ORIGINAL RESEARCH **Open access**

A Compact UAV Swarm Detection Network Using Depthwise Separable Convolution and Attention Mechanisms

Thabo Nkosi[1*], Lerato Molefe[1], Sipho Dlamini[2], Ayanda Mokoena[3], Kabelo Ndlovu[4]

Abstract

Addressing the challenges of insufficient detection precision and excessive computational demands in current anti-UAV systems, this study introduces a lightweight approach for detecting UAV swarms using You Only Look Once Version X (YOLOX). The technique applies depthwise separable convolution to streamline and refine the network structure, substantially decreasing the overall parameter count with only a minor trade-off in accuracy. In addition, a Squeeze-and-Excitation (SE) module is embedded within the backbone network to boost feature extraction performance. A Convolutional Block Attention Module (CBAM) is added to the feature fusion stage, allowing the network to prioritize key features while diminishing the influence of less relevant ones. Furthermore, Distance-IoU (DIoU) is used in place of Intersection over Union (IoU) in the regression loss computation to enhance model training, and data augmentation methods are used to enlarge the training dataset for superior detection performance. Experiments reveal that the proposed method achieves a mean Average Precision (mAP) of 82.32%, improving on the baseline by nearly 2% while requiring only about 1/10th the parameters of YOLOX-S and occupying a model size of 3.85 MB. Consequently, this solution offers a compact, high-accuracy model appropriate for deployment on diverse edge computing platforms.

Keywords Object detection, Attention mechanism, Unmanned Aerial Vehicle (UAV) swarm, Lightweight model, Data augment

*Correspondence:
Thabo Nkosi
thabo.nkosi@uct.ac.za
[1] Department of AI for Maize and Sorghum Genetic Trait Prediction, Faculty of Agriculture, University of Cape Town, Cape Town, South Africa
[2] Department of Smart Pest Surveillance, Faculty of Agriculture, University of the Witwatersrand, Johannesburg, South Africa
[3] Department of Machine Learning for Yield Gap Analysis, Faculty of Agriculture, University of Pretoria, Pretoria, South Africa
[4] Department of Precision Water Recycling, Faculty of Agriculture, Stellenbosch University, Stellenbosch, South Africa

Introduction

Over recent years, the swift progress and widespread adoption of Unmanned Aerial Vehicle (UAV) technology across civilian and military sectors have driven a sharp rise in UAV swarm-based applications. At this stage, most investigations focus on developing unmanned aerial systems that support UAV teamwork, enable independent navigation for multiple UAVs, and enable pursuit-evasion scenarios [1].

In contemporary conflicts with heavy UAV deployment, the need for robust anti-UAV capabilities has become increasingly pressing [2]. Nevertheless, present-day anti-UAV measures fall short when confronting coordinated UAV swarms [3]. While individual or small groups of UAVs can be handled through tactics such as physical interception, signal jamming, hijacking, or direct elimination, large-scale swarms pose far greater challenges. Timely long-range detection of approaching swarms, followed by size assessment and continuous tracking, becomes essential.

This is an open access article distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License (CC BY-NC-SA 4.0), which permits non-commercial use, distribution, reproduction, and adaptation in any medium or format, provided that appropriate credit is given to the original author(s) and source, a link to the license is provided, and any changes made are indicated. Adapted material must be distributed under the same license.

© 2025 The Author(s). | Published by Agro Intelligence Research Press

Hence, advancing UAV swarm detection and tracking technologies lays the foundation for full battlefield awareness, rational decision-making, and effective countermeasures. Given the strict demands of anti-UAV swarm platforms for precise, rapid target positioning, traditional radar or passive sensing approaches often suffer significant disruption from environmental noise, leading to erroneous or overlooked detections. As a result, leveraging computer vision for UAV swarm identification and monitoring carries substantial scientific importance.

This work emphasizes UAV detection tailored for anti-UAV operations. In the suggested framework, the detection of a UAV swarm triggers the immediate deployment of ground-based sensors to record video footage, enabling fast, precise target identification for follow-up actions.

Existing detection algorithms typically target medium-sized objects, yet low-flying UAVs appear as tiny elements in images with minimal texture details. This scarcity of information, particularly amid cluttered surroundings, complicates feature extraction and increases risks of false positives or negatives. To strengthen detection across multiple scales and challenging environments while respecting resource constraints, the current study develops a compact UAV swarm detection framework enhanced with attention mechanisms. Dataset expansion via data augmentation is performed to enrich training variety. Moreover, depthwise separable convolution [4] compresses the core network design, producing a solution that balances required accuracy with minimal resource consumption.

Training and evaluation utilize the UAVSwarm dataset [5]. Results confirm that the developed method achieves an mAP of 82.32%, with parameter volume reduced to roughly 1/10th that of YOLOX-S and a total size of only 3.85 MB. In equivalent test conditions, its accuracy exceeds MobileNetv3-yolov4, GhostNet-Yolov4, YOLOv4-Tiny, YOLOX-Tiny, and YOLOX-Nano by 15.59%, 15.41%, 1.78%, 0.58%, and 1.82%, respectively. Network parameter totals and overall model dimensions also demonstrate strong efficiency.

Key contributions of this paper include:

(1) Application of depthwise separable convolution for model compression, resulting in a nano-scale architecture dedicated to lightweight UAV swarm detection.

(2) Integration of a Squeeze-and-Excitation (SE) module [6] into the backbone for superior object feature capture. Inserting a Convolutional Block Attention Module (CBAM) [7] into the feature fusion component focuses on salient features and reduces noise from unimportant elements.

(3) Replacement of Intersection over Union (IoU) with Distance-IoU (DIoU) [8] for regression loss during training to support better optimization. Simultaneously, Mosaic [9] and Mixup [10] augmentation strategies expand the dataset, yielding enhanced detection results.

Related Work

Swarm intelligence techniques are vital for multi-UAV operations involving obstacle avoidance, duty assignment, route optimization, and formation adjustments. Object detection stands as a fundamental computer vision challenge. Earlier traditional techniques relied mainly on hand-crafted features [11], resulting in poor adaptability and high resource usage. Recently, the explosive growth of deep learning across industries has driven extensive exploration of learning-based detection algorithms.

Related work for UAV swarm

Numerous studies currently explore UAV platforms enabling cooperation, autonomous multi-UAV movement, and pursuit-evasion dynamics. To facilitate reliable information exchange in swarms, Cheriguene *et al.* [12] developed COCOMA, a power-saving multicast routing system. It creates a distribution tree for efficient data delivery from a single origin to all

© 2025 The Author(s). | Published by Agro Intelligence Research Press

swarm members, thereby shortening inter-UAV distances, reducing energy expenditure, and increasing operational duration. Tzoumas *et al.* [13] introduced a dynamic space partition (DSP) control strategy for heavy-lift UAV swarms tasked with wide-area surveillance in fire response missions. Sastre *et al.* [14] designed and tested multiple scheduling algorithms for swarm drone launches, demonstrating reliable performance within practical time constraints. Sastre *et al.* [15] further advanced a safe, collision-free launch protocol for UAV groups. Findings confirm notable gains in temporal efficiency and the complete elimination of collision hazards.

Related work for UAV detection

Deep learning object detectors fall into two primary groups: two-stage and one-stage systems. Two-stage detectors first employ a region proposal network to identify potential bounding boxes, then extract features from each proposal for classification and refinement. Prominent examples include R-CNN [16], Fast R-CNN [17], and Faster R-CNN [18]. One-stage detectors, however, rely on a unified network and regression approach to perform classification and localization simultaneously, thereby eliminating the need for separate proposal generation. Notable instances are the You Only Look Once (YOLO) family, Single Shot Detector (SSD) [19], and RetinaNet [20].

Driven by advances in visual computing, extensive efforts have targeted vision-based UAV identification. Hu *et al.* [21] first adapted YOLOv3 for UAV tasks. Their method incorporates the final four feature scales for multi-resolution analysis to capture richer structural details. It dynamically adjusts anchor counts based on UAV dimensions derived from input resolution, boosting small-target performance without sacrificing processing speed. Sun *et al.* [22] created TIB-Net by embedding a cyclic pathway into the Extremely Tiny Face Detector (EXTD) framework to strengthen small-object representation. A spatial attention component was also incorporated into the backbone to improve small UAV positioning and overall effectiveness. Ma *et al.* [23] combined attention blocks with PP-YOLO and switched to the Mish activation to counter vanishing gradients, yielding marked gains in precision. Yavariabdi *et al.* [24] introduced FastUAV-NET, an extended YOLOv3-tiny architecture optimized for embedded systems. Expanding backbone depth and width allows comprehensive local and global feature learning from video input, achieving higher accuracy with lower latency. Liu *et al.* [25] adopted the EfficientNet-Lite backbone in YOLOv5s to reduce parameters, applied adaptive spatial fusion to mitigate accuracy drops, and added angular constraints to the loss function to align predicted and actual orientations, thereby accelerating convergence. Wang *et al.* [26] utilized Se-ResNet by incorporating SeNet attention into the feature extractor, strengthening channel-wise relationships and target emphasis to overcome background interference for low-altitude UAVs. Differences between the proposed solution and prior UAV detection systems are summarized in **Table 1**.

