Lightweight Aerial Target Detection Algorithm with Enhanced Small Target Perception

Guang Wang, Yanming Zhang, Qiang Ai, *Member, IAENG*

***Abstract*—Addressing the issue of low detection accuracy for small objects in drone aerial imagery, this paper improves the YOLOv8n framework. It introduces the Multi-Scale Feature Aggregation Pyramid (MFPN) and the Efficient Local Attention Module (ELAM) to enhance feature representation and positional accuracy. By replacing SE attention with CBAM and adopting the Mish activation function, channel responsiveness and the nonlinearity of output features are enhanced. Additionally, the Wise-IoUv3 loss function replaces CIoU, dynamically adjusting gradient weights based on the quality of anchor boxes, thereby reducing the impact of low-quality predictions and geometric penalties in overlapping areas. Validation on the VisDrone-2019 dataset demonstrates that the proposed improvements significantly enhance the mAP, proving an enhanced detection performance and generalization capability of the model.**

***Index Terms*—YOLOv8n, UAV detection, multi-scale features, small target, lightweight algorithms**

I. Introduction

WITH the rapid advancement of Unmanned Aerial Vehicle (UAV) technology, UAVs are increasingly utilized in diverse fields such as agricultural monitoring, disaster rescue, environmental protection, and urban management [1]–[3]. Owing to their high flexibility, low cost, and extensive area coverage, UAVs are becoming indispensable tools for a variety of tasks [4], [5]. However, detecting small targets in UAV aerial images continues to present numerous challenges.

Since UAVs typically operate at high altitudes, the targets in the captured images are often small and easily overlooked [6]. Furthermore, the background in aerial images is typically rich and complex, often obscuring the targets [7]. In certain scenarios, the high density of targets results in mutual occlusion [8]. These challenges necessitate enhanced capabilities for small target detection in UAV aerial imagery.

In recent years, the rapid development of deep learning technology has significantly advanced the detection of small targets using UAVs [9]. Researchers have introduced various methods and techniques in this domain, thereby driving continuous advancements. Existing main detection algorithms can be broadly classified into two categories based on their approach to candidate region generation: ”single-stage” and ”two-stage” algorithms [10]–[12].

Single-stage detection algorithms, such as YOLO (You Only Look Once), SSD (Single Shot MultiBox Detector), and RetinaNet, are notable [13]–[15]. These algorithms simultaneously perform target classification and localization using raw features, without generating candidate regions [16]. While this approach significantly enhances detection speed, it may compromise accuracy [17]. Although faster, these algorithms often exhibit less precision compared to those requiring candidate region generation [18].

The YOLO algorithm conducts target localization and classification in a single forward pass, whereas SSD performs detection across feature maps of different scales, excelling in detecting small targets [19], [20]. RetinaNet utilizes the Focal Loss function to effectively address class imbalance issues, thus maintaining high detection speed and enhancing accuracy [21], [22].

Two-stage detection algorithms, including Fast R-CNN [23], Faster R-CNN [24], and SPP-Net [25], have demonstrated superior performance in detecting small targets with UAV imagery. These algorithms initially generate candidate regions, followed by precise target classification and localization within these areas. Although this approach often yields higher detection accuracy than single-stage algorithms, it comes at the cost of reduced detection speed.

To improve the detection accuracy of small targets in UAV imagery, numerous researchers have proposed effective techniques. For example, Pan et al. [26] implemented the ASFF (Adaptive Spatial Feature Fusion) module in YOLOv4, which adaptively fuses features from various levels, thereby enhancing the detection capability for small targets. Zhang et al. [27], the ECA (Efficient Channel Attention) mechanism was incorporated into YOLOv5, enhancing the model's capability to extract features from small targets through automatic learning of channel importance.

Alaftekin et al. [28] utilized the Mish activation function, replacing the traditional Leaky ReLU in YOLOv4, which enhanced the model's nonlinear expression capability and thereby improved detection accuracy. Additionally, Zhang et al. [29] integrated the GhostNet architecture into YOLOv5, effectively leveraging inter-layer information through dense connections to boost model performance.

Lin et al. [30] implemented the CBAM (Convolutional Block Attention Module) in YOLOX, employing the DIoU_NMS algorithm to optimize the selection of predicted boxes, significantly enhancing the model's accuracy and performance. Wang et al. [31], an IoU-aware branch was introduced in YOLOv4 to optimize the IoU loss function, thereby improving the bounding box regression accuracy for small targets.

Lastly, Chu et al. [32] adopted a multi-scale feature fusion method in YOLOv5, constructing a feature pyramid to perform detection at various scales, significantly enhancing the detection capabilities for small targets.

Based on the analysis presented above and addressing the

Manuscript received August 11, 2024; revised November 11, 2024.

Guang Wang is an Associate Professor at the School of Software, Liaoning Technical University, Huludao, Liaoning, 125105, China (e-mail: 275469783@qq.com).

Yanming Zhang is a Postgraduate Student at the School of Software, Liaoning Technical University, Huludao, Liaoning, 125105, China (Corresponding author to provide email: 824762260@qq.com).

Qiang Ai is a Postgraduate Student at the College of Computer, Qinghai Normal University, Xining, Qinghai, 810008, China (phone: +8617813140425; e-mail: qiang.ai@outlook.com).

challenges inherent in drone aerial image target detection, this study has enhanced the YOLOv8n model, referencing current research to develop a lightweight aerial target detection algorithm. The specific improvements are as follows:

- Building upon EfficientNetV2 [33], this study proposes a lightweight backbone network. By replacing the SE attention module with CBAM in MBConv, we enhance the network's response to critical channels. Additionally, substituting the traditional activation function with the Mish function improves the nonlinearity of the output features, smooths gradients, and avoids saturation phenomena, thereby enhancing activation effects.
- To address the issue of rapid reduction in feature map dimensions, this study introduces the Multi-Scale Feature Aggregation Pyramid (MFPN), which optimizes feature fusion. Additionally, the introduction of a new detection scale, H_2, enhances the detection capabilities for small objects.
- Devise Multi-Scale Feature Fusion Module (MSFF), which incorporates the Multi Convolution Module (MC) and the Feature Enhancement Module (FE) to enrich feature diversity and enhance detection accuracy. The structure processes channel and spatial attentions in parallel, preserving multi-scale detail information, thus significantly improving the detection capabilities for small targets in drone aerial imagery.
- To enhance the capability of capturing local semantics in shallow feature maps, this study introduces the Efficient Local Attention Module (ELAM). ELAM utilizes bidirectional average pooling and parallel 1D convolutions to process sequence signals, combined with larger convolution kernels and Group Normalization (GN), effectively enhancing the expression of positional information and significantly improving the model's ability to handle local features.
- The Half-Wavelet Attention Block (HWAB) [34] is introduced to enhance feature extraction through wavelet transformations and Dual Attention Units (DAU). HWAB processes input features through segmentation, transformation, and inverse transformation, thus improving their representational capacity. Through feature fusion and residual connections, these enhancements ultimately boost the model's ability to capture fine details.
- Replace the CIoU loss function with Wise-IoUv3 [35], which utilizes a dynamic non-monotonic focusing mechanism to enhance detection performance. This mechanism adjusts gradient weights based on the quality of anchor boxes, effectively reducing the impact of low-quality predictions. Furthermore, the loss function diminishes reliance on geometric penalties when there is significant overlap in box heights, thereby improving the model's generalization ability and overall performance.

The document is organized as follows: Chapter 2 describes the overall structure of the methods and the implementation details of each module. Chapter 3 validates the effectiveness of these methods through experimental tests. Chapter 4 concludes the discussion.

II. Methodology

A. Lightweight Backbone Network

In modern computer vision tasks, object detection is widely applied. However, as deep learning models become increasingly complex and large, the demand for computational resources also grows. To achieve efficient object detection in resource-limited environments, the design of lightweight networks is particularly critical. YOLOv8, an efficient object detection model, employs the improved CSP-Darknet53 as its backbone network. Despite its excellent performance, further lightweight optimization is still required for UAV aerial target detection scenarios. EfficientNetV2, through a compound scaling strategy that adjusts the depth, width, and resolution of the network, achieves an optimized balance between performance and efficiency, with a higher computational efficiency compared to CSPDarknet53. Therefore, this study replaces the backbone network of YOLOv8 with the improved EfficientNetV2 to significantly reduce the model's parameters and computational load while maintaining high detection accuracy, thus enhancing computational efficiency and real-time performance.

