sensors

MDPI

Article

SOD-YOLOv8—Enhancing YOLOv8 for Small Object Detection in Aerial Imagery and Traffic Scenes

Boshra Khalili and Andrew W. Smyth *

Department of Civil Engineering and Engineering Mechanics, Columbia University, New York, NY 10027, USA; bk2898@columbia.edu

* Correspondence: aws16@columbia.edu

Abstract: Object detection, as a crucial aspect of computer vision, plays a vital role in traffic management, emergency response, autonomous vehicles, and smart cities. Despite the significant advancements in object detection, detecting small objects in images captured by high-altitude cameras remains challenging, due to factors such as object size, distance from the camera, varied shapes, and cluttered backgrounds. To address these challenges, we propose small object detection YOLOv8 (SOD-YOLOv8), a novel model specifically designed for scenarios involving numerous small objects. Inspired by efficient generalized feature pyramid networks (GFPNs), we enhance multi-path fusion within YOLOv8 to integrate features across different levels, preserving details from shallower layers and improving small object detection accuracy. Additionally, we introduce a fourth detection layer to effectively utilize high-resolution spatial information. The efficient multi-scale attention module (EMA) in the C2f-EMA module further enhances feature extraction by redistributing weights and prioritizing relevant features. We introduce powerful-IoU (PIoU) as a replacement for CIoU, focusing on moderate quality anchor boxes and adding a penalty based on differences between predicted and ground truth bounding box corners. This approach simplifies calculations, speeds up convergence, and enhances detection accuracy. SOD-YOLOv8 significantly improves small object detection, surpassing widely used models across various metrics, without substantially increasing the computational cost or latency compared to YOLOv8s. Specifically, it increased recall from 40.1% to 43.9%, precision from 51.2% to 53.9%, $mAP_{0.5}$ from 40.6% to 45.1%, and $mAP_{0.5:0.95}$ from 24% to 26.6%. Furthermore, experiments conducted in dynamic real-world traffic scenes illustrated SOD-YOLOv8's significant enhancements across diverse environmental conditions, highlighting its reliability and effective object detection capabilities in challenging scenarios.

Keywords: small object detection; YOLOv8; feature pyramid network; bounding box regression; attention mechanism

Citation: Khalili, B.; Smyth, A.W. SOD-YOLOv8—Enhancing YOLOv8 for Small Object Detection in Aerial Imagery and Traffic Scenes. *Sensors* **2024**, *24*, 6209. https://doi.org/10.3390/s24196209

Academic Editor: Steve Vanlanduit

Received: 13 August 2024
Revised: 10 September 2024
Accepted: 24 September 2024
Published: 25 September 2024

1. Introduction

Object detection in computer vision plays a crucial role across various fields, including autonomous vehicles [1–3], traffic scene monitoring [4,5], enhancing intelligent driving systems [6], and facilitating search and rescue missions [7]. Accurate detection of small objects such as pedestrians, vehicles, motorcycles, bicycles, traffic signs, and lights is crucial for safe navigation and decision-making in autonomous vehicles and intelligent driving systems [1,3]. Furthermore, detecting small objects enhances traffic flow management, pedestrian safety, and overall traffic scene analysis. This capability is essential for improving urban planning and transportation systems [4,5].

As the cost of UAV production decreases and flight control techniques advance, these small, flexible devices are increasingly used for intelligent traffic monitoring [8]. UAVs typically operate at higher altitudes to capture broader views, which reduces the apparent size of ground objects, due to greater distances. This distance complicates object detection within captured images [8]. Despite significant progress in object detection, detecting small

objects such as pedestrians, motorcycles, bicycles, and vehicles in urban traffic remains challenging, due to their size, varied shapes, and cluttered backgrounds. This challenge is further amplified when working with limited hardware resources in computer vision and object detection.

Small objects, which occupy a small portion of an image and have a lower resolution and less distinct visual characteristics compared to larger objects, are more challenging to detect accurately. Moreover, shallow layers in networks such as YOLOv8 may filter out essential spatial details required for detecting these small objects, resulting in data loss. Additionally, smaller objects can be overshadowed by larger ones during feature extraction, potentially causing the loss of relevant details crucial for accurate detection. Overcoming these challenges is crucial for improving overall detection accuracy and reliability in real-world scenarios.

Despite advancements in recent models, existing small object detection methods in UAV aerial photography and traffic scenarios still encounter substantial challenges. Many of these methods primarily focus on feature fusion, but often neglect crucial inner block connections, resulting in less effective feature integration. Additionally, conventional attention mechanisms fail to adequately account for interactions among spatial details and are limited by the narrow receptive field of 1×1 kernel convolutions embedded in their architecture. This restriction impacts local cross-channel interaction and comprehensive contextual information modeling. Furthermore, existing IoU-based bounding box regression methods frequently encounter enlargement issues, which can negatively affect both accuracy and convergence speed during training. While these methods often perform well on controlled datasets, they struggle to generalize across diverse environments and dynamic lighting conditions in real-world settings.

In this paper, to tackle the persistent challenges of detecting small objects in UAV aerial photography and urban traffic scenes, we propose a novel model based on YOLOv8. Our approach integrates an optimized GFPN, inspired by efficient-RepGFPN [9], into YOLOv8. This enhancement incorporates skip connections and queen fusion structures to improve efficacy, without significantly increasing computational complexity or latency. Additionally, the introduced C2f-EMA module enhances feature extraction by redistributing feature weights using the EMA attention mechanism [10]. Unlike other attention mechanisms, it overcomes the mentioned limitations by incorporating 3×3 convolutions and a parallel structure. We also include a fourth detection layer to effectively utilize high-resolution spatial details. Furthermore, given the significant impact of bounding box regression in object detection, we employ the PIoU method, which enhances performance and reduces convergence time by incorporating an improved penalty term and attention mechanism. In addition, we conducted experiments using real-world traffic scenes captured by cameras mounted on buildings, evaluating a variety of environments, lighting conditions, and dynamic scenarios, including nighttime and crowded settings. This approach tested the generalization abilities of the small object detection methods beyond controlled datasets. Our key contributions are as follows:

- We enhance the YOLOv8 architecture with multi-path fusion inspired by the efficient RepGFPN in DAMO-YOLO models, improving small object detection by better fusing features across different levels and adding a fourth detection layer for high-resolution spatial details.
- We integrate the C2f-EMA structure into the network, using the efficient multi-scale attention module to improve feature extraction by prioritizing relevant features and enhancing the network's ability to detect objects of various sizes.
- We replace the original CIoU with PIoU, which improves bounding box regression by focusing on moderate-quality anchor boxes, simplifying calculations, and enhancing the detection accuracy with faster convergence.
- We validated our model through visual analyses in challenging scenarios and experiments using real-world traffic images, demonstrating the effectiveness of our approach in enhancing small object detection.

The structure of the remaining sections of this paper is as follows: Section 2 discusses related work. Section 3 provides an overview of the YOLOv8 network architecture. Section 4 details the proposed enhanced YOLOv8. Section 5 covers our experimental setup and result analysis. Finally, Section 6 concludes the paper.

2. Related Work

Small object detection has been a significant challenge in the field of computer vision, particularly in UAV aerial photography and traffic scenarios. This section reviews mainstream object detection algorithms, recent advancements in small object detection, and specific enhancements made to the YOLO framework.

Mainstream object detection algorithms predominantly use deep learning techniques, categorized into two types: two-stage and one-stage methods. Two-stage methods process candidate frames with a classifier and perform deep learning on corresponding frames [11]. Typical two-stage detection algorithms include R-CNN [11], fast R-CNN [12], and faster R-CNN [13]. The R-CNN family is a classic two-stage algorithm known for high detection accuracy but that faces challenges such as slow speed, training complexity, and optimization [14]. One-stage detectors, like the YOLO series [15,16] and SSD [17], use a single neural network to predict box coordinates and categories in one pass. Consequently, single-stage networks excel in applications where speed is crucial. However, they sacrifice some accuracy. Despite advancements in speed, these methods struggle with accuracy due to the multi-scale nature of objects and the prevalence of small objects in UAV and traffic scenes.

Recent research has focused on improving small object detection in UAV aerial photography and traffic scenarios, which is challenging due to their lower resolution and less distinct visual characteristics compared to larger objects. Studies have explored diverse backbone architectures to enhance feature representation, reduce false positives, and extract relevant features from complex backgrounds in UAV imagery.

Liu et al. [18] introduced a model for small target detection in UAV images, addressing leakage and false positives by integrating ResNet units and optimizing convolutional operations to expand the network's receptive field. Liu et al. [19] proposed CBSSD, a specialized detector for small object detection in UAV traffic images. By integrating ResNet50's lower-level features with VGG16, CBSSD improves feature representation, enhances object recognition accuracy, and reduces false positives under challenging lighting conditions. In addition, some recent works have used vision transformers [20,21]. Additionally, Liu et al. [22] utilized multi-branch parallel feature pyramid networks (MPFPN) and SSAM for detecting small objects in UAV images. Their approach enhances feature extraction through MPFPN for deep layer detail recovery, while SSAM reduces background noise, significantly boosting accuracy. Experimental results on the VisDrone-DET dataset [23] showcased their method's competitive performance.

Adaptations and optimizations within the YOLO framework have also been explored to address challenges in small object detection. Lai et al. [24] introduced STC-YOLO, a specialized variant of YOLOv5 designed for challenging traffic sign detection. Their improvements included refined downsampling, a dedicated small object detection layer, and a CNN-based feature extraction module with multi-head attention. STC-YOLO demonstrated a significant 9.3% improvement in mean average precision (mAP) compared to YOLOv5 on benchmark datasets.

Further enhancements have been made to YOLOv8, focusing on improving backbone architectures, integrating attention mechanisms to focus on relevant features and suppress irrelevant ones, and modifying loss functions. Shen et al. [25] introduced DS-YOLOv8 to enhance small object detection within images by integrating deformable convolution C2f (DCN_C2f) and self-calibrating shuffle attention (SC_SA) for adaptive feature adjustment, alongside wise-IoU [26] and position regression loss to boost performance. Experimental results across diverse datasets showed significant enhancements in $mAP_{0.5}$. Wang et al. [8] improved YOLOv8 for UAV aerial photography with a BiFormer attention mechanism to focus on important information and FFNB for effective multiscale feature fusion. This

resulted in a significant 7.7% increase in mean detection accuracy over baseline models, surpassing widely-used alternatives in detecting small objects. This marked substantial progress in UAV object detection, though it necessitates further optimization due to the increased computational complexity from additional detection layers.

