Dean&Francis

ISSN 2959-6157

AERIAL OBJECT DETECTION SYSTEM WITH DEEP LEARNING

Yinfeng Liu[1] and

Xiaocan Ouyang[2]

[1]School of Beijing City University, Beijing, China
[2]School of Bangor University, Bangor, UK
Corresponding author: xcy23jkt@bangor.ac.uk

Abstract:

Nowadays micro/mini drones become highly accessible to the person from all walks of life. The statement mentioned above poses enormous safety hazards and regulatory challenges. Due to the smaller radar reflection cross-sectional area of unauthorized drones, difficult to detect by radio detection system, which may interfere the normal takeoff and landing progress of aircraft or leak the location information of facilities. In recent years, deep learning methods have made good progress in the field of small object detection. Therefore, we suggest, in this paper, a drone detection method that integrates deep learning-based classification and localization tasks. Using YOLO v8(You Only Look Once Version 8), deep learning neural network, and adjusting its architecture and parameters to better adapt to small object detection such as micro/mini drones. In addition, to train the neural network model in this article to classify detected aerial aims, we selected a multi class flying object dataset that includes birds, drones, helicopters, and fixed wing aircraft, among which some may be potential threats.

Keywords: Small target detection; Self-attention mechanism; Convolutional neural network; YOLO V8; Expansion convolution.

1. Introduction

With the rapid development of drone technology, various types of unmanned aerial vehicles (UAVs) are being manufactured. The International Civil Aviation Organization (ICAO) uses this general term to refer to any unmanned aircraft system (UAS). Functionally, drones are categorized into FPV (First Person View) racing drones and aerial photography drones. Structurally, they are classified as multi-rotor drones, vertical take-off and landing (VTOL) drones, fixed-wing drones, and single-rotor drones. According to the classification by NATO (North Atlantic Treaty Organization), drones weighing no more than 2 kilograms are referred to as micro drones, those weighing less than 25 kilograms are classified as small drones, and those weighing less than 150 kilograms fall into Category I drones[1].

Drone activities worldwide are becoming increasingly active. According to the Civil Unmanned Aerial Vehicle Development Report by the Securities Economic Research Institute, the number of registered drones in China has been increasing annually. By the end of 2023, the total number of registered drones in

China reached 1.267 million, representing a 32.2% year-on-year increase from the end of 2022[2]. However, the relative ease with which the public can access micro/mini drones poses significant challenges to safety and confidentiality. For instance, in airport environments, low-flying birds, illegal or unauthorized drones[3], or other aircraft during takeoff and landing may conflict with flight paths, disrupting normal aircraft operations and potentially leading to safety incidents. Micro/mini drones could also be acquired by malicious actors to expose images of sensitive national facilities or carry out sabotage activities. Based on these examples, I believe it is crucial to propose a drone detection system capable of classifying and locating illegal drones.

Drone detection or counter-UAS technologies are generally divided into four categories: RF signal analysis systems[4], radio signal detection systems (radar)[5], acoustic sensors[6], and electro-optical/thermal imaging sensors. RF analyzers can detect radio communication signals between drones and ground control stations, capturing their current positions and operator locations. Radio signal detection systems, i.e., radar, can precisely detect and locate drones. However, due to drones' low-altitude flight, slow speed, and small radar cross-section, distinguishing them from noise and clutter is challenging. Acoustic sensors can detect the sound emitted by drones and calculate their direction but are susceptible to environmental noise. Electro-optical and thermal imaging sensors are suitable for detecting small, fast-moving objects at low altitudes, enabling visual detection and classification of drones. Nevertheless, their performance is significantly affected by weather, lighting, and other environmental factors[7].

In this paper, we propose a system based on the YOLO v8 neural network model. By adjusting the model's hyperparameters, replacing network modules, and optimizing its structural framework, we enhance its capability for detecting small targets. This model is trained on a dataset comprising images of birds, drones, helicopters, and fixed-wing aircraft captured by electro-optical camera sensors. The aim is to accurately detect micro/mini drones and similar objects, enabling more efficient and rapid identification of potential security threats in a given airspace.

