Small Object Detection in Aerial Drone Imagery based on YOLOv8

Junyu Pan, Yujun Zhang*

***Abstract*—In recent years, the utilization of unmanned aerial vehicles (UAVs) for aerial target detection has gained significant attention due to their high-altitude perspective and maneuverability, which offer novel opportunities and tremendous potential in this field. However, detecting targets in UAV aerial images remains highly challenging due to the presence of numerous small targets with limited feature information, as well as issues like target occlusion and complex backgrounds that severely impact detection accuracy. To address these challenges, we propose a detection model called BDC-YOLOv8 that aims to enhance accuracy for small targets while minimizing computational complexity. Specifically, we augment the YOLOv8 architecture by incorporating a dedicated detection head tailored for small targets to improve performance when encountering such objects. Additionally, we restructure the neck network of the model to better extract and fuse feature information from targets with significant scale variations. Furthermore, we introduce the concept of DynamicHead to enhance the detection head by incorporating various attention mechanisms suitable for our task ahead of the original detection head, thereby enhancing the model's capability to detect objects of different scales and complex backgrounds. Moreover, we introduce Convolutional Block Attention Module (CBAM) to identify regions of interest in densely populated areas. Extensive experiments conducted on the VisDrone2019 dataset yield promising results where our model achieves a mean Average Precision (mAP) score of 38% and an AP50 score of 59.6%. Compared to the original YOLOV8 model, improvements are observed with increases in mAP by 2.5% and AP50 by 3.7%, respectively. Notably, our model demonstrates a significant enhancement in detecting small targets with an increase in APs evaluation metric by 4.1%.**

***Index Terms*—Object detection, Small objects, Attention mechanism, Feature fusion.**

I. Introduction

THE unmanned aerial vehicle (UAV) technology has progressively matured over the past decade, rendering it widely applicable in diverse fields such as industry, agriculture, and military due to its portability, efficiency, and ease of deployment. Object detection plays a pivotal role in UAV missions. However, detecting small objects during UAV aerial photography poses a significant challenge compared to conventional-sized objects. Small objects often lack distinctive features and are prone to occlusion by other objects, thereby impeding accurate detection by models. Moreover, UAV aerial datasets frequently exhibit complex backgrounds and severe occlusions that introduce interference and hinder the detection of small objects. Henceforth, it is imperative to develop an efficient and effective network model specifically tailored for small object detection.

In recent years, significant advancements have been made in both deep learning and object detection techniques. The most cutting-edge object detection algorithms can be categorized into two main types: single-stage detectors and two-stage detectors. Single-stage detectors encompass the renowned YOLO series [1, 2] , SSD [3], and RetinaNet [4], while two-stage detectors include the RCNN series [5–7]. Although two-stage detectors generally achieve higher accuracy compared to single-stage detectors, they often encounter challenges related to training difficulty and detection speed. With the remarkable performance of transformer models [8] in natural language processing, researchers have begun incorporating transformers into computer vision [9–12], resulting in substantial improvements. However, transformers also introduce higher computational costs and deployment challenges, rendering them unsuitable for real-time tasks such as target detection and tracking in UAV aerial imagery. Many existing real-time detectors prioritize both accuracy and detection speed by utilizing CNN-based methods; however, these detectors exhibit suboptimal performance when dealing with small objects amidst complex backgrounds, making them less suitable for small object detection in UAV aerial imagery.

The proposed BDC-YOLOv8 algorithm is an enhancement of YOLOv8. Firstly, the neck network of YOLOv8 is reconstructed to incorporate bidirectional feature fusion for improved retention of low-level features. Additionally, a dedicated small object detection head is introduced to significantly enhance the accuracy in detecting small objects. Furthermore, the original detection head is improved by integrating multiple attention mechanisms, effectively enhancing the model's ability to handle targets with varying scales and complex backgrounds. To facilitate identification of regions of interest within images, a Convolutional Block Attention module (CBAM) is integrated into the backbone network, resulting in a slight reduction in computational complexity while improving accuracy.

II. Related work

Object detection algorithms have made significant advancements to date, achieving high accuracy on numerous conventional object detection datasets. However, their performance remains suboptimal for datasets containing small and dense objects due to the inadequate capabilities of current algorithms in detecting such objects accurately. While the initial YOLO series and RCNN series algorithms have shown improvements in detecting general objects, they fall short when it comes to datasets with small objects. To address this issue, researchers have proposed various methods. For

Manuscript received March 26, 2024; revised July 9, 2024. This work was supported by the Key Laboratory of Internet of Things Application Technology on Intelligent Construction, Liaoning Province (2021JH13/10200051)

Junyu Pan is a graduate student of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China. (e-mail: 1838719677@qq.com).

Yujun Zhang is a Professor of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China. (e-mail: 1997zyj@163.com).