Wang et al. [27] improved YOLOv8 for detecting targets in remote sensing images, focusing on complex backgrounds and diverse small targets. They introduced a small target detection layer and incorporated a C2f-E structure using the EMA attention module. Experimental results on the DOTAv1.0 dataset [28] demonstrated a notable 1.3% increase in $mAP_{0.5}$ to 82.7%, highlighting significant advancements in target detection accuracy. However, their approach introduces increased computational complexity. Xu et al. [29] introduced YOLOv8-MPEB, specialized for small target detection in UAV images, addressing scale variations and complex scenes. Enhancements included replacing CSPDarknet53 [30] with MobileNetV3 for efficiency, integrating efficient multi-scale attention in C2f for better feature extraction, and incorporating a bidirectional feature pyramid network (BiFPN) [31] in the neck segment for enhanced adaptability. Experimental results on a custom dataset demonstrated that YOLOv8-MPEB achieved a 91.9% mAP, a 2.2% improvement over the standard YOLOv8, while reducing parameters by 34% and model size by 32%. However, accurately detecting dense small targets remains a challenge.

Despite the advancements in the reviewed studies, small object detection methods still face challenges in UAV aerial photography and traffic scenarios. These methods primarily focus on feature fusion but often neglect inner block connections. In contrast, our approach integrates an optimized GFPN, inspired by the efficient-RepGFPN, into YOLOv8. This enhancement incorporates skip connections and queen fusion structures to improve efficacy, without significantly increasing computational complexity or latency. Additionally, the introduced C2f-EMA module enhances feature extraction by redistributing feature weights using the EMA attention mechanism. Unlike other attention mechanisms, it overcomes limitations such as neglecting interactions among spatial details and the limited receptive field of 1×1 kernel convolution, which limits local cross-channel interaction and contextual information modeling.

Furthermore, our method avoids the enlargement issues common in other bounding box regression methods. The used PIoU loss function effectively guides anchor boxes during training, resulting in faster convergence and demonstrating its effectiveness. While existing methods perform well in controlled datasets, they often struggle to generalize across diverse environments and dynamic lighting conditions in real-world settings. In this paper, we experimented with real-world traffic scenes captured by building-mounted cameras, assessing diverse environments, lighting conditions, and dynamic scenarios such as nighttime and crowded scenes. This challenged the generalization capabilities of the small object detection method beyond controlled datasets.

3. Introduction of YOLOv8 Detection Network

As shown in Figure 1, the YOLOv8 architecture consists of three main elements: the backbone, neck, and detection layers. Each of these components will be introduced in the subsequent sections.

3.1. Backbone Layer

The architecture of YOLOv8 is based on the CSPDarknet53 [30] backbone, employing five downsampling stages to extract distinct scale features. It improves the information flow and stays lightweight by using the C2f module instead of the cross-stage partial (CSP) module [32]. The C2f module includes dense and residual structures for better gradient flow and feature representation. The backbone also includes a spatial pyramid pooling fast (SPPF) module, which captures features at multiple scales to boost detection performance. The SPPF layer reduces computational complexity and latency, while optimizing feature extraction [33].

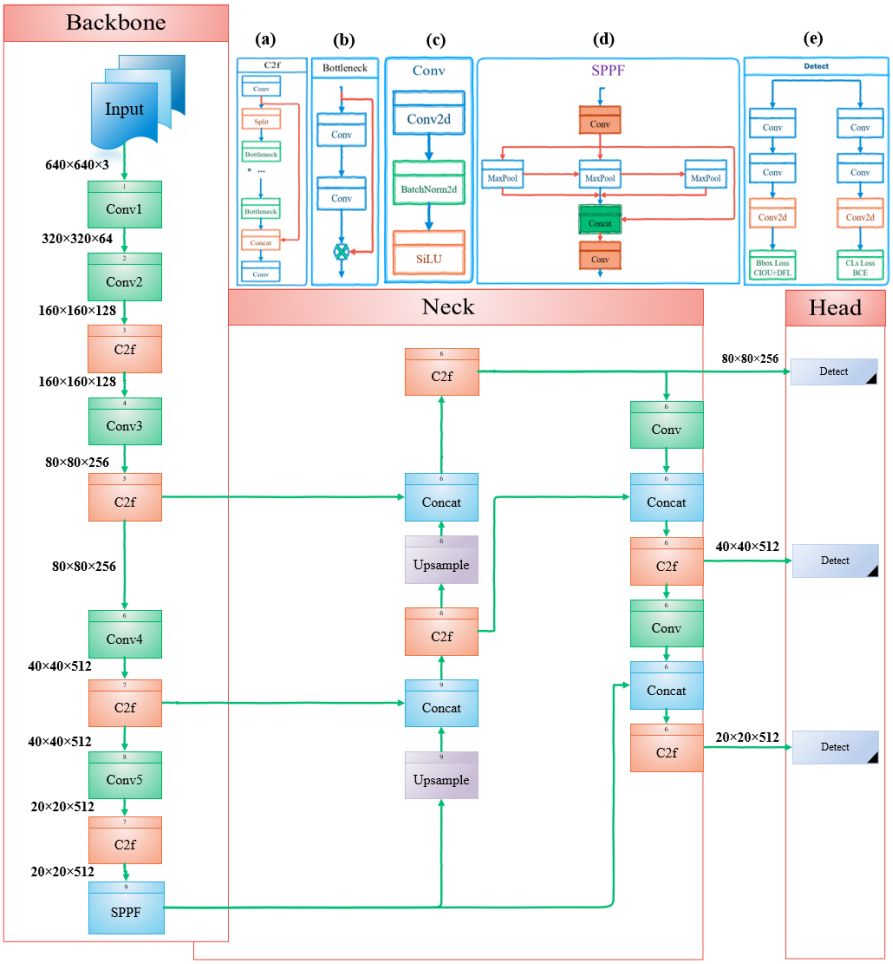

Figure 1. The network structure of YOLOv8, including the following modules: (**a**) C2F; (**b**) Bottleneck; (**c**) Convolution (conv); (**d**) Spatial Pyramid Pooling Fast (SPPF); and (**e**) Detection Layer.

3.2. Neck Layer

For multi-scale feature fusion, YOLOv8's neck uses a feature pyramid network (FPN) [34] and path aggregation network (PANet) [35]. The FPN enhances hierarchical feature fusion, improving object detection across various scales through a top-down pathway, while the PANet enhances feature representation and information reuse with a bottom-up pathway, though it increases computational cost. Combining FPN–PANet structures and C2f modules integrates feature maps of various scales, merging both shallow and deep information.

3.3. Detection Head Layer

YOLOv8, a state-of-the-art object detection model, enhances accuracy and robustness by using a task-aligned assigner [36] instead of traditional anchors. This assigner dynamically categorizes samples as positives or negatives, refining the model's ability to detect objects accurately. The detection head features a decoupled structure with separate branches for object classification and bounding box regression. For classification, it employs binary cross-entropy loss (BCE Loss). For regression, it uses a combination of distribution focal loss (DFL) [37] and complete intersection over union (CIoU) [38] loss. These efficient loss functions are crucial for precise object localization, further boosting the model's performance.

Bounding box loss functions aim to accurately localize objects by penalizing differences between predicted and ground truth bounding boxes. IoU-based loss functions [39] are crucial for bounding box regression in the detection layer. IoU loss measures the overlap between predicted and ground truth boxes by comparing the ratio of their intersection area

to their union area. However, its gradient diminishes when there is no overlap, making it less effective in such cases. Various IoU-based loss functions have been developed, with different methodologies and constraints. CIoU, used in YOLOv8, minimizes the normalized distance between the center points of predicted and ground truth boxes and includes an aspect ratio penalty term. This approach improves convergence speed and overall performance.

4. Method

As shown in Figure 1, the YOLOv8 architecture features a CSPDarknet53 backbone with C2f modules, an FPN–PANet neck, and CIoU-based loss functions for detection. We introduce three pivotal enhancements to improve small object detection in our study. First, we enhance the feature fusion within the YOLOv8 architecture's neck to better retain crucial spatial details typically filtered out by shallower layers. This modification aims to mitigate information loss, especially for smaller objects overshadowed by larger ones during feature extraction. Second, we propose the C2f-EMA module, integrating an EMA attention mechanism to prioritize relevant features and spatial details across different channels. This method enhances feature extraction efficiency by redistributing feature weights effectively. Finally, we use PIoU as an improved bounding box regression metric, replacing CIoU. PIoU incorporates a penalty term that minimizes the Euclidean distance between the corresponding corners of predicted and ground truth boxes, offering a more intuitive measure of similarity and stability in box regression tasks. These methods contribute to enhancing the accuracy and robustness of our small object detection framework. The enhanced structure depicted in Figure 2 is utilized in this paper.

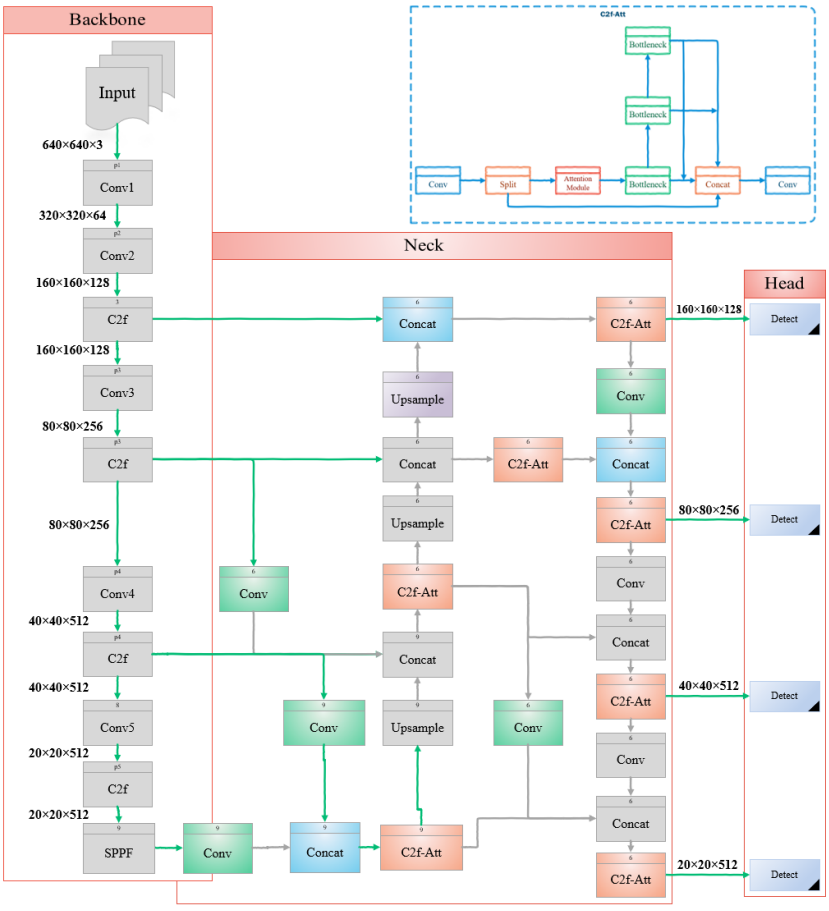

Figure 2. Proposed improved YOLOv8 for small object detection, with original YOLOv8 in gray and highlighted improved modules.