1) CBAM-MBConv Module:

The MBConv (Mobile-inverted Bottleneck Convolution) is a core convolution module in EfficientNetv2, while CBAM (Convolutional Block Attention Module) represents an innovative attention mechanism that integrates spatial and channel attentions. This study innovatively designs the CBAM-MBConv module by combining these two modules, aimed at enhancing feature extraction capabilities and overall model performance.

The MBConv module was first introduced in MobileNetV2 and has been widely used in the EfficientNet series, with its main structure shown in Fig. 1(a). This module employs an innovative inverted residual structure that expands and then compresses the feature maps to reduce both parameters and computational load. The introduced depthwise separable convolutions, which are divided into depthwise and pointwise convolutions, significantly reduce the computational complexity.In EfficientNetV2, to further reduce parameter count and optimize computational efficiency, the Fused-MBConv was designed for use in the shallower layers of the network. Its main structure shown in Fig. 1(c).

CBAM is a lightweight attention mechanism that enhances feature representation by combining channel and spatial attentions, with its main structure displayed in Fig. 2.

As depicted in Fig. 3, channel attention processes the input feature maps through parallel max pooling and average pooling layers, compressing the dimensions from C×H×W to C×1×1. The features are then processed by a shared multi-layer perceptron (MLP) structure which initially reduces the number of features per channel to 1/R of the original through dimension reduction. After dimension reduction, the MLP expands back to the original number of channels and reweights the importance of each channel. This design not only optimizes parameter efficiency but also enhances learning and expression of key feature channels. The processed features are activated by the ReLU function, producing two activated results. These results are element-wise added, and through a sigmoid activation function, the final output of the channel attention is generated. This output is then multiplied

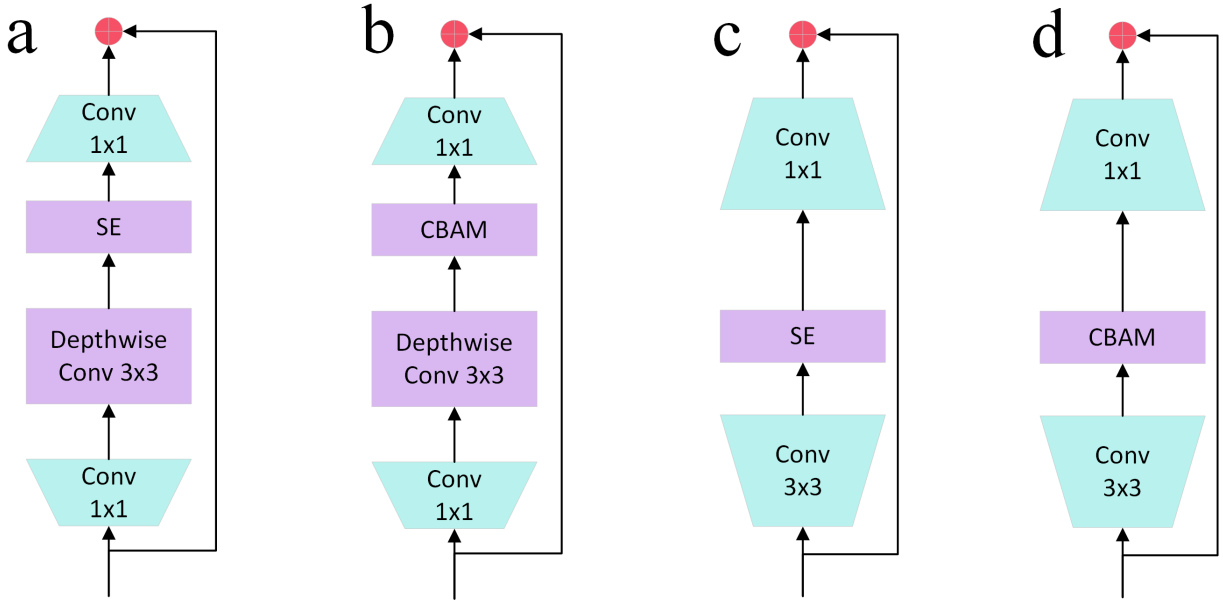


Fig. 1: Structure of MBConv. (a) MBConv. (b) CBAM-MBConv. (c) Fused-MBConv. (d) CBAM-Fused-MBConv

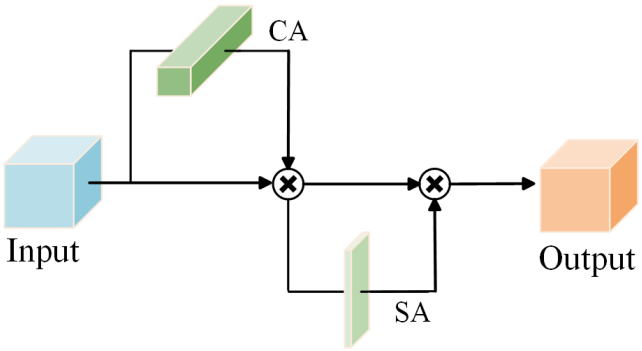


Fig. 2: Structure of CBMA.

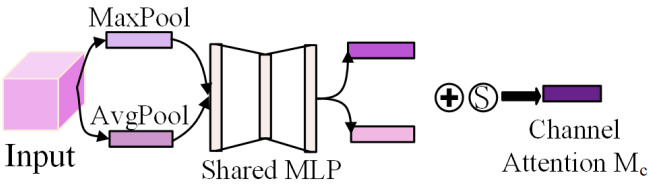


Fig. 3: Structure of CA.

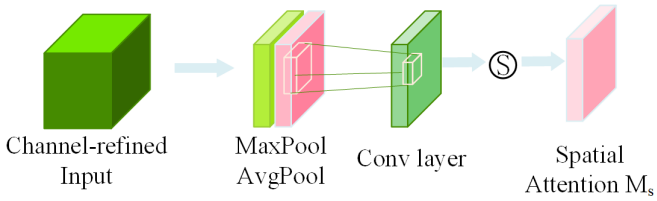


Fig. 4: Structure of SA.

back with the original image, restoring the dimensions to C×H×W.

As shown in Fig. 4, spatial attention processes the output from channel attention through max pooling and average pooling to produce two feature maps of size 1×H×W. These feature maps are then concatenated using a Concat operation. They are transformed into a single-channel feature map via a 7×7 convolution, and this map is subsequently processed with a sigmoid function to generate the final spatial feature map. Lastly, this spatial feature map is multiplied by the original image, restoring the dimensions to C×H×W.

The improved CBAM-MBConv replaces the SE attention module with the CBAM attention module. The CBMA module combines spatial and channel attentions, outperforming the SE module's singular channel attention design. Initially, the CBAM module applies spatial attention to focus on critical spatial regions to extract contextual information, allowing the network to more accurately locate information-rich areas in aerial images. Subsequently, the channel attention mechanism encodes global information for each channel and adjusts the feature responses based on inter-channel dependencies. This dual attention strategy enables the CBAM module to enhance features by not only boosting responses of crucial channels but also retaining information about key spatial locations, resulting in a more detailed and comprehensive feature representation.

2) Improving Activation Functions:

EfficientNetV2 employs the Swish activation function, which excels in many tasks. However, in complex scenarios, the Mish activation function offers superior nonlinear transformations, enhancing model performance. This study replaces the Swish activation function in EfficientNetV2 with Mish to enhance the model's feature extraction capabilities. The definition of the Swish activation function is as follows:

$$\text{Swish}(x) = x \cdot \sigma(x) \tag{1}$$

Swish is known for its smoothness and non-monotonic properties, which aid gradient-based optimization processes. However, in complex scenarios, Swish exhibits some drawbacks such as: (1) Gradient vanishing problem: In extreme negative regions, Swish's gradient approaches zero, potentially hindering the training of deep networks. (2) Saturation phenomenon: Swish's output tends to become linear

in extreme positive regions, possibly limiting the model's ability to express non-linearity. (3) Computational complexity: Swish relies on the sigmoid function, which increases computational complexity. In contrast, Mish addresses these deficiencies through its unique formula design. The computation process of Mish is as follows:

$$\text{Mish}(x) = x \cdot \tanh(\ln(1 + e^x)) \tag{2}$$

Firstly, $\ln(1 + e^x)$ transforms the input x into a nonlinear function, ensuring that the output remains within a certain range. This is followed by the hyperbolic tangent function $\tanh$, which further enhances nonlinearity. Finally, by multiplying by the input x, the scale information is preserved while introducing further nonlinear transformations. Through these operations, the Mish function effectively enhances the nonlinearity of the output features, smooths the gradients, avoids saturation phenomena, and improves the activation effects.

