2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

LightDrone-YOLO: A Novel Lightweight and Efficient Object Detection Network for Unmanned Aerial Vehicles

Xin Li[1,2†] [0009-0002-2142-2301] , Tianze Zhang[2,3†] [0009-0007-7788-1626], Yifan Lyu[1,2] , Zhixuan Miao[1,2] [0009-0006-4721-6150] and Gang Shi[1,2] [0009-0000-2488-147X] (✉)

[1] College of Computer Science and Technology, Xinjiang University, Urumqi 830046, China
[2] Xinjiang Key Laboratory of Signal Detection and Processing, Urumqi 830046, China
[3] Faculty of Science, The University of Melbourne, Melbourne, VIC 3010, Australia
shigang@xju.edu.cn
†These authors contributed equally to this work.

Abstract. In recent years, the application of unmanned aerial vehicles (UAVs) has grown exponentially in various fields due to their convenience. These vehicles have become ubiquitous in numerous fields, including environmental monitoring, agricultural management, urban planning, traffic monitoring, and emergency rescue, playing an instrumental role in these domains. However, target detection from the perspective of a drone is fraught with challenges. These challenges include the difficulty of detecting small targets, interference from lighting and background in complex scenes, and limited hardware resources. To address these challenges, we have enhanced the YOLOv8 model and introduced a lightweight and efficient target detection model specifically designed for the perspective of a drone, named LightDrone-YOLO. Firstly, a specialised layer is incorporated into the model for the purpose of enhancing detection of small targets. Secondly, a lightweight multi-scale feature fusion neck (LMFF-Neck) is designed to reduce the number of parameters and computational complexity of the model and improve the fusion of multi-scale features. Thirdly, we improved the C2f module and renamed it C2f-MFEM, which is designed to enhance feature extraction. Finally, the spatial feature weighting fusion (SFWF) module was designed to accurately select the most valuable spatial information during the multi-scale feature fusion process. Experimental results on the Visdrone 2021 dataset demonstrate the effectiveness of the proposed method, and the mean accuracy (mAP) is substantially improved. In the validation and test datasets, the proposed method demonstrated superiority over other prevalent lightweight models, with mAP50 reaching 40.8% and 32.5%.

Keywords: YOLOv8, Feature fusion, Aerial Images, Object detection.

1 INTRODUCTION

Object detection has become a fundamental component of computer vision, playing a crucial role in a multitude of applications [2] , including autonomous driving, surveillance systems, and aerial photography with drones [3] .

Convolutional neural networks (CNNs) have driven a paradigm shift in the field of computer vision, with deep learning leading advancements in object detection. These models autonomously learn multi-level features, enabling intricate visual feature extraction without the need for manual algorithms. Two-stage networks, such as RCNNs [4] , generate candidate regions first, then classify and regress them. Single stage networks, including YOLO [5] and SSD [6] , perform end-to-end classification and regression directly, achieving faster detection. Selecting the most appropriate method for a given application requires a careful balancing of accuracy and real-time performance.

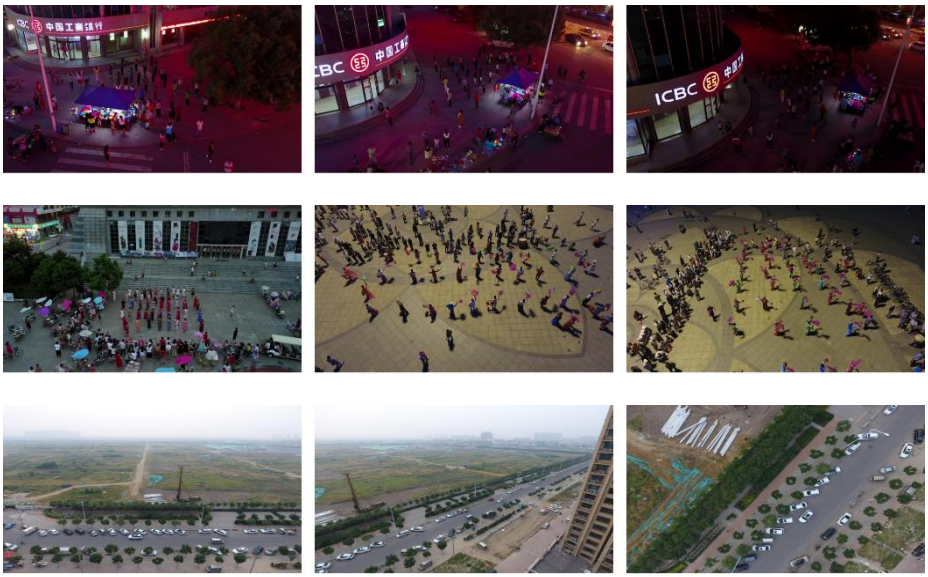

Fig. 1. Sample images taken from Visdrone2021 [1]. These images describes the main problems of object detection in UAV images.

In recent years, the use of drones (UAVs) has experienced significant growth in various fields due to their convenience, particularly in the domain of computer vision, where they offer distinct advantages in capturing large-scale high-resolution image and video data [7] .