4.1. Improved GFPN for Multilevel Feature Integration

In YOLOv8, crucial spatial details are primarily encoded in the network's shallower layers. However, these layers often filter out less prominent details, leading to significant data loss for small object detection. Additionally, smaller objects may be overshadowed by larger ones during feature extraction, resulting in a gradual loss of information and the potential disappearance of relevant details. To address these challenges, this study introduces an enhanced feature fusion method in the neck of the YOLOv8 architecture. This method focuses on preserving and effectively utilizing important information from the shallower layers, thereby enhancing the overall detection accuracy, especially for small objects.

The FPN merges features of different resolutions extracted from a backbone network. It begins with the highest-resolution feature map and progressively combines features from higher to lower resolutions using a top-down approach. The PAFPN improves on the FPN by adding a bottom-up approach that enhances the bidirectional information flow. It merges features from lower to higher network layers, prioritizing spatial detail preservation, even with increased computational demands.

The BiFPN [31] enhances object detection by integrating features across different resolutions bidirectionally, using both bottom-up and top-down pathways. This method optimizes multi-scale feature utilization, simplifies the network by reducing computational complexity, and includes skip connections at each level. These connections allow for adaptable use of input features, enhancing feature fusion across scales and details [31]. However, deep stacking of BiFPN blocks may cause gradient vanishing during training, potentially affecting overall network performance [40].

Prior methods mainly focused on combining features, without considering inner block connections. In contrast, the GFPN introduces skip connections and queen fusion structures. It employs skip-layer and cross-scale connections to enhance feature combination. The GFPN implements skip connections in two forms: $\log_2(n)$-link and dense-link.

The $\log_2(n)$-link method optimizes information transmission by allowing the l^{th} layer at level k to receive feature maps from up to $\log_2(l) + 1$ preceding layers. These skip connections help mitigate gradient vanishing during back-propagation by extending the shortest gradient distance from one layer to approximately $1 + \log_2(n)$ layers [9]. This extension facilitates more effective gradient propagation over longer distances, potentially enhancing the scalability of deeper networks.

In contrast, the dense-link method ensures that each scale feature P_k^l at level k receives feature maps from all preceding layers up to the l^{th} layer. This promotes a robust information flow and the integration of features across multiple scales. In the GFPN, this structure facilitates seamless connectivity between layers, enhancing feature reuse and improving the network's efficiency in tasks like object detection. During back-propagation, the dense connectivity in the GFPN supports efficient transmission of feature information across the network hierarchy. Dense-link and $\log_2(n)$-link configurations are illustrated in Figure 3.

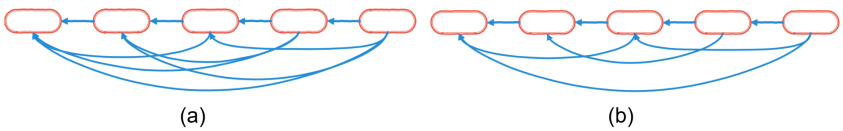

Figure 3. skip-layer links: (**a**) dense-link: concatenates features from all preceding layers; (**b**) $\log_2 n$-link: concatenates features from up to $\log_2(l) + 1$ layers at each level.

Another significant improvement in the GFPN is the queen-fusion module, which facilitates cross-scale connections to enhance adaptability to multi-scale variations. This module utilizes a 3×3 convolution to merge features across different scales, gathering input features from diagonally adjacent nodes above and below, to minimize information loss during feature fusion. Implementing this approach enhances the network's capability to handle multi-scale variations, potentially improving the overall performance robust-

ness. Figure 4 illustrates various methods for integrating features across different layers, including the FPN, PANet, BiFPN, and GFPN.

In YOLOv8, integrating the PAFPN with C2f modules effectively combines feature maps across scales, thereby enhancing object detection capabilities. This study aimed to enhance YOLOv8's small object detection using advanced feature fusion techniques. However, replacing the PAFPN with the GFPN in YOLOv8 improves precision, while introducing a higher latency compared to the PAFPN-based model.

This paper introduces an enhanced and efficient GFPN, depicted in Figure 5, inspired by efficient-RepGFPN [41]. By integrating it into YOLOv8, the model achieves superior performance, without significantly increasing computational complexity or latency. The efficient-RepGFPN reduces complexity by parameterizing and eliminating additional upsampling operations in queen-fusion. Furthermore, it upgrades the feature fusion module to a CSPNet, enhancing the merging of features across different scales [41].

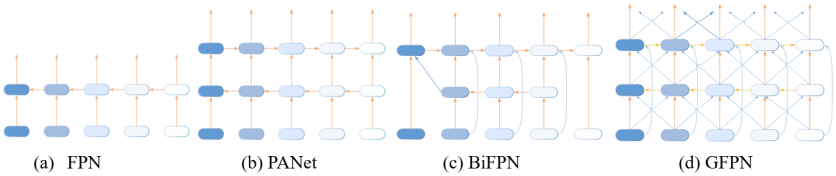

Figure 4. Different feature pyramid network designs: (**a**) FPN uses a top-down strategy; (**b**) PANet enhances FPN with a bottom-up pathway; (**c**) BiFPN integrates cross-scale pathways bidirectionally; (**d**) GFPN includes a queen-fusion style pathway and skip-layer connections.

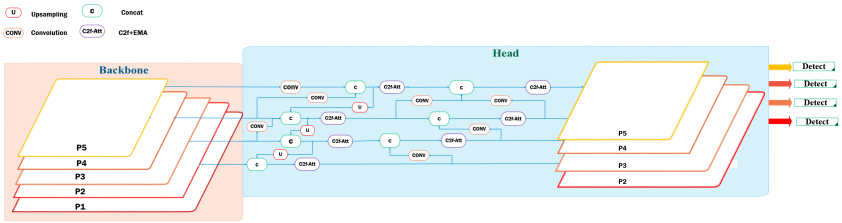

Figure 5. Enhanced and efficient GPFN structure.

In the feature fusion block of the GFPN architecture, we replace the conventional 3 × 3 convolution-based feature fusion with C2f-EMA, incorporating an attention mechanism. This module merges high-level semantic features with low-level spatial details, thereby enhancing the representation and detection accuracy of small objects. These modifications maintain the GFPN's ability to improve feature interaction and efficiency by effectively managing both types of information in the neck section. Inspired by the efficient-RepGFPN, we also reparametrize and eliminate additional upsampling operations in queen-fusion. Ultimately, these enhancements improve the efficiency and effectiveness of YOLOv8 for object detection tasks, without significantly increasing computational complexity or latency.

We enhance the network's capability by adding a detection layer, which involves integrating feature maps from P2 alongside those from P3 to P5, which are used in YOLOv8. This enhancement significantly improves the network's ability to detect small objects. As depicted in Figure 5, P2 with a resolution of 320 × 320 plays a crucial role in preserving the finer spatial details essential for improving small object detection. Additionally, a detection head is introduced, prompting the network structure to focus more on features related to small objects.

This approach not only provides higher-resolution details but also enhances feature fusion, offers comprehensive contextual information, and enables precise localization. By leveraging features from both fine and coarse scales simultaneously, the network achieves accurate detection of small objects, effectively capturing finer details. The enhanced architectural design illustrated in Figure 5 is implemented and evaluated in this study.

4.2. Embedding Efficient Multi-Scale Attention Mechanism in C2f

The C2f module in YOLOv8 enhances gradient flow and detection accuracy by dynamically adjusting channel numbers through split and concatenate operations, optimizing feature extraction while managing computational complexity [27]. It incorporates convolutional and residual structures to deepen network training and address the vanishing gradient problem, thereby improving feature extraction.

The C2f-EMA module introduced in this paper enhances feature extraction by redistributing feature weights using the EMA attention mechanism. This mechanism prioritizes relevant features and spatial details across different channels within the image. Figure 6 illustrates the EMA structure, which partitions input features into groups, processes them through parallel subnetworks, and integrates them with advanced aggregation techniques. This enhancement significantly boosts the representation of small, challenging objects and enhances the efficiency of the network backbone.

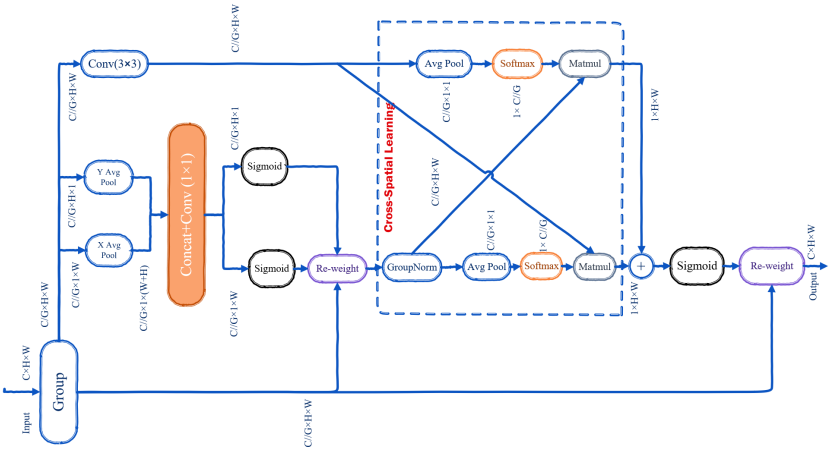

Figure 6. Efficient multi-scale attention mechanism.

The EMA module employs feature grouping to partition the input feature map X along the channel dimension into G sub-features, denoted as $X = [X_1, X_2, \ldots, X_G]$, where each $X_i \in \mathbb{R}^{\frac{C}{G} \times H \times W}$. This approach enables specialized feature extraction and representation by allowing the network to learn different semantics or characteristics within each group. Additionally, it optimizes the CNNs by reducing computation.

The EMA module uses a parallel subnetworks approach to efficiently capture multi-scale spatial information and cross-channel dependencies. It features two parallel branches: the 1×1 branch, with two routes, and the 3×3 branch, with one route. In the 1×1 branch, each route employs 1D global average pooling to encode channel information along the horizontal and vertical spatial directions. These operations produce two encoded feature vectors representing global information, which are then concatenated along the height direction. A 1×1 convolution layer is subsequently applied to the concatenated output to maintain channel integrity and capture cross-channel interactions by blending information across different channels. The outputs are split into two vectors, refined using non-linear sigmoid functions to adjust attention weights in a 2D binomial distribution. Channel-wise attention maps are then combined through multiplication within each group, enhancing the interactive features across channels.

EMA differs from traditional attention methods by addressing issues such as neglecting interactions among spatial details and the limited scope of 1×1 kernel convolution, accomplished through the inclusion of a 3×3 convolution branch. This branch uses a single route with a 3×3 convolution kernel to capture multi-scale spatial information. Additionally, the output of the 1×1 branch undergoes 2D global average pooling to encode global spatial information. The pooled output is integrated with the transformed output

from the 3 × 3 branch, aligning dimensions to enhance feature aggregation through both spatial information sources.