3) Improved Backbone Network:

This study has designed a lightweight backbone network based on CBMA-MBConv, taking inspiration from EfficientNetV2. CBMA-MBConv is utilized in the deeper layers of the network, while Fused-CBMA-MBConv is applied in the shallower layers. The specific designs for each stage are detailed in Table I.

TABLE I: Backbone Network Design

Stage	Module	Stride	Channel	Layers
1	Conv	2	16	1
2	Fused-CBAM-MBConv	1	32	2
3	Fused-CBAM-MBConv	2	64	4
4	CBAM-MBConv	2	128	6
5	CBAM-MBConv	1	256	3

B. Multi-scale Feature Fusion Structure

Yolov8 integrates Path Aggregation Network (PAN) and Feature Pyramid Network (FPN) to construct a network structure featuring both top-down and bottom-up pathways. In this structure, feature fusion effectively complements shallow spatial information with deep semantic information. However, each convolutional layer in the backbone network has a stride of 2, reducing the size of the output feature maps to one quarter of the input image. This significant downscaling results in less smooth transitions between adjacent layers, particularly evident in the extraction of target features from UAV aerial images.

To address this issue, this study introduces a new feature fusion architecture, the Multi-Scale Feature Aggregation Pyramid (MFPN). This structure aims to tackle the rapid reduction in feature map size caused by large strides in the backbone network. The implementation involves downsampling or upsampling the feature maps output from the backbone network to match the dimensions of intermediate layer feature maps. Subsequently, the feature maps are merged through an average-weighted fusion operation, addressing the channel multiplication issue caused by Concat operations, and further multi-scale features are extracted through a 3×3 convolution layer.

To enhance the detection capabilities for small targets, this study introduces a new detection scale, H_2, into the model. Positioned in the shallower layers of the network, H_2 has a smaller receptive field, which preserves richer local spatial semantics. Combining the H_2 detection head with existing larger-scale detection heads effectively reduces the loss of local semantics caused by scale expansion, thereby improving detection performance for small targets. Although this addition increases computational and memory overheads, it significantly enhances the accuracy of detecting small targets. The overall structure of the model's neck before and after improvements is shown in Fig. 5(b).

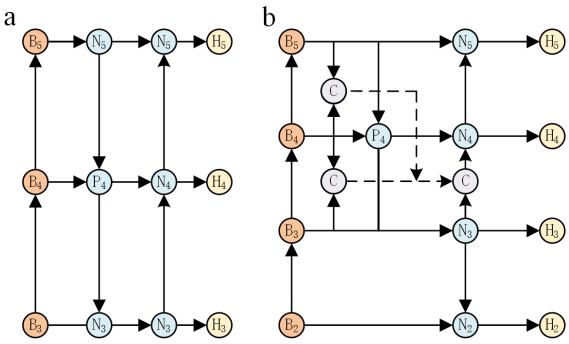


Fig. 5: Structure of Neck. (a) PAN-FPN. (b) MFPN.

C. Multi-Scale Feature Fusion Module (MSFF)

In the YOLOv8 architecture, the Spatial Pyramid Pooling Fast (SPPF) structure is applied to the last layer of the backbone network to enhance multi-scale feature extraction and object detection performance. This module significantly accelerates processing speed through consecutive max pooling operations. However, in practical applications such as UAV aerial imagery, due to complex image features and the presence of small objects, the original model may lose some spatial information. Furthermore, max pooling might retain noise as significant features, leading to false detections.

This study introduces a new feature fusion module, the Multi-Scale Feature Fusion (MSFF), as illustrated in Fig. 6. The module consists of multiple convolutional(MC) modules and feature enhancement(FE) modules working in parallel. The MSFF module outputs a feature space rich in information, as shown in Fig. 6, including multi-scale, channel-level, and spatial-level feature information.

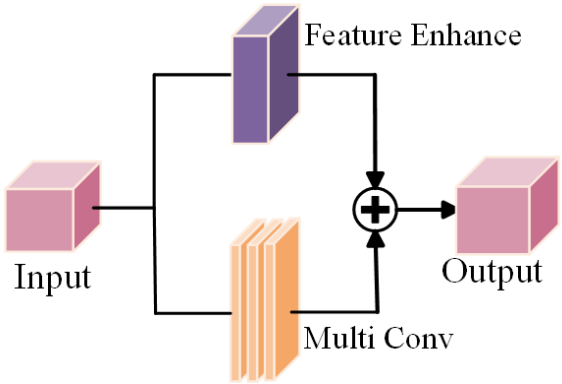


Fig. 6: Structure of MSFF.

The FE Module consists of channel-spatial attention, serving as an improvement over the traditional SE module. Its

structure is shown in Fig. 7. Unlike the SE module, which focuses solely on channel-wise feature representations, the FE module considers both channel and spatial aspects of feature representation. This allows the model to perform attention calculations simultaneously on channel and spatial dimensions, enabling more comprehensive extraction and representation of feature information.

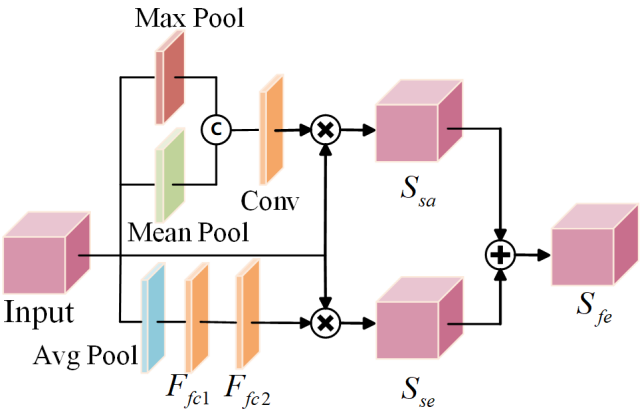


Fig. 7: Structure of FE Module.

The FE module performs parallel computations of channel and spatial attention weights. The channel attention component, SE, first performs global average pooling on each channel to produce a feature space of C×1×1. This is followed by the calculation of excitation weights through two fully connected layers. Finally, these weights are multiplied by the feature space to produce the enhanced feature space S_{se}. The specific formula is as follows:

$$S_{se} = \mathbf{X} \cdot (F_{fc2}(F_{fc1}(\mathbf{X}))) \tag{3}$$

$$F_{fc1} = F_{Relu}(F_{linear}(\mathbf{y})) \tag{4}$$

$$F_{fc2} = F_{Sigmoid}(F_{linear}(\mathbf{y})) \tag{5}$$

Here, F_{fc1} represents the first fully connected layer performing a linear transformation followed by the ReLU activation function, and F_{fc2} represents the second fully connected layer performing a linear transformation and then normalizing the weight parameters using the Sigmoid function. The symbol '·' denotes element-wise multiplication across corresponding channels.

The Spatial Attention (SA) Module performs max pooling and average pooling across all channels of the feature space to produce two 1×H×W feature maps. These feature maps are then concatenated using a Concat operation, followed by convolution using a convolution kernel. The specific formula is as follows:

$$S_{sa} = \mathbf{X} \cdot F_{conv}^{7\times7}(F_{cat}(F_{mean}(\mathbf{X}), F_{max}(\mathbf{X}))) \tag{6}$$

Ultimately, the output of the complete FE module is obtained by adding the output feature maps from the SE and SA modules, as Equation (7) show.

$$S_{fe} = S_{se} + S_{sa} \tag{7}$$

The structure of the MC module is shown in Fig. 8. This module helps expand the receptive field and provides richer feature information. Unlike traditional multi-scale models, the MC module utilizes Depthwise Separable Convolutions (DWConv) instead of dilated or varied scale convolutions. DWConv decomposes standard convolutions into depth-wise and pointwise convolutions, reducing the number of parameters and computational complexity. By employing DWConv, the MC module significantly enhances precision while maintaining minimal parameters and computational complexity. Despite the reduced complexity, the MC module still achieves substantial precision improvements.