instance, RetinaNet [4], introduced by Lin et al., effectively tackles the problem of class imbalance in object detection and enhances the detection performance for small objects; however, it also introduces higher computational complexity. Cascade-RCNN [7], presented by Cai et al., further improves model performance by cascading multiple RCNN modules. Centernet [13], proposed by Duan et al., is a center-point-based object detection algorithm that enhances the detection performance for small and dense objects while compromising accuracy for larger ones. FPN [14], suggested by Lin et al., provides multi-scale feature representations enabling adaptability to different scale requirements of detected objects. DETR [11], introduced by Xia et al., applies transformers to object detection tasks and improves accuracy for occluded and densely packed objects; nevertheless, its performance on small objects remains unsatisfactory.The DETR series consistently achieves state-of-the-art results on the widely used COCO dataset for object detection. However, it still faces significant challenges, such as high computational complexity and slow training speeds, which render DETR unsuitable for real-time detection applications. To address these issues, Zhu et al. proposed Deformable DETR [15], which focuses its attention module on a small set of key sampling points around the reference. These modifications effectively reduce computational load while enhancing performance and decreasing training time by a factor of ten. Roh et al. introduced Sparse DETR [16], which selectively updates the tokens expected to be referenced by the decoder; applying auxiliary detection loss to the selected tokens in the encoder improves performance while minimizing computational overhead. Wang et al. proposed Anchor DETR [17], introducing a novel query mechanism for Transformer-based object detection and designing an attention variant that reduces memory costs while achieving comparable or superior performance compared to standard attention in DETR, thereby improving runtime speed as well. Li et al., presented DN-DETR [18], introducing a new denoising training method to expedite DETR training process efficiently.Zhang et al., proposed DINO [19], an advanced end-to-end object detector that utilizes contrastive denoising training method along with hybrid query selection method incorporating anchor initialization and ”look forward twice” scheme for box prediction.These innovations significantly enhance both performance and efficiency over previous models similar to DETR, effectively reducing model size and pretraining data requirements while achieving superior results.

The YOLO series models have achieved remarkable success in the field of computer vision through continuous development, and their enhanced algorithms have been extensively applied across various domains with commendable outcomes [20, 21]. In comparison to its predecessors, YOLOv8 exhibits superior accuracy while maintaining a lightweight design. Additionally, YOLOv8 offers models of varying sizes (n, s, m, l, and x) to cater to diverse tasks; its network architecture comprises a backbone, neck, and head.

A. Backbone

The backbone network plays a pivotal role in object detection tasks, as it is responsible for extracting meaningful features from input images to provide informative data for subsequent operations. Commonly employed backbone networks encompass ResNet [22], DenseNet [23], Shufflenet [24], SwinTransformer [9], etc., which have exhibited robust feature extraction capabilities when applied across diverse models. YOLOv8 employs an enhanced CSPDarknet53 as its backbone network, derived from the previous iteration, YOLOv5, ensuring both lightweight implementation and improved detection accuracy. It incorporates residual connections and bottleneck structures to reduce network size while enhancing performance. Furthermore, the C3 module in YOLOv5 is substituted with the C2f module in YOLOv8, achieving further lightweight design while retaining the SPPF (Spatial Pyramid Pooling) module utilized in YOLOv5 and other related architectures.

B. Neck

The neck network is positioned between the backbone network and the detection heads, primarily responsible for processing the feature maps extracted by the backbone network at different stages to cater to diverse task requirements. Commonly employed neck networks encompass FPN [14], NasFPN [25], and BiFPN [26]. In YOLOv8, the neck module adopts PAN-FPN [27] methodology, effectively extracting and integrating multi-scale features through operations like feature pyramid networks and feature fusion. This enhancement elevates object detection performance and robustness, enabling superior adaptability of the model towards objects with varying scales and sizes in complex scenes.

C. Detection head

The detection heads [28] in object detection tasks are responsible for processing the feature maps generated by the backbone network and neck network to accurately determine the precise location and class of objects. In YOLOv8, significant modifications have been made to the detection heads compared to its predecessor, transitioning from coupled heads to a decoupled head structure that is currently widely adopted. This structural change effectively separates the classification and detection processes, resulting in improved flexibility, trainability, and generalization performance. Consequently, these enhancements greatly enhance the model's applicability across diverse scenarios.

III. Method Introduction

YOLOv8 has demonstrated its accuracy and reliability on various object detection datasets, showcasing multiple detection layers of different scales to effectively detect objects of diverse sizes. However, there is room for improvement in detecting small objects. Therefore, we adopt YOLOv8 as the baseline model and propose enhancements to enhance its precision in detecting small objects. Our approach is evaluated on the VisDrone2019 dataset, which comprises UAV aerial photography images capturing a significant proportion of small objects along with complex backgrounds and severe occlusions. These factors can potentially result in false positives and false negatives during object detection; hence necessitating further algorithm optimization and model improvements to augment the accuracy of small object detection.