Unlike traditional attention methods that use basic averaging, EMA integrates attention maps from parallel subnetworks using a cross-spatial learning approach. It uses matrix dot product operations to capture relationships between individual pixels, enriching the global context across all pixels. Specifically, the EMA module combines the global and local spatial information from its parallel 1 × 1 and 3 × 3 branches to enhance feature representation. This approach effectively captures long-range dependencies and multi-scale spatial details, improving overall feature aggregation.

SoftMax is then applied to the outputs to generate 2D Gaussian maps, highlighting relevant features and modeling long-range dependencies. This process is repeated for the second spatial attention map, using 2D global average pooling and sigmoid functions to preserve precise spatial positional information. Finally, feature maps obtained from spatial attention weights within each group are aggregated. The resulting output feature map retains the original input size, ensuring efficiency and effectiveness for integration into architectures. The final output is a redistributed feature map that captures pixel-level pairwise relationships and highlights the global context across all pixels and channels, allocating higher weights to more relevant features and spatial details.

In this paper, we introduce the C2f-EMA as a replacement for C2f, redistributing the feature map using the EMA structure to assign higher weights to more relevant features and spatial details within images. This enhancement aims to improve the detection performance, especially for small objects with very fine details due to their size. C2f-EMA includes an initial convolution, split function, EMA module, and parallel processing, collectively enhancing the network's overall performance. As shown in Figure 7, this mechanism operates within the second residual block of the C2f.

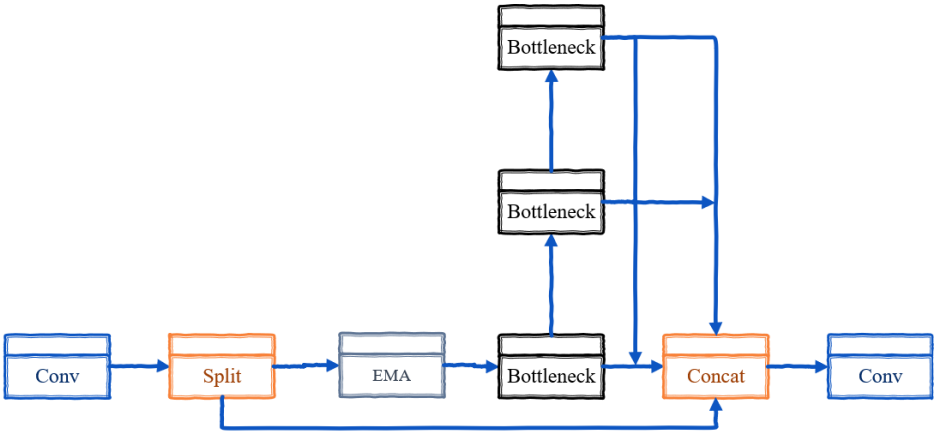

Figure 7. C2f-EMA.

4.3. Improved Bounding Box Loss Function

Bounding box loss functions penalize discrepancies between predicted and ground truth bounding box parameters to enhance object localization. IoU-based loss functions are essential for this purpose, measuring overlap as the ratio of intersection over union. However, their effectiveness diminishes when there is no overlap, resulting in negligible gradients. Several IoU-based loss functions have been developed to address this limitation, each presenting unique approaches and specific limitations. CIoU, implemented in YOLOv8, considers the distance between box centers and differences in aspect ratios. This refinement enhances the convergence speed and overall performance. However, the aspect ratio penalty term in CIoU may not sufficiently account for size variations between boxes of the same aspect ratio but different dimensions. Moreover, CIoU involves computationally intensive inverse trigonometric functions, which could pose drawbacks in real-time applications. The equation for CIoU is presented in Equation (1):

$$L_{\mathrm{CIOU}} = L_{\mathrm{IOU}} + \frac{d^2}{c^2} + v,\ v \quad = \frac{4}{\pi^2}\left(\arctan\left(\frac{w_{\mathrm{gt}}}{h_{\mathrm{gt}}}\right) - \arctan\left(\frac{w}{h}\right)\right)^2 \tag{1}$$

where d represents the Euclidean distance between the center points of the predicted and ground truth bounding boxes. Additionally, c represents a normalization factor, typically representing the diagonal length of the smallest enclosing box that contains both the predicted and ground truth bounding boxes. v represents the aspect ratio penalty, which accounts for discrepancies in aspect ratios between the predicted and ground truth boxes. (w, h) and $(w_{\mathrm{gt}}, h_{\mathrm{gt}})$ represent the width and height of the predicted and ground truth bounding box, respectively.

Efficient intersection over union (EIoU) adjusts CIoU by using distinct penalty terms for width and height rather than a shared aspect ratio penalty, aiming for a more precise measurement of differences between anchor box and target box dimensions. Equation (2) presents the EIoU formula.

$$L_{\mathrm{EIOU}} = L_{\mathrm{IOU}} + \frac{d^2}{c^2} + \frac{(w_{\mathrm{pred}} - w_{\mathrm{gt}})^2}{w_c^2} + \frac{(h_{\mathrm{pred}} - h_{\mathrm{gt}})^2}{h_c^2} \tag{2}$$

where w_c and h_c represent the width and height of the smallest enclosing bounding box, respectively. Despite addressing size discrepancies, EIoU encounters challenges such as anchor box enlargement during regression and slow convergence. This issue is critical in object detection models that use IoU-based loss functions, where optimization may inadvertently enlarge anchor boxes, instead of precisely converging them to target sizes, thereby reducing localization precision [42]. CIoU and EIoU losses use the term $R_D = \frac{d^2}{c^2}$, where d is the diagonal length of the intersection between the anchor and target boxes, and c is the diagonal length of the smallest enclosing box covering both. The gradient of R_D with respect to d is $\frac{2d}{c^2}$, meaning R_D decreases as c increases. The problem is when the boxes do not overlap, enlarging the anchor box increases c, reducing R_D, and thus lowering the CIoU and EIoU losses without improving the overlap. Using c as the denominator in the penalty term is flawed, allowing loss reduction through anchor box size manipulation rather than overlap improvement, indicating the need for a revised penalty term to better handle non-overlapping boxes [42].

Wise intersection over union (WIoU) [26] introduces a dynamic, non-monotonic focusing mechanism in bounding box regression. This mechanism prioritizes anchor boxes with moderate quality and reduces harmful gradients from low-quality examples. WIoU uses the aspect ratio and the distance between predicted and ground truth boxes as penalty terms. It evaluates anchor box quality dynamically by comparing each box's quality to the average quality of all boxes in the batch, giving more attention to those with moderate quality. The WIoU calculation is given by Equation (3).

$$L_{\mathrm{WIoUv3}} = \frac{\beta}{\delta \alpha^{\beta-\delta}} \cdot e^{\left(\frac{(x-x_{gt})^2+(y-y_{gt})^2}{w_{gt}^2+h_{gt}^2}\right)}, \quad \beta = \frac{L_{\mathrm{IOU}}^*}{\overline{L}_{\mathrm{IOU}}}, \quad L_{\mathrm{IOU}} \in [0, +\infty) \tag{3}$$

The non-monotonic attention function of WIoU is denoted by β, while δ and α serve as hyperparameters that regulate its gradient. The operation $*$ denotes the detach operation, and L_{IOU} indicates the average L_{IOU} value across all anchor boxes within a batch. WIoU introduces attention-based predicted box loss and focusing coefficients. However, it relies on multiple hyperparameters, posing challenges in optimization for diverse datasets.

We use PIoU [42] as a replacement for the CIoU in the original network. The details of the PIoU method are outlined in Algorithm 1. The penalty term in PIoU enhances bounding box regression by minimizing the Euclidean distance between corresponding corners of predicted and ground truth boxes. This approach offers a more intuitive measure of similarity and proves effective for both overlapping and non-overlapping boxes. Unlike traditional IoU-based loss functions, PIoU mitigates the issue of box enlargement, ensuring precise and stable box regression. The simulated results in Figure 8 demonstrate its effectiveness. Figure 8 illustrates an experiment evaluating anchor box regression using various loss

functions. The CIoU loss function exhibited a continuous enlargement of anchor boxes from epochs 25 to 75 and failed to achieve full convergence to the ground truth anchor box by epoch 150. In contrast, the anchor box guided solely by the penalty term in the PIoU loss function, without consideration of the attention function, did not show enlargement issues during training, as observed in epochs 25 and 75. By epoch 75, it demonstrated almost complete convergence to the ground truth bounding box, reaching a perfect fit by epoch 150. PIoU loss uses a non-monotonic attention layer to enhance focus on medium- and high-quality anchor boxes. By prioritizing moderate-quality stages in anchor box regression, PIoU improves the object detector performance. The non-monotonic attention function $u(\lambda q)$ is controlled by the parameter λ. PIoU simplifies the tuning process by requiring only one hyperparameter. The penalty factor P is replaced with q, which indicates anchor box quality on a scale from 0 to 1. When $q = 1$ (meaning $P = 0$), the anchor box perfectly aligns with the target box. As P increases, q decreases, signifying lower-quality anchor boxes.

Algorithm 1: PIoU bounding box regression

Input:
- Two arbitrary convex shapes: $A, B \subseteq \mathbb{R}^n$
- Width and height of input image: w, h
- Width and height of ground truth box: $w_{\text{gt}}, h_{\text{gt}}$
- Coordinates of bounding box 1: $b1_{x1}, b1_{x2}, b1_{y1}, b1_{y2}$
- Coordinates of bounding box 2: $b2_{x1}, b2_{x2}, b2_{y1}, b2_{y2}$
- IoU (Intersection over Union)

Output:
- Powerful-IoU

Steps:

1. Calculate Absolute Differences for Widths:
   - $dw1 = |\min(b1_{x2} - b1_{x1}) - \min(b2_{x2} - b2_{x1})|$
   - $dw2 = |\min(b1_{x2} - b1_{x1}) - \min(b2_{x2} - b2_{x1})|$
2. Calculate Absolute Differences for Heights:
   - $dh1 = |\min(b1_{y2} - b1_{y1}) - \min(b2_{y2} - b2_{y1})|$
   - $dh2 = |\min(b1_{y2} - b1_{y1}) - \min(b2_{y2} - b2_{y1})|$
3. Compute Parameter P:

$$P = \frac{1}{4}\left(\frac{dw1 + dw2}{w_{\text{gt}}} + \frac{dh1 + dh2}{h_{\text{gt}}}\right)$$

