	FPS
GhostNet-YOLOv4	66.91	11.00	42.40	83	12
MobileNetv3-YOLOv4	66.73	11.30	53.70	90	11
YOLOv4-Tiny	80.54	5.87	22.40	45	22
YOLOX-Tiny	81.74	5.03	13.70	55	18
YOLOX-Nano	80.50	0.89	3.70	71	14
Ours	82.32	0.93	3.85	71	14

The experimental results that are obtained on the computational time are tabulated in Table 6. It should be said that, due to limitations in the hardware platform, our experiment did not achieve the effect in the original paper, but we believe that the proposed method will have lower latency and higher processing speed under the condition of higher computing power. Table 6 illustrates that, compared with the baseline model YOLOX-Nano, the proposed model achieves higher recognition accuracy under approximately the same inference

time. Thus, our model can also meet the requirements of being real-time and is suitable for applications that require low latency. However, the YOLOX-Tiny and YOLOv4-Tiny networks are quicker than the proposed model, as they run 18 FPS and 22 FPS, respectively. Even though the proposed architecture is slower than these two models, significantly, it provides higher detection accuracy. Therefore, the proposed model is a lightweight model with high detection accuracy and suitable for various edge computing devices.

In order to verify the detection effect of the UAV detection model proposed in this paper in actual scenes, some images in the dataset are selected for detection, and the comparison diagram of the detection effect is shown in Figure 14. As shown in Figure 14a, in an environment with a simple background, most models can successfully identify UAVs. The proposed method can detect more small and distant objects than the baseline model YOLOX-Nano. As shown in Figure 14b, when the background is complex or UAVs are densely distributed, there are many undetected phenomena with the other models. The algorithm in this paper shows better detection performance and can accurately detect UAVs when occlusion occurs among objects. As shown in Figure 14c, when the image resolution is low, the algorithm in this paper can still accurately detect UAVs with complex backgrounds and provide higher confidence scores. Through comparison, it can be demonstrated that our method has improved detection accuracy and confidence, which shows that it can satisfy the requirements of being lightweight and providing higher accuracy, meeting the requirements of industrial applications.