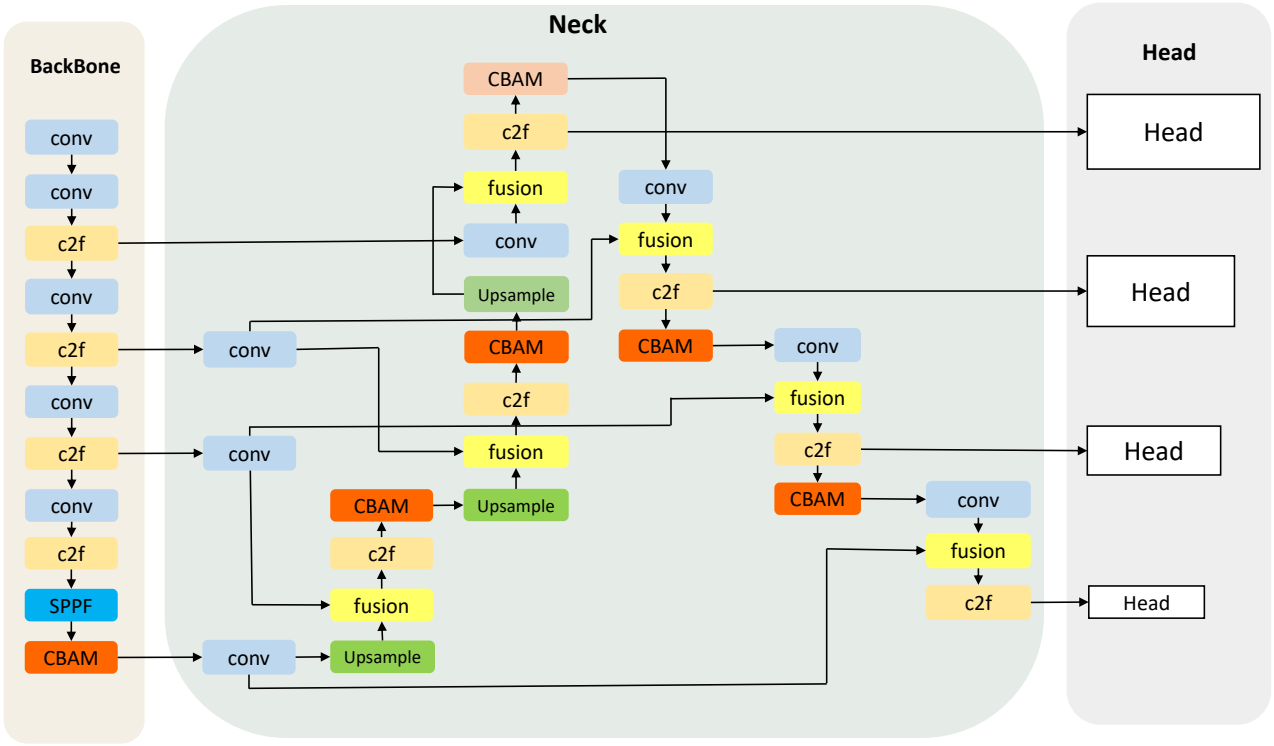

Fig. 1: Overall improved architecture diagram

A. *Overall architecture*

To address these challenges, we have enhanced the original YOLOv8 model as depicted in Figure 1. In the neck network, we have identified that the feature extraction method of the original model is not suitable for effectively detecting small targets due to its unidirectional structure which tends to overlook low-dimensional features, resulting in missed detections of small targets. To tackle this issue and simultaneously preserve more intricate details and edge information while enhancing feature fusion, we propose a novel architecture for the neck network. Taking inspiration from Bidirectional Feature Pyramid Network (BiFPN), we reconstructed the neck network of the baseline model. The figure illustrates that our feature fusion module adopts a bidirectional feature pyramid network approach. With this enhancement, our redesigned neck network demonstrates superior capabilities in extracting features compared to the baseline model. It significantly enhances the model's ability to detect objects at various scales, thereby improving both robustness and performance.

In addition, in order to further enhance the detection ability of the model for features of different scales, the concept of DynamicHead is introduced, and the original detection head is expanded by fusing multiple attention mechanisms. This enhancement improves the performance when dealing with objects of different scales.

The CBAM attention mechanism was ultimately integrated into the model, effectively enhancing its feature representation capabilities during the experiment. These CBAM attention mechanisms selectively emphasize crucial information and significantly improve the model's ability to detect small objects. These enhancements render the model more adaptable to complex environments, resulting in a more satisfactory performance in object detection tasks.

B. *BIFPN*

Given the presence of objects exhibiting varying scales within the dataset, such as nearby cars and distant pedestrians, the original model encounters challenges in effectively detecting these objects when faced with significant scale changes. It tends to prioritize larger objects while disregarding smaller ones, resulting in missed detections. This is a prevalent issue observed in contemporary object detection algorithms. To address this problem, we have redesigned the neck network by incorporating the concept of Weighted Bidirectional Feature Pyramid Network (BIFPN). This approach enhances both feature extraction and fusion capabilities of the original neck network, thereby improving the model's ability to detect objects at different scales. The bidirectional structure enables the model to acquire more low-dimensional features, effectively reducing missed detections of small objects and making it more suitable for our task.

Due to the varying resolutions of input features, their contributions to the fused output features are typically unequal. To address this issue, BIFPN introduces learnable weights that capture the importance of different input features and applies multi-scale feature fusion iteratively in both top-down and bottom-up directions. This approach effectively mitigates the problem of missing detection of small targets caused by scale changes. Traditional FPNs often neglect low-dimensional information due to unidirectional information flow, resulting in a loss of semantic details for small targets. Similar to its predecessor YOLOv5, YOLOv8 utilizes an enhanced PAN-FPN structure, which is a pyramidal feature

extraction network. PAN-FPN combines bottom-up and top-down feature propagation on top of FPN by upsampling low-resolution feature maps while simultaneously downsampling high-resolution feature maps and connecting them to form pathways. In this process, each layer's information is fused with adjacent layers' information to minimize information loss and retain more detailed information, thereby enhancing detection accuracy. Our modified structure also adopts this bidirectional sampling approach as depicted in FIG 2.Firstly, it eliminates single-input nodes that lack feature fusion and make minimal contributions to feature extraction. Subsequently, it establishes a connection from the original input nodes to the output nodes and assigns additional weights to each input, enabling the network to learn the significance of individual input features. This adaptive integration of features across layers mitigates conflicting information between them, resulting in a slight increase in computational overhead but significantly enhancing feature extraction capabilities while minimizing semantic information loss for small objects during feature fusion.

When merging features of varying resolutions, it is customary to standardize them to the same resolution before combining. However, due to their disparate resolutions, the contributions of these input features differ, thereby necessitating unequal treatment. BIFPN addresses this issue by assigning an additional weight to each input, as depicted in Equation 1. This enables the network to learn and leverage the significance of each input through a weighted fusion approach.

$$O = \sum_i \frac{w_i}{\epsilon + \sum_j w_j} \cdot I_i \tag{1}$$

Here, $w_i \geq 0$ is ensured by applying the ReLU activation function after each w_i. The learning rate e is set to a small value of 0.0001 to avoid numerical instability. Each normalized weight value is also between 0 and 1, and since there is no softmax operation involved, the efficiency is much higher.