4. Calculate Custom Loss L:

$$L = 1 - \text{IoU} - e^{-P^2}$$

5. Calculate the Focal Loss by Adding an Attention Layer:

$$q = e^{-P}, \quad q \in (0, 1]$$

$$u(x) = 3x \cdot e^{-x^2}$$

$$L_{PIoU} = u(\lambda q) \cdot L$$

$$L_{PIoU} = 3 \cdot (\lambda q) \cdot e^{-(\lambda q)^2} \cdot L$$

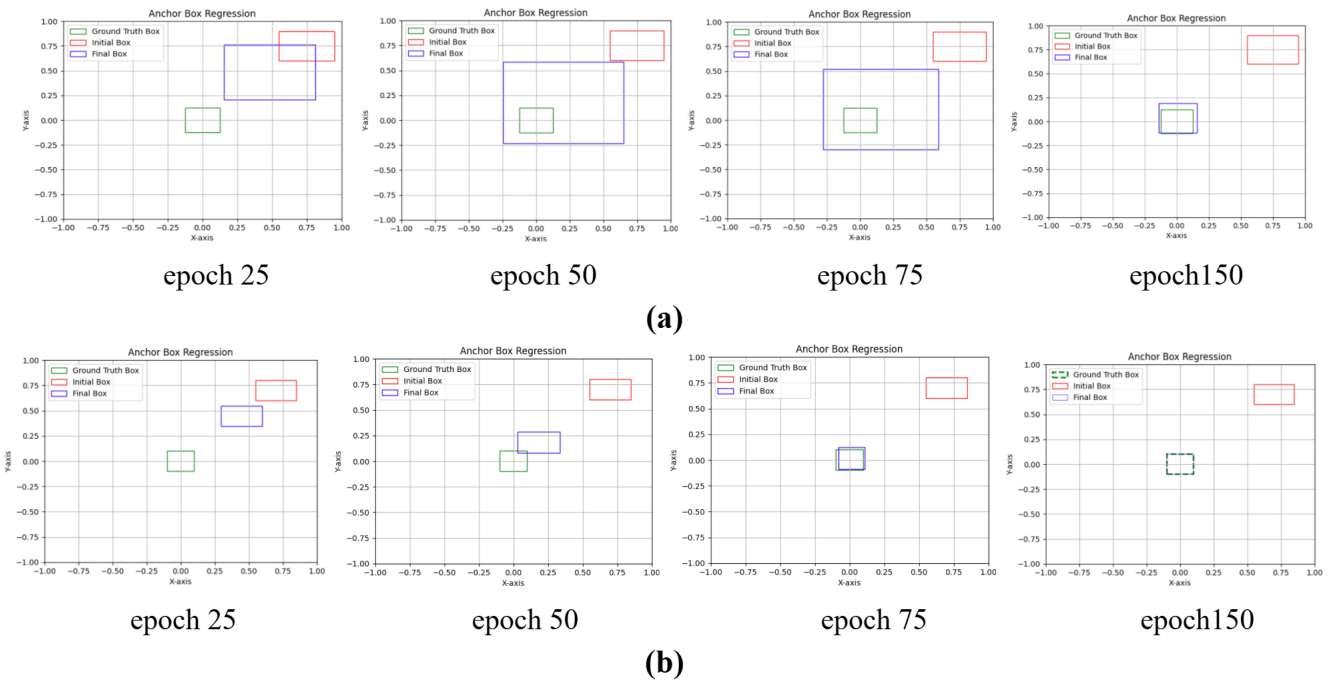

Figure 8. Anchor box regression process guided by (**a**) complete IoU-based loss function (CIoU), (**b**) penalty term in powerful-IoU (PIoU) loss function without attention function.

5. Results

This section begins with an introduction to the dataset utilized in this paper, followed by detailing the experimental environment and training strategy. It further outlines the evaluation metrics employed to assess the model's performance. The effectiveness of the proposed approach is then demonstrated through a comparative analysis with state-of-the-art models, using YOLOv8 as the baseline. Furthermore, the section includes an evaluation of the model's performance in challenging real-world scenarios, such as detecting distant objects and small objects positioned far from the camera.

5.1. Dataset

The VisDrone2019 dataset [23], a prominent collection of UAV aerial photography, was developed by Tianjin University's Lab of Machine Learning and Data Mining in collaboration with the AISKYEYE data mining team. It comprises 288 video clips totaling 261,908 frames and 10,209 static images. These visuals were captured using various drone-mounted cameras, showcasing diverse scenarios across more than a dozen cities throughout China. The dataset is exceptionally rich, featuring a wide range of geographic locations, environmental settings, and object types. Geographically, the dataset covers footage from 14 different cities across China, offering a comprehensive spectrum of scenes from urban to rural landscapes. It includes a diverse array of objects such as pedestrians, cars, bicycles, and more. Additionally, the dataset spans various population densities, ranging from sparse to densely crowded areas, and captures images under different lighting conditions, including both daytime and nighttime scenes. One distinguishing feature of the VisDrone2019 dataset is its inclusion of numerous small objects of varying sizes, depicted from different angles and within various scenes. This diversity increases the dataset's complexity and difficulty compared to other computer vision datasets. Figure 9 illustrates the process of manually annotating objects in the VisDrone2019 dataset.

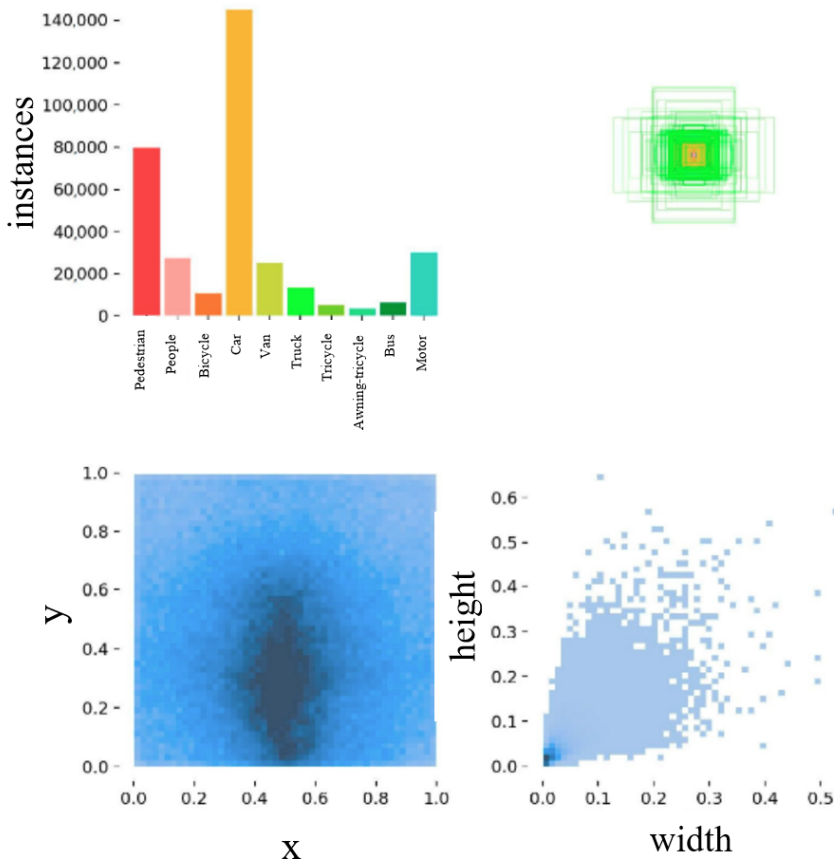

Figure 9. Information regarding the manual annotation process for objects in the VisDrone2019 dataset.

5.2. Experimental Environment and Training Strategies

In this study, YOLOv8s was selected as the baseline model for investigation and further enhancements. The model was trained on the VisDrone dataset using an NVIDIA RTX A6000 GPU (48 GB) on Linux, utilizing PyTorch 2.2.1 and CUDA 12.1. Training involved optimizing key parameters, running for 200 epochs with the stochastic gradient descent (SGD) optimizer [43] set to a momentum of 0.932. The initial learning rate started at 0.01 and decayed gradually to 0.0001. A batch size of 32 was chosen for efficient memory usage and stable training, with input images resized to 640 × 640 pixels. A weight decay of 0.0005 was also applied to prevent overfitting and improve model generalization.

5.3. Evaluation Metrics

To assess the detection performance of our enhanced model, we utilized several evaluation metrics: precision, recall, mAP0.5, mAP0.5 : 0.95, and the number of model parameters. The specific formulas for these metrics are provided in this section.

Precision is the metric that represents the ratio of true positives to the total predicted positives, as defined by Equation (4):

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4}$$

True positives (TP) is the number of instances where the model accurately predicts a positive instance. False positives (FP) is the number of instances where the model incorrectly predicts a positive instance. False negatives (FN) is the number of instances where the model fails to predict a positive instance.

Recall measures the ratio of correctly predicted positive samples to all actual positive samples, as defined by Equation (5):

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5}$$

Average precision (AP) represents the area under the precision–recall curve, calculated using Equation (6):

$$\text{AP} = \int_0^1 \text{Precision}(\text{Recall})\, d(\text{Recall}) \tag{6}$$

Mean average precision (mAP) represents the average AP value across all categories, indicating the model's overall detection performance across the entire dataset. This calculation is defined by Equation (7):

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^{N} \text{AP}_i \tag{7}$$

where AP_i represents the average precision value for the category indexed by i, and N denotes the total number of categories in the training dataset.

mAP$_{0.5}$ is the average precision calculated at an IoU threshold of 0.5.

mAP$_{0.5:0.95}$ is calculated across IoU thresholds from 0.5 to 0.95, with values computed at intervals of 0.05.

5.4. Experiment Results

This section presents a comprehensive evaluation of the SOD-YOLOv8 model through targeted experiments. We began by comparing the PIoU loss function with other common loss functions in YOLOv8s. Next, we assessed the integration of the GFPN structure with the EMA and other attention modules. We then evaluated SOD-YOLOv8s against various YOLO variants (YOLOv3, YOLOv5s, and YOLOv7) and widely used models (Faster R-CNN, CenterNet, Cascade R-CNN, and SSD). Ablation studies validated the contributions of each enhancement. Visual experiments with the VisDrone2019 dataset demonstrated the model's effectiveness in diverse scenarios, including distant, high-density, and nighttime conditions. Finally, real-world traffic scene evaluations highlighted the model's applicability and performance in challenging environments with cameras mounted on buildings at significant distances from the objects of interest.

5.4.1. Comparative Evaluation of Bounding Box Regression

To evaluate the impact of PIoU, we conducted comparative experiments on YOLOv8s using PIoU and other common loss functions under consistent training conditions. As shown in Table 1, PIoU achieved the best detection performance. Specifically, it improved $\text{mAP}_{0.5}$ by 1.1%, $\text{mAP}_{0.5:0.95}$ by 0.2%, precision by 1.6%, and recall by 0.4% compared to CIoU. Additionally, the simpler loss function of PIoU made model tuning easier, demonstrating its potential as an efficient and effective bounding box regression method.

Table 1. Detection results of YOLOv8s with different bounding box loss functions, shown as percentages (best outcomes in bold).