Related work for lightweight network

Even though the R-CNN, YOLO, and SSD families of algorithms demonstrate strong results in object detection, they typically involve high computational demands and sizable model footprints. This limits their suitability for scenarios with restricted resources, such as constrained processing capabilities, memory capacity, and/or energy usage [27]. As intelligent mobile devices evolve toward edge deployment and greater portability, a range of lightweight object-detection algorithms has emerged. These aim to preserve solid detection capabilities on hardware with modest specifications, aligning with the growth of smart device technologies.

The foundational concept behind MobileNet [4, 28, 29] relies on depthwise separable convolution. It replaces standard convolution filters with depthwise convolutions for feature extraction and employs pointwise convolutions to merge features, thereby reducing parameter count and computational cost. ShuffleNet [30, 31], introduced by Zhang X. *et al.* in 2018, focuses on reducing the cost of extensive pointwise convolutions in MobileNet by combining group convolutions with channel shuffling. Tan *et al.* [32] presented MnasNet in 2019, with its main advances being a multi-objective optimization approach and a factored hierarchical search space that balances accuracy and inference latency while increasing layer diversity. Han *et al.* [33] developed GhostNet in 2020. This begins with fewer convolutions applied to the input for standard convolution, producing output features with reduced channels as an internal feature map. Each channel then undergoes a linear transformation to produce the corresponding Ghost feature maps. These are then combined with

© 2025 The Author(s). | Published by Agro Intelligence Research Press

the internal feature map to yield the final GhostNet convolution output. Xiong *et al.* [34] introduced MobileDets in 2021, a lightweight object detection network built on Neural Architecture Search (NAS) tailored for detection tasks, reaching state-of-the-art results on mobile accelerators.

Table 1. A comparative overview of UAV detection methods.

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

Materials and Methods

Overview

An object detection model can typically be broken down into backbone, neck, and head components, as illustrated in **Figure 1**. The backbone handles feature extraction, the neck integrates multi-scale features from the backbone outputs, and the head handles classification and regression tasks.

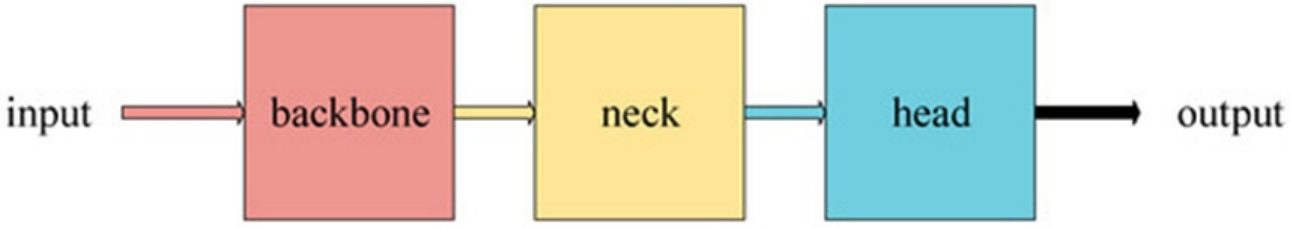

Figure 1. Overall structure of the object detection model.

Backbone network

CSPDarkNet [35] serves as the backbone for the UAV swarm detection model, primarily composed of convolutional layers and CSP structures, detailed in **Table 2**. Initially, a 640 × 640 RGB three-channel input image is fed into the network, where the Focus module modifies its dimensions and channel count. Subsequently, four stacked ResBlock body modules extract features. Within the final Resblock body module, the SPP module applies max pooling with varying kernel sizes to expand the network's receptive field. The CSPDarkNet ultimately produces feature maps from the 2nd, 3rd, and 4th Resblock body modules, sized 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	

© 2025 The Author(s). | Published by Agro Intelligence Research Press

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

In the ResBlock body module, the CSPLayer resembles a residual design. The input undergoes multiple 1×1 and 3×3 convolutional layers for n iterations, after which the outcome is concatenated with the original input to form the output, as depicted in **Figure 2a**. To better support UAV swarm target detection, an adaptation from the MobileNet V3 model [29] incorporates a Squeeze-and-Extraction (SE) module [6] into the CSPLayer, shown in **Figure 2b**. The UAV detection model uses the SiLU activation function. SiLU features an unbounded upper limit, a lower bound, and smooth non-monotonic behavior. It enables quicker convergence in training, with the following formula:

$$SiLU\,(x) = x \cdot sigmoid\,(x)\,, \quad (1)$$

© 2025 The Author(s). | Published by Agro Intelligence Research Press

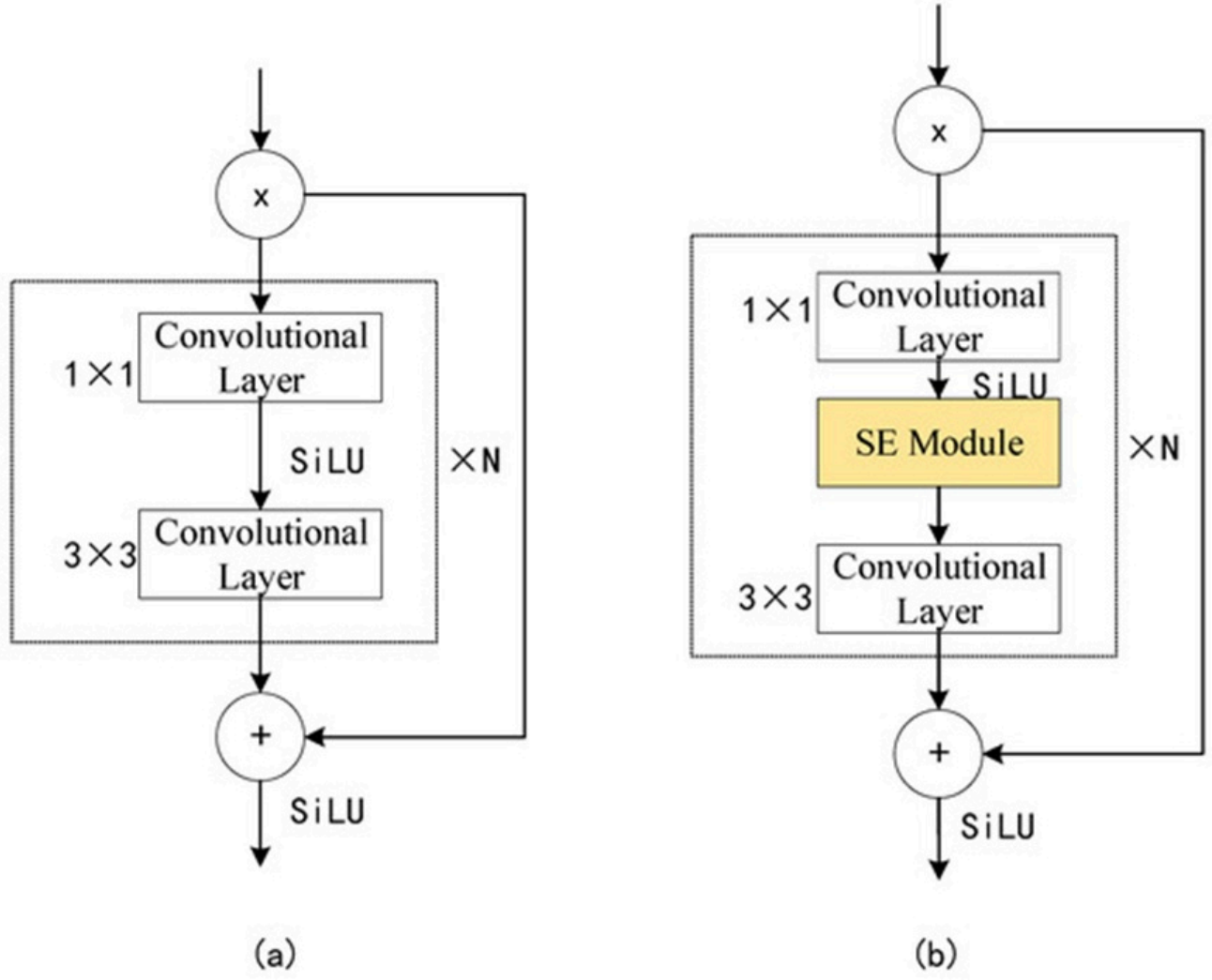

Figure 2. (a) Original CSPLayer structure, and (b) CSPLayer structure with SE module.

More precisely, the SE module assigns varying weight coefficients to each channel based on inter-channel correlations. These weights are multiplied by the prior features and added to the original features to enhance them. As illustrated in **Figure 3**, the SE attention process unfolds as follows: The extracted feature $X \in \mathbb{R}^{H' \times W' \times C'}$ is first transformed into $U \in \mathbb{R}^{H \times W \times C}$ via the function Ftr. Global average pooling (·) then summarizes each channel's global information into a descriptor. This descriptor is refined through (·,) for adaptive calibration to improve weight accuracy. Ultimately, the enhanced feature results from scaling the original feature with the computed weights via $F(\cdot, \cdot)$.