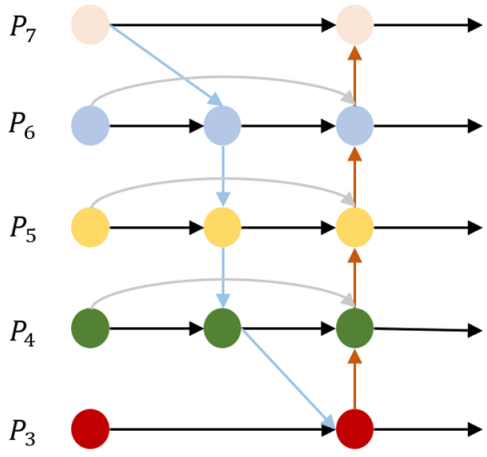

Fig. 2: Improved feature fusion

The YOLOv8 baseline model incorporates three detection layers and heads of varying scales to accurately detect objects of different sizes. Our research specifically focuses on datasets containing a significant number of small objects, which pose challenges for accurate identification. To tackle this issue, we have designed a dedicated detection head exclusively for small objects. This additional small object detection head, combined with the restructured network model, effectively mitigates the impact caused by drastic variations in object size.

C. DynamicHead detection head

In aerial drone imagery, targets often exhibit significant scale variations and complex backgrounds. To address these challenges, we introduced the concept of a dynamic detection head (DynamicHead) to enhance YOLOv8's detection capabilities. We integrated three different attention mechanisms into the detection head: scale-aware attention for feature levels, spatial-aware attention for spatial positions, and task-aware attention for output channels. These attention mechanisms establish global dependencies, effectively expand the receptive field, and gather more contextual information. We believe these mechanisms are crucial for detecting targets in aerial drone imagery, significantly mitigating the impact of target occlusion and complex backgrounds. However, attention mechanisms introduce higher computational costs compared to traditional CNNs, increasing both training and inference overhead. Additionally, using multiple attention mechanisms leads to a substantial increase in computational burden. Therefore, we cannot directly incorporate these attention mechanisms into the detection head. To address this, we introduced the concept of DynamicHead, aiming to improve the model's detection performance for targets with significant scale variations and complex backgrounds while minimizing the increase in computational complexity.

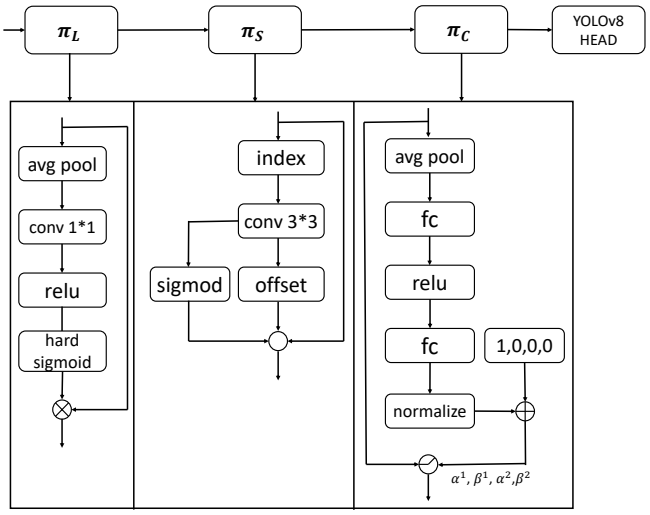

Fig. 3: Improvement of detection head

DynamicHead treats the input as a three-dimensional tensor: level × space × channel, where level represents the feature level, space represents the product of the width and height of the feature map (HW), channel denotes the number of channels, which can be expressed as $\mathcal{F} \in R^{l \times s \times c}$. In response to this attention, if we directly use the fully connected layer, it can be described as $W(\mathcal{F}) = \pi(\mathcal{F}) \cdot \mathcal{F}$. However, connecting three attention mechanisms in this way will lead to a sharp increase in computational complexity,

making it impractical. A feasible solution is to transform the above attention mechanisms into three serial attention mechanisms, each focusing on a single dimension, as shown in Equation 2.

$$W(\mathcal{F}) = \pi_C\left(\pi_S\left(\pi_L(\mathcal{F})\cdot\mathcal{F}\right)\cdot\mathcal{F}\right)\cdot\mathcal{F} \tag{2}$$

The functions $\pi_L(\cdot), \pi_S(\cdot)$and $\pi_C(\cdot)$ respectively represent attention on the three dimensions: L, S, and C.

1) Scale-aware Attention $\pi_L(\cdot)$: The attention module with scale awareness dynamically integrates features based on their semantic importance across different scales, aligning feature maps with target scales by operating on the level dimension. By enhancing attention on the level dimension, it further improves scale-awareness in target detection, effectively enhancing object detection for various scales and overall performance. The mathematical representation of this module is provided in Formula 3.

$$\pi_L(\mathcal{F})\cdot\mathcal{F} = \sigma\left(f\left(\frac{1}{SC}\sum_{S,C}\mathcal{F}\right)\right)\cdot\mathcal{F} \tag{3}$$

In this case, f(·) approximates a linear function using a 1x1 convolutional layer. $\sigma(x) = \max\left(0, \min\left(1, \frac{x+1}{2}\right)\right)$ is the hard-sigmoid function.