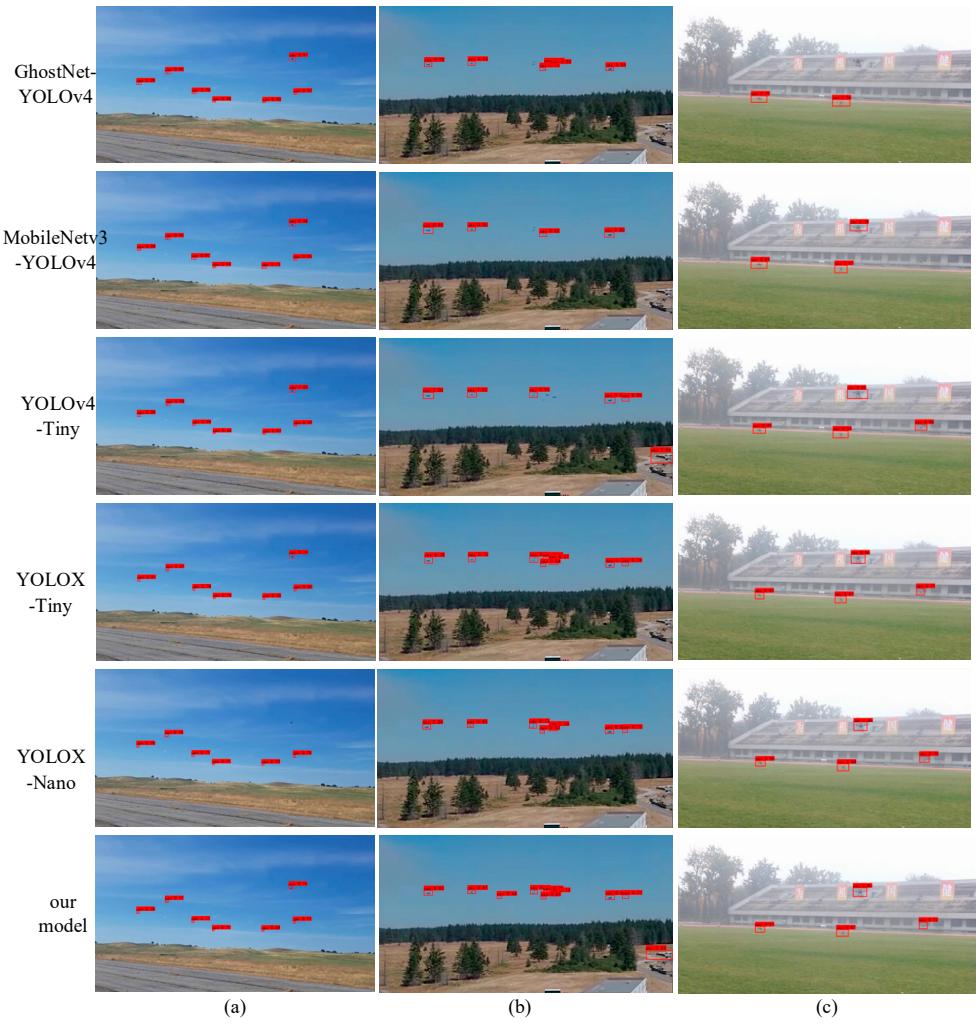


Figure 14. Comparison of detection results of different models.

In addition to the construction of an anti-UAV system, the algorithm proposed in this paper can also be applied as a swarm intelligence algorithm to achieve UAV swarm formation, multi-UAV cooperation, etc. By deploying lightweight object detection methods, UAVs in the swarm can quickly and accurately obtain the position and status of other partners, so that they can adjust themselves in time according to the swarm intelligence algorithm. Swarms of UAVs may have enhanced performance during performing some missions where having coordination among multiple UAVs may enable broader mission coverage and provide more efficient operating performance. Moreover, the proposed method will greatly help to improve the performance of UAV swarm systems including total energy, average end-to-end delay, packets delivery ratio, and throughput. According to the analysis, the application of the lightweight model provided by this paper may reduce the total energy demand and average end-to-end delay of the UAV swarm system, while increasing the packets delivery ratio and throughput of the system. This means that the UAV swarm system may be able to achieve higher data transmission efficiency at a lower cost, thus, better performing tasks.

5. Conclusions

This paper proposes a lightweight UAV swarm detection model integrating an attention mechanism. First, the structure of the network is simplified and optimized by using the depthwise separable convolution method, which greatly reduces the total number of parameters of the network. Then, a SE module is introduced into the backbone network to improve the model's ability to extract object features; the introduction of a CBAM in the feature fusion network makes the network pay more attention to important features and suppress unnecessary features. Finally, in the training process, a loss function based on DIoU can better describe the overlapping information and make the regression faster and more accurate. In addition, two data augmentation technologies are used to expand the UAVSwarm dataset to achieve better UAV detection. The proposed model is a lightweight model with high detection accuracy and only 3.85 MB in size, which is suitable for embedded devices and mobile terminals. In conclusion, the real-time performance and accuracy of the UAV swarm detection model proposed in this paper meet the requirements of rapid detection of UAVs in real environments, which has practical significance for the construction of anti-UAV systems. In our future work, we will continue to study and optimize the improvement strategy, so that it can achieve better recognition accuracy and real-time performance under the premise of minimizing the complexity of the model.

Author Contributions: Conceptualization, C.W.; methodology, C.W.; software, L.M.; validation, L.M., Q.G. and J.W.; formal analysis, Q.G., X.L. and F.D.; investigation, L.M.; resources, C.W.; data curation, L.M.; writing—original draft preparation, L.M.; writing—review and editing, C.W., Q.G. and T.W.; visualization, L.M., L.W. and E.W.; supervision, C.W.; project administration, C.W.; funding acquisition, C.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China (Grant No. 61703287, 62173237, and 61972016), the Scientific Research Program of Liaoning Provincial Education Department of China (Grant Nos. LJKZ0218, LJKMZ20220556, and JYT2020045), the Young and Middle-aged Science and Technology Innovation Talents Project of Shenyang of China (Grant No. RC210401), the Doctoral Scientific Research Foundation of Shenyang Aerospace University (Grant No. 22YB03).

Data Availability Statement: Not applicable.

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

References

1. Tang, J.; Duan, H.; Lao, S. Swarm intelligence algorithms for multiple unmanned aerial vehicles collaboration: A comprehensive review. *Artif. Intell. Rev.* **2022**, 1–33. [CrossRef]
2. Zhang, D.; Cheng, Y.; Lin, Q.; Yu, G.F.; Xiao, L.W. Key Technologies and Development Trend of UAV Swarm Operation. *China New Telecommun.* **2022**, *24*, 56–58.