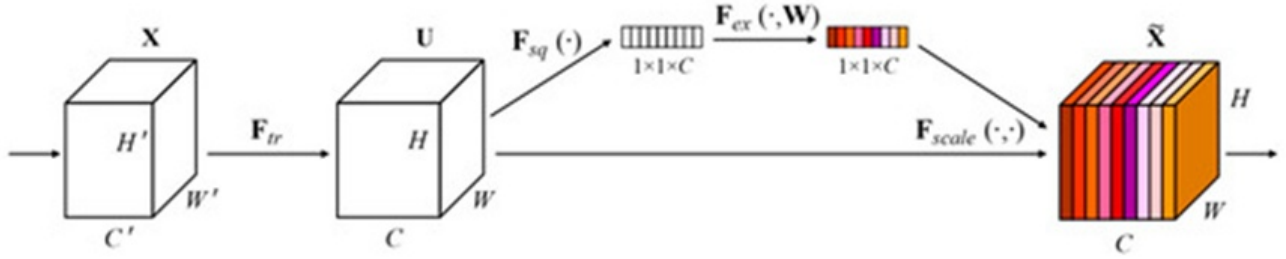

Figure 3. Diagrammatic sketch of the SE module.

© 2025 The Author(s). | Published by Agro Intelligence Research Press

Neck network

The three feature layers from CSPDarkNet are forwarded to the neck for further refinement and integration. The neck in the UAV swarm detection model builds upon the Path Aggregation Network (PANet) [36]. Input feature maps are resized via convolution before being fused via upsampling and downsampling. The detailed architecture appears in **Figure 4**.

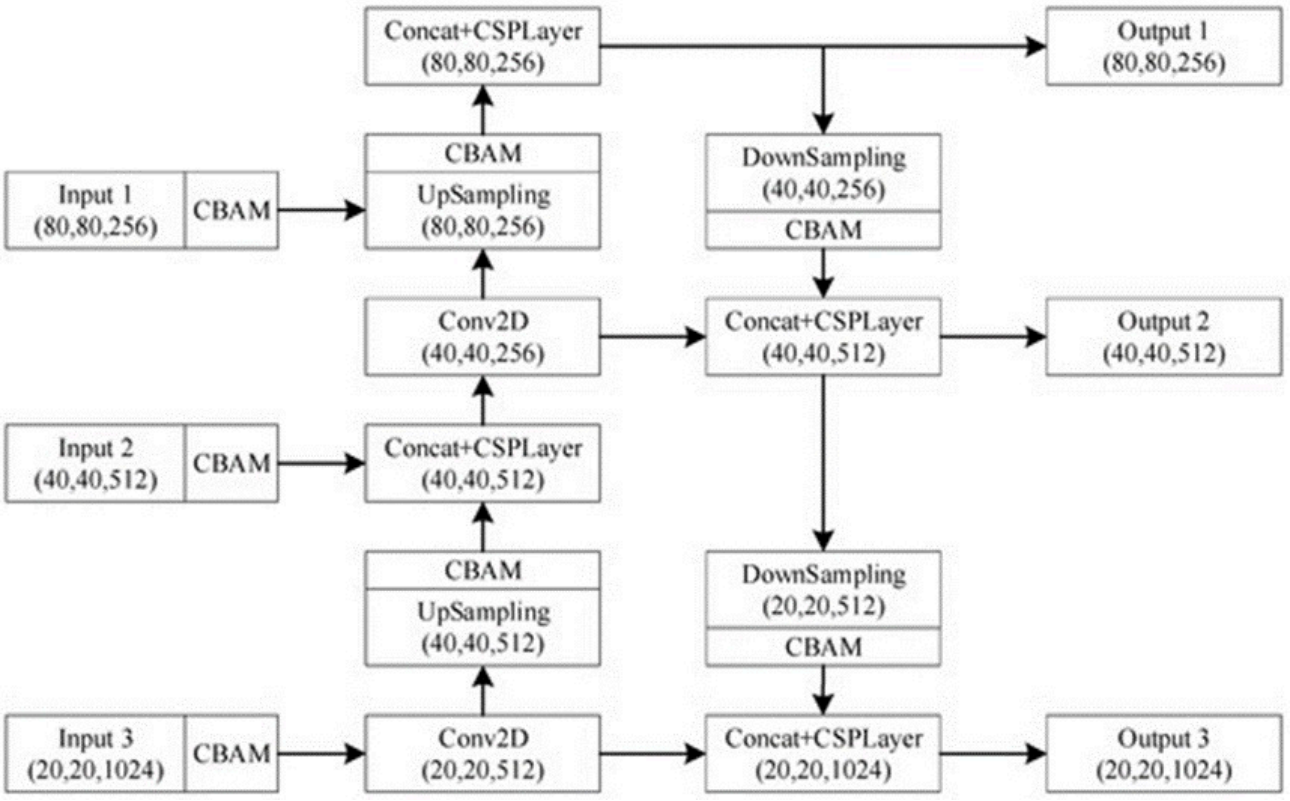

Figure 4. The structure of the neck network.

To elevate the model's UAV detection effectiveness, a Convolutional Block Attention Module (CBAM) [7] is first integrated with the three backbone-derived feature maps before neck input. CBAM is also applied after each upsampling and downsampling step in the PANet. CBAM functions as a straightforward yet powerful attention mechanism for convolutional networks, merging channel and spatial dimensions. Upon receiving a feature map, it sequentially applies the Channel Attention Module (CAM) followed by the Spatial Attention Module (SAM). The computation is given by **Eq. 2**:

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

As presented in **Figure 5**, within the CAM, an input feature map $F \in \mathbb{R}C \times H \times W$ first receives Global Average Pooling (GAP) and Global Maximum Pooling (GMP) across its spatial dimensions. These are fed into a shared Multilayer Perceptron (MLP), summed, and passed through a Sigmoid to generate a one-dimensional channel attention vector $Mc \in \mathbb{R}C \times 1 \times 1$. This vector is then multiplied by F to produce the channel-weighted map Fc. For the SAM, the CAM output Fc undergoes pooling, followed by a 7 × 7 convolution and Sigmoid activation to create a two-dimensional spatial attention vector $Ms \in \mathbb{R}1 \times H \times W$. Multiplication of Fc and Ms yields the final feature map Fa.