2) Spatial-aware Attention $\pi_S(\cdot)$: The module for spatial perception attention enhances the ability to discriminate between different spatial positions. Given the high dimensionality of S, it is imperative to decompose this module into two sequential steps: firstly, employing variable convolution for learning sparsity; and subsequently, aggregating features across multiple levels at the same spatial position in the spatial dimension, as illustrated in Formula 4. Geometric transformations of the target correspond to distinct spatial positions, and augmenting attention in the spatial dimension amplifies the target detector's capacity for perceiving space.

$$\pi_S(\mathcal{F})\cdot\mathcal{F} = \frac{1}{L}\sum_{l=1}^{L}\sum_{k=1}^{K} w_{l,k}\cdot\mathcal{F}(l; p_k + \Delta p_k; c)\cdot\Delta m_k \tag{4}$$

Among them, K represents the number of sparse sampling positions. $p_k + \Delta p_k$ performs a position offset to focus on discriminative regions. Δm_k is a self-learned factor that measures the importance of position p_k. All of these can be learned from the input features of the intermediate levels in F.

3) Task-aware Attention $\pi_C(\cdot)$: To enhance generalization in joint learning and target representation, we propose a task-aware attention module that dynamically adjusts feature channels in the channel dimension to cater to different tasks. Each channel corresponds to a specific task, thereby improving the perception capability of target detection across various tasks (as depicted in formula 5). By increasing attention within the channel dimension, we can effectively strengthen the perception capability of target detection for diverse tasks.

$$\begin{aligned} &\pi_C(\mathcal{F})\cdot\mathcal{F} = max(\nu) \\ &\nu = \alpha^1(\mathcal{F})\cdot\mathcal{F}_c + \beta^1(\mathcal{F}), \alpha^2(\mathcal{F})\cdot\mathcal{F}_c + \beta^2(\mathcal{F}) \end{aligned} \tag{5}$$

We use $\left[\alpha^1, \alpha^2, \beta^1, \beta^2\right]^T = \theta(\cdot)$ as super functions to learn to control the activation threshold. In addition, we introduce the $\theta(\cdot)$ function, which works similar to Dynamic ReLU. Firstly, we perform global pooling on the L × S dimensions, then pass through two fully connected layers, one normalization layer, and finally normalize the output using the shifted sigmoid function. Through these optimizations and enhancements, our model can better adapt to different data and task requirements.

D. CBAM

In the context of drone-based aerial target detection, the presence of intricate and dynamic backgrounds poses a significant challenge to the model's capacity for accurate target detection. To effectively tackle this challenge, we have incorporated an attention mechanism known as CBAM (Convolutional Block Attention Module) into our model. The primary objective behind integrating this module is to aid the model in mitigating the impact of cluttered background information and enhancing its focus on efficiently extracting target objects.

The CBAM module is designed to enhance attention in a lightweight manner and consists of two sub-modules: the Channel Attention Module (CAM) and the Spatial Attention Module (SAM). Figure 4 illustrates the structural diagram of CBAM, showcasing its composition. The CAM module calculates the channel attention map by considering information along the channel dimension, while the SAM module computes the spatial attention map by taking into account information along the spatial dimension. These two attention maps are then multiplied with the input feature map to facilitate adaptive refinement of features.

In our experiments, we successfully integrated the CBAM module into both the backbone and neck networks of the model, resulting in enhanced feature representation capability and improved detection performance. Despite a slight increase in computational overhead, this optimization measure effectively enhances target detection tasks in complex backgrounds, thereby increasing robustness and reliability. Furthermore, this enhancement not only improves target identification accuracy but also expands its applicability to a wider range of real-world scenarios by enhancing generalization capability.

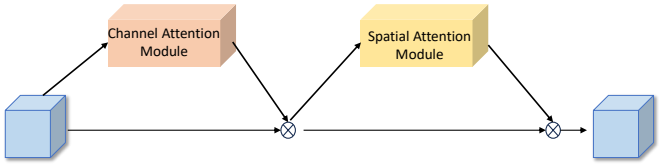

Fig. 4: CBAM network architecture

IV. Experimental Design and Implementation

A. Dataset Introduction

In our experiments, we utilized the VisDrone2019 DET dataset, meticulously curated by the AISKYEYE team at Tianjin University's Machine Learning and Data Mining Laboratory in China. This comprehensive dataset comprises 8629 images encompassing 10 distinct categories and a total of 343205 annotations. It showcases diverse scenarios captured under varying weather conditions and lighting

settings using an array of drones. Spanning thousands of kilometers across 14 urban locations, it encompasses both urban and rural environments. The object categories within this dataset include pedestrians, vehicles, bicycles, among others, with scene densities ranging from sparse to crowded environments. Moreover, the dataset presents numerous challenging instances such as complex dense scenes and partially occluded objects.

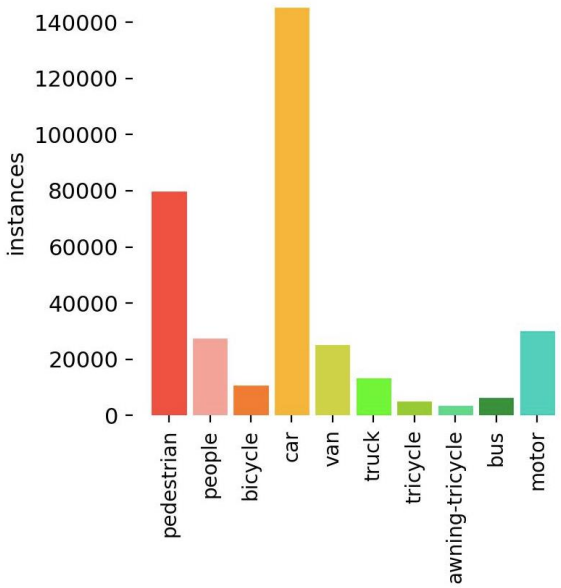

Fig. 5: The number of labels for each category in VisDrone2019 dataset.