However, target detection based on the perspective of UAVs still faces several challenges: (1) UAV-captured images inevitably contain a large number of small targets [8] , which are difficult to detect due to their limited area and susceptibility to noise interference; (2) the motion of UAVs causes image shaking, resulting in blurred targets and unclear textures [9] ; (3) the dynamic range of scenes captured by UAVs is susceptible to factors such as inadequate lighting and background interference, which exacerbates the difficulty of detection. (4) the limited hardware resources of UAVs pose a challenge in supporting target detection models with high computational and storage demands [10] .

To address these issues, we propose a lightweight multi-feature fusion network, LightDrone-YOLO. This network achieves a balance between speed and accuracy in

object detection through cross-scale feature fusion and an attention mechanism. It is designed to improve the object detection results of drone images. The contributions of this study are as follows:

1. We have proposed a lightweight multi-feature fusion network for drone images, LightDrone-YOLO, which is based on YOLOv8. The network embeds two newly designed modules based on the baseline model to improve the object detection performance.
2. We have designed a lightweight multi-scale feature fusion neck structure, which incorporates DySample [11] to enhance the connection between the object and the context. This part is specially optimized for unmanned aerial scenes, and a small target detection layer is introduced to improve the detection accuracy of small targets and reduce the number of parameters in the model.
3. The C2f-MFEM module is designed to enhance the backbone feature extraction effect by integrating the proposed Multifaceted Fusion Excitation Module with the C2f module.
4. In addressing the challenge of feature loss that may arise in cross-scale fusion, the Spatial Feature Weighted Fusion Detection Head has been developed. This innovation enhances detection accuracy by weighting and fusing the spatial features of neighboring layers to select the most effective spatial information.

2 RELATED WORK

2.1 High performance UAV image detection model

In recent years, the field of drone image target detection has witnessed significant advancements, largely driven by the rapid development of deep learning technology. In pursuit of optimal detection performance, numerous scholars have conducted extensive research to promote technological innovation.

One notable approach is the incorporation of the BiFPN [12] (Bi-directional Feature Pyramid Network) structure into a model combined with a small target detection layer, as seen in DMA-YOLO [13] . This refinement enhances the detection performance of small targets in drone images. However, it introduces a substantial increase in the complexity of the feature fusion component of the model neck, leading to a significant computational burden.

In addressing the challenge of target detection in blurred images, Li et al. [14] proposed the DREB-Net, a novel approach designed to mitigate motion blur in drone images. The network introduces a specialized component, the Blur Restoration Auxiliary Branch (BRAB), which enhances the detection of targets in blurry conditions by restoring critical image details. This design is specifically tailored to address the unique challenges posed by motion blur, offering an effective solution for enhancing detection performance in such images.

The CEASC model [15] introduces a context-enhanced sparse convolutional layer (CESC), a pioneering innovation that addresses the challenge of inadequate integration of contextual information in scenarios where sparse convolution is employed for the

processing of small objects. This model enhances the accuracy of detection through the utilization of global contextual features. However, it is important to note that global contextual information may not always be reliable or readily available. Additionally, while the CEASC model effectively reduces computational costs during inference through sparse convolutions, its complex network structure and large data volume still pose a significant challenge to computational resources during the training phase. In summary, while these methods have enhanced detection accuracy and processing efficiency, there are still issues such as over-weighting of the model and weak adaptability.

2.2 Development of lightweight UAV image detection model

Due to the limitations of the computing speed and memory capacity of unmanned aerial vehicle (UAV) processors, the deployment of most high-performance UAV image target detection on edge devices such as UAVs is not currently feasible. Many scholars have also studied this basis and designed lightweight UAV image detection algorithms.

LUD-YOLO [16] has designed a lightweight ASFF [17] module and greatly reduced the number of model parameters and computational complexity through pruning. While the model's speed is ensured, significant room for enhancement remains in terms of accuracy. Additionally, the pruning method employed by this model exhibits limited generalization capabilities, resulting in substantial variations in performance across different datasets. This challenge hinders the attainment of consistent and dependable results in practical applications.

The Drone-TOOD [18] model enhances the decomposition capability of the task by introducing ETDA, using the Classification Module for category prediction and the Location Module for location prediction. However, this improvement also introduces the complexity of parameter tuning, because the introduction of different network branches and loss functions makes the model tuning process more cumbersome.

The Drone-YOLO [19] model has been shown to achieve multi-scale information fusion through the design of a multilayer PAFPN structure and a sandwich module, leading to significant improvements in the spatial and semantic information of the target. This enhancement has notably improved the model's accuracy. However, due to the multiple feature fusions and up-and-down transmissions, it has increased the computational complexity of the model and may introduce redundant or irrelevant information. Consequently, while Drone-YOLO demonstrates efficacy in terms of accuracy, it incurs a substantial cost in terms of inference time and real-time performance.

The LODNU [20] model enhances the accuracy of multiscale object detection by incorporating an adaptive scale weighted feature fusion module (ASWFF). However, when the number of layers in the feature pyramid is excessive or when there is a considerable disparity between the layers, ASWFF may encounter challenges in effectively fusing all pertinent features. This vulnerability to disruption by outliers or noise is a salient limitation of the model.