© 2025 The Author(s). | Published by Agro Intelligence Research Press

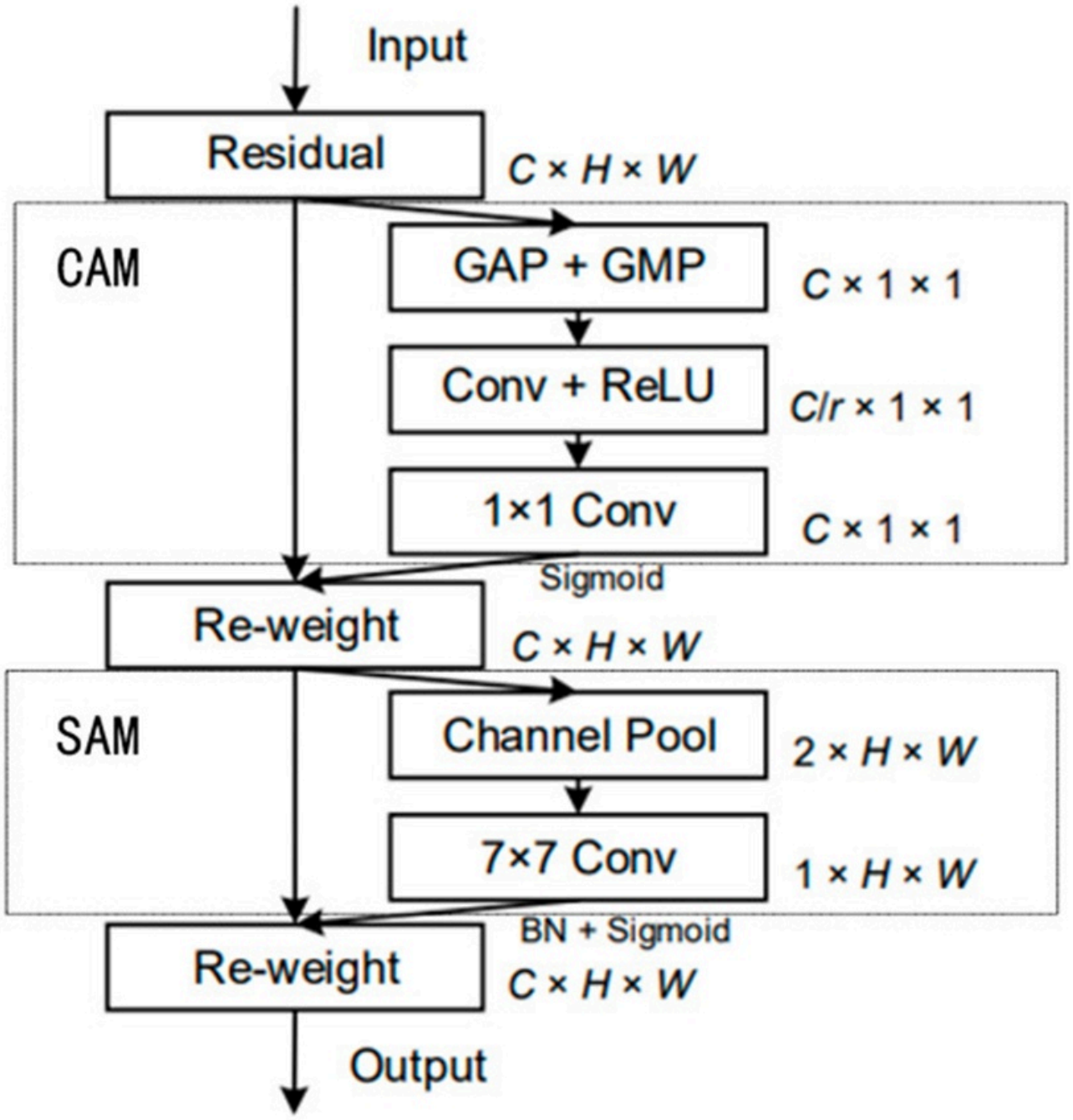

Figure 5. Diagrammatic sketch of the CBAM.

Head network

Once feature fusion and enhancement are complete, the head network performs classification and regression on the multi-scale feature layers, producing the final detection outputs. Its structure is displayed in **Figure 6**. The head of this UAV detection model comprises two convolutional branches [35]: one dedicated to object classification and category prediction, and the other to objectness scoring and bounding-box coordinate regression. Consequently, each feature layer generates three types of predictions:

© 2025 The Author(s). | Published by Agro Intelligence Research Press

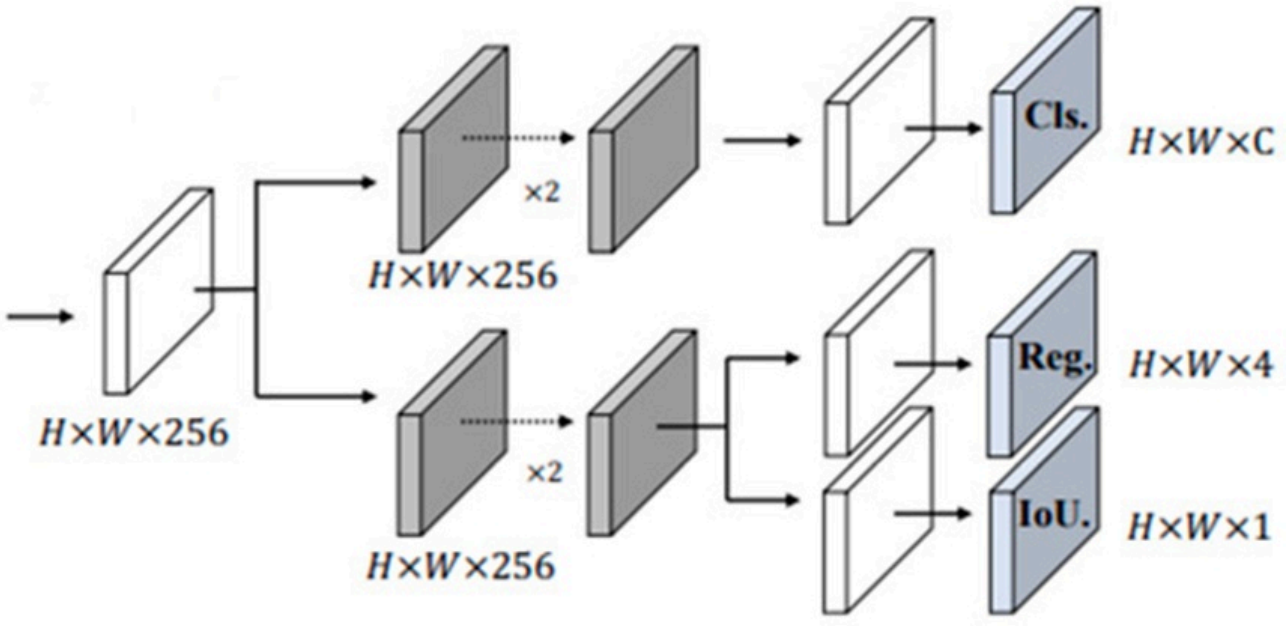

Figure 6. The structure of the head network.

(1) Reg (h, w, 4): Predicts target location details. The four values represent x, y, w, and h, where x and y denote the center coordinates of the predicted box, and w and h indicate its width and height.

(2) Obj (h, w, 1): Determines whether the predicted box corresponds to a foreground or background object. After being processed through the Sigmoid function, it yields a confidence score for the presence of an object in each box. Values nearer to 1 indicate higher target probability.

(3) Cls (h, w, num_classes): Identifies the specific object category, assigning a score per class. These scores are converted to confidence levels via the sigmoid function.

The three prediction outputs are combined to produce a final prediction tensor per feature layer of (h, w, 4 + 1 + num_classes). The first four elements in the last dimension cover regression parameters for each point; the fifth assesses object presence; and the remaining num_classes elements specify the object category.

Lightweight model

The fundamental purpose of a lightweight model is to overcome the restrictions imposed by limited storage capacity and power usage, which hinder conventional neural networks when deployed on resource-constrained hardware. To address the high computational demands of standard deep convolutional neural networks, this study emphasizes strategies to reduce model complexity and computational requirements without sacrificing object detection precision. Since depthwise separable convolution [4] offers an efficient way to shrink model dimensions while preserving strong feature extraction performance, this work integrates it to streamline and enhance the UAV swarm detection network.

Standard convolution handles both input filtering and feature combination simultaneously in one operation. In contrast, depthwise separable convolution separates these tasks into two distinct layers. This separation substantially reduces network parameters and calculations, with only a minor impact on accuracy. Specifically, depth-separable convolution restructures the conventional process into depthwise convolution followed by pointwise convolution. Depthwise convolution applies an individual filter per input channel, while pointwise convolution employs a 1 × 1 kernel to merge the resulting outputs. Illustrations of standard convolution and depthwise separable convolution appear in **Figures 7a and 7b**,

© 2025 The Author(s). | Published by Agro Intelligence Research Press

respectively. This paper replaces traditional convolutions with depthwise separable ones throughout the UAV detection network to lower both parameter count and processing load.

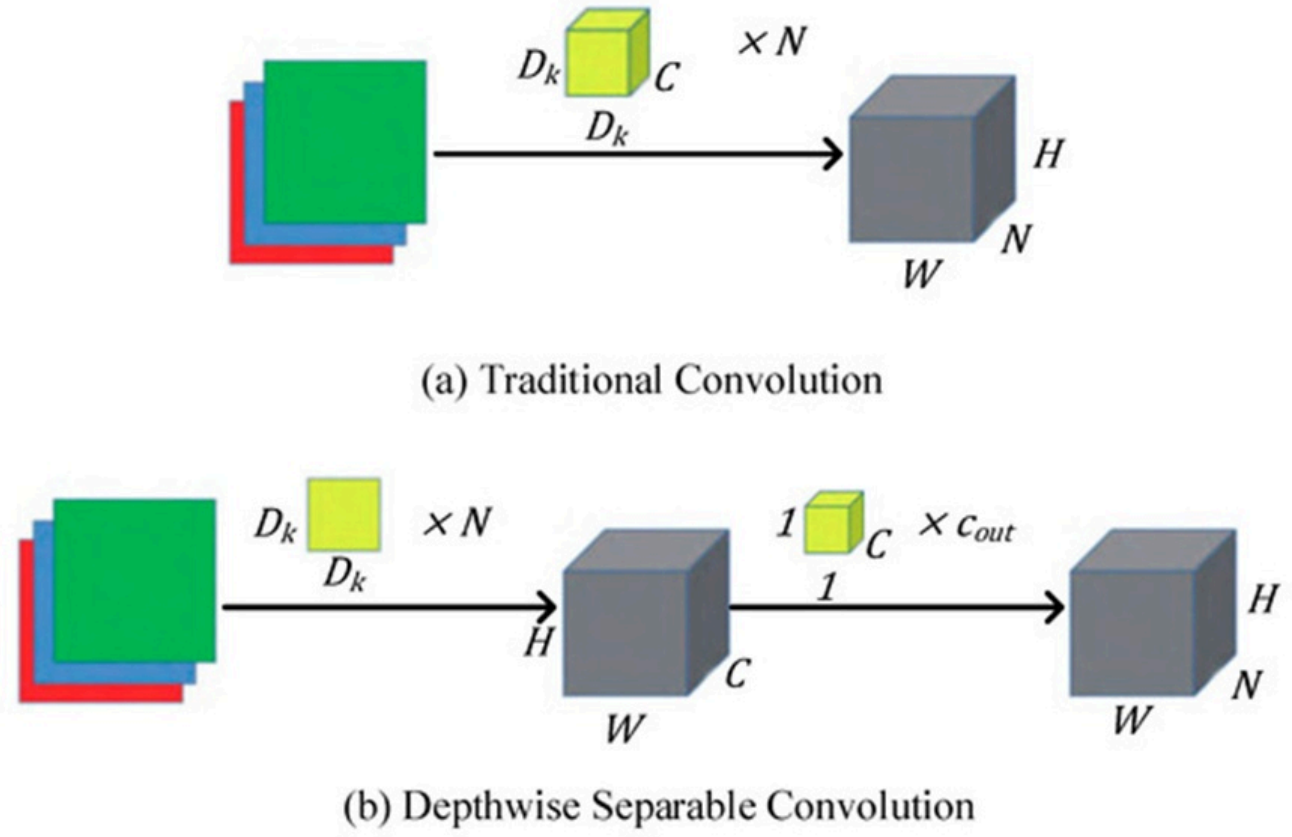

Figure 7. Comparison of the two convolution methods.