2. YOLO v8 Algorithm Model Design and Analysis

Due to their relatively small size, unmanned aerial vehicles (UAVs) can easily find cover at low altitudes, while those at high altitudes are far from the photoelectric equipment of counter-UAS systems. Consequently, images of these UAVs often feature a low pixel ratio. Therefore, in UAV detection systems, the detection targets face challenges such as small size and extensive occlusion compared to the background. This makes small target detection under complex weather conditions and environments more difficult than general object detection.

YOLOv8 employs a single-stage object detection approach, simultaneously predicting bounding boxes and categories within a single network. It balances recognition accuracy with speed, making it a real-time and efficient algorithm among single-stage recognition methods. The variants include YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x. Table 1 presents the results of YOLOv8's pre-trained models on the COCO dataset [8]. As shown in Table 1, as the model scales up from small to large, the network size, depth, and performance increase, while speed decreases. Comparative analysis reveals that the YOLOv8n model offers the fastest detection speed, better meeting real-time requirements. It also demands fewer computational resources, making it easier to deploy on devices with limited capabilities.

Given the specific requirements of this study, the system must achieve effective target detection with low latency (or fast detection speed). YOLOv8n demonstrates flexibility, high detection accuracy, and low latency when handling complex scenarios, ensuring rapid responses. Considering the dual demands of real-time performance and accuracy, we selected YOLOv8n as the base model for this study and optimized it further to enhance its precision.

Table 1 Training results of different Yolo v8 models

model	mAP50-95	Paras/M	FLOP/B	Time/ms
YOLO v8n	37.3	3.2	8.7	0.99
YOLO v8s	44.9	11.2	28.6	1.20
YOLO v8m	50.2	25.9	25.9	1.83
YOLO v8l	52.9	43.7	43.7	2.39
YOLO v8x	53.9	68.2	257.8	3.53

3. Improved YOLO v8 Algorithm Model Construction

To address the issue of lower recognition performance of the YOLO v8 model in complex open-field environments, this paper designs the YOLO v8-final object detection model for detecting high- and low-altitude flying objects in complex weather conditions, with its network structure shown in Fig.1. The improvements include:

- Focus Module Integration: Reduces computational complexity while preserving critical spatial information for subsequent feature extraction and object detection.
- C3STR Module Addition: Based on the Swin Transformer network, this module enhances precision in capturing small targets.
- SPPCSPC Replacement: Substitutes the Spatial Pyramid Pooling Fast (SPPF) module with the Cross-Stage Partial Connection Spatial Pyramid Pooling (SPPCSPC) module to boost feature extraction capability and computational efficiency.
- Additional Detection Head: Amplifies feature pixel resolution to improve sensitivity toward minuscule targets.
- Depthwise Separable Convolution (DWConv) Integration: Replaces partial standard convolution modules to achieve network lightweighting.
- C2 Module Optimization: Substitutes selected C2F modules with C2 modules, further reducing model size while enhancing feature extraction capability. These optimizations enable the model to operate within the 24GB VRAM constraints of the experimental RTX 4090 GPU.