Metrics	Precision%	Recall%	mAP$_{0.5}$%	mAP$_{0.5:0.95}$%
CIOU	51.2	40.1	40.6	24
EIOU	49.8	39.9	40	23
WIOU v1	51.4	40	40.7	23.7
WIOU v2	51.9	40	40.7	23.8
WIOU v3	52.6	40	41.2	**24.2**
MPDIOU [44]	52.1	39	40.7	23.9
PIoU ($\lambda = 1.2$)	**52.8**	**40.5**	**41.7**	**24.2**

5.4.2. Comparative Experiment of Attention Mechanisms

To evaluate the effectiveness of integrating the GFPN structure with the EMA attention mechanism, we incorporated three widely used attention modules—CBAM [45], CA [46], and SE [47]—at the same position within the GFPN structure. This setup allowed for a direct comparison in our experiments, as detailed in Table 2. The experimental results demonstrated that training with the GFPN-EMA combination consistently outperformed the GFPN configurations with CBAM, CA, and SE in terms of $mAP_{0.5}$ values. Specifically, the GFPN-EMA model exhibited significant improvements across most object categories, particularly in the pedestrian, people, car, bus, and motor classes, as well as overall $mAP_{0.5}$. As shown in Table 2, these findings highlighted EMA's efficacy in improving small object detection accuracy in the VisDrone dataset. EMA achieved this by addressing spatial interactions, overcoming the limitations of 1×1 kernel convolution through the integration of a 3×3 convolution for multi-scale spatial information, and employing cross-spatial learning to merge attention maps from parallel subnetworks. This approach effectively combined global and local spatial contexts.

Table 2. Detection results of GFPN-YOLOv8s with various attention mechanisms presented as percentages. (Best results are highlighted in bold).

Models	Pedestrian	People	Bicycle	Car	Van	Truck	Bus	Motorcycle	$mAP_{0.5}$
YOLO-GFPN-SE	50.9	42.6	15.0	83.5	45.1	39.2	53.4	51.3	42.9
YOLO-GFPN-CBAM	48.8	40.3	14.6	82.9	46.0	38.8	55.4	49.3	42.3
YOLO-GFPN-CA	51.0	42.3	**17.8**	83.8	**47.5**	**40.1**	57.3	52.7	43.9
YOLO-GFPN-EMA	**51.0**	**43.3**	16.7	**83.8**	46.7	39.1	**60.9**	**52.8**	**44.2**

5.4.3. Comparison with Different Mainstream Models

When comparing SOD-YOLOv8s with other YOLO variants such as YOLOv3, YOLOv5s, and YOLOv7, SOD-YOLOv8s was remarkably efficient. Despite having a small model size of 11.5 million parameters, SOD-YOLOv8s achieved the highest accuracy metrics. It outperformed YOLOv3, YOLOv5s, and YOLOv7, despite YOLOv3 and YOLOv5s having larger model sizes, ranging from 12.0 million to 18.3 million parameters. The YOLOv8 model was adapted into various scales (YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x) by adjusting width and depth, each progressively consuming more resources to improve detection performance. We conducted comparisons between SOD-YOLOv8s and the different scales of YOLOv8 to further validate the performance of our proposed approach. Based on the information presented in Table 3, despite its lower parameter count of 11.5 million, SOD-YOLOv8s achieved the highest recall (43.9%), $mAP_{0.5}$ (45.1%), and $mAP_{0.5:0.95}$ (26.6%). In contrast, YOLOv8m, which had 25.9 million parameters, achieved lower accuracy metrics. This indicates that SOD-YOLOv8s is efficient in terms of computing capacity and model size, while also performing well in object detection tasks.

This study conducted a comparative experiment to evaluate the performance of SOD-YOLOv8s against widely adopted models, including faster R-CNN, CenterNet, Cascade R-CNN, and SSD. In faster R-CNN [13], the region proposal network (RPN) [13] relies on backbone network features to generate region proposals. However, due to a lower feature map resolution for small objects, the RPN may struggle to accurately localize them, leading to potential missed detections. Cascade R-CNN [48] enhances detection performance through a multilevel architecture, albeit at the cost of increased computational complexity and training difficulty. CenterNet [49] simplifies the architecture with an anchor-free, center-point approach but faces challenges in precisely locating small objects in crowded or obscured scenes. These challenges arise from ambiguous object centers, interference from larger objects, complex backgrounds obscuring object centers, and sensitivity to pixel-level inaccuracies. Additionally, SSD's performance decreases on smaller objects compared to larger ones, as its shallow neural network layers may lack the detailed high-level features necessary for accurate small object prediction. According to the data provided in Table 4, the SOD-YOLOv8s model achieved

the highest performance with an $AP_{0.5}$ of 45.1% and $AP_{0.5:0.95}$ of 26.6% compared to the other models such as CenterNet, Cascade R-CNN, SSD, and faster R-CNN.

Table 3. Different YOLO models' results, presented as percentages.(The best-performing outcomes are highlighted in bold).

Models	Model's Size	Backbone	Precision	Recall	$mAP_{0.5}$	$mAP_{0.5:0.95}$	Time/ms	Parameter/10^6
YOLOv3 [30]	-	Darknet-53	53.6	43.2	42	23.1	209	18.3
YOLOv5s	-	CSP-Darknet-53	46.7	34.8	34.7	19.2	13.9	12.0
YOLOv7 [50]	-	ELAN	51.5	42.3	40.1	21.8	71.5	**1.7**
YOLOv8 [51]	YOLOv8n	CSP-Darknet-53	44.0	33.2	33.5	19.5	**6.7**	4.2
	YOLOv8s		51.1	39.1	39.6	23.8	7.8	11.1
	YOLOv8m		**55.8**	42.6	44.5	**26.6**	16.8	25.9
SOD-YOLOv8s	-	CSP-Darknet-53	53.9	**43.9**	**45.1**	**26.6**	17.7	11.5

Table 4. Results from different widely used models, presented as percentages.(The best-performing outcomes are highlighted in bold).

Models	Backbone	$AP_{0.5}$	$AP_{0.5:0.95}$
Faster R-CNN [13]	ResNet	37.8	21.5
Cascade R-CNN [48]	ResNet	39.4	24.2
CenterNet [49]	ResNet50 [52]	39.1	22.8
SSD [17]	MobileNetV2 [53]	33.7	19
SOD-YOLOv8s	CSP-Darknet-53	**45.1**	**26.6**

5.4.4. Ablation Experiments

To validate the efficacy of each enhancement approach proposed in this study, ablation experiments were conducted on the baseline model. The results from these tests, shown in Table 5, demonstrate that each enhancement significantly improved the detection performance across various categories. Introducing PIoU for bounding box regression enhanced localization, without enlargement issues, leading to a significant 1.1% increase in $mAP_{0.5}$. This improvement was particularly beneficial for categories such as pedestrian, people, and bicycle. Integrating an enhanced GFPN and incorporating a new small object detection layer into the YOLOv8 network resulted in a significant 2.9% increase in $mAP_{0.5}$. This enhancement demonstrated substantial performance improvements across all categories, including pedestrian, bicycle, car, van, and motor, highlighting GFPN's effectiveness in capturing multi-scale features. Furthermore, integrating the C2f-EMA module, which utilized the EMA attention mechanism, and replacing C2f with it within the neck layers increased the $mAP_{0.5}$ by 0.5%. This enhancement notably benefited categories such as people, motor, and truck, demonstrating its effectiveness in improving detection across various categories. According to Table 6, our proposed efficient model enhanced the object detection performance significantly, without adding significant computational cost or latency compared to YOLOv8s. It improved recall from 43% to 44%, precision from 45% to 46%, $mAP_{0.5}$ from 40% to 45.1%, and $mAP_{0.5:0.95}$ from 20% to 26.6%.

Table 5. Comparative experiments between the enhanced model and YOLOv8s across various categories, with percentages presented (best-performing outcomes highlighted in bold).

Models	Pedestrian	People	Bicycle	Car	Van	Truck	Bus	Motorcycle	$mAP_{0.5}$
YOLOv8s	43.5	34.2	14.9	79.5	45.0	40.3	58.1	45.4	40.6
YOLOv8s-PIoU	46	38.2	15.6	80.4	45.8	39	60.2	47	41.7
YOLOv8s-PIoU-GFPN	52.8	44.0	17.7	**84.2**	**47.7**	39.4	60.8	53.2	44.6
YOLOv8s-PIoU-GFPN-EMA	**53.1**	**44.5**	**18.2**	83.9	47.1	**41.0**	**60.9**	**53.8**	**45.1**

Table 6. Detection results following the adoption of different improvement strategies, presented as percentages.

Baseline	PIoU	GFPN	EMA	Precision	Recall	$mAP_{0.5}$	$mAP_{0.5:0.95}$	Detection Time/ms	Parameter/10^6
✓				51.2	40.1	40.6	24	7.8	11.1
✓	✓			52.8	40.5	41.7	24.2	7.4	11.1
✓	✓	✓		52.7	44.3	44.6	26.3	11.5	11.5
✓	✓	✓	✓	53.9	43.9	45.1	26.6	11.6	11.5

Figure 10 depicts the evaluation metrics for SOD-YOLOv8 and YOLOv8s across 200 training epochs. Our model outperformed YOLOv8s in precision and $mAP_{0.5}$ starting around epoch 15 and stabilized after 50 epochs. This illustrates that SOD-YOLOv8 significantly enhanced the detection performance, particularly for small and challenging objects, without introducing significant complexity.

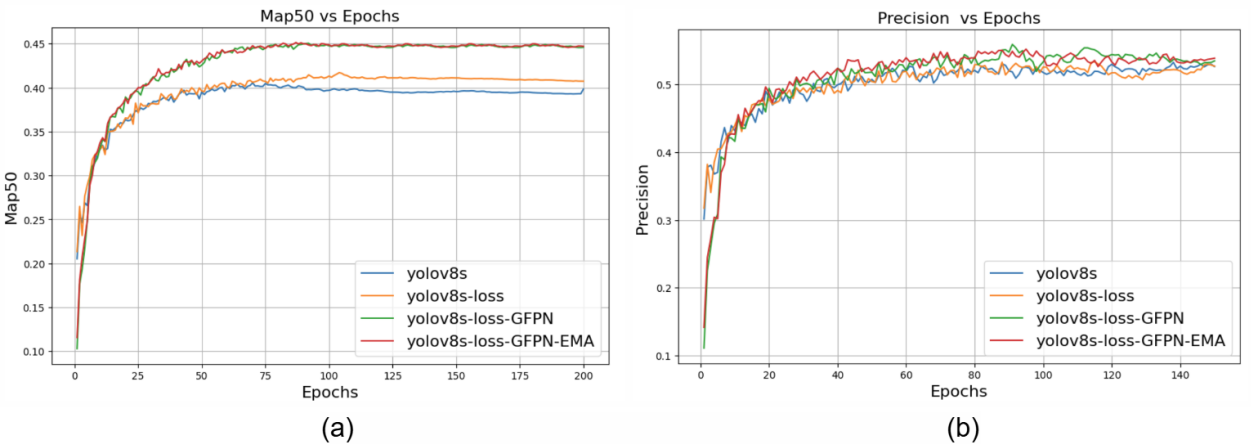

Figure 10. (**a**) Training progress plot comparing YOLOv8s-GFPN-EMA, YOLOv8s-GFPN, and YOLOv8s based on $mAP_{0.5}$ (**b**) and precision.