Analysis of total convolution kernel parameters and overall computational operations helps quantify the inner product workload. Suppose N sets of convolutions sharing kernel dimensions $Dk \times Dk \times C$ are performed on the input image, producing a feature map of size $W \times H \times N$. In this case, the parameter counts and operation volumes for both approaches are detailed in **Table 3**. Depthwise separable convolution effectively reduces network scale and computational cost. By decomposing the convolution into separate filtering and combining stages, using kernels of equal width and height to create fixed-size feature maps, the required computations decrease by a factor of $1/N + N+1/Dk2$. This reduction forms the foundation for achieving a lightweight architecture.

Table 3. Comparison of the two convolution methods.

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

Model training

During training of the UAV swarm detection model, Distance-IoU (DIoU) [8] replaces Intersection over Union (IoU) for the regression loss calculation, facilitating improved model optimization. Moreover, Mosaic [9] and Mixup [10] data

© 2025 The Author(s). | Published by Agro Intelligence Research Press

augmentation methods are employed to enlarge the dataset and boost UAV detection performance.

Data augmentation

In computer vision, data augmentation acts as a technique to broaden the available training dataset. It modifies existing images to address limited sample quantities and thereby increases overall training data volume. The UAVSwarm dataset used in this experiment has a small number of training instances and highly similar scenes. Therefore, Mosaic and Mixup algorithms are integrated into the image preprocessing stage. These techniques increase sample diversity and enrich target backgrounds, helping prevent overfitting during training while strengthening the model's detection accuracy and generalization.

Figure 8 demonstrates the enhancement produced by the Mosaic algorithm. Initially, four separate images undergo random cropping, scaling, and rotation, then are combined into a single unified image that serves as the training input. Importantly, bounding box coordinate data from the original images is preserved, ensuring the new composite image retains accurate UAV bounding box locations. This delivers dual benefits: object scales are reduced to better support small-target detection requirements, and background diversity is increased, granting the UAV swarm detection model greater resilience in intricate environments.



Figure 8. The enhancement effect of Mosaic.

Originally developed for image classification, the Mixup algorithm randomly picks two images from a batch and blends them at a specific ratio to produce novel training samples. Compared with Mosaic, Mixup is far more efficient, involving negligible extra computation while markedly accelerating model execution. Its mathematical formulation is given by:

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

Where (xi,yi) and (xj,yj) denote two randomly picked samples along with their labels, $(\tilde{x},\tilde{y})$ represent the resulting mixed samples and labels used for network training, and λ indicates the mixing coefficient. **Eq. 3** shows that Mixup linearly combines samples via this coefficient. The visual outcome of the Mixup algorithm is illustrated in **Figure 9**.

© 2025 The Author(s). | Published by Agro Intelligence Research Press



Figure 9. The enhancement effect of Mixup.

Loss function

Network training aims to minimize the loss function so that predicted boxes more closely align with ground-truth boxes, yielding a stronger, more reliable model. For object detection, the loss function must capture the spatial proximity between predicted and actual boxes, confirm target presence within predictions, and verify object category correctness. As generated by the detection head, the overall loss comprises three distinct components, defined in **Eq. 4**:

$$Loss = Loss_{Reg} + Loss_{Obj} + Loss_{Cls}\ , \quad (4)$$

(1) Regression loss ($LossReg$) quantifies positional discrepancies between the predicted box and ground truth. The model's predicted x, y, w, and h values define the box's location, and loss is derived from the DIoU metric comparing the prediction and ground truth. **Figure 10** presents the working principle of DIoU-based regression loss, and the associated equation is **Eq. 5**:

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

Where b stands for the center coordinates of the predicted box and bgt for those of the ground truth; d measures the Euclidean distance between these centers; and c denotes the diagonal length of the smallest rectangle enclosing both boxes. While IoU calculates the overlap proportion, it returns zero whenever boxes do not intersect. Additionally, when one box lies entirely within another without size change, IoU remains constant, complicating optimization. DIoU resolves these shortcomings and delivers a more effective evaluation for regression.

© 2025 The Author(s). | Published by Agro Intelligence Research Press

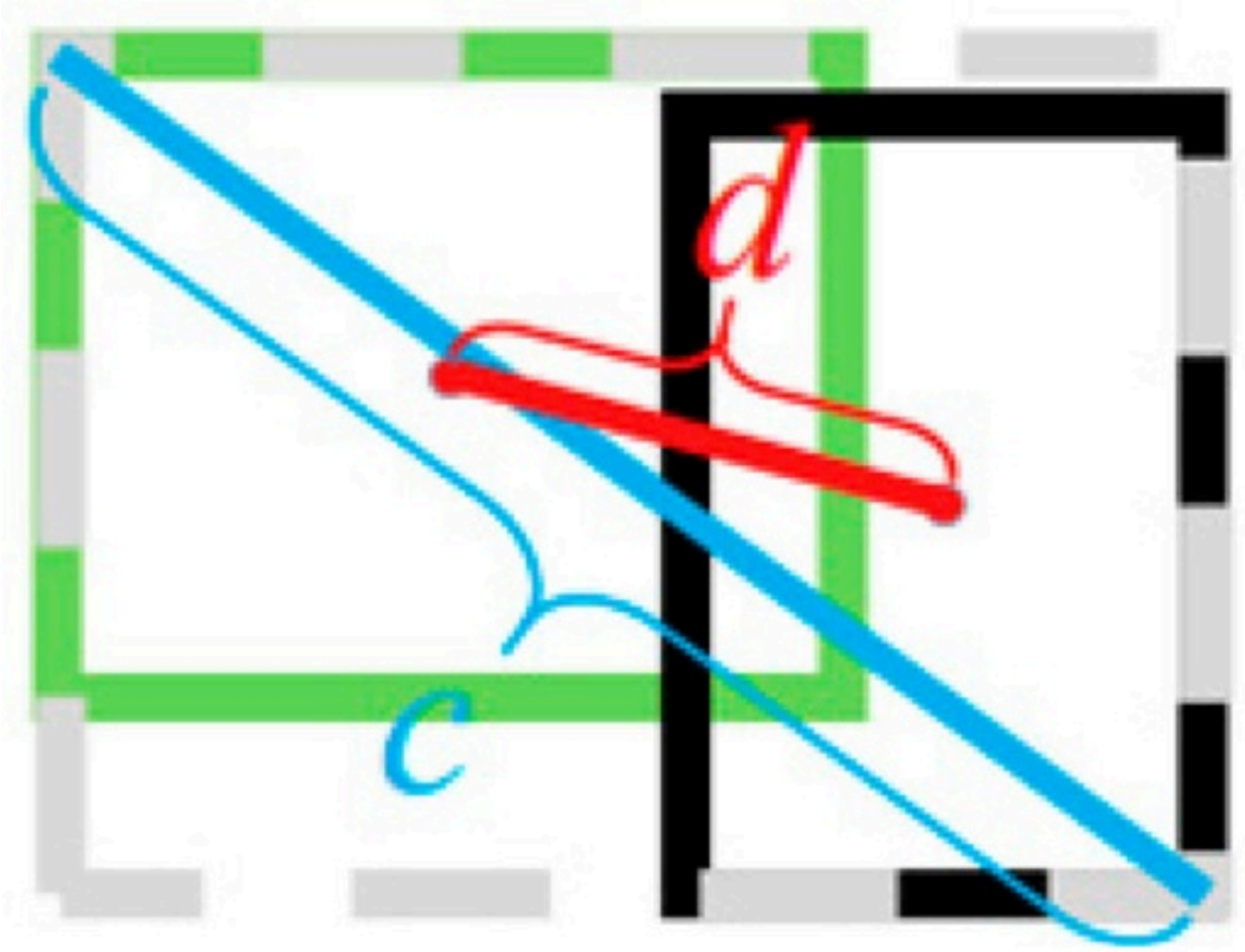

Figure 10. Schematic diagram of DIoU.