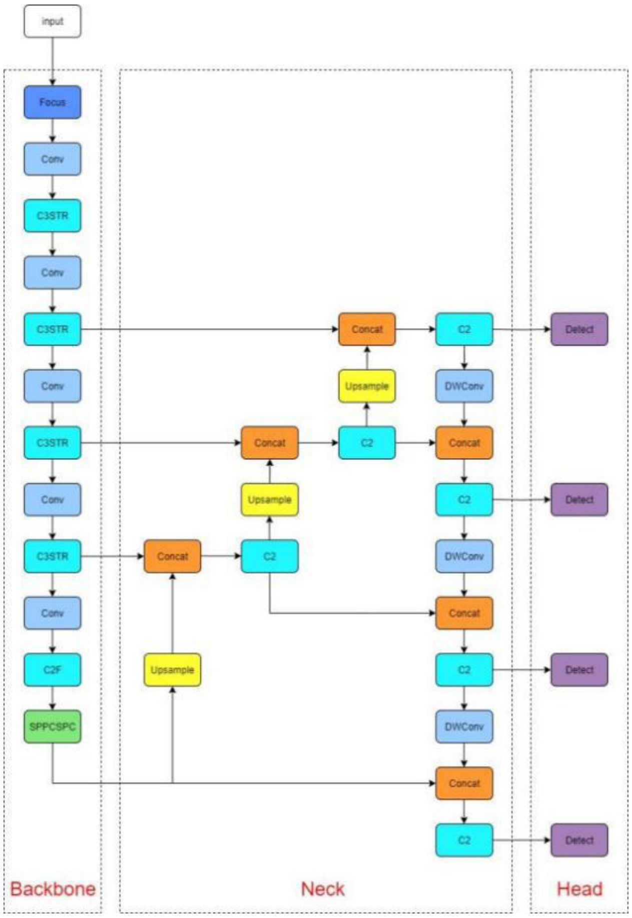


Fig. 1 YOLO v8n-final Convolutional network

3.1 Focus module

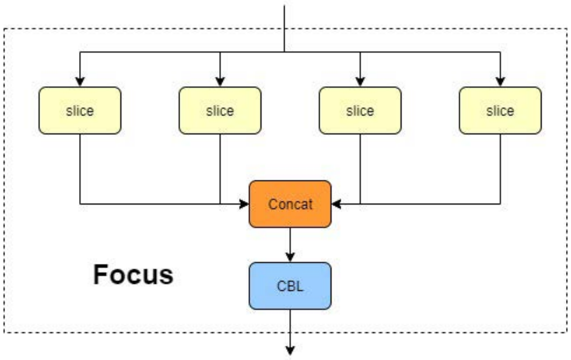


Fig.2 The convolutional network of the Focus model

The network structure of the Focus layer is shown in Fig.2. The Focus layer first performs a slicing operation on the input feature map. Specifically, it samples the input feature map at every other pixel along both the width and height dimensions, resulting in four feature maps with halved resolution. These four feature maps maintain the same number of channels as the input but have their spatial dimensions (width and height) halved. Next, the Focus layer concatenates these four sliced feature maps along the channel dimension. Since the channel count of each individual feature map remains unchanged, the concatenated feature map has four times the original number of channels, while the spatial dimensions (width and height) remain halved. Finally, the Focus layer performs a convolution operation on the concatenated feature map. This convolution, typically using a 3x3 kernel and often followed by Batch Normalization and an activation function (such as SiLU), further extracts features and can adjust the output channel count as required. By optimizing the feature extraction process, this layer significantly improves the model's detection performance [9].

3.2 C3STR module

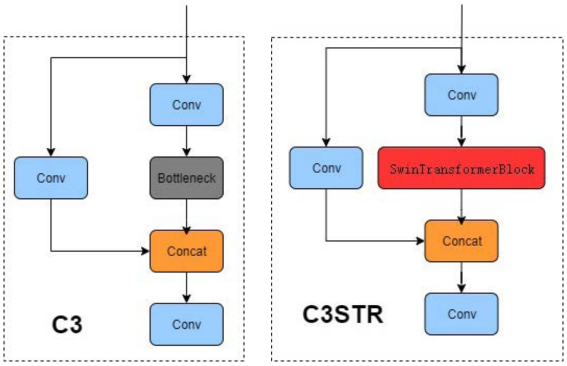


Fig.3 Comparison between the C3 and the C3STR

In the fields of deep learning and computer vision, Transformer-based detection algorithms have become a revolutionary model architecture. They primarily utilize the self-attention mechanism to process sequential data, enhancing the model's ability to understand temporal and spatial relationships. The self-attention module, as the core of the Transformer architecture, enables the model to capture long-range dependencies more efficiently and accurately when processing sequence data like images and text, thereby significantly improving the performance of detection tasks [10].

Swin Transformer is an innovative model based on the Transformer architecture, specifically designed for computer vision tasks. By introducing multi-scale processing and a window-based self-attention mechanism, it effectively overcomes the limitations of traditional Transformer models when processing image data. The core idea of the Swin Transformer lies in decomposing the image into multiple sub-windows and applying the self-attention mechanism within these sub-windows. This reduces computational complexity while maintaining the model's sensitivity to global information [11].

Swin Transformer excels in various computer vision tasks such as image recognition, object detection, and semantic segmentation. Firstly, it adopts a hierarchical structure, similar to the multi-level feature extraction in CNNs, enabling it to capture visual features from low-level to high-level. This hierarchical structure progressively reduces the resolution of the feature maps while increasing the receptive field at each layer, allowing the model to handle image information at different scales. Secondly, Swin Transformer introduces a strategy called "shifted windows" for self-attention computation [12].