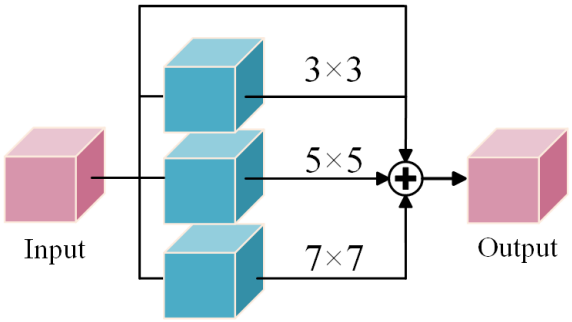


Fig. 8: Structure of MC Module.

The input to the MC module is represented as $[x_1, \ldots, x_C] \in \mathbb{R}^{C\times H\times W}$, denoted as $\mathbf{X}$. The multi-scale feature space mapping is obtained from the MC module. The specific formula is as follows:

$$H_i = [h_1^i, h_2^i, \ldots, h_C^i] \in \mathbb{R}^{C\times H\times W}, \quad i = 1, 2, 3 \tag{8}$$

$$H = H_1 + H_2 + H_3 + \mathbf{X} \tag{9}$$

In this context, Equation (8) describes the convolution operation on the feature space, with convolution kernels of channel count C and sizes 3x3, 5x5, and 7x7 respectively. In Equation (9) , the symbol "+" represents feature fusion, which enhances the information content of the feature maps while maintaining the same number of channels.

After parallel computations by the FE and MC modules, the output feature space S of the MSFF module can be described as follows:

$$S = H + S_{fe} \tag{10}$$

In the MSFF module, the FE and MC modules operate in parallel. Unlike traditional cascading connections, this parallel configuration prevents the loss of multi-scale feature information, ensuring that the output feature space encompasses multi-scale, channel-level, and spatial-level features. Consequently, incorporating the MSFF module significantly enhances the detection capabilities for small aerial targets.

D. Efficient Local Attention Module

The shallow feature maps of the network have a smaller receptive field and richer local semantics, capturing low-level features of the image such as edges, textures, and colors, which typically exhibit significant locality. Consequently, this study introduces an Efficient Local Attention Module (ELAM) into the shallow local information enhancement module to better capture and utilize these local features, thereby enhancing model performance.

The core idea of the ELAM, illustrated in Fig. 9, is to enhance the expression of positional information to improve the efficiency of feature extraction. Initially, positional

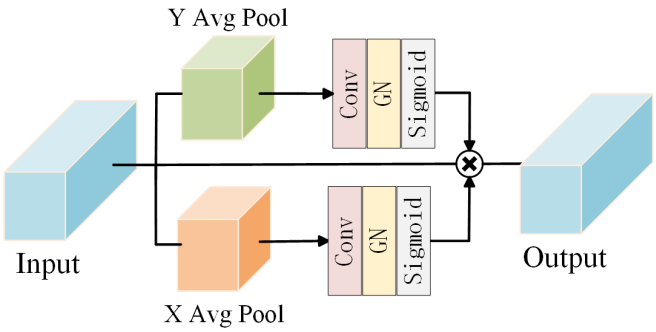


Fig. 9: Structure of ELAM.

information embeddings are extracted from sequence signals within channels, which are more suitably processed by 1D convolution than by traditional 2D convolution. 1D convolution is not only lighter but also more effective in processing sequence signals. In contrast, although traditional CA methods using 2D convolution enhance some feature extraction capabilities, the limitations of 1×1 convolution kernels result in constraints on feature extraction.

To overcome these limitations, the ELAM employs 1D convolution kernels of size 5 or 7 to enhance positional information embedding. This design enables ELAM to more effectively capture spatial dependencies within areas of interest. Subsequently, the module processes the enhanced positional information using Group Normalization (GN), resulting in horizontal and vertical positional attention maps. Specifically, horizontal and vertical positional information, processed by 1D convolutions denoted as F_h and F_w respectively, and Group Normalization, produces positional attention maps y_h and y_w. These attention maps are then combined with the input feature maps to produce the final output Y of the ELA module. The specific computation formula is as follows:

$$y_h = \sigma(Gn(F_h(z_h))) \tag{11}$$

$$y_w = \sigma(Gn(F_w(zw))) \tag{12}$$

$$Y = x_c \times y_h \times y_w \tag{13}$$

E. Half-Wavelet Attention Module

The architecture of the Half-Wavelet Attention Module (HWAB) is a modification of the Dual Attention Unit (DAU). The DAU extracts features using both channel and spatial attentions and integrates these features to capture richer contextual information. HWAB further incorporates wavelet transformations to enhance the efficacy of feature extraction. The structure of HWAB is illustrated in Fig. 10 and includes the following steps:

1) Feature Splitting: The input feature map $f_{\text{in}} \in \mathbb{R}^{C\times H\times W}$ is initially split along the channel dimension into two parts, f_{identity} and f_t, each sized $\mathbb{R}^{\frac{C}{2}\times H\times W}$. This division aims to reduce computational complexity while preserving essential contextual information. f_{identity} retains the original domain features, which are not subjected to wavelet transformation, while f_t undergoes Discrete Wavelet Transformation (DWT) to produce the wavelet domain features f_w.

2) Discrete Wavelet Transform (DWT): The feature f_t undergoes DWT to produce the wavelet domain features $f_w \in \mathbb{R}^{2C\times\frac{H}{2}\times\frac{W}{2}}$, transforming the feature maps into the frequency domain for more efficient feature extraction.

3) Weighted Wavelet Features: The wavelet domain features f_w are processed through the Dual Attention Unit (DAU) to obtain weighted wavelet features $\hat{f}_w \in \mathbb{R}^{2C\times\frac{H}{2}\times\frac{W}{2}}$, where DAU enhances feature expression by integrating channel and spatial attentions.

4) Inverse Wavelet Transform (IWT): The weighted wavelet features $\hat{f}_w$ are transformed back to the original size using IWT to produce $\hat{f}_t$.

5) Feature Fusion and Residual Features: $\hat{f}_t$ is combined with f_{identity}, processed through a 3x3 convolution layer and Parametric ReLU (PReLU) to create the residual features f_r.

6) Output Features: Finally, the shortcut features are added to the residual features f_r through a 1x1 convolution layer to produce the output features $f_{\text{out}} \in \mathbb{R}^{C\times H\times W}$, incorporating wavelet attention information.

F. Self-Attention Dynamic Detection Head

The default YOLOv8 model includes three detection heads located in the deeper layers, potentially resulting in the loss of shallow features and insufficient semantic representation in deep feature maps during small object detection tasks. Due to the richer semantic and more accurate positional information in shallow feature maps, researchers have attempted to enhance small object detection by adding additional detection heads to capture more shallow features. However, variations in scale complicate these heads' ability to comprehensively understand information across all scales. Addressing these challenges, this study introduces a Self-Attention Dynamic Detection Head (SDHead).

SDHead processes the input feature map of size $C\times H\times W$ with scale-aware, spatial-aware, and task-aware capabilities. The computation formula for its attention function is as follows:

$$W(F) = \pi_C(\pi_S(\pi_L(F)\cdot F)\cdot F)\cdot F \tag{14}$$

$$\pi_L(F)\cdot F = \sigma\left(f\left(\frac{1}{SC}\sum_{SC}F\right)\right)\cdot F \tag{15}$$

$$\pi_S(F)\cdot F = \frac{1}{L}\sum_{l=1}^{L}\sum_{k=1}^{K} w_{l,k} \cdot F(l;p_k+\Delta p_k;c)\cdot\Delta m_k \tag{16}$$

$$\pi_C(F)\cdot F = \max\begin{pmatrix}\alpha^1(F)\cdot F_C+\beta^1(F),\\ \alpha^2(F)\cdot F_C+\beta^2(F)\end{pmatrix} \tag{17}$$

In SDHead, $f(\cdot)$ is approximated by a 1x1 convolution layer as a linear function. The activation function $\sigma(x) = \max(0,\min(1,x+12))$ is an adapted sigmoid function. The spatial-aware attention module, leveraging feature fusion, targets critical discriminative regions between spatial locations and feature layers. This module employs deformable convolutions to enable sparse learning of attention and aggregates cross-layer features at the same spatial locations. The number of sparse sampling locations is denoted by K, and the position $p_k+\Delta p_k$ is adjusted by a spatial offset Δp_k to focus on discriminative regions, with Δm_k representing the