(2) Object loss ($LossObj$) addresses the binary question of whether an object occupies the predicted box. Based on the head network outputs, it distinguishes between target presence and absence. All feature points aligned with ground truths count as positive samples, and the rest as negative. Binary Cross-Entropy loss is then computed from these positive/negative predictions.

(3) Classification loss ($LossCls$) measures inaccuracies in category assignment. Feature points from model predictions have their category outputs extracted, after which the Binary Cross-Entropy loss is derived by comparing them against the true ground truth categories.

Results and Discussion

The suggested UAV swarm target-detection model was developed on the PyTorch deep learning platform. Input images are resized to 640×640, with a batch size of 12 images during training. A total of 100 epochs were executed without any pre-trained weights. Mosaic and Mixup augmentation techniques were active for the initial 70 epochs and then deactivated for the remaining 30 epochs. Stochastic Gradient Descent (SGD) was used as the optimizer, with a learning rate of 0.01. Performance was evaluated using mean Average Precision (mAP), parameter quantity, model size, latency, and Frame Per Second (FPS). Experiments were conducted on a system with dual Intel Xeon E5 2.40 GHz CPUs, one NVIDIA GTX 1080 Ti GPU, and 32 GB of RAM.

© 2025 The Author(s). | Published by Agro Intelligence Research Press

UAVSwarm dataset

Wang *et al.* [5] assembled 72 UAV image sequences and manually labeled them, producing a new UAV swarm dataset, UAVSwarm, designed for multi-object UAV detection and tracking tasks. The dataset contains 12,598 images, with a maximum UAV count per image of 23. It splits into 36 training sequences (6844 images) and 36 test sequences (5754 images). The version employed here mostly filters out non-UAV items (such as bird groups), yet several scenes involve complex scenes that result in UAV occlusions. **Figure 11** displays example images and corresponding annotation details from the dataset.

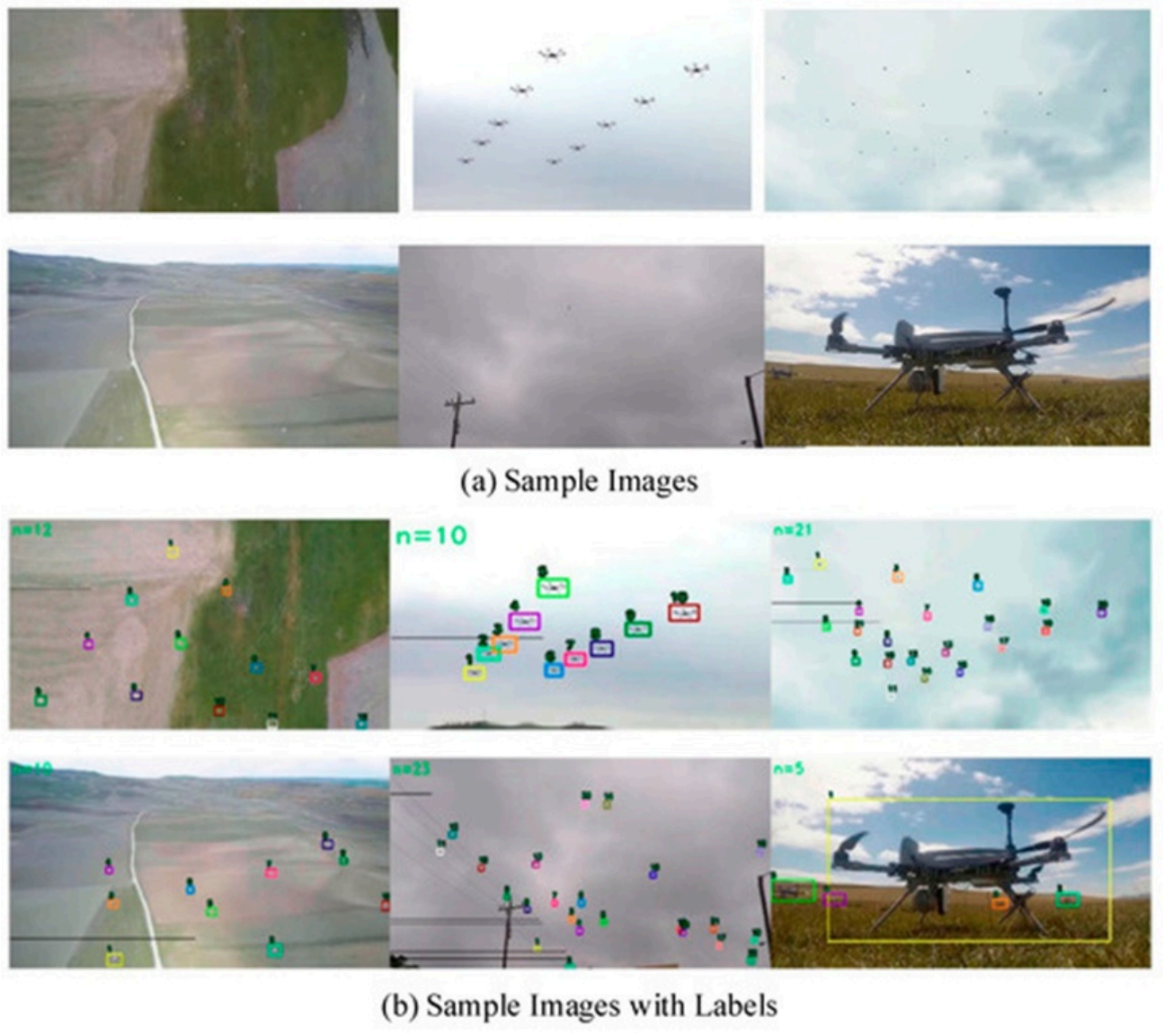

Figure 11. Samples of the UAVSwarm dataset.

Ablation experiment

Initially, the YOLOX framework was simplified and refined using depthwise separable convolution, yielding a nano-level network. To assess the lightweight module's value, this research trained three scaled-down YOLOX versions—YOLOX-S, YOLOX-Tiny, and YOLOX-Nano—under uniform training conditions. It evaluated them on their respective test sets to compare outcomes. **Table 4** indicates that the UAV detection precision of these three YOLOX variants on the test data

© 2025 The Author(s). | Published by Agro Intelligence Research Press

exceeds 80%. In terms of precision versus scale within the YOLOX family, findings follow the standard trend in object detection systems: greater layer counts and parameter volumes yield enhanced feature capture and generalization, resulting in higher recognition rates. Setting the DIoU threshold to 0.5 yields nearly identical mAP scores across the three networks, despite wide variations in model dimensions and total parameter counts. Specifically, the nano network's size and parameter total equal roughly 1/10th of the S version. This illustrates that, given identical hardware, the compact nano version processes more input images and reduces costs while maintaining the required precision. Hence, YOLOX-Nano served as the reference model here.

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

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

Table 5 indicates that the suggested model achieves 82.32% mAP on the UAVSwarm test data, marking roughly a 2% gain over the reference, while requiring only about 40 Kb more in parameters and overall size. Furthermore, adding SE and CBAM components and adjusting the loss functions improved mAP relative to the reference, confirming the benefit of these three components.

Table 5. Ablation experiment based on YOLOX-Nano.

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

Loss progression graphs for multiple network variants across training are shown in Figure 12, with Epoch on the horizontal axis and Loss on the vertical axis. Observations show comparable convergence patterns across the YOLOX family of models. Training loss falls steeply at first, then reduces steadily and levels off as epochs increase. In the end, the suggested network reaches the smallest loss, validating the suitability and impact of the training setup and settings for elevating detection performance.