Fig. 3 shows the network structures of C3 and C3STR. The C3STR module inherits from the C3 module. While the C3 module can effectively fuse residual features and occupies less GPU memory compared to the C2F module, its Bottleneck module cannot achieve feature interaction for small targets. This paper replaces the Bottleneck module in the C3 module with the Swin Transformer Block to construct the C3STR module. This enhances the feature interaction capability for small targets while retaining the C3 module's ability to fuse residual features. The improved model employs the C3STR module, leveraging its shifted window and hierarchical structure to enhance the resolution of input feature maps, progressively expand the receptive field of aircraft features, and achieve feature interaction for aircraft between neighboring pixels.

3.3 SPPCSPC module

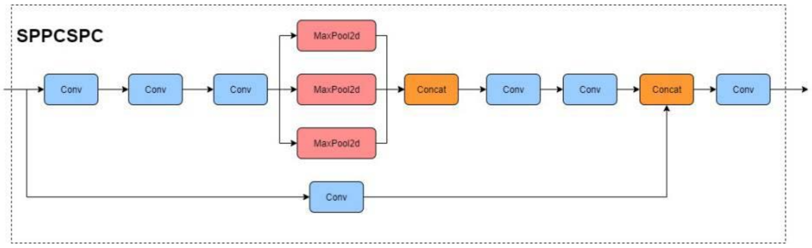


Fig.4 SPPCSPC Internal convolutional network

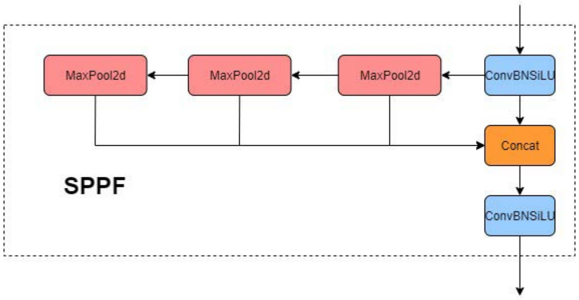


Fig.5 SPPF internal convolutional network

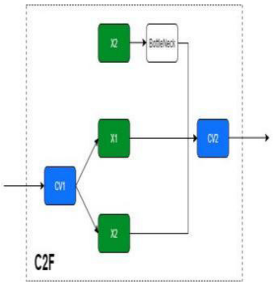
Fig.4 and Fig.5 illustrate the network structures of the SPPCSPC and SPPF layers, respectively. The SPPCSPC module utilizes an SPP layer to capture feature information at different scales, enhancing the model's detection capability for multi-scale targets. Combined with the CSP module, SPPCSPC maintains high performance while reducing computational load and improving the model's inference speed [13].

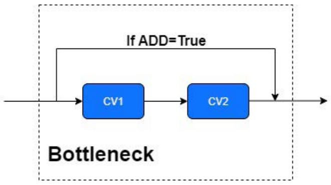
3.4 DWConv module

The main advantage of depthwise separable convolution lies in reducing computational load and parameter count, while simultaneously improving model efficiency and speed. This is because, in depthwise convolution, each channel only needs to convolve with one filter, unlike traditional convolution where each channel needs to convolve with all filters**.** The dimensionality reduction applied to input channels by pointwise convolution leads to a lower parameter count. Consequently, it offers significant advantages in terms of reduced model size and computational requirements, while typically maintaining strong performance levels [14].

Depthwise separable convolution (DWConv) consists of two parts: depthwise convolution and pointwise convolution. In depthwise convolution, each input channel is convolved with a separate filter (kernel). This means each input channel generates a corresponding output channel. Depthwise convolution is mainly used to capture the spatial information of the input data. Pointwise convolution is a 1×1 convolution operation that convolves all channels of the input at each position. Pointwise convolution can be seen as a convolution operation performed on the channel dimension of the input data, without involving spatial information. It is used to linearly combine the feature maps generated by depthwise convolution from the individual channels. DWConv reduces parameter count and computational complexity by splitting the traditional convolution operation into two steps: depthwise convolution and pointwise convolution, while maintaining the model‘s expressive power and performance.