B. Experimental setup

In terms of hardware, we utilized an Xeon(R) Platinum 8350C CPU, 42GB of memory, and a GeForce RTX 3090 GPU. For software, we employed PyTorch 1.11.0 and Python 3.8, running on a Linux operating system.

The VisDrone2019 dataset contains images with varying resolutions. To achieve better results, we adjusted the input image size to 1500x1500. We set the batch size to 2 and used the default learning rate for YOLOv8. YOLOv8 offers five different models of increasing sizes, denoted as n, s, m, l, and x. Larger models generally provide higher detection accuracy but also increase training time and computational complexity. For convenience in experiments, we did not use the largest YOLOv8x model but instead utilized the YOLOv8s model. Its performance was evaluated using the validation set from the VisDrone2019 dataset.

C. Comparative experiments

The VisDrone2019 dataset comprises images with varying resolutions. In order to enhance the performance, we adjusted the input image size to 1500x1500 pixels. The batch size was set to 2, while the default learning rate for YOLOv8 was employed. YOLOv8 offers a range of models denoted as n, s, m, l, and x, with increasing sizes. Although larger models generally yield higher detection accuracy, they also lead to longer training time and increased computational complexity. For experimental convenience purposes in this study, we opted not to utilize the largest YOLOv8x model but instead employed the YOLOv8s model. Its performance evaluation was conducted using the validation set from the VisDrone2019 dataset.

TABLE I: Compare different categories pairwise

Method	ImageSize	mAP	AP50
YOLOv8s	1280*1280	35.5	55.9
Cascade-RCNN	1280*1280	16.1	32.0
DPNet	1280*1280	29.6	54.0
RRNet	1280*1280	29.1	55.8
TPH-YOLOv5	1280*1280	35.7	57.3
SMPNet	1280*1280	36.0	59.5
ours	1280*1280	38.0	59.6

We conducted detection on 10 different categories in the VisDrone2019 dataset, encompassing pedestrian, people, bicycle, car, van, trunk, tricycle, awning-tricycle, bus and motor. Notably, there are overlapping categories such as pedestrian and people or tricycle and awning-tricycle. These targets are typically captured from a drone's perspective and often appear very small; thus models can only extract limited features. Additionally, the presence of numerous highly similar categories poses challenges for accurate detection. Consequently, we compared the performance of our enhanced model with other mainstream models (as presented in Table 2). Our experimental results demonstrate that our improved model achieves commendable performance across most categories while validating its accuracy.

D. Ablation experiments

To delve deeper into the impact of each component on the experimental results, we conducted a series of experiments using the YOLOv8s backbone network in the same experimental environment. By comparing different evaluation metrics, including mean average precision (mAP), 50% Intersection over Union (IoU) precision (AP50), and small object precision (APs), we obtained a series of insightful results. These findings contribute to a more comprehensive understanding of the roles of each component in object detection tasks. Table 3 presents the results of our ablation experiments. Through systematic analysis and comparison, we can better discern the contribution of each component to model performance, thereby providing a solid reference basis for further research and optimization.

TABLE III: Ablation experiments

	BIFPN	DyHead	CBAM	mAP	AP50	APs
YOLOv8s	-	-	-	35.5	55.9	24.7
YOLOv8s	√	-	-	36.5	58	27.1
YOLOv8s	-	√	-	36.2	57.5	26.8
YOLOv8s	-	-	√	35.8	56.3	25.5
YOLOv8s	√	√	√	38.0	59.6	28.8

E. Experimental results visualization

Our BDC-YOLOv8 underwent a series of experiments on the VisDrone2019 dataset, effectively showcasing its ex-

TABLE II: Compare different categories pairwise

Methods	all	pedestrian	people	bicycle	car	van	trunk	tricycle	awning-tricycle	bus	motor
YOLOv8s	35.5	34.1	23.8	18.4	66.3	42.3	37.2	27.9	17.1	54.6	33.6
Cascade-RCNN	16.1	16.3	6.2	4.2	37.2	20.4	17.1	14.5	12.4	24.3	14.9
PPNet	29.1	30.4	14.9	13.7	51.4	36.1	35.2	28.0	19.0	44.2	25.9
DPNet	29.6	32.3	16.0	12.9	51.5	39.8	30.7	**30.7**	18.4	38.5	28.0
TPH-YOLOv5	35.7	28.0	14.9	14.2	67.6	45.0	**44.8**	25.1	**20.5**	55.7	27.8
ours	**38.0**	**38.8**	**27.0**	**21.5**	**68.9**	**46.3**	36.8	29.0	17.6	**56.8**	**37.2**