© 2025 The Author(s). | Published by Agro Intelligence Research Press

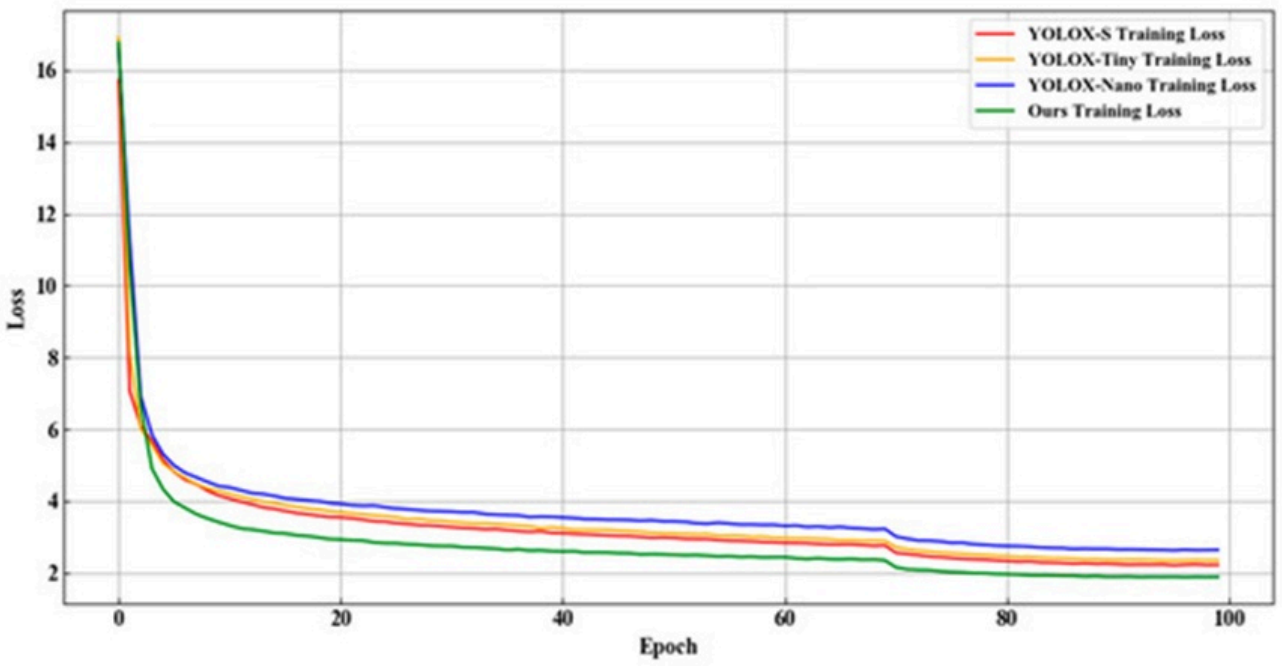

Figure 12. Loss curves of the different networks.

Regarding the UAVSwarm collection, featuring UAVs as classic small targets, the network outlined here boosted the Precision and Recall of the reference model by 0.63% and 1.84%, respectively, at the 0.5 threshold, as shown in **Figure 13**. This Recall rise implies the chosen refinements successfully amplify the model's handling of positive foreground targets.

© 2025 The Author(s). | Published by Agro Intelligence Research Press

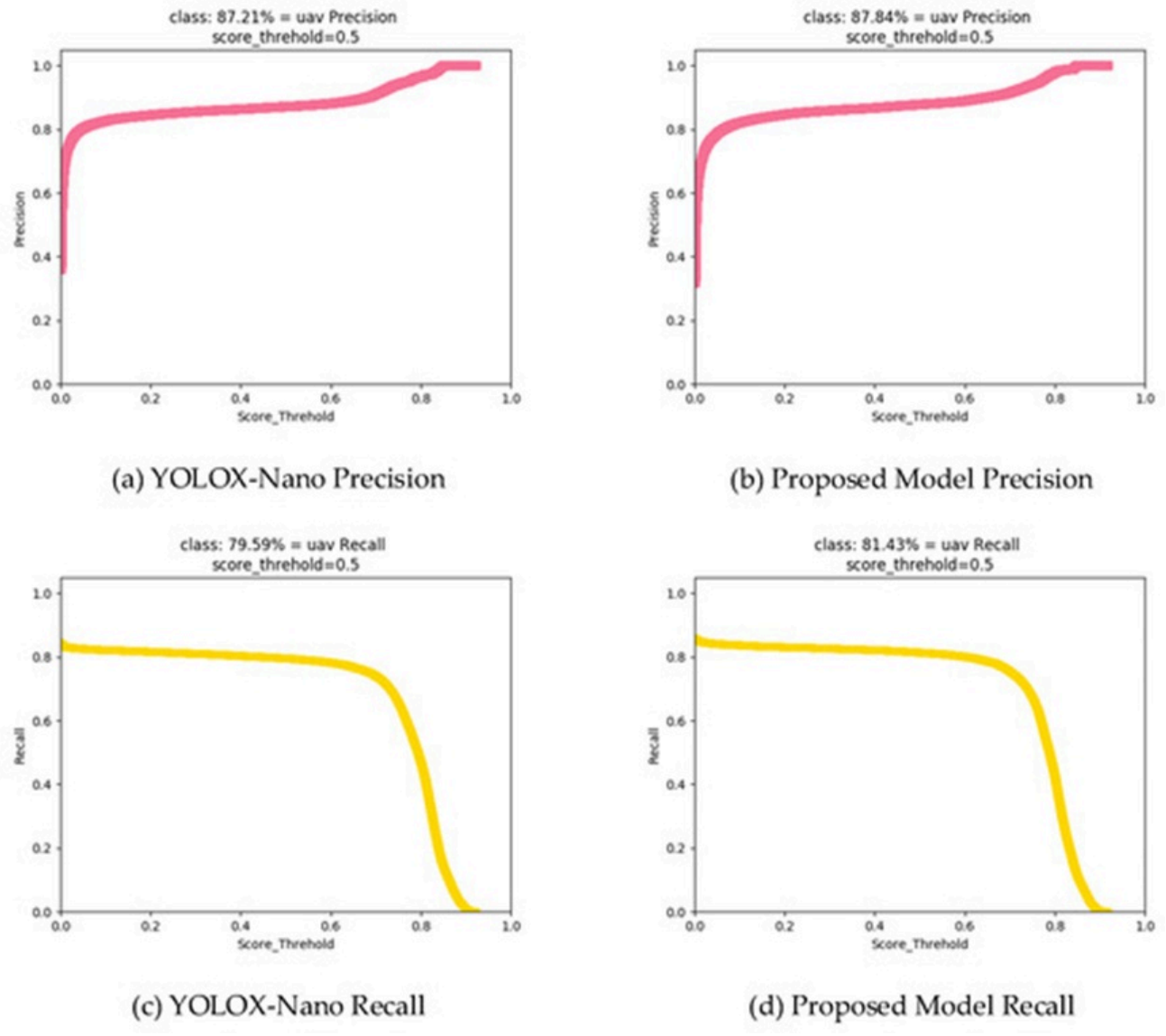

Figure 13. Precision and Recall of UAV detection.

Comparison experiment

To provide a balanced view of the UAV swarm detection network in this work, other compact YOLO architectures were trained with equivalent hyperparameters and underwent side-by-side review. Outcomes feature in **Table 6**. Under the shared test set, the introduced UAV detection approach hits 82.32% mAP, exceeding MobileNetv3-yolov4 by 15.59%, GhostNet-YOLOv4 by 15.41%, YOLOv4-Tiny by 1.78%, YOLOX-Tiny by 0.58%, and YOLOX-Nano by 1.82%. At once, it holds the best totals for network parameters and model dimensions.

Table 6. Comparison experiment of different networks.

Method	FPS	Latency/ms	Model size/Mb	Params ($\times 10^6$)	mAP@0.5 (%)
GhostNet-YOLOv4	12	83	42.40	11.00	66.91
MobileNetv3-YOLOv4	11	90	53.70	11.30	66.73

© 2025 The Author(s). | Published by Agro Intelligence Research Press

YOLOv4-Tiny	22	45	22.40	5.87	80.54
YOLOX-Tiny	18	55	13.70	5.03	81.74
YOLOX-Nano	14	71	3.70	0.89	80.50
Ours	14	71	3.85	0.93	82.32

Processing duration findings from tests are listed in **Table 6**. Owing to hardware constraints, results fell short of those reported in the source paper, but with greater resources, the method should yield shorter delays and quicker runs. **Table 6** shows that the new model achieves better identification rates than the YOLOX-Nano reference while matching inference durations. Therefore, it meets real-time needs and supports low-delay applications. Still, YOLOX-Tiny and YOLOv4-Tiny achieve faster rates at 18 FPS and 22 FPS. Despite lagging these, the design delivers clearly superior accuracy. Thus, it stands as a compact model offering strong precision ideal for multiple edge platforms.

To examine the real-world performance of this UAV detection approach, a subset of dataset images was tested, and the detection outcomes are shown in **Figure 14**. Per **Figure 14a**, simple settings allow most systems to spot UAVs, but the new technique identifies additional small or remote ones beyond the YOLOX-Nano reference. In **Figure 14b**, amid busy backgrounds or crowded UAVs, alternatives miss many detections, whereas this algorithm excels, correctly spotting UAVs even with overlaps. **Figure 14c** shows that at reduced resolutions, it still pinpoints UAVs in challenging settings with high confidence. Comparisons affirm gains in accuracy and reliability, showing it balances compactness with strong performance for practical deployment.