3.5 C2 and C2F module





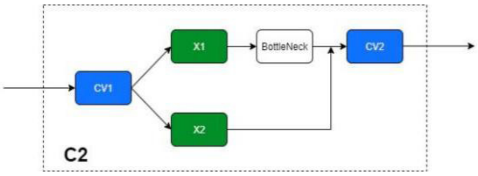


Fig.6 C2F convolutional network Fig.7 Bottleneck network Fig.8 C2 convolutional network

In C2F (Fig.6), the Sequential structure composed of Bottlenecks (Fig.7) in the original C2 module (Fig.8) is replaced with a ModuleList, shifting the operation from sequential to parallel. While the C2 module has fewer parameters and superior feature extraction capability compared to the C2F module, its gradient flow information propagation performance is weaker than that of C2F [15]. Therefore, C2F modules are retained in the backbone network in order to obtain feature maps characterized by both high resolution and rich semantic information , thereby improving object detection accuracy. The C2F modules in the neck are replaced with C2 modules to reduce parameter count and optimize feature extraction capability, enhancing the network’s computational efficiency.

4. Experiment and Result Analysis

To validate the effectiveness of the YOLO v8-final model in detecting aircraft under various weather conditions, the experiment utilized a public dataset provided by AhmedMohsen on the drone-detection-new Computer Vision Project website. This dataset contains nearly 12,000 images featuring fixed-wing aircraft, drones, birds, and helicopters. Comparative experiments were conducted to verify the contribution of each improvement strategy to the model’s detection performance.

4.1 Evaluation Indicators

The experiments in this paper use mean Average Precision (mAP50), Precision, and Recall to evaluate the accuracy of the detection model.

Its definition is as shown in the formula

$$P = \frac{TP}{TP + FP} \tag{1}$$

$$R = \frac{TP}{TP + FN} \tag{2}$$

$$AP = \int_0^1 p(r)dr \tag{3}$$

$$\text{mAP} = \frac{1}{m}\sum_{i=1}^{m} APi \tag{4}$$

In the formula: TP represents the number of positive samples predicted to be in the positive class; FP represents the number of negative samples predicted as positive classes; FN is the number of positive samples predicted to be of the negative class.

4.2 Experimental Environment and Parameter Settings

The experimental operating system was Ubuntu, and the specific configuration details are presented in Table 2. During network training, the batch size was set to 8, with the input image resolution configured at 640×640. Training was performed on a GPU for a total of 100 epochs. Throughout the training process, stochastic gradient descent was employed to optimize the parameters of the network model. Parameters including the initial learning rate, momentum, and weight decay were all maintained as the default parameters from the original YOLOv8n model.

Table 2 Experimental environment configuration

Configuration	Parameter
Development environment	Anaconda+Pycharm
CPU	12 vCPU Intel(R) Xeon(R) Platinum 8352V CPU @ 2.10GHz
GPU	RTX 4090(24GB)
Operating system	ubuntu20.04
Operating environment	CUDA11.8+Pytorch2.0.0
Programming language	Python

4.3 Comparative Experiment

4.3.1 Training Results of YOLO v8n Model

The results of the original YOLO v8n are shown in Fig.9 and Table 3.