3. Cai, J.; Wang, F.; Yang, B. Exploration of UAV cluster defense technology. *Aerodyn. Missile J.* **2020**, *12*, 5.
4. Andrew, 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.
5. Wang, C.; Su, Y.; Wang, J.; Wang, T.; Gao, Q. UAVSwarm Dataset: An Unmanned Aerial Vehicle Swarm Dataset for Multiple Object Tracking. *Remote Sens.* **2022**, *14*, 2601. [CrossRef]
6. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. *IEEE T. Pattern Anal.* **2020**, *42*, 2011–2023. [CrossRef] [PubMed]
7. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I. CBAM: Convolutional Block Attention Module. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2018; pp. 3–19.
8. 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, New York, NY, USA, 7–12 February 2020; pp. 12993–13000.
9. Bochkovskiy, A.; Wang, C.-Y.; Liao, H. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
10. Zhang, H.; Cisse, M.; Dauphin, Y.N.; David, L.P. mixup: Beyond Empirical Risk Minimization. *arXiv* **2017**, arXiv:1710.09412.
11. Zou, Z.; Shi, Z.; Guo, Y.; Ye, J. Object Detection in 20 Years: A Survey. *arXiv* **2019**, arXiv:1905.05055v2.
12. Cheriguene, Y.; Bousbaa, F.Z.; Kerrache, C.A.; Djellikh, S.; Lagraa, N.; Lahby, M.; Lakas, A. COCOMA: A resource-optimized cooperative UAVs communication protocol for surveillance and monitoring applications. *Wireless Netw.* **2022**, 1–17. [CrossRef]
13. Tzoumas, G.; Pitonakova, L.; Salinas, L.; Scales, C.; Richardson, T.; Hauert, S. Wildfire detection in large-scale environments using force-based control for swarms of UAVs. *Swarm Intell.* **2022**, 1–27. [CrossRef]
14. Sastre, C.; Wubben, J.; Calafate, C.T.; Cano, J.C.; Manzoni, P. Safe and Efficient Take-Off of VTOL UAV Swarms. *Electronics* **2022**, *11*, 1128. [CrossRef]
15. Sastre, C.; Wubben, J.; Calafate, C.T.; Cano, J.C.; Manzoni, P. Collision-free swarm take-off based on trajectory analysis and UAV grouping. In Proceedings of the 2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Belfast, UK, 9 August 2022; pp. 477–482.
16. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
17. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
18. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
19. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Cheng, Y.; Alexander, C.B. SSD: Single Shot MultiBox Detector. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 21–37.
20. Lin, T.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2980–2988.
21. Hu, Y.; Wu, X.; Zheng, G.; Liu, X. Object Detection of UAV for Anti-UAV Based on Improved YOLO v3. In Proceedings of the 2019 Chinese Control Conference (CCC), Guangzhou, China, 27–30 July 2019; pp. 8386–8390.
22. Sun, H.; Yang, G.; Shen, J.; Liang, D.; Liu, N.; Zhou, H. TIB-Net: Drone Detection Network with Tiny Iterative Backbone. *IEEE Access* **2020**, *8*, 130697–130707. [CrossRef]
23. Ma, J.; Yao, Z.; Xu, C.; Chen, S. Multi-UAV real-time tracking algorithm based on improved PP-YOLO and Deep-SORT. *J. Comput. Appl.* **2022**, *42*, 2885–2892.
24. Yavariabdi, A.; Kusetogullari, H.; Celik, T.; Cicek, H. FastUAV-NET: A Multi-UAV Detection Algorithm for Embedded Platforms. *Electronics* **2021**, *10*, 724. [CrossRef]
25. Liu, B.; Luo, H. An Improved Yolov5 for Multi-Rotor UAV Detection. *Electronics* **2022**, *11*, 2330. [CrossRef]
26. Wang, C.; Shi, Z.; Meng, L.; Wang, J.; Wang, T.; Gao, Q.; Wang, E. Anti-Occlusion UAV Tracking Algorithm with a Low-Altitude Complex Background by Integrating Attention Mechanism. *Drones* **2022**, *6*, 149. [CrossRef]
27. Yang, Y.; Liao, Y.; Lin, C.; Ni, S.; Wu, Z. A Survey of Object Detection Algorithms for Lightweight Convolutional Neural Networks. *Ship Electron. Eng.* **2021**, *41*, 31–36.
28. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L. 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.
29. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.
30. 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.
31. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2018; pp. 122–138.

32. Tan, M.; Chen, B.; Pang, R.; Vasudevan, V.; Vasudevan, V.; Sandler, M.; Howard, A.; Le, Q.V. MnasNet: Platform-Aware Neural Architecture Search for Mobile. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 2815–2823.
33. 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 (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 1577–1586.
34. Xiong, Y.; Liu, H.; Gupta, S.; Akin, B.; Bender, G.; Wang, Y.; Kindermans, P.J.; Tan, M.; Singh, V.; Chen, B. MobileDets: Searching for Object Detection Architectures for Mobile Accelerators. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 3824–3833.
35. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
36. Liu, S.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.

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