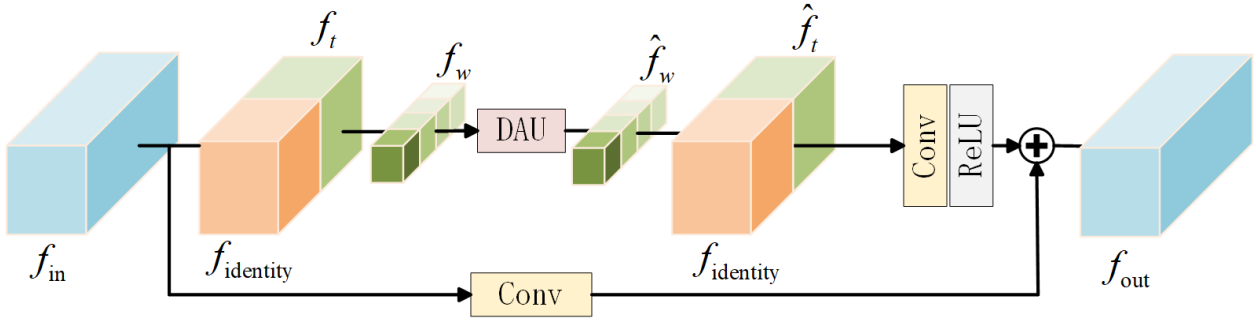


Fig. 10: Structure of HWAB.

importance weights at position p_k, learned from intermediate input features.

Task-aware attention dynamically adjusts to support tasks across different scales. F_C represents the feature slice of channel C. The function $\left[\alpha^1, \beta^1, \alpha^2, \beta^2\right]^T = \theta(\cdot)$ is a hyper-function that learns activation thresholds.

G. WIoUv3 Loss Function

The diagram of the loss function parameters is shown in Fig. 11, where "Real box" represents the ground truth or label box, and "Predicted box" represents the bounding box predicted by the algorithm. The coordinates $(b_{cx}^{gt}, b_{cy}^{gt})$ denote the center of the ground truth box, while (b_{cx}, b_{cy}) denote the center of the predicted box. The dimensions w_{gt} and h_{gt} represent the width and height of the ground truth box, respectively, while w and h represent the width and height of the predicted box. The terms c_w and c_h denote the width and height of the minimal bounding box that encompasses both the predicted and ground truth boxes. The Euclidean distances $\rho(w, w_{gt})$ and $\rho(h, h_{gt})$ measure the differences in width and height between the predicted and ground truth boxes, respectively. These parameters are used to calculate the discrepancy between the predicted and ground truth boxes, aiding in the optimization of model predictions.

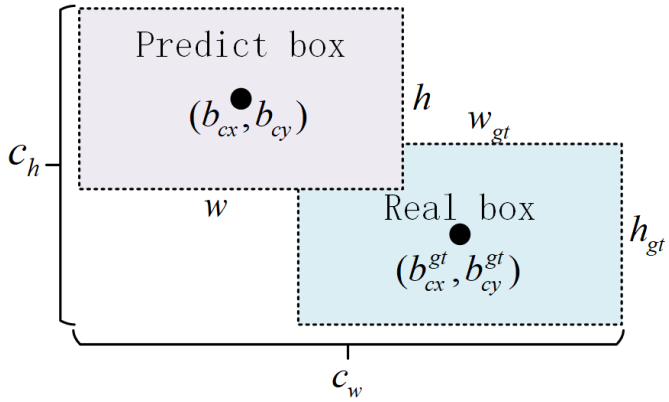


Fig. 11: Parameters of Loss Function.

YOLOv8 employs the Complete Intersection over Union (CIoU) loss function, whose formula is presented in Equation (18). However, CIoU's performance in drone aerial imagery applications is suboptimal for several reasons. Firstly, CIoU uses the aspect ratio of the predicted and ground truth boxes as a penalization factor. If the aspect ratios are the same, but the widths and heights differ, no penalty is applied, which can be detrimental to recognition accuracy. Furthermore, the computational cost of CIoU is significant, especially due to the calculations involving inverse trigonometric functions, which can slow down the detection speed.

$$L_{\mathrm{CIoU}} = 1 - \mathrm{IoU} + \frac{\rho^2(b, b_{\mathrm{gt}})}{c_w^2 + c_h^2} + \frac{4}{\pi^2}\left(\arctan\frac{h_{\mathrm{gt}}}{w_{\mathrm{gt}}} - \arctan\frac{h}{w}\right)^2 \tag{18}$$

The Wise-IoU loss function incorporates a dynamic non-monotonic focusing mechanism, significantly enhancing the detection performance of the algorithm. Wise-IoU is available in three versions: WIoUv1, WIoUv2, and WIoUv3. The calculation formula for WIoUv1 is presented in Equations (19) to (21). As the foundational version, WIoUv1 introduces distance as a metric for attention. When there is a certain overlap between the target and predicted boxes, it reduces the penalties based on geometric measurements, thereby improving the model's generalization ability.

$$L_{\mathrm{WIoUv1}} = R_{\mathrm{WIoU}} \times L_{\mathrm{IoU}} \tag{19}$$

$$R_{\mathrm{WIoU}} = \exp\left(\frac{(b_{cx}^{gt} - b_{cx})^2 + (b_{cy}^{gt} - b_{cy})^2}{c_w^2 + c_h^2}\right) \tag{20}$$

$$L_{\mathrm{IoU}} = 1 - \mathrm{IoU} \tag{21}$$

WIoUv3 is an enhancement over WIoUv1, detailed in Equations (22) to (24). It defines the quality of anchor boxes using the outlier value β, which is used to construct a non-monotonic focusing factor r. A higher β value indicates lower anchor box quality, resulting in a smaller r value and reduced gradient gains, thereby minimizing harmful gradients caused by low-quality anchor boxes. WIoUv3 employs a judicious gradient gain allocation strategy, dynamically adjusting the weights of high and low-quality anchor boxes in the loss function. This approach ensures the model focuses on medium-quality samples, thereby enhancing overall performance.

$$L_{\mathrm{WIoUv3}} = r \times L_{\mathrm{WIoUv1}} \tag{22}$$

$$r = \frac{\beta}{\delta \alpha^{\beta - \delta}} \tag{23}$$

$$\beta = \frac{L_{\mathrm{IoU}}^*}{L_{\mathrm{IoU}}} \in [0, +\infty) \tag{24}$$

In this study, the WIoUv3 loss function replaces the CIoU previously used in YOLOv8. This substitution aims to address specific challenges effectively. On one hand, WIoUv3