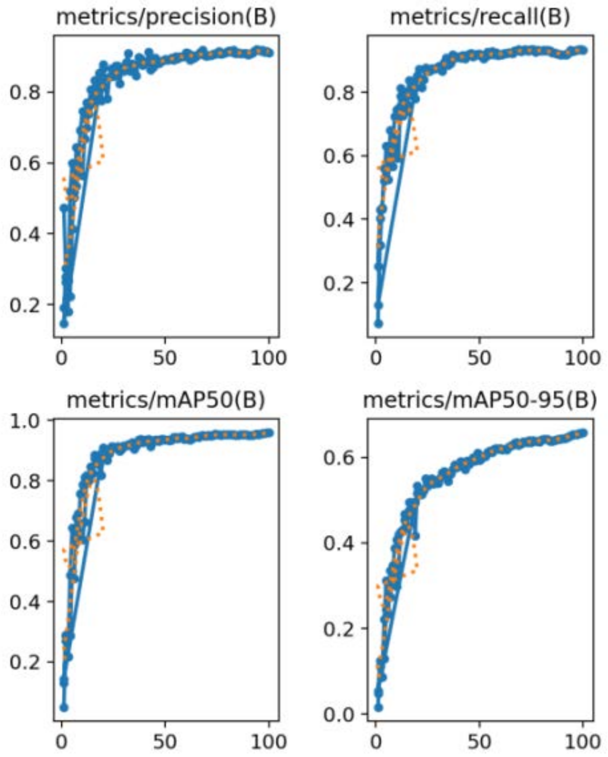


Fig.9 YOLO train result

4.3.2 Training Results of YOLO v8n-swin Model

As shown in the Fig.10, this model incorporates a Swin-Transformer dynamic attention mechanism module into its backbone network, with the training evaluation results displayed in the accompanying Fig.11 and table 3.

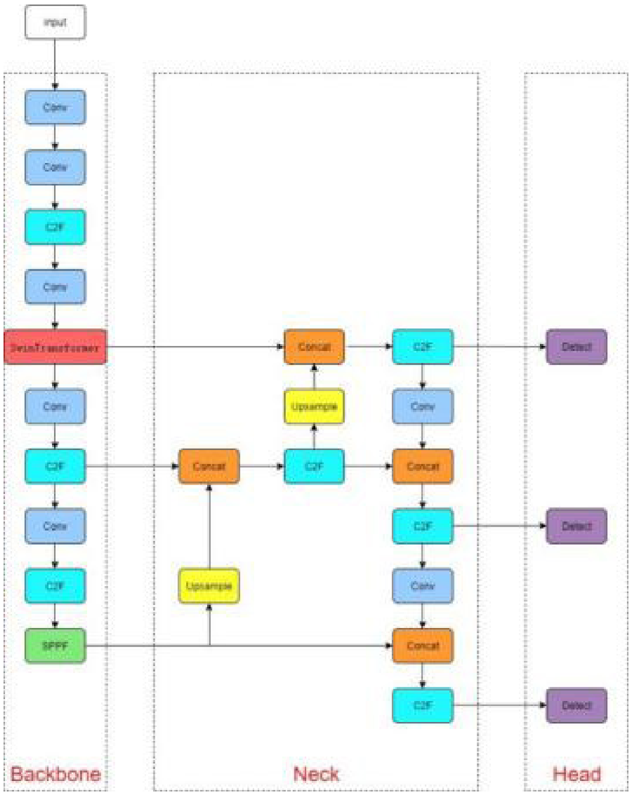


Fig.10 YOLO v8n-swin convolutional network

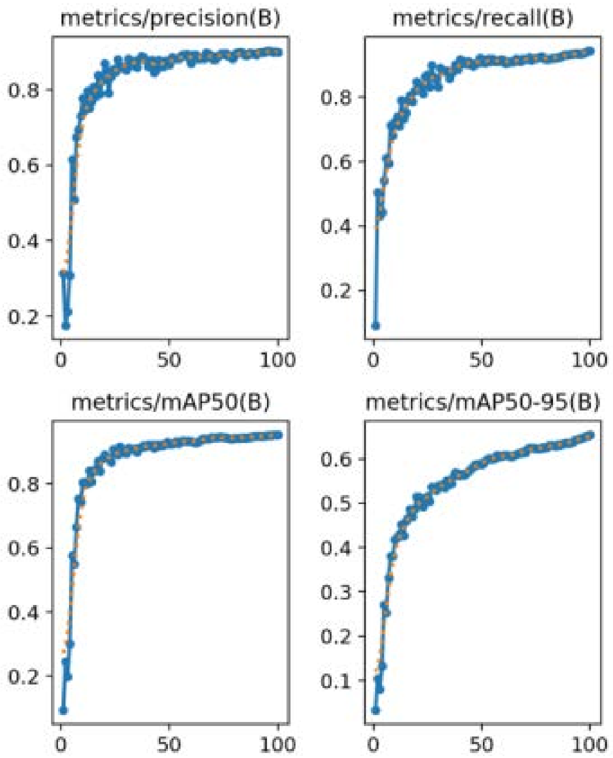


Fig.11 YOLO v8n-swin train result

4.3.3 Training Results of YOLO v8n-small Model

As shown in the Fig.12, an additional detection head has been added to the YOLO v8n network architecture in this model, with its training results and evaluation metrics presented in the accompanying Fig.13 and table 3.