The SCA-YOLO [21] model introduces the SCA module, a core module that employs a fusion of spatial and coordinate attention mechanisms, along with a small object detection layer. These enhancements have led to the remarkable performance of SCA-YOLO. However, the increased computational demands of the SCA module and the

small target detection layer have also led to a significant reduction in its detection speed, resulting in its inability to meet real-time requirements on certain devices with limited computing capabilities. In summary, while these methods have achieved substantial progress in enhancing detection performance, the majority of them continue to grapple with the trade-off between computational efficiency and real-time performance.

3 METHOD

In this section, we will provide a comprehensive description of the proposed methodology. The objective of our research is to explore the design of a lightweight and efficient UAV target detection network model. This model is primarily employed for UAV target detection, and thus, it is predominantly based on the characteristics of UAV images to enhance its performance. The primary aspects of the model are as follows:1) In order to enhance the detection performance of small targets, we incorporated shallow features, thereby enabling the optimization of small targets by shallow detection heads.2) Based on the characteristics of UAV images and our analysis, we designed a lightweight multi-scale feature fusion neck structure. This structure not only reduces the number of model parameters and computational complexity, but also improves the fusion effect of features at different scales. 3) The C2f module in the backbone was improved, and the feature extraction effect was enhanced by adding the MFEM module. 4) A spatial feature weighting fusion module was proposed to improve the ability of the detection head to screen important spatial information. As illustrated in Fig. 2, the modified model structure is depicted.

3.1 Small Object Detection Layer

Shallow feature maps have small receptive fields, limiting overlap and capturing fine-grained details, effective for small object detection. They also compensate for information loss during downsampling, preserving contextual information. Deep features, closer to the output layer, are rich in semantic information reflecting overall image characteristics but weak in capturing small object details due to low resolution.

To enhance object detection, optimizing convolutional network structures and feature map utilization across dimensions is crucial. This leverages shallow details and integrates deep semantic information for accurate detection.

The YOLOv8 model utilises its P3 layer for the detection of small objects measuring 80×80 pixels, with effective identification of objects exceeding 8×8 pixels. However, the model encounters challenges in accurately identifying objects smaller than 8 pixels.

To address this limitation, the proposed model is augmented by adding a P2 layer of 160×160 pixels, with the aim of enhancing the capture of complex details and optimising small object detection to be larger than 4×4 pixels.

3.2 Lightweight Multi-Scale Feature Fusion Neck

In the YOLOv8 model, the Neck layer, which is conventionally configured as a PAFPN structure, incorporates three detection layers and is predominantly employed for feature fusion and detection. Nevertheless, to enhance the detection of small objects in the dataset, we have incorporated the feature maps of the shallow network and refined the model structure.

As demonstrated in Fig. 2, the YOLOv8 backbone network undergoes four downsampling operations during feature extraction, with the number of channels doubling after each downsampling, resulting in four times as many channels for the deepest features as for the shallow ones. These feature layers are directly passed into the FPN structure, complicating the subsequent multi-scale feature fusion process and increasing the number of network parameters and computational overhead.

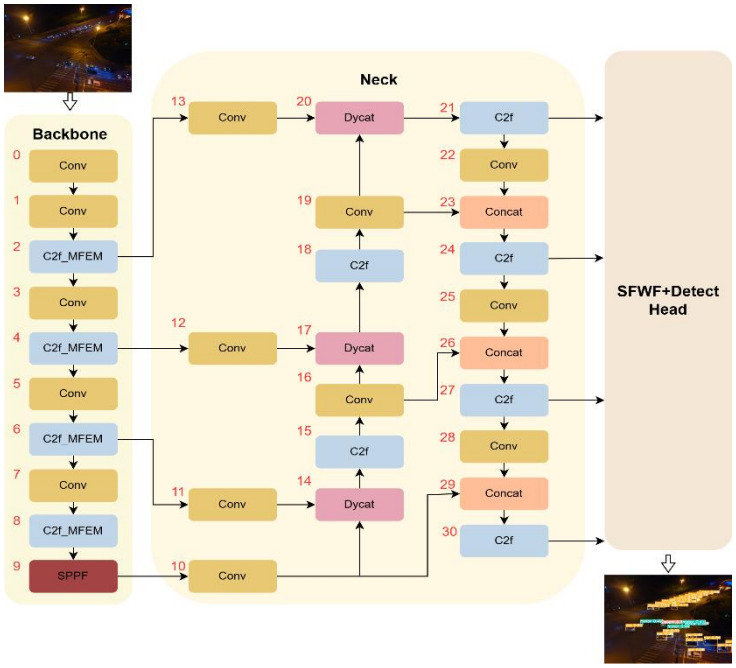

Fig. 2. Overall model structure

To balance the relationship between model performance and accuracy, we have made the following improvements:

First, an analysis of the characteristics of shallow and deep features was conducted. Shallow features have a small receptive field and high spatial resolution, which is suitable for capturing the details of small objects. In contrast, deep features have a large receptive field and rich semantic information, but the corresponding feature map resolution is low, which is more suitable for capturing the information of medium and large objects. However, in practical application scenarios, such as UAV images, small-sized

objects account for a very large proportion. As a result, the role of deep features in capturing objects is greatly reduced, but their computational cost is very high, which affects the performance of the model.