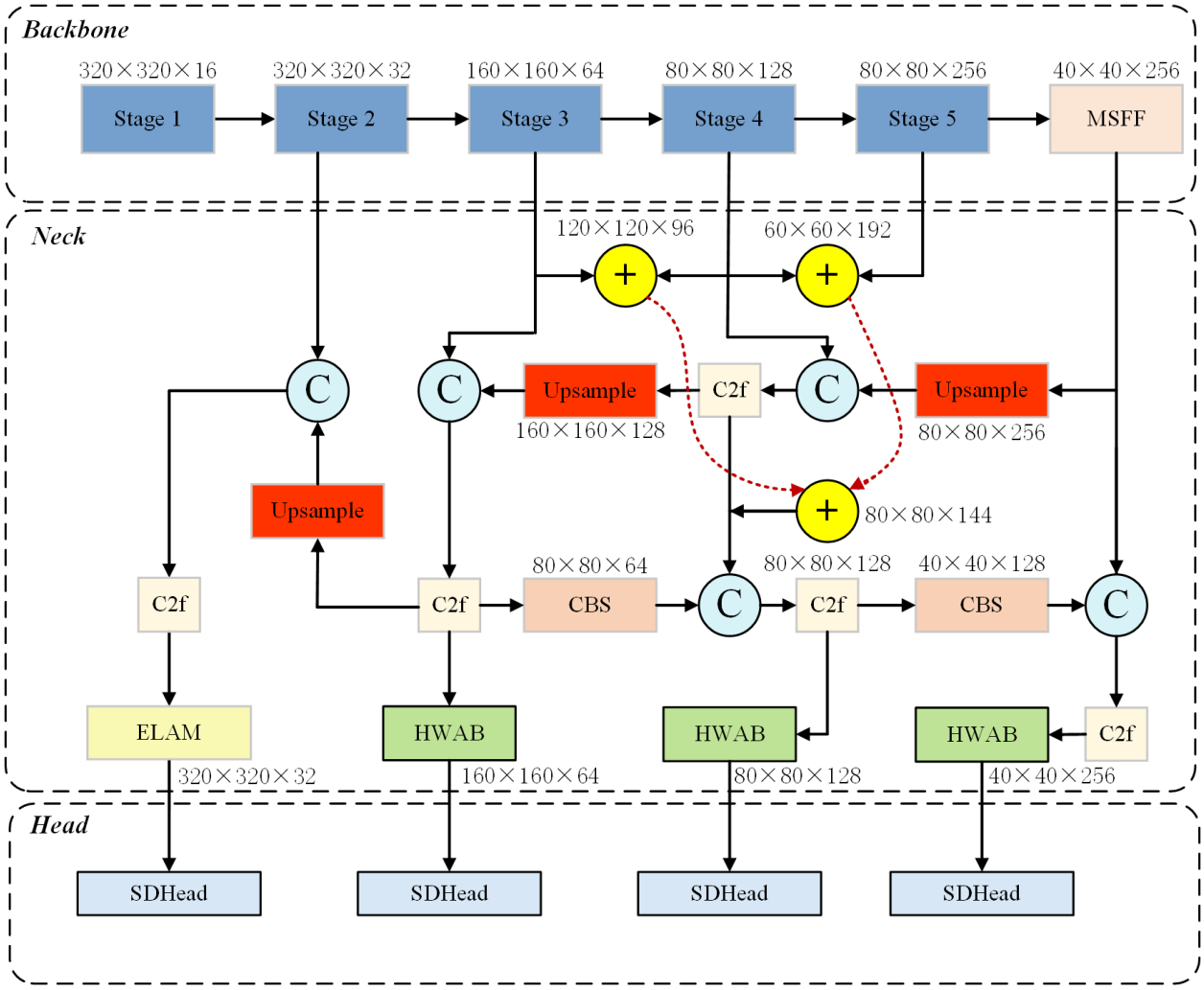


Fig. 12: Structure of Our Network.

mitigates the excessive penalization of distance and aspect ratio, especially when the quality of training data annotations is low, through its dynamic non-monotonic focusing mechanism. On the other hand, when there is a high overlap between the predicted and target boxes, WIoUv3 lessens the penalty on geometric factors, thereby enhancing the model's generalization ability with minimal training intervention.

H. Network Structure

The structure of the algorithm proposed in this paper is illustrated in Figure 12. Incorporating the modules discussed earlier, we have modified the YOLOv8n network to implement a lightweight target detection algorithm that enhances the perception of small targets.

III. EXPERIMENTS

A. Dataset

The experiments in this paper are primarily based on the VisDrone-2019 [36] dataset, one of the mainstream drone aerial imagery datasets, collected by the Machine Learning and Data Mining team at Tianjin University. The VisDrone-2019 dataset includes 6,471 training images, 548 validation images, and 1,610 test images. This dataset covers 10 categories, all of which are common entities in aerial tasks such as cars, pedestrians, and bicycles.

TABLE II: Experimental Hardware Environment

Parameter	Experimental Environment
CPU	Intel(R) Xeon(R) Platinum 8370C
GPU	RTX 3090 (24G)
Operating System	Ubuntu 20.04
Graphics Driver Version	520.56
CUDA Version	11.8
Python Version	3.11.5
Deep Learning Framework	Pytorch 2.0.1+cu118

B. Experimental Environment and Parameters

The experiments are based on the YOLOv8n network as the baseline, with the specific settings shown in Table II. All network training was conducted using the Adam optimizer with an initial learning rate of 0.01, a weight decay of 0.0005, a batch size of 16, and 300 epochs. The input image size was set to 640×640. All experiments employed default data augmentation methods. Ablation and comparative experiments were conducted under the same settings, without additional configurations or training.

C. Evaluation Metrics

This study evaluates the model's detection performance using three metrics: Mean Average Precision (mAP), the

TABLE III: Performance Comparison of Different Models on the VisDrone2019 Dataset

Method	mAP@0.5 (%)	mAP@0.5:0.95 (%)	FLOPs (GB)	Params (M)
YOLOv3-spp	36.8	22.6	155.4	62.7
YOLOv3-tiny	16.0	7.0	12.8	8.8
YOLOv3	35.3	17.0	154.6	61.6
YOLOv5n	24.0	12.3	4.1	1.8
YOLOv5s	29.7	16.3	15.7	7.2
YOLOv5m	37.1	20.9	47.9	20.9
YOLOv5l	36.7	21.0	107.6	46.2
YOLOv7-tiny	36.9	19.1	13.2	6.1
YOLOv7	47.6	26.3	103.2	36.7
YOLOv8n	33.8	19.7	8.2	3.1
YOLOv8s	38.3	22.9	28.4	11.1
YOLOv7X+ [37]	41.2	25.3	56.8	23.5
DUCAF-Net [38]	39.38	23.10	-	-
Ours	40.3	24.2	19.7	7.9

TABLE IV: Detection results after the introduction of different improvement strategies

Method	New Backbone	MSFF	MFPN+ELAM	HWAB	SDHead	WIoUv3	mAP@0.5 (%)	Params (M)	FLOPs (GB)
A							33.8	3.1	8.2
B	✓						34.1	2.8	6.2
C		✓					35.2	3.25	8.4
D			✓				38.1	5.5	10.8
E				✓			35.1	3.8	8.5
F					✓		35.3	3.9	8.1
G						✓	34.1	3.1	8.2
H	✓	✓	✓	✓	✓	✓	40.3	7.9	19.7

Note: ✓ represents the presence of the corresponding improvement strategy. The best results are indicated in bold.

number of parameters (Params), and Floating Point Operations (FLOPs).

Where mAP is used to assess the overall performance of the network model, with mAP@0.5 representing the mean Average Precision at an IoU threshold of 0.5. FLOPs indicate the number of floating-point operations per second, understood as the computational speed, which can be used to evaluate hardware performance.

The relevant formulas are as follows:

$$P = \frac{TP}{TP+FP} \tag{25}$$

$$R = \frac{TP}{TP+FN} \tag{26}$$

$$AP = \int_0^1 P(R)\mathrm{d}R \tag{27}$$

$$mAP = \frac{\sum_{i=1}^{k} APi}{k} \tag{28}$$

$$FLOPs = Ci \times K^2 \times C \times W \times H \tag{29}$$

D. Comparison Experiment

To demonstrate the effectiveness of the proposed algorithm in detecting small targets in drone aerial imagery, we conducted a comparative analysis using the VisDrone dataset against classical mainstream models. As shown in Table III, the values in bold represent the best results for each category across all algorithms.

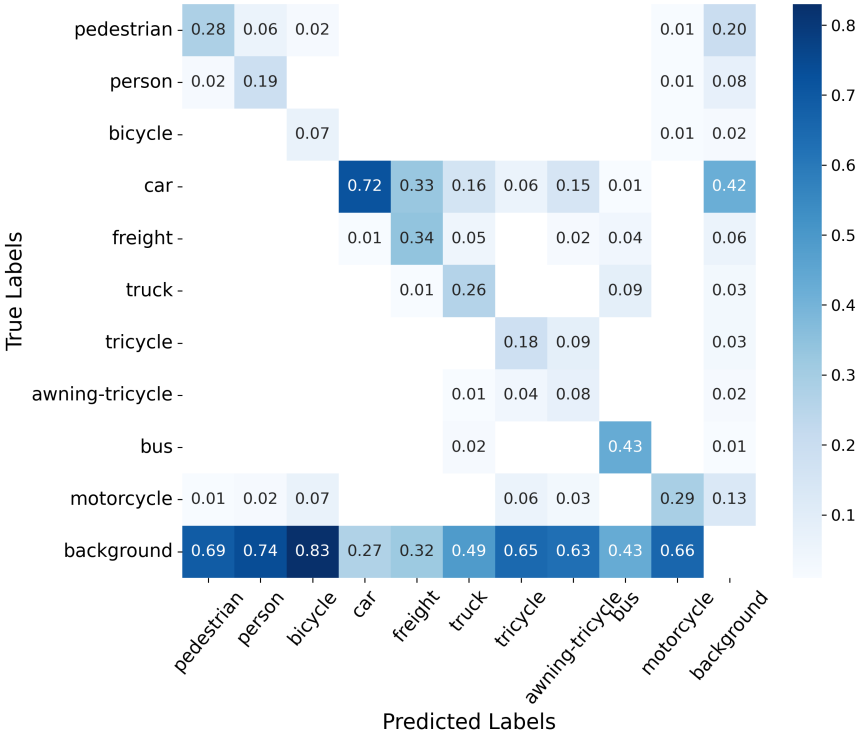
E. Ablation Experiments

To validate the effectiveness of the improved algorithm, a series of ablation experiments were conducted on the baseline model YOLOv8n using the VisDrone-2019 dataset, as shown in Table IV. Various enhancement strategies led to different degrees of improvement in mAP@0.5, with method G, which integrates all introduced strategies, achieving the best overall performance at 30.5% mAP@0.5. With the introduction of these strategies, there was an increase in the number of model parameters and floating-point operations (FLOPs), indicating a trade-off between model complexity and performance enhancement. Nevertheless, the increase in parameters and FLOPs is relatively modest compared to the significant improvements in detection accuracy.