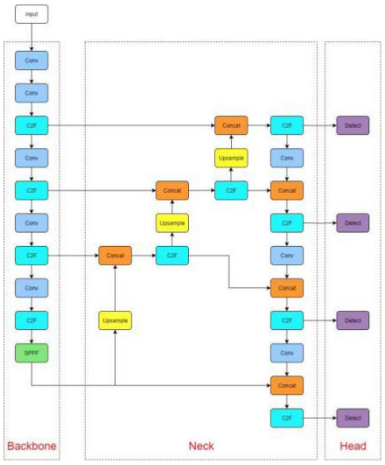


Fig.12 YOLO v8n-small convolutional network

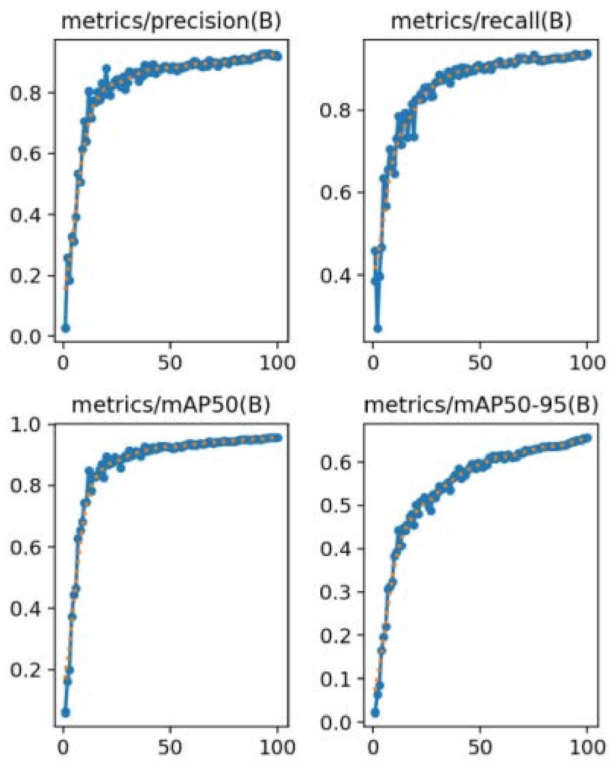


Fig.13 YOLO v8n-small train result

4.3.4 Training Results of YOLO v8n-final Model

The final training results of YOLO v8n-final are shown in Fig.14 and Table 3.

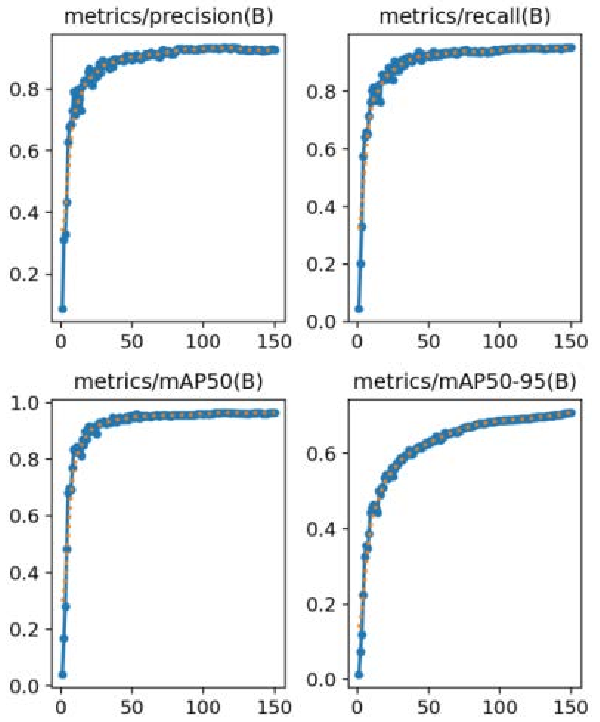


Fig.14 YOLO v8n-final train result

Table 3 Comparison of results from different models

model	mAP50	Precision	Recall
YOLO v8n	*0.955*	*0.913*	*0.933*
YOLO v8n-swin	0.952	0.900	0.944
YOLO v8n-small	0.955	0.923	0.935
YOLO v8n-final	0.960	0.935	0.950

5. Conclusion

Based on YOLO v8n, this paper proposes a drone small-target detection algorithm named YOLO v8n-final, which partially addresses issues including false positives, high miss rates, and poor real-time performance in micro-drone target detection. The algorithm enables small-target detection under complex weather conditions on devices with limited computational resources.