To address this challenge, we propose a modification to the standard convolutional layer, introducing a 1×1 convolutional layer to the output position of the backbone feature map. This adjustment is accompanied by a uniform adjustment of the number of channels to 256 [22]. This modification serves to expand the number of channels in the P2 layer to 256, while concurrently reducing the number of channels in the P5 and P4 layers to 256. This adjustment is made with the intention of maintaining a reasonable number of channels for the feature pyramid [22] input. This refinement enhances the significance of shallow features, facilitating the recognition of small targets, while concurrently reducing the number of deep features and the computational demands. Concurrently, we employ the DySample technique to resize the feature maps. Subsequently, we concatenate the feature maps of disparate layers through the concat operation, ensuring the utilization of the high resolution of shallow features and the extensive semantic information of deep features.

Experimental statistics demonstrate that the number of parameters in the enhanced four-layer feature PAFPN is approximately half that of the three-layer PAFPN. This design facilitates the extraction of more detection details from shallow features, enhancing the model's detection performance. Additionally, it retains sufficient deep semantic features to ensure the model's detection capability for objects of varied sizes. Consequently, this optimization enhances the performance of the YOLOv8 model in practical application scenarios.

3.3 C2f-MFEM

In the context of image detection tasks, the primary challenge in achieving accurate object detection is often the obscurity of features due to noise interference.

To address this issue, we have ingeniously integrated the C2f-MFEM module into the YOLOv8 backbone network. Specifically, we have innovatively incorporated a multi-fusion excitation module (MFEM) into the residual block of C2f, which first performs global average and global maximum pooling operations on the input feature map. The average pooling strategy has been demonstrated to effectively capture the overall average information of the feature map, while the maximum pooling method has been shown to focus sharply on the most prominent feature extremes. This two-pronged pooling method has been shown to comprehensively capture feature representations at different levels and significantly enhance the model's sensitivity to diverse information. As illustrated in Fig. 3, the modified C2f structure is depicted.

Subsequently, a shared multi-branch fully connected layer is employed to compress the pooling results and aggregate the outputs of these branches. This step ensures a comprehensive integration of information from different pooling strategies, thereby enabling the model to comprehend diverse data characteristics in greater detail. Subsequently, the results of the multiple branches are concatenated and excited through a fully connected layer (FC) to obtain channel weight information [23] . Finally, these weights are used to weight the original features, effectively suppressing the interference

of background noise and further improving the accuracy and robustness of object detection. The configuration of the MFEM module is illustrated in Fig. 4.

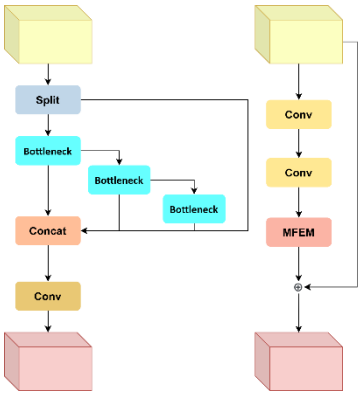

Fig. 3. C2f-MFEM block

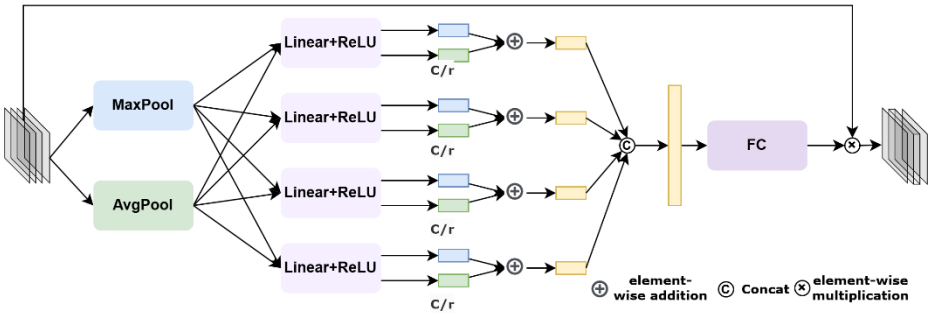

Fig. 4. MFEM

3.4 Spatial Feature Weighted Fusion Detection Head

In the FPN architecture, although multi-scale feature fusion endows the network with the ability to capture rich information, it inevitably introduces redundant contextual information. This redundant information often covers many areas that are not directly related to the detection target, thus introducing too much noise and weakening the model's performance in object recognition tasks. To address this challenge, we propose the Spatial Feature Weighted Fusion (SFWF) module. The SFWF module aims to precisely filter out the most valuable spatial information during the multi-scale feature fusion process.

As demonstrated in Fig. 5, the SFWF module accepts features at the target layer scale and features at the neighboring layer scales. Through a series of downsampling and upsampling operations, shallow detailed features and deep semantic features are integrated into the target layer scale. This process ensures the comprehensiveness of

the information and realizes feature fusion across scales. The integration of features from different layers is achieved through a process known as feature stitching, which effectively fuses low-level fine spatial information with high-level abstract semantic information. This fusion enhances the diversity and richness of features, with low-level features providing detailed spatial details and high-level features containing deep semantic connotations. The combination of these features enables the model to effectively handle object detection challenges of various scales and types.