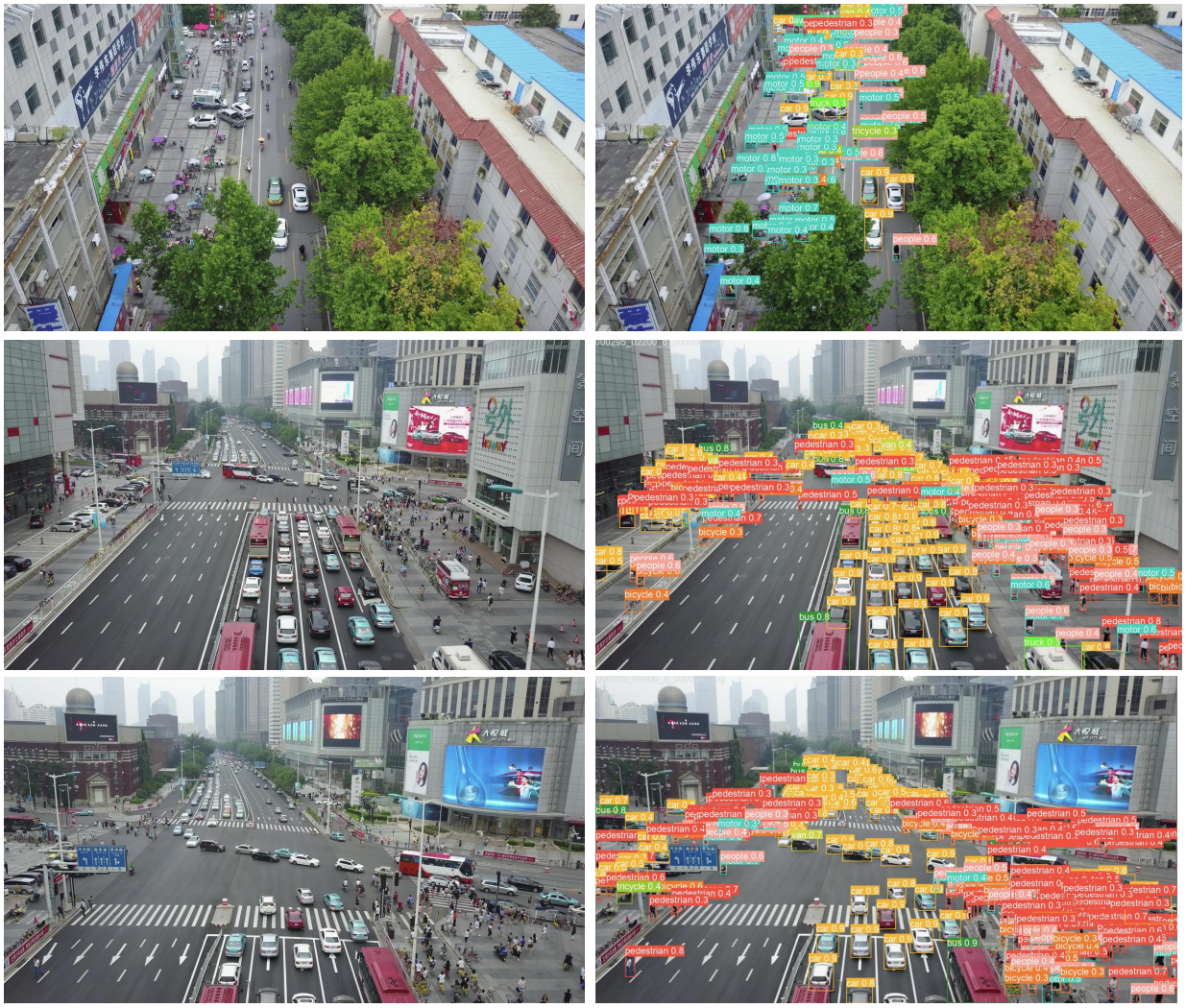

Fig. 6: Results Visualization

ceptional performance in object detection, particularly when dealing with small and occluded objects. The visualized results depicted in Figure 6 vividly illustrate the detection outcomes achieved by our model. It is evident that our approach attains higher levels of detection accuracy when confronted with diminutive and densely packed objects. In comparison to the original model, our methodology exhibits superior detection performance and enhanced stability in addressing these arduous tasks. These findings not only validate the efficacy of our model but also provide significant guidance and insights for future research endeavors aimed at optimization.

In our dataset, there are several categories that share similarities, such as pedestrians and crowds, bicycles and motorcycles, tricycles and open tricycles, among others. When viewed from a drone's aerial perspective, these small objects have limited distinguishing features. As a result, they can be easily missed or falsely detected. Figure 7 demonstrates the effectiveness of our model in reducing false alarms when dealing with different categories. These improvements not only enhance the accuracy of the model but also strengthen its reliability in complex scenarios.

V. Conclusion

In this paper, we propose an enhanced model based on YOLOv8, a widely adopted framework in the field, aiming to improve detection accuracy and capability for small targets in aerial drone imagery. Through empirical validation, our