© 2025 The Author(s). | Published by Agro Intelligence Research Press

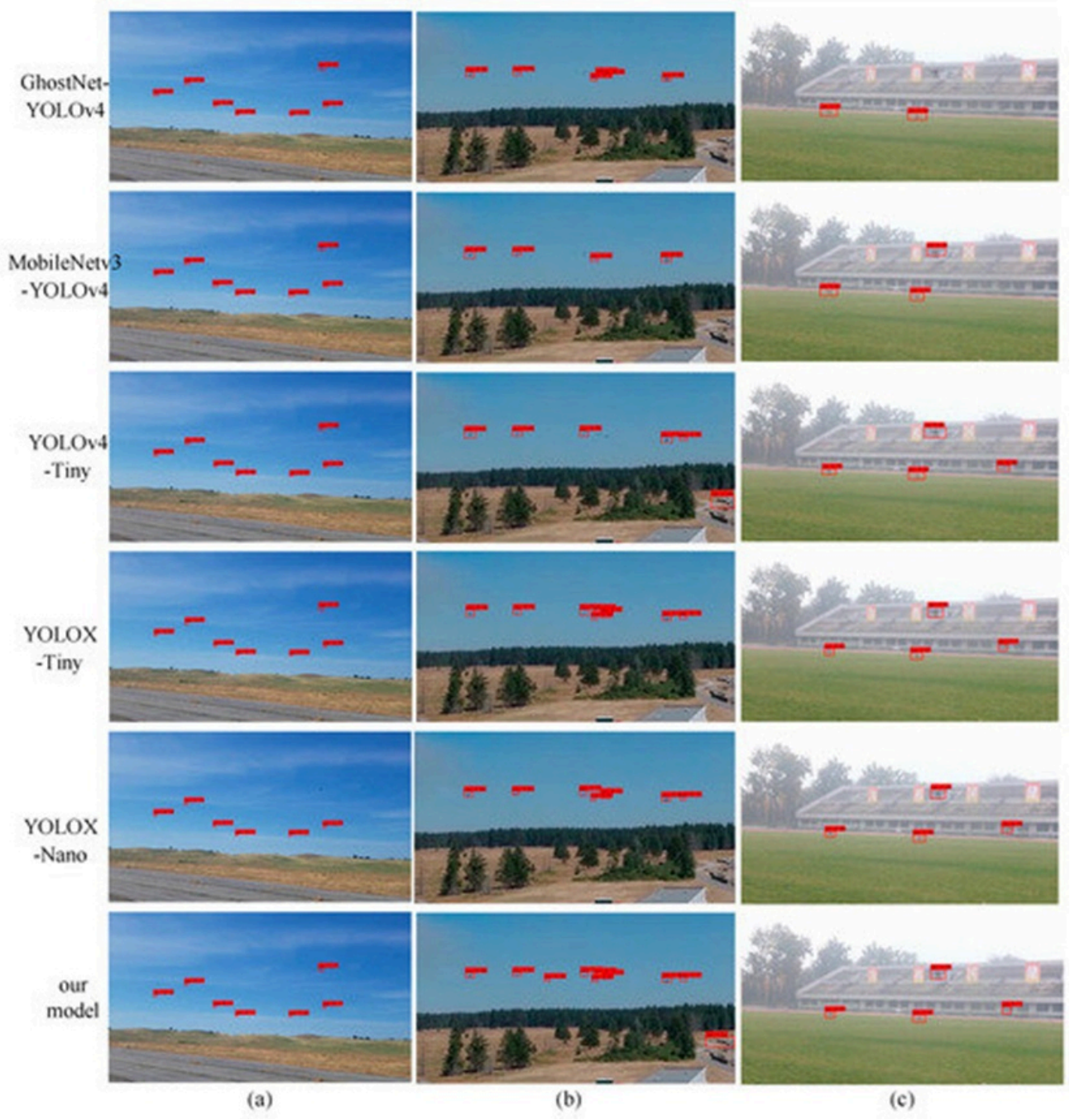

Figure 14. Comparison of detection results of different models.

Aside from anti-UAV setups, this paper's algorithm is well-suited for swarm intelligence applications in UAV group shaping, joint operations among multiple UAVs, and related activities. Lightweight detection enables swarm members to swiftly and precisely learn peer locations and states, enabling prompt adaptations via intelligence rules. Coordinated groups can deliver greater reach and improved efficiency for multi-unit tasks. Moreover, the technique should markedly improve UAV swarm metrics, including total power consumption, average end-to-end latency, packet success rate, and data throughput. Evaluations suggest this lightweight model may reduce energy consumption and transmission delays in swarms while increasing packet rates and throughput. Consequently, the system could gain superior data handling at lower expense, excelling in assignments.

© 2025 The Author(s). | Published by Agro Intelligence Research Press

Conclusion

This study presents a lightweight UAV swarm detection framework that incorporates an attention mechanism. To begin with, the network architecture is streamlined and optimized by adopting depthwise separable convolutions, significantly reducing the overall number of network parameters. Subsequently, an SE module is embedded into the backbone network to enhance the model's ability to extract object features. In addition, integrating CBAM into the feature fusion network enables the model to emphasize critical features while reducing the influence of redundant information. During the training stage, a DIoU-based loss function is employed, which better captures overlap information and improves both speed and regression precision. Furthermore, two data augmentation techniques are applied to enlarge the UAVSwarm dataset, thereby enhancing UAV detection performance.

The proposed approach achieves high detection accuracy while maintaining a lightweight structure of only 3.85 MB, making it highly suitable for embedded platforms and mobile devices. Overall, the UAV swarm detection model developed in this work meets the real-time and accuracy requirements for rapid UAV detection in practical environments, demonstrating significant value for the development of anti-UAV systems. In future research, further efforts will focus on refining and optimizing the proposed improvement strategy to achieve superior recognition accuracy and faster real-time performance while keeping model complexity as low as possible.

Acknowledgements

None

Conflict of interest

None

Financial support

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).

Ethics statement

None

Received: 04 Nov 2023 Revised: 09 Jan 2024 Accepted: 25 Jan 2024
Published online: 25 November 2025

Rights and permissions

© 2025 The Author(s). | Published by Agro Intelligence Research Press

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at https://creativecommons.org/licenses/by-nc-sa/4.0/.

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.

2. Zhang D, Cheng Y, Lin Q, Yu GF, Xiao LW. Key technologies and development trend of UAV swarm operation. China New Telecommun. 2022;24:56-8.

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, et al. 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.

6. Hu J, Shen L, Albanie S, Sun G, Wu E. Squeeze-and-excitation networks. IEEE Trans Pattern Anal Mach Intell. 2020;42:2011-23.

7. Woo S, Park J, Lee JY, Kweon I. CBAM: Convolutional block attention module. In: European Conference on Computer Vision; Cham, Switzerland; 2018. p. 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; 2020. p. 12993-3000.

9. Bochkovskiy A, Wang CY, Liao H. YOLOv4: Optimal speed and accuracy of object detection. arXiv. 2020;arXiv:2004.10934.

10. Zhang H, Cisse M, Dauphin YN, David LP. 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 FZ, Kerrache CA, Djellikh S, Lagraa N, Lahby M, et al. COCOMA: A resource-optimized cooperative UAVs communication protocol for surveillance and monitoring applications. Wirel Netw. 2022:1-17.

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.

14. Sastre C, Wubben J, Calafate CT, Cano JC, Manzoni P. Safe and efficient take-off of VTOL UAV swarms. Electronics. 2022;11:1128.

15. Sastre C, Wubben J, Calafate CT, Cano JC, 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; 2022. p. 477-82.

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; 2014. p. 580-7.

17. Girshick R. Fast R-CNN. In: Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV); Santiago, Chile; 2015. p. 1440-8.

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;39:1137-49.

19. Liu W, Anguelov D, Erhan D, Szegedy C, Cheng Y, Alexander CB. SSD: Single shot MultiBox detector. In: European Conference on Computer Vision; Cham, Switzerland; 2016. p. 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; 2017. p. 2980-8.

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; 2019. p. 8386-90.

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;8:130697-707.

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;42:2885-92.

24. Yavariabdi A, Kusetogullari H, Celik T, Cicek H. FastUAV-NET: A multi-UAV detection algorithm for embedded platforms. Electronics. 2021;10:724.

25. Liu B, Luo H. An improved Yolov5 for multi-rotor UAV detection. Electronics. 2022;11:2330.

26. Wang C, Shi Z, Meng L, Wang J, Wang T, Gao Q, et al. Anti-occlusion UAV tracking algorithm with a low-altitude complex background by integrating attention mechanism. Drones. 2022;6:149.

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;41:31-6.

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; 2018. p. 4510-20.

29. Howard A, Sandler M, Chu G, Chen LC, Chen B, Tan M, et al. Searching for MobileNetV3. In: Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV); Seoul, Republic of Korea; 2019. p. 1314-24.

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; 2018. p. 6848-56.

31. Ma N, Zhang X, Zheng HT, Sun J. ShuffleNet V2: Practical guidelines for efficient CNN architecture design. In: European Conference on Computer Vision; Cham, Switzerland; 2018. p. 122-38.

32. Tan M, Chen B, Pang R, Vasudevan V, Vasudevan V, Sandler M, et al. 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; 2019. p. 2815-23.

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; 2020. p. 1577-86.

© 2025 The Author(s). | Published by Agro Intelligence Research Press

34. Xiong Y, Liu H, Gupta S, Akin B, Bender G, Wang Y, et al. 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; 2021. p. 3824-33.

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; 2018. p. 8759-68.

© 2025 The Author(s). | Published by Agro Intelligence Research Press