5.4.5. Visual Assessment

We conducted visual experiments to evaluate our model's detection performance. Our analysis included various metrics such as confusion matrices and inference test results. To validate the effectiveness of our method in challenging real-world scenarios, we performed inference tests using images captured by a camera mounted on the 12th floor of a building. This scenario involved capturing images from a high vantage point, posing challenges in detecting numerous small objects in a crowded traffic scene at an intersection.

VisDrone2019 Dataset Results

To visualize the performance of SOD-YOLOv8s, we utilized a confusion matrix. This matrix organizes predictions into a format where each row corresponds to instances of a true class label, and each column corresponds to instances predicted by the model. Diagonal elements indicate correct predictions, where the predicted class matches the actual class. Off-diagonal elements represent incorrect predictions, where the predicted class does not match the actual class.

Figure 11 demonstrates the improved detection performance of SOD-YOLOv8s across most object categories. The confusion matrix shows lighter shades in the last row compared to YOLOv8s, indicating reduced misclassifications of objects as background. However, challenges remained in accurately identifying bicycles, tricycles, and awning-tricycles, which were often mislabeled as background. Despite these issues, SOD-YOLOv8s shows darker shades along the main diagonal, indicating an overall increase in correctly detected objects.

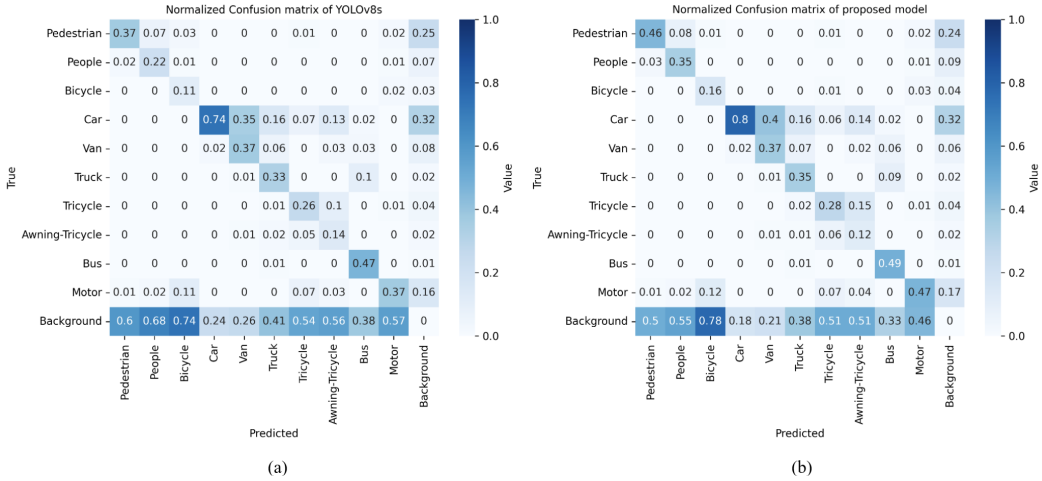

Figure 11. (**a**) Confusion matrix of YOLOv8s; (b) confusion matrix of proposed model.

As depicted in Figure 12, we assessed the efficacy of SOD-YOLOv8 across three challenging scenarios within the Visdrone dataset: nighttime conditions, crowded scenes with high-density objects, and scenes with distant objects. Remarkably, across all three scenarios, notable improvements were observed. In the nighttime scenario, illustrated in the first row of Figure 12, objects were detected with higher IoU values, and a greater number of smaller objects were successfully identified. In the second scenario, depicted in the second row of Figure 12, SOD-YOLOv8 demonstrated superior performance by successfully detecting numerous small objects located at the corners of intersections, a task which YOLOv8s struggled with. Similarly, in the third scenario involving objects positioned farther from the camera, SOD-YOLOv8s excelled in detecting objects with higher IoU values and successfully identifying a greater number of smaller objects. These results demonstrated the substantial improvements provided by SOD-YOLOv8s across different environmental conditions, indicating its reliability and effectiveness in detecting objects in challenging scenarios.

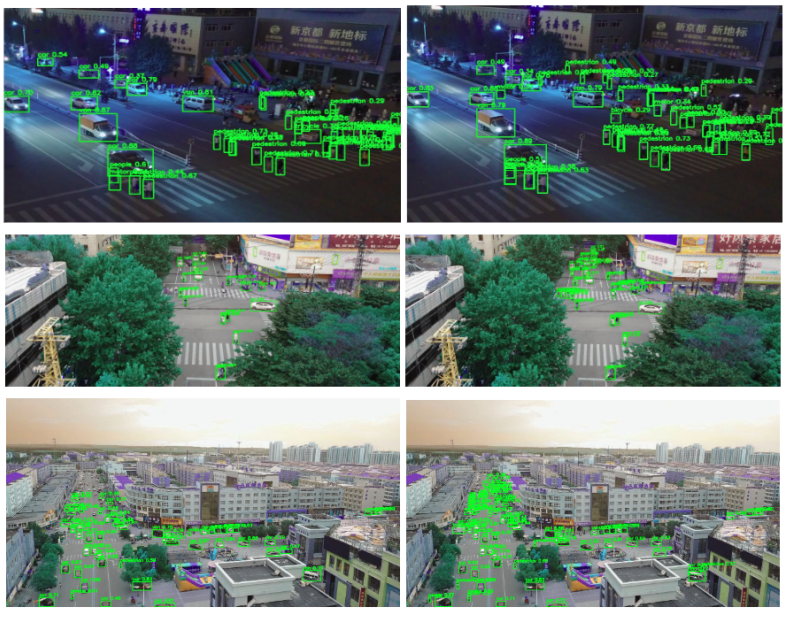

Figure 12. Inference results for (**a**) YOLOv8s and (**b**) SOD-YOLOv8s across diverse scenarios including distant and high-density objects, as well as nighttime scenarios, using the VisDrone2019 dataset.

Real Dataset Results

This section evaluates the model's performance in dynamic, real-world challenging scenarios where a camera was mounted on a building at a significant distance from the objects of interest. To assess the applicability and generalization of the proposed SOD-YOLOv8 model, we conducted inference experiments using real-world data from a traffic scene scenario. The image data were primarily captured by NSF PAWR COSMOS testbed cameras [54–56] mounted on the 12th floor of Columbia's Mudd building (Figure 13), overlooking the intersection of 120th St. and Amsterdam Ave. in New York City. Images were specifically selected from this vantage point to utilize its elevated perspective and greater distance from the street. This viewpoint posed a unique challenge for object detection, requiring enhanced perception due to the reduced scale of objects, including various vehicle types and pedestrians.

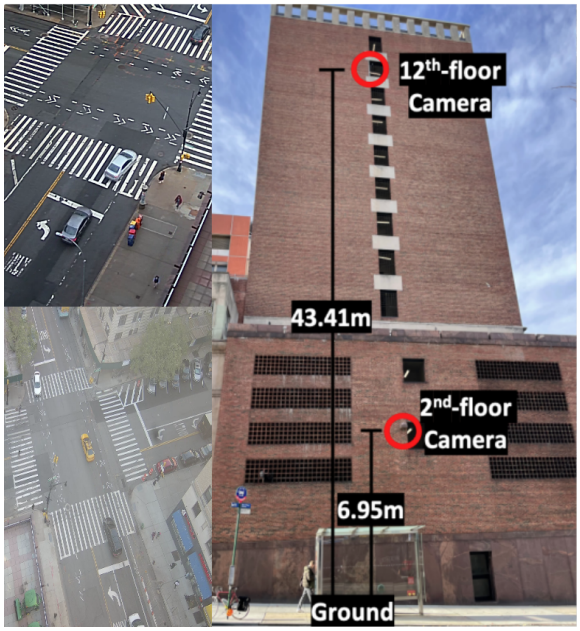

Figure 13. The perspective captured by COSMOS cameras on the 12th floor of Columbia's Mudd building overlooking the intersection [56].

As depicted in Figure 14, we assessed SOD-YOLOv8's performance in three challenging real-world traffic scenarios using images from cameras on the 12th floor, such as crowded scenes with high-density objects, distant objects, and nighttime conditions. Significant improvements were observed across all three scenarios. In the first scenario, shown in the top row of Figure 14, SOD-YOLOv8 outperformed YOLOv8s by successfully detecting numerous small-scale pedestrians at the corners of intersections, a task where YOLOv8s struggled. In the second scenario, with distant objects, SOD-YOLOv8 showed superior performance, achieving higher IoU values and effectively detecting more small objects. In the nighttime scenario, shown in the third row of Figure 14, SOD-YOLOv8 achieved higher IoU values for detected objects and identified more small objects compared to the YOLOv8s baseline model, despite challenging lighting conditions. These results illustrate the substantial improvements achieved by SOD-YOLOv8 across diverse environmental conditions, highlighting its reliability and effective object detection capabilities in challenging scenarios.

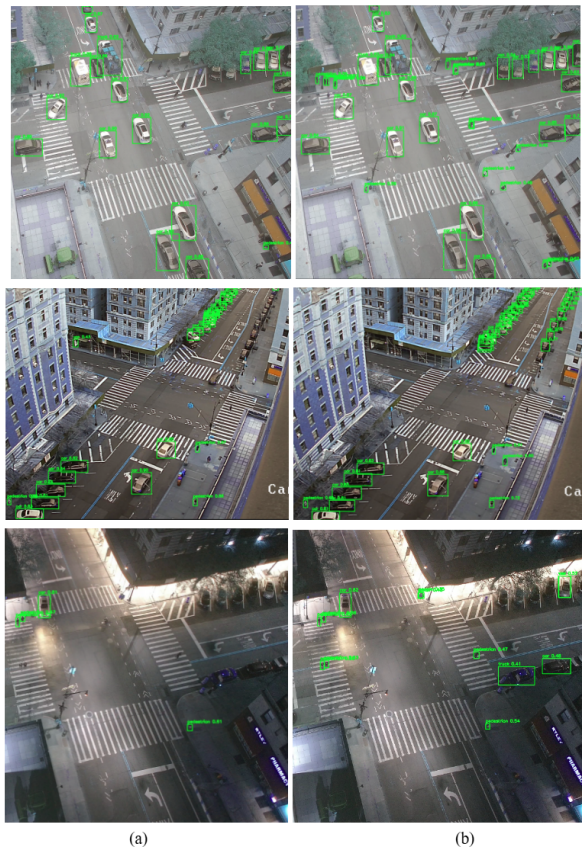

Figure 14. Inference results for (**a**) YOLOv8s and (**b**) SOD-YOLOv8s across various scenarios, including scenes with distant and high-density objects, as well as nighttime scenarios, using the traffic scene dataset.