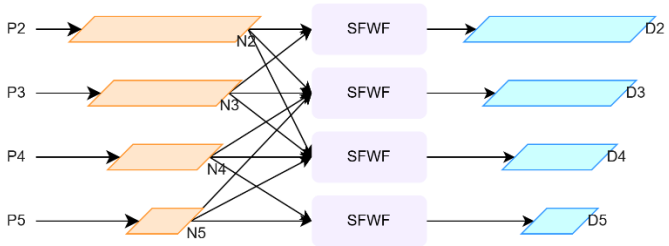

Fig. 5. SFWF-Head

Subsequently, a 1×1 convolution kernel is employed to extract $H \times W$ feature maps corresponding to each scale of features. The spatial weights of each feature map are then calculated using the softmax function. These weights are precisely assigned to the corresponding scale feature maps to achieve weighted processing of the feature maps. Finally, the weighted feature maps are summed to obtain the fused feature map. This process enhances not only the utilization efficiency of the features but also the model's object recognition ability in complex scenes.

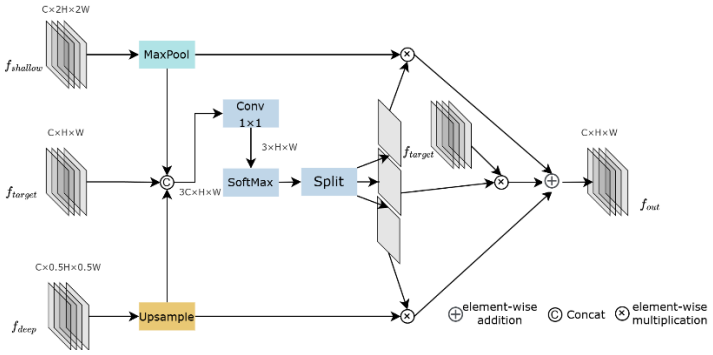

Fig. 6. SFWF block

As demonstrated in Fig. 6, assuming that the input of the feature is $[f_{shallow} \in R^{C\times 2H\times 2W}, f_{deep} \in R^{C\times 0.5H\times 0.5W}, f_{target} \in R^{C\times H\times W}]$ and the intermediate feature is $[f'_{shallow} \in R^{C\times H\times W}, f'_{deep} \in R^{C\times H\times W}, f'_{target} \in R^{3C\times H\times W}]$, the neighboring layer features are adjusted by up-sampling and down-sampling, and the formula is expressed as:

$$f'_{shallow} = MaxPool(f_{shallow}) \tag{1}$$

$$f'_{deep} = Upsample(f_{deep}) \tag{2}$$

$$f'_{target} = Concat(f'_{shallow}, f_{target}, f'_{deep}) \tag{3}$$

Subsequently, the spatial weight information corresponding to the feature layer is obtained by employing a 1 × 1 convolution and softmax operation on $f'_{target} \in R^{3C\times H\times W}$. The result is then split and multiplied to obtain the weighted feature $[f''_{shallow} \in R^{C\times H\times W}, f''_{deep} \in R^{C\times H\times W}, f''_{target} \in R^{3C\times H\times W}]$, which is expressed by the formula:

$$f''_{shallow} = f'_{shallow} \times SoftMax\left(\mathrm{Conv}(f'_{target})\right)[0] \tag{4}$$

$$f''_{target} = f'_{target} \times SoftMax\left(\mathrm{Conv}(f'_{target})\right)[1] \tag{5}$$

$$f''_{deep} = f'_{deep} \times SoftMax\left(\mathrm{Conv}(f'_{target})\right)[2] \tag{6}$$

Finally, the desired output feature is obtained by means of an addition operation, which is expressed by the following formula:

$$f_{out} = f''_{shallow} + f''_{target} + f''_{deep} \tag{7}$$

4 EXPERIMENTS

4.1 Experimental environment

The experimental platform utilized in this study is Ubuntu 22.04, an operating system that incorporates a NVIDIA A40 graphics card, with a total graphics memory capacity of 48 GB per card. The deep learning framework employed is PyTorch 2.0.0, and the Python version is 3.8, along with CUDA version 11.7.

4.2 Datasets and experimental details

The Visdrone2021 dataset, developed by Tianjin University's Machine Learning and Data Mining Laboratory, is used in this experiment. It is one of China's most extensive and complex aerial drone photography datasets, capturing various daily life scenes with 10 categories from 14 cities. It covers diverse altitudes, weather, lighting conditions, and objects with varying occlusion and deformation. The dataset includes 6,471 training, 548 validation, and 3,190 test images (with a challenging subset of 1,580). Image categories encompass cars, pedestrians, buses, bicycles, tricycles, boxcars, trucks, vans, and people, totaling 2.6 million labels. Reflecting real-world drone scenarios, it aligns with this study's background and objectives.

The experimental setup involved 200 epochs of training per model, with a batch size of 16 and $640\ x\ 640$ images. The optimization algorithm utilizes AdamW without the use of a pre-trained model. The learning rate was 0.01, and mosaic data augmentation was employed.