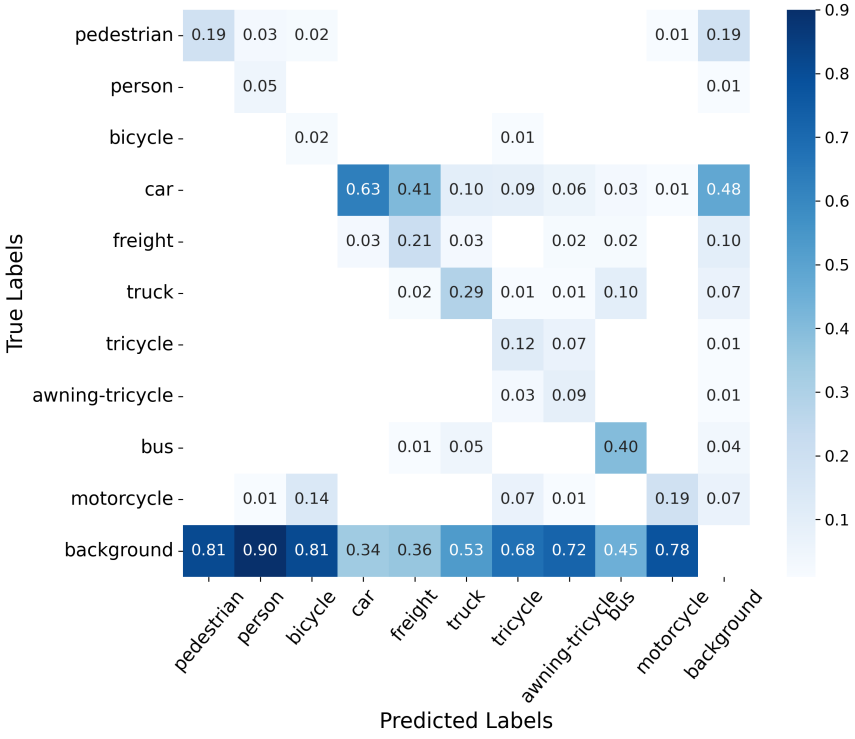
F. Interpretability Analysis

This study constructed confusion matrices and generated heatmaps to interpret the model's inference process, systematically assessing diagnostic accuracy. We demonstrated the effectiveness of the improved method and the YOLOv8n model in category recognition. As shown in Fig. 13, the rows in the confusion matrices represent actual categories, while the columns represent predicted categories. Values on the diagonal indicate the percentage of correct classifications, while off-diagonal values show the proportion of misclassifications.

As depicted in Figure 13a, the confusion matrix of the proposed method has darker areas along the diagonal, indicating enhanced performance in accurately predicting target categories. As shown in Figure 13b, the baseline model displayed a higher proportion of misclassifying small targets such as humans, bicycles, and awning tricycles as background, signifying significant miss rates for these categories. Although the improved model has reduced miss rates for these categories, the proportion of correct predictions remains to be improved. Moreover, the model excels in handling larger objects, such as buses, with varying degrees of improved detection accuracy.



(a) Confusion Matrix Plot of Our Model



(b) Confusion Matrix Plot of YOLOv8n.

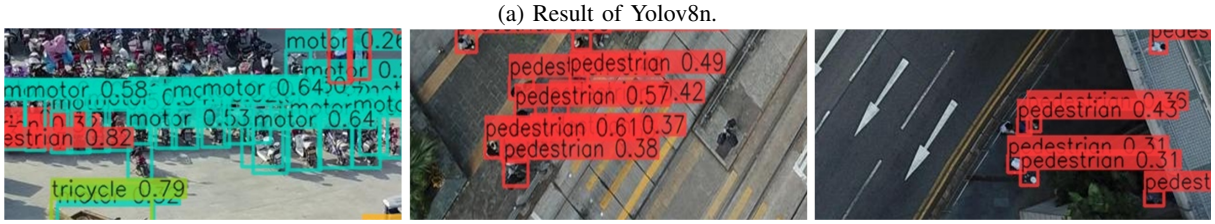
Fig. 13: Confusion Matrix Plot.

G. Vision Analysis

To evaluate the performance of different algorithms in real-world scenarios, we conducted tests using the VisDrone-2019 dataset and recorded the results. The final test outcomes are shown in Fig. 14 and Fig. 15, where subfigure (a) illustrates the detection results of YOLOv8n, and subfigure (b) presents the results of our proposed algorithm.

As seen in Fig. 14, YOLOv8n exhibited a higher number of missed detections, particularly for small targets such as pedestrians and motorcycles, while our algorithm significantly reduced the miss rate. Additionally, in Fig. 15, YOLOv8n demonstrated more false detections, especially for small objects like cars and pedestrians, whereas our algorithm effectively minimized these errors. In the third

(a) Result of Yolov8n.



(b) Result of Our Model.

Fig. 14: Scenarios for Obvious Missed Detection.

(a) Result of Yolov8n.

(b) Result of Our Model.

Fig. 15: Scenarios for Obvious Error Detection.

comparison in Fig. 15, YOLOv8n mistakenly detected a ground-level container as a truck, while our method avoided this error, as it did not wrongly classify the container as a truck due to its enhanced ability to detect small objects and recognize the absence of essential truck features like the cab and wheels.

Overall, our algorithm achieved notable improvements in detection accuracy across various types of objects.

IV. Conclusion

This paper presents an improved drone aerial target detection algorithm based on the YOLOv8n, significantly enhancing the accuracy of small target detection through innovative strategies. Initially, a shallow feature enhancement module merges spatial and semantic information from different layers, substantially increasing detection accuracy for small targets such as pedestrians. Moreover, by embedding mid-scale feature synthesis layers and skip connections into the traditional feature pyramid structure, the algorithm smooths feature transitions and reduces feature loss. Additionally, the incorporation of the MSFF module effectively extracts and merges features, lowering the miss rate.

In terms of efficiency, the algorithm utilizes partial and depth-wise separable convolutions, reducing computational complexity and ensuring rapid inference. The HWAB attention mechanism further refines the focus on critical information, enhancing overall performance. Testing on the VisDrone-2019 dataset shows that this algorithm significantly outperforms the baseline model, especially in detecting small targets. Compared to other classic algorithms, this algorithm has lower parameters and higher detection accuracy, demonstrating extensive potential for applications.

Future work will focus on optimizing computational efficiency and inference time to better adapt to the deployment needs of edge devices, aiming to enhance the efficiency and accuracy of drone aerial target detection.

References

[1] N. Mohamed, J. Al-Jaroodi, I. Jawhar, A. Idries, and F. Mohammed, "Unmanned aerial vehicles applications in future smart cities," *Technological forecasting and social change*, vol. 153, p.119293, 2020.

[2] H. Yao, R. Qin, and X. Chen, "Unmanned aerial vehicle for remote sensing applications—a review," *Remote Sensing*, vol. 11, no. 12, p.1443, 2019.

[3] L. Kang, Z. Lu, L. Meng, and Z. Gao, "Yolo-fa: Type-1 fuzzy attention based yolo detector for vehicle detection," *Expert Systems with Applications*, vol. 237, p.121209, 2024.

[4] S. A. H. Mohsan, N. Q. H. Othman, Y. Li, M. H. Alsharif, and M. A. Khan, "Unmanned aerial vehicles (uavs): Practical aspects, applications, open challenges, security issues, and future trends," *Intelligent Service Robotics*, vol. 16, no. 1, pp109–137, 2023.