6. Author Contribution

All the authors contributed equally and their names were listed in alphabetical order.

References

[1] Drone Wars UK. https://dronewars.net/. Last accessed 2020/05/23.

[2] Tang Xuxia, Yang Shan, Cao Hanmin. Low-Altitude Economy Industry Special Series II: Market Space, Technology Trends, and Industrial Chain Opportunities of eVTOL Power Systems - 250417.

[3] Wu Xiangchen, Gu Zenan, Qiu Jingyun. Application of Reconnaissance-Strike Coordination Scheme for ‘Low-Slow-Small’ UAV Defense Systems. Instrumentation Users, 2023, 30(04): 27-31.

[4] Hao Mingming, Sheng Huaijie, Chen Mingjian. Simulation Model of Repeater Jamming System Based on RF Signals. Journal of Detection & Control, 2019, 41(01): 53-58.

[5] Tian Shengxiang, Luo Long, Liu Zhanshuang, et al. Discussion on the Status and Development of Intelligent Traceability Systems for UAV Countermeasure Technology. In: Intelligent Information Processing Industrialization Branch, China Hi-Tech Industrialization Association. Proceedings of the 16th National Conference on Signal and Intelligent Information Processing and Application. [Publisher not specified], 2022: 545-548. DOI:10.26914/c.cnkihy.2022.053450.

[6] Zong Bo, Bao Jiabin, Fu Jiajia, Tang Wencai, He Jialin, Li Da. Challenges and Opportunities Brought by ‘Low-Slow-Small’ UAVs to Nuclear Security Management. Radiation Protection Bulletin, 2022, 42(03): 33-38.

[7] Chen Yu, Ren Yilin. Counter-Unmanned Aerial Vehicle Warfare: A New Global Battlefield for Attack and Defense. China National Defense News, 2025-03-18.

[8] Mi Zeng, Lian Zhe. Research Review of YOLO Methods for General Object Detection. Computer Engineering and Applications, 2024, 60(21): 38-54.

[9] Li Changwei. Design and Implementation of a Single-Target UAV Tracker Based on YOLOv5s [Master’s thesis]. Xihua University, 2023. DOI:10.27411/d.cnki.gscgc.2023.000210.

[10] Bai Junqing, Wang Mengting, Shen Shouting. Vehicle Small Object Detection Algorithm in UAV Remote Sensing Images Based on YOLOv5. Science Technology and Engineering, 2025, 25(12): 5110-5118.

[11] Qi Jianbo. Research on Small Object Detection Algorithms in Remote Sensing Images Based on Deep Learning [Master’s thesis]. Qingdao University of Science and Technology, 2024. DOI:10.27264/d.cnki.gqdhc.2024.000139.

[12] Chen Zhexuan, Gao Xuelian, Song Jiayu, et al. Few-Shot Foreign Object Detection on Power Transmission Lines Fusing Dual Encoding and Meta-Learning. Chinese Journal of Scientific Instrument, 2025, 46(03): 193-205. DOI:10.19650/j.cnki.cjsi.J2413568.

[13] Chai Rui. Research on Small Target Detection in Aerial Images via Reconstructing SPPCSPC and Optimizing Downsampling [Master’s thesis]. Liaoning Technical University, 2024. DOI:10.27210/d.cnki.glnju.2024.000132.

[14] Ping Lujing, Ma Xing, Mu Chunyang, et al. Grasp Detection Algorithm Based on Depthwise Separable Convolution Residual Module. Transducer and Microsystem Technologies, 2025, 44(05): 133-137. DOI:10.13873/J.1000-9787(2025)05-0133-05.

[15] Liu Ning, Ouyang Ze, Ma Wenyuan, et al. Improved YOLOv5-FGC Recognition Algorithm for Surface Defect Detection of Steel Parts. Manufacturing Automation, 2024, 46(12): 24-33.