4.3 Assessment of indicator

The evaluation metrics employed in this study encompass mean accuracy (mAP), average precision (AP), precision (P), recall (R), giga floating-point calculations metrics (GFLOPs), and model parameters (Params). The following formulas are employed to calculate precision (P) and recall (R):

$$P = \frac{TP}{TP+FP} \tag{8}$$

$$P = \frac{TP}{TP+FN} \tag{9}$$

Table 1. The ablation study results of the algorithm on VisDrone2021 dataset.

Datasets	Model	P	R	mAP50	mAP95	GFLOPs	Params
Test	YOLOv8-n	38.5	28.7	26.6	14.8	8.2	3.0
	YOLOv8-n+P2	39.5	30.8	28.9	16.4	40.8	3.8
	YOLOv8-n+P2+LMFF-Neck	42.2	32.5	30.7	17.4	17.1	2.2
	YOLOv8-n+P2+LMFF-Neck+C2f-MFEM	42.5	33.2	31.1	17.7	17.3	2.2
	YOLOv8-n+P2+LMFF-Neck+C2f-MFEM+SFWF	43.6	33.6	32.5	18.3	18.0	2.4
Val	YOLOv8-n	43.7	33.8	33.3	19.2	8.2	3.0
	YOLOv8-n+P2	46.9	36.0	36.6	21.9	40.8	3.8
	YOLOv8-n+P2+LMFF-Neck	49.7	37.9	39.3	23.3	17.1	2.2
	YOLOv8-n+P2+LMFF-Neck+C2f-MFEM	50.0	39.2	40.2	24.1	17.3	2.2
	YOLOv8-n+P2+LMFF-Neck+C2f-MFEM+SFWF	50.3	39.7	40.8	24.3	18.0	2.4

In these equations, TP is equivalent to the number of samples that were correctly predicted to be positive, FP is equivalent to the number of samples that were incorrectly predicted to be positive, and FN is equivalent to the number of samples that were incorrectly predicted to be negative.

The formulas for calculating the average precision (AP) and mean average precision (mAP) are as follows:

$$AP = \int_0^1 p(x)dx \tag{10}$$

$$mAP = \frac{1}{K\sum_{i=1}^{K} AP_i} \tag{11}$$

The parameter K indicates the number of categories, and AP is the average precision of each category.

GFLOPs are utilized to quantify the computational complexity inherent to the training of a model.

Params is used to measure the consumption of computational memory resources.

4.4 Ablation experiment

We conducted an experimental study on the VisDrone2021 dataset to evaluate the impact of the enhanced modules in the LightDrone-YOLO model on the detection of objects from the perspective of a drone. A summary of the experimental data is presented in Table 1, illustrating the changes in results with the addition of P2, LMFF-Neck, C2f-MFEM and SFWF.

The preliminary findings suggest that incorporating P2 leads to a substantial enhancement in the model's detection performance, as demonstrated by an increase in the accuracy of the most stringent metric, mAP95, to 16.4%, thereby substantiating the importance of employing spatially rich feature maps for effective small object detection in this specific task.

Secondly, the efficacy of the LMFF-Neck module was demonstrated by its enhancement of the mAP50 by 1.8% and the mAP95 by 1% in comparison to the baseline model. This enhancement was achieved while maintaining the capacity of feature representation by decreasing the number of channels. The GFLOPs were reduced from 40.8 to 17.1, representing a decrease of more than 50%, thereby achieving an effective balance of performance and efficiency.

Furthermore, the implementation of the C2f-MFEM module has been demonstrated to enhance the mAP50 and mAP95 by 0.4% and 0.3%, respectively, suggesting that the module is efficacious in optimising the feature extraction of the backbone network.

Finally, the SFWF module, which was integrated into the detection head, enhanced the mAP50 metric by 1.4%. This module enhances the model's capacity to integrate contextual information and multi-level features, thereby further enhancing the model's detection performance. Concurrently, the computational demands of this module are minimal, with an increase of less than 1 GFLOPs and a 0.2M increase in the number of parameters.

4.5 Comparative experiments

In order to evaluate the efficacy of the proposed methodology, comparative experiments were conducted on the Vis-drone2021 test dataset. The performance of the proposed method was then compared with state-of-the-art algorithms, including Fast R-CNN, Faster R-CNN, Cascade R-CNN, RetinaNet, CenterNet, DMNet, HRDet+, MSC-CenterNet, YOLOv3-LITE and LightUAV-YOLO.

As shown in Table 2, the outcomes demonstrated the efficacy of the proposed method, with significant advancements observed. For pedestrian detection, the method attained 30.1% mAP50, comparable to LightUAV-YOLO and superior to Fast R-CNN and Faster R-CNN. In the domain of person detection, the proposed method surpassed all competitors except DMNet and YOLOv3-LITE. The vehicle detection performance exhibited a remarkable distinction. For Car, the method attained 73.9% mAP50, marginally higher than LightUAV-YOLO and notably higher than other methods. Notably, our method outperforms other algorithms in detecting buses with an mAP50 of 55.0%.

Table 2. Comparison of different algorithms on the Visdrone2021 test dataset.