[5] N. Sharma and R. D. Garg, "Advanced transportation safety using real-time gis-based alarming system for animal-prone zones and pothole areas," *Journal of transportation engineering, Part A: Systems*, vol. 149, no. 4, p.04023003, 2023.

[6] H. Gupta and O. P. Verma, "Monitoring and surveillance of urban road traffic using low altitude drone images: a deep learning approach," *Multimedia Tools and Applications*, vol. 81, no. 14, pp19 683–19 703, 2022.

[7] Y. Long, G.-S. Xia, S. Li, W. Yang, M. Y. Yang, X. X. Zhu, L. Zhang, and D. Li, "On creating benchmark dataset for aerial image interpretation: Reviews, guidances, and million-aid," *IEEE Journal of selected topics in applied earth observations and remote sensing*, vol. 14, pp4205–4230, 2021.

[8] X. Luo, Y. Wu, and L. Zhao, "Yolod: A target detection method for uav aerial imagery," *Remote Sensing*, vol. 14, no. 14, p.3240, 2022.

[9] A. Bouguettaya, H. Zarzour, A. Kechida, and A. M. Taberkit, "Vehicle detection from uav imagery with deep learning: A review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 11, pp6047–6067, 2021.

[10] P. Mittal, R. Singh, and A. Sharma, "Deep learning-based object detection in low-altitude uav datasets: A survey," *Image and Vision computing*, vol. 104, p.104046, 2020.

[11] L. P. Osco, J. M. Junior, A. P. M. Ramos, L. A. de Castro Jorge, S. N. Fatholahi, J. de Andrade Silva, E. T. Matsubara, H. Pistori, W. N. Gonçalves, and J. Li, "A review on deep learning in uav remote sensing," *International Journal of Applied Earth Observation and Geoinformation*, vol. 102, p.102456, 2021.

[12] G. Wang, Y. Chen, P. An, H. Hong, J. Hu, and T. Huang, "Uav-yolov8: A small-object-detection model based on improved yolov8 for uav aerial photography scenarios," *Sensors*, vol. 23, no. 16, p.7190, 2023.

[13] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using yolo: Challenges, architectural successors, datasets and applications," *multimedia Tools and Applications*, vol. 82, no. 6, pp9243–9275, 2023.

[14] D.-S. Bacea and F. Oniga, "Single stage architecture for improved accuracy real-time object detection on mobile devices," *Image and Vision Computing*, vol. 130, p.104613, 2023.

[15] A. Vijayakumar and S. Vairavasundaram, "Yolo-based object detection models: A review and its applications," *Multimedia Tools and Applications*, pp1–40, 2024.

[16] Y. Dai, X. Li, F. Zhou, Y. Qian, Y. Chen, and J. Yang, "One-stage cascade refinement networks for infrared small target detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp1–17, 2023.

[17] S. Wang, Y. Wang, Y. Chang, R. Zhao, and Y. She, "Ebse-yolo: high precision recognition algorithm for small target foreign object detection," *IEEE Access*, vol. 11, pp57 951–57 964, 2023.

[18] M. Li, S. Cheng, J. Cui, C. Li, Z. Li, C. Zhou, and C. Lv, "High-performance plant pest and disease detection based on model ensemble with inception module and cluster algorithm," *Plants*, vol. 12, no. 1, p.200, 2023.

[19] M. T. Hosain, A. Zaman, M. R. Abir, S. Akter, S. Mursalin, and S. S. Khan, "Synchronizing object detection: Applications, advancements and existing challenges," *IEEE Access*, 2024.

[20] K. Li, Y. Wang, and Z. Hu, "Improved yolov7 for small object detection algorithm based on attention and dynamic convolution," *Applied Sciences*, vol. 13, no. 16, p.9316, 2023.

[21] Z. Yu, H. Huang, W. Chen, Y. Su, Y. Liu, and X. Wang, "Yolo-facev2: A scale and occlusion aware face detector," *Pattern Recognition*, vol. 155, p.110714, 2024.

[22] C. Zhao, X. Shu, X. Yan, X. Zuo, and F. Zhu, "Rdd-yolo: A modified yolo for detection of steel surface defects," *Measurement*, vol. 214, p.112776, 2023.

[23] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp1440–1448.

[24] Y. Wang, S. M. A. Bashir, M. Khan, Q. Ullah, R. Wang, Y. Song, Z. Guo, and Y. Niu, "Remote sensing image super-resolution and object detection: Benchmark and state of the art," *Expert Systems with Applications*, vol. 197, p.116793, 2022.

[25] X. Ai, Q. He, and P. Zhang, "Analysis of deep learning object detection methods," in *Third International Conference on Machine Learning and Computer Application (ICMLCA 2022)*, vol. 12636. SPIE, 2023, pp271–279.

[26] C. Pan, J. Chen, and R. Huang, "Medical image detection and classification of renal incidentalomas based on yolov4+ asff swin transformer," *Journal of Radiation Research and Applied Sciences*, vol. 17, no. 2, p.100845, 2024.

[27] D.-Y. Zhang, W. Zhang, T. Cheng, X.-G. Zhou, Z. Yan, Y. Wu, G. Zhang, and X. Yang, "Detection of wheat scab fungus spores utilizing the yolov5-eca-asff network structure," *Computers and Electronics in Agriculture*, vol. 210, p.107953, 2023.

[28] M. Alaftekin, I. Pacal, and K. Cicek, "Real-time sign language recognition based on yolo algorithm," *Neural Computing and Applications*, vol. 36, no. 14, pp7609–7624, 2024.

[29] Y. Zhang, W. Cai, S. Fan, R. Song, and J. Jin, "Object detection based on yolov5 and ghostnet for orchard pests," *Information*, vol. 13, no. 11, p.548, 2022.

[30] J. Lin, D. Yu, R. Pan, J. Cai, J. Liu, L. Zhang, X. Wen, X. Peng, T. Cernava, S. Oufensou *et al.*, "Improved yolox-tiny network for detection of tobacco brown spot disease," *Frontiers in Plant Science*, vol. 14, p.1135105, 2023.

[31] H. Wang, Y. Jin, H. Ke, and X. Zhang, "Ddh-yolov5: improved yolov5 based on double iou-aware decoupled head for object detection," *Journal of Real-Time Image Processing*, vol. 19, no. 6, pp1023–1033, 2022.

[32] J. Chu, Y. Li, H. Feng, X. Weng, and Y. Ruan, "Research on multi-scale pest detection and identification method in granary based on improved yolov5," *Agriculture*, vol. 13, no. 2, p.364, 2023.

[33] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *International conference on machine learning*. PMLR, 2021, pp10 096–10 106.

[34] C.-M. Fan, T.-J. Liu, and K.-H. Liu, "Half wavelet attention on m-net+ for low-light image enhancement," in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp3878–3882.

[35] Z. Tong, Y. Chen, Z. Xu, and R. Yu, "Wise-iou: bounding box regression loss with dynamic focusing mechanism," *arXiv preprint arXiv:2301.10051*, 2023.

[36] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, and H. Ling, "Detection and tracking meet drones challenge," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp7380–7399, 2021.

[37] S. Li and W. Liu, "Small target detection model in aerial images based on yolov7x+." *Engineering Letters*, vol. 32, no. 2, pp436–443, 2024.

[38] Y. Bai, Z. Li, J. Wu, and X. Yu, "Ducaf-net: An object detection method for uav imagery." *Engineering Letters*, vol. 31, no. 4, pp1374–1382, 2023.

Guang Wang is an associate professor at the Software College, Liaoning Technical University. He participated in the completion of 2 projects of the National Science Natural Fund. The main completer of the completed research projects won the provincial scientific research awards above 3.

His research interests include: intelligent data processing, big data technology and image recognition research.

Yanming Zhang is a postgraduate student at the Software College, Liaoning Technical University. His reasearch interests include: deep learning, computer vision, target detection.

Qiang Ai(M'24) is currently a postgraduate student at the College of Computer, Qinghai Normal University. He received his bachelor's degree from Software College, Liaoning Technical University. He became a Member (M) of IAENG in 2024. He has been worked at the Institute of Information Engineering, Chinese Academy of Sciences. Now he is a collaborating scholar of Urban Computing Laboratory, Aisess (Dalian) Computer Services Co., Ltd.

His research interests include: spatial-temporal big data, knowledge graph and intelligent transportation system.