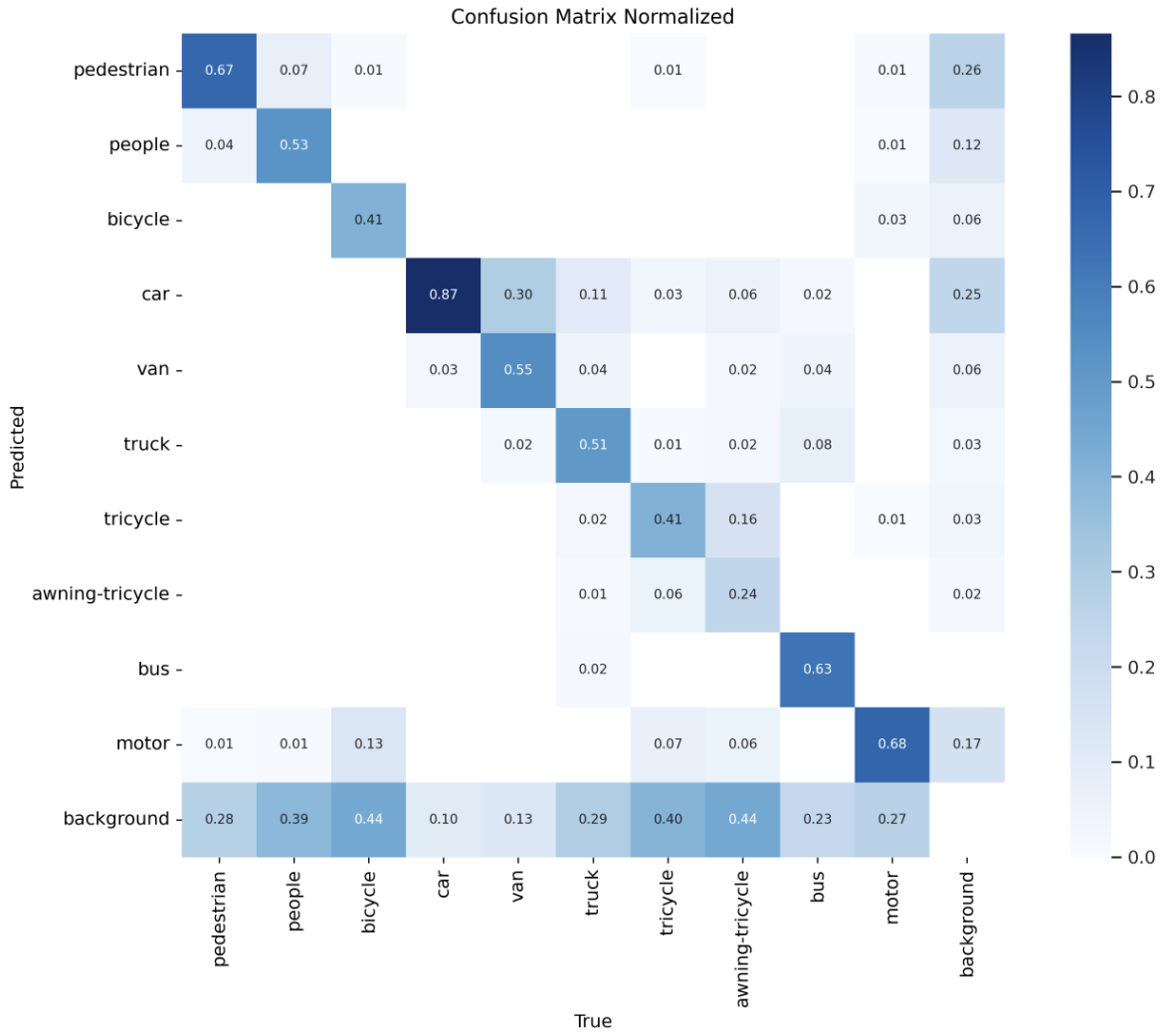

Fig. 7: Confusion matrix

model demonstrates its effectiveness and robustness, particularly in enhancing the detection performance for small targets. We conducted experiments using the VisDrone2019 dataset as our experimental subject.

Firstly, we introduce the concept of Bi-directional Feature Pyramid Network (BiFPN) to restructure the neck network of the model, enabling it to obtain richer contextual information. Secondly, we incorporate new layers and detection heads specifically designed for small target detection, resulting in improved detection accuracy when dealing with small targets. Thirdly, we adopt the idea of DynamicHead detection head with multiple attention mechanisms to enhance the original detection head. The new detection head effectively unifies scale perception, spatial perception, and task perception, thereby enhancing detection accuracy. Finally, we integrate CBAM (Convolutional Block Attention Module) attention mechanisms into both the backbone and neck networks to improve the model's perception capability.

Compared to the baseline model, our improved model achieves a 2.5% increase in mean Average Precision (mAP), a 3.7% increase in AP50, and a 4.1% increase in APs. However, challenges such as small target misses and category misclassification still persist. Therefore, our future research will focus on further optimizing the model while concurrently enhancing detection accuracy for diminutive targets, particularly within lightweight model configurations.

References

[1] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7464–7475, 2023.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, 2016.

[3] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pp. 21–37, Springer, 2016.

[4] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2980–2988, 2017.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, 2014.

[6] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448, 2015.

[7] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6154–6162, 2018.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[9] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using

shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.
[10] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European Conference on Computer Vision*, pp. 213–229, Springer, 2020.
[11] Z. Xia, X. Pan, S. Song, L. E. Li, and G. Huang, "Vision transformer with deformable attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4794–4803, 2022.
[12] L. Zhu, X. Wang, Z. Ke, W. Zhang, and R. W. Lau, "Biformer: Vision transformer with bi-level routing attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10323–10333, 2023.
[13] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "Centernet: Keypoint triplets for object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6569–6578, 2019.
[14] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125, 2017.
[15] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020.
[16] B. Roh, J. Shin, W. Shin, and S. Kim, "Sparse detr: Efficient end-to-end object detection with learnable sparsity," *arXiv preprint arXiv:2111.14330*, 2021.
[17] Y. Wang, X. Zhang, T. Yang, and J. Sun, "Anchor detr: Query design for transformer-based detector," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 2567–2575, 2022.
[18] F. Li, H. Zhang, S. Liu, J. Guo, L. M. Ni, and L. Zhang, "Dn-detr: Accelerate detr training by introducing query denoising," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13619–13627, 2022.
[19] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, "Dino: Detr with improved denoising anchor boxes for end-to-end object detection," *arXiv preprint arXiv:2203.03605*, 2022.
[20] W. Zhang, Y. Zhao, Y. Guan, T. Zhang, Q. Liu, and W. Jia, "Green apple detection method based on optimized yolov5 under orchard environment [j]," *Engineering Letters*, vol. 31, no. 3, 2023.
[21] X. Zhang and Y. Tia, "Improved yolov5s traffic sign detection.," *Engineering Letters*, vol. 31, no. 4, 2023.
[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.
[23] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, 2017.
[24] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6848–6856, 2018.
[25] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "Nas-fpn: Learning scalable feature pyramid architecture for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7036–7045, 2019.
[26] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10781–10790, 2020.
[27] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8759–8768, 2018.
[28] X. Dai, Y. Chen, B. Xiao, D. Chen, M. Liu, L. Yuan, and L. Zhang, "Dynamic head: Unifying object detection heads with attentions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7373–7382, 2021.