Method	PED	PER	BC	Car	Van	Truck	TRI	ATRI	Bus	MO	mAP50
Fast R-CNN [24]	21.4	15.6	6.7	51.7	29.5	19.0	13.1	7.7	31.4	20.7	21.7
FasterR-CNN [24]	20.9	14.8	7.3	51.0	29.7	19.5	14.0	8.8	30.5	21.2	21.8
Cascade R-CNN [24]	22.2	14.8	7.6	54.6	31.5	21.6	14.8	8.6	34.9	21.4	23.2
RetinaNet [24]	13.0	7.9	1.4	45.5	19.9	11.5	6.3	4.2	17.8	11.8	13.9
CenterNet [25]	22.6	20.6	14.6	59.7	24.0	21.3	20.1	17.4	37.9	23.7	26.2
DMNet [1]	28.5	20.4	15.9	56.8	37.9	30.1	22.6	14.0	47.1	29.2	30.3
HRDet+ [26]	28.6	14.5	11.7	49.4	37.1	35.2	28.8	21.9	43.3	23.5	28.0
MSC-CenterNet [1]	33.7	15.2	12.1	55.2	40.5	34.1	29.2	21.6	42.2	27.5	31.1
YOLOv3-LITE [27]	34.5	23.4	7.9	70.8	31.3	21.9	15.3	6.2	40.9	32.7	28.5
LightUAV-YOLO [28]	30.5	18.9	9.9	73.5	37.9	33.9	16.6	16.6	52.7	30.3	32.1
Ours	30.1	19.1	19.1	73.9	36.7	33.3	17.8	18.9	55.0	30.1	32.5

As demonstrated in Table 3, our proposed method demonstrates superior performance in comparison to several contemporary lightweight models in the Visdrone2021 validation dataset, attaining a mAP50 of 40.8%. Specifically, it surpassed Drone-YOLO(nano) by 2.7%, LightUAV-YOLO by 1.0%, LE-YOLO by 1.5%, YOLOv5-n by 7.9%, YOLOv5-s by 0.2%, YOLOv8-n by 7.5%, YOLOv8-s by 0.3%, and YOLOv11-n by 11.3%. With 24.3% mAP50:95, it was competitive, especially considering its lightweight nature (2.4M parameters).

Table 3. Experimental results on VisDrone2021 val datasets.

Method	mAP50	mAP95	Params
Drone-YOLO(nano) [19]	38.1	22.7	3.05
LightUAV-YOLO [28]	39.8	24.1	2.2
LE-YOLO [29]	39.3	22.7	2.1
LUDY-N [16]	35.2	-	2.8
LUDY-S [16]	41.7	-	10.3
YOLOv5-n [30]	32.9	18.6	2.5
YOLOv5-s [30]	39.3	23.4	9.1
YOLOv8-n	33.3	19.2	3.0
YOLOv8-s	39.5	23.5	11.1
YOLOv11-n	29.5	17.4	2.58
Ours	40.8	24.3	2.4

5 CONCLUSIONS

In the task of object detection in UAV aerial images, we face a series of challenges, mainly including small targets, complex backgrounds, light interference, and limited hardware resources. These issues affect the detection accuracy of the detection model, while making it particularly difficult to find a balance between efficiency and performance. To address these challenges, this study proposes a model called LightDrone-YOLO by improving it based on the YOLOv8 architecture.

Specifically, we significantly enhance the model's ability to detect small targets by introducing a specialized small target detection layer P2. In addition, the design of the LMFF-Neck structure effectively reduces the number of parameters and operations of the model and improves the computational efficiency. Meanwhile, the combination of the MFEM module and the C2f module further strengthens the feature extraction effect of the backbone network, enabling the model to capture target features more accurately. Finally, we designed the SFWF module and combined it with the detection head to effectively mitigate the feature loss problem that may occur during the cross-scale fusion process.

Although the LightDrone-YOLO model has made significant progress in target detection for UAV aerial scenes, it still has some limitations. Especially when multiple small targets overlap, the detection accuracy of the model may be affected. Therefore, in our future research work, we will continue to improve and optimize the LightDrone-YOLO model with the aim of achieving higher detection accuracy and stronger generalization ability.

References

1. Cao, Y., He, Z., Wang, L., Wang, W., Yuan, Y., Zhang, D., Zhang, J., Zhu, P., Van Gool, L., Han, J., et al.: VisDrone-DET2021: The vision meets drone object detection challenge results. In: Proceedings of the IEEE/CVF International conference on computer vision, pp. 2847–2854. IEEE (2021)
2. Jia, X., Tong, Y., Qiao, H., Li, M., Tong, J., Liang, B.: Fast and accurate object detector for autonomous driving based on improved YOLOv5. Scientific reports 13(1), 9711 (2023)
3. Teja, Y.D.: Static object detection for video surveillance. Multimedia Tools and Applications 82(14), 21627–21639 (2023)
4. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems 28 (2015)
5. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779–788. IEEE (2016)
6. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: Single shot multibox detector. In: Proceedings of the 14th European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, October 11–14, 2016, Part I, pp. 21–37. Springer (2016)
7. Ding, J., Xue, N., Xia, G.S., Bai, X., Yang, W., Yang, M.Y., Belongie, S., Luo, J., Datcu, M., Pelillo, M., et al.: Object detection in aerial images: A large-scale benchmark and challenges. arXiv preprint arXiv:2102.12219 (2021)

8. Li, K., Wan, G., Cheng, G., Meng, L., Han, J.: Object detection in optical remote sensing images: A survey and a new benchmark. ISPRS Journal of Photogrammetry and Remote Sensing 159, 296–307 (2020)
9. Zhuang, J., Dai, M., Chen, X., Zheng, E.: A faster and more effective cross-view matching method of UAV and satellite images for UAV geolocalization. Remote Sensing 13(19), 3979 (2021)
10. Zhao, H., Zhang, H., Zhao, Y.: YOLOv7-SEA: Object detection of maritime UAV images based on improved YOLOv7. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 233–238 (2023)
11. Liu, W., Lu, H., Fu, H., and Cao, Z.: Learning to Upsample by Learning to Sample. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 6027–6037 (2023)
12. Chen, J., Mai, H.S., Luo, L., Chen, X., Wu, K.: Effective Feature Fusion Network in BIFPN for Small Object Detection. In: Proceedings of the IEEE International Conference on Image Processing (ICIP), pp. 699–703 (2021)
13. Li, Y., Feng, Y., Zhou, M., Xiong, X., Wang, Y., Qiang, B.: DMA-YOLO: Multi-scale object detection method with attention mechanism for aerial images. The Visual Computer 40(6), 4505–4518 (2024)
14. Li, Q., Zhang, Y., Fang, L., Kang, Y., Li, S., and Zhu, X.X.: DREB-Net: Dual-stream Restoration Embedding Blur-feature Fusion Network for High-mobility UAV Object Detection. arXiv preprint arXiv:2410.17822 (2024)
15. Du, Bowei, Huang, Yecheng, Chen, Jiaxin, Huang, Di: Adaptive sparse convolutional networks with global context enhancement for faster object detection on drone images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13435–13444. (2023)
16. Fan, Qingsong, Li, Yiting, Deveci, Muhammet, Zhong, Kaiyang, Kadry, Seifedine: LUD - YOLO: A novel lightweight object detection network for unmanned aerial vehicle. Information Sciences 686, 121366 (2025)
17. Du, Bowei, Huang, Yecheng, Chen, Jiaxin, Huang, Di: Adaptive sparse convolutional networks with global context enhancement for faster object detection on drone images. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 13435–13444. (2023)
18. Liu, Songtao, Huang, Di, Wang, Yunhong: Learning spatial fusion for single - shot object detection. arXiv preprint arXiv:1911.09516 (2019)
19. Zhang, Zhengxin: Drone - YOLO: an efficient neural network method for target detection in drone images. Drones 7(8), 526 (2023)
20. Chen, Naiyuan, Li, Yan, Yang, Zhuomin, Lu, Zhensong, Wang, Sai, Wang, Junang: LODNU: lightweight object detection network in UAV vision. The Journal of Supercomputing 79(9), 10117–10138 (2023)
21. Zeng, Shuang, Yang, Wenzhu, Jiao, Yanyan, Geng, Lei, Chen, Xinting: SCA - YOLO: A new small object detection model for UAV images. The Visual Computer 40(3), 1787–1803 (2024)
22. Lin, Tsung - Yi, Dollár, Piotr, Girshick, Ross, He, Kaiming, Hariharan, Bharath, Belongie, Serge: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2117–2125. (2017)
23. Hu, Jie, Shen, Li, Sun, Gang: Squeeze - and - excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132–7141. (2018)

24. Yu, W., Yang, T., Chen, C.: Towards resolving the challenge of long - tail distribution in UAV images for object detection. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp. 3258–3267 (2021)
25. Albaba, B. M., Ozer, S.: Synet: An ensemble network for object detection in uav images. In: 2020 25th International conference on pattern recognition (ICPR), pp. 10227–10234. IEEE (2021)
26. Du, D., Zhu, P., Wen, L., Bian, X., Lin, H., Hu, Q., Peng, T., Zheng, J., Wang, X., Zhang, Y., et al.: VisDrone-DET2019: The vision meets drone object detection in image challenge results. In: Proceedings of the IEEE/CVF international conference on computer vision workshops, pp. 0–0 (2019)
27. Zhao, H., Zhou, Y., Zhang, L., Peng, Y., Hu, X., Peng, H., Cai, X.: Mixed YOLOv3-LITE: A lightweight real-time object detection method. Sensors 20(7), 1861 (2020)
28. Lyu, Y., Zhang, T., Li, X., Liu, A., Shi, G.: LightUAV-YOLO: a lightweight object detection model for unmanned aerial vehicle image. The Journal of Supercomputing 81(1), 105 (2025)
29. Yue, M., Zhang, L., Huang, J., Zhang, H.: Lightweight and efficient tiny - object detection based on improved YOLOv8n for UAV aerial images. Drones 8(7), 276 (2024)
30. Jocher, G., Stoken, A., Chaurasia, A., Borovec, J., Kwon, Y., Michael, K., Changyu, L., Fang, J., Skalski, P., Hogan, A., et al.: ultralytics/yolov5: v6. 0-YOLOv5n'Nano'models, Roboflow integration, TensorFlow export, OpenCV DNN support. Zenodo (2021)