6. Conclusions

Detecting small-scale objects in UAV images presents significant challenges, which can reduce the overall effectiveness. To address these issues, we introduced SOD-YOLOv8, a specialized object detection model designed for UAV aerial photography and traffic scenes dominated by small objects. Built upon YOLOv8, this model integrates enhanced multi-path fusion inspired by the GFPN architecture of DAMO-YOLO models, facilitating effective feature fusion across layers and simplifying the architecture through reparameterization. By leveraging a high-resolution fourth layer and incorporating a C2f-EMA structure, SOD-YOLOv8 prioritizes small objects, enhances feature fusion, and improves precise localization. In addition, PIoU is used as a replacement for CIoU, the IoU-based loss function in YOLOv8.

The SOD-YOLOv8 model outperformed widely used models such as CenterNet, Cascade R-CNN, SSD, and faster R-CNN across various evaluation metrics. Our efficient model significantly enhanced the object detection performance, without substantially increasing the computational cost or detection time compared to YOLOv8s. It improved the recall from 40.1% to 43.9%, precision from 51.2% to 53.9%, $mAP_{0.5}$ from 40.6% to 45.1%, and $mAP_{0.5:0.95}$ from 24% to 26.6%. In real-world traffic scenarios captured by building-mounted cameras, SOD-YOLOv8 achieved higher IoU values and identified more small objects than YOLOv8s, even under challenging conditions like poor lighting and crowded backgrounds. These capabilities make it ideal for applications such as UAV-based traffic monitoring.

However, challenges remain in deploying small object detection methods in resource-constrained environments. While attention mechanisms and complex feature fusion improve performance in controlled settings, they may struggle with generalization across diverse environments and conditions, complicating real-world deployment and mainte-

nance. In this study, given the promising results of the used PIoU method on the VisDrone dataset and real-world traffic scenes, which involved numerous small objects, future research will prioritize evaluating PIoU across various datasets. We plan to assess the model's performance in adverse weather conditions to enhance its adaptability and robustness across diverse scenarios, while also creating a comprehensive annotated dataset and conducting additional experiments to further validate its performance. Additionally, efforts will focus on refining the GFPN architecture and exploring alternative processing methods.

Author Contributions: Conceptualization, B.K. and A.W.S.; Methodology, B.K. and A.W.S.; Software, B.K.; Validation, B.K. and A.W.S.; Investigation, A.W.S.; Resources, B.K. and A.W.S.; Writing—original draft, B.K.; Writing—review and editing, A.W.S.; Visualization, B.K.; Supervision, A.W.S.; Project administration, A.W.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Center for Smart Streetscapes, an NSF Engineering Research Center, under grant agreement EEC-2133516.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Acknowledgments: The authors are grateful to Eric Valasek and Nicholas D'Andre from Gridmatrix for motivating the small object detection problem through a number of discussions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-view 3D oBject Detection Network for Autonomous Driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6526–6534.
2. Alqarqaz, M.; Younes, M.B.; Qaddoura, R. An Object Classification Approach for Autonomous Vehicles Using Machine Learning Techniques. *World Electr. Veh. J.* **2023**, *14*, 41. [CrossRef]
3. Lim, Y.; Tiang, S.S.; Lim, W.H.; Wong, C.H.; Mastaneh, M.; Chong, K.S.; Sun, B. Object Detection in Autonomous Vehicles: A Performance Analysis. In Proceedings of the International Conference on Mechatronics and Intelligent Robotics, Singapore, 22–23 August 2023; Springer Nature: Singapore, 2023; pp. 277–291.
4. Feng, J.; Wang, J.; Qin, R. Lightweight detection network for arbitrary-oriented vehicles in UAV imagery via precise positional information encoding and bidirectional feature fusion. *Int. J. Remote Sens.* **2023**, *44*, 4529–4558. [CrossRef]
5. Chuai, Q.; He, X.; Li, Y. Improved Traffic Small Object Detection via Cross-Layer Feature Fusion and Channel Attention. *Electronics* **2023**, *12*, 3421. [CrossRef]
6. Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved Techniques for Training GANs. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Barcelona, Spain, 5–11 December 2016.
7. Alsamhi, S.H.; Shvetsov, A.V.; Kumar, S.; Shvetsova, S.V.; Alhartomi, M.A.; Hawbani, A.; Rajput, N.S.; Srivastava, S.; Saif, A.; Nyangaresi, V.O. UAV computing-assisted search and rescue mission framework for disaster and harsh environment mitigation. *Drones* **2022**, *6*, 154. [CrossRef]
8. Wang, G.; Chen, Y.; An, P.; Hong, H.; Hu, J.; Huang, T. UAV-YOLOv8: A small-object-detection model based on improved YOLOv8 for UAV aerial photography scenarios. *Sensors* **2023**, *23*, 7190. [CrossRef] [PubMed]
9. Jiang, Y.; Tan, Z.; Wang, J.; Sun, X.; Lin, M.; Li, H. Giraffedet: A heavy-neck paradigm for object detection. *arXiv* **2022**, arXiv:2202.04256.
10. Ouyang, D.; He, S.; Zhang, G.; Luo, M.; Guo, H.; Zhan, J.; Huang, Z. Efficient multi-scale attention module with cross-spatial learning. In Proceedings of the ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes, Greece, 4–10 June 2023; pp. 1–5.
11. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
12. Girshick, R. Fast R-CNN. *arXiv* **2015**, arXiv:1504.08083.
13. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems Montreal, QC, Canada, 7–12 December 2015; pp. 21–37.
14. Deng, Z.; Sun, H.; Zhou, S.; Zhao, J.; Lei, L.; Zou, H. Multi-scale object detection in remote sensing imagery with convolutional neural networks. *ISPRS J. Photogramm. Remote Sens.* **2018**, *145*, 3–22. [CrossRef]

15. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; IEEE Computer Society: Washington DC, USA, 2016; pp. 779–788.
16. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
17. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 21–37.
18. Liu, M.; Wang, X.; Zhou, A.; Fu, X.; Ma, Y.; Piao, C. UAV-YOLO: Small object detection on unmanned aerial vehicle perspective. *Sensors* **2020**, *20*, 2238. [CrossRef]
19. Liu, W.; Qiang, J.; Li, X.; Guan, P.; Du, Y. UAV image small object detection based on composite backbone network. *Mob. Inf. Syst.* **2022**, *2022*, 7319529. [CrossRef]
20. Yang, B.; Wang, X.; Xing, Y.; Cheng, C.; Jiang, W.; Feng, Q. Modality Fusion Vision Transformer for Hyperspectral and LiDAR Data Collaborative Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 2024, *early access*. [CrossRef]
21. Tummala, S.; Kadry, S.; Bukhari, S.A.C.; Rauf, H.T. Classification of brain tumor from magnetic resonance imaging using vision transformers ensembling. *Curr. Oncol.* **2022**, *29*, 7498–7511. [CrossRef] [PubMed]
22. Liu, Y.; Yang, F.; Hu, P. Small-object detection in UAV-captured images via multi-branch parallel feature pyramid networks. *IEEE Access* **2020**, *8*, 145740–145750. [CrossRef]
23. Zhu, P.; Wen, L.; Du, D.; Bian, X.; Fan, H.; Hu, Q.; Ling, H. Detection and Tracking Meet Drones Challenge. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 7380–7399. [CrossRef] [PubMed]
24. Lai, H.; Chen, L.; Liu, W.; Yan, Z.; Ye, S. STC-YOLO: Small object detection network for traffic signs in complex environments. *Sensors* **2023**, *23*, 5307. [CrossRef] [PubMed]
25. Shen, L.; Lang, B.; Song, Z. DS-YOLOv8-based object detection method for remote sensing images. *IEEE Access* **2023**, *11*, 125122–125137. [CrossRef]
26. Tong, Z.; Chen, Y.; Xu, Z.; Yu, R. Wise-IoU: Bounding box regression loss with dynamic focusing mechanism. *arXiv* **2023**, arXiv:2301.10051.
27. Wang, H.; Yang, H.; Chen, H.; Wang, J.; Zhou, X.; Xu, Y. A remote sensing image target detection algorithm based on improved YOLOv8. *Appl. Sci.* **2024**, *14*, 1557. [CrossRef]
28. Xia, G.S.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A large-scale dataset for object detection in aerial images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18-23 June 2018; pp. 3974–3983.
29. Xu, W.; Cui, C.; Ji, Y.; Li, X.; Li, S. YOLOv8-MPEB small target detection algorithm based on UAV images. *Heliyon* **2024**, *10*, e29501. [CrossRef]
30. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
31. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
32. Wang, C.Y.; Liao, H.Y.M.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 13–19 June 2020; pp. 390–391.
33. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef] [PubMed]
34. Lin, T.-Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
35. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
36. Feng, C.; Zhong, Y.; Gao, Y.; Scott, M.R.; Huang, W. TOOD: Task-Aligned One-Stage Object Detection. In Proceedings of the 2021 IEEE International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 3490–3499.
37. Li, X.; Wang, W.; Wu, L.; Chen, S.; Hu, X.; Li, J.; Tang, J.; Yang, J. Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection. *arXiv* **2020**, arXiv:2006.04388.
38. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-iou loss: Faster and better learning for bounding box regression. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12993–13000.
39. Yu, J.; Jiang, Y.; Wang, Z.; Cao, Z.; Huang, T. Unitbox: An advanced object detection network. In Proceedings of the 24th ACM International Conference on Multimedia, Melbourne, Australia, 28 October–1 November 2016; pp. 516–520.
40. Kang, M.; Ting, C.M.; Ting, F.F.; Phan, R.C.W. Bgf-yolo: Enhanced yolov8 with multiscale attentional feature fusion for brain tumor detection. *arXiv* **2023**, arXiv:2309.12585.
41. Xu, X.; Jiang, Y.; Chen, W.; Huang, Y.; Zhang, Y.; Sun, X. Damo-yolo: A report on real-time object detection design. *arXiv* **2022**, arXiv:2211.15444.

42. Liu, C.; Wang, K.; Li, Q.; Zhao, F.; Zhao, K.; Ma, H. Powerful-IoU: More straightforward and faster bounding box regression loss with a nonmonotonic focusing mechanism. *Neural Netw.* **2024**, *170*, 276–284. [CrossRef]
43. Loshchilov, I.; Hutter, F. SGDR: Stochastic Gradient Descent with Warm Restarts. In Proceedings of the 5th International Conference on Learning Representations (ICLR), Toulon, France, 24–26 April 2017.
44. Siliang, M.; Yong, X. Mpdiou: A loss for efficient and accurate bounding box regression. *arXiv* **2023**, arXiv:2307.07662.
45. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
46. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.
47. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
48. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into High Quality Object Detection. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6154–6162.
49. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Keypoint triplets for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27–28 October 2019; pp. 6569–6578.
50. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. 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, Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.
51. Reis, D.; Kupec, J.; Hong, J.; Daoudi, A. Real-time flying object detection with YOLOv8. *arXiv* **2023**, arXiv:2305.09972.
52. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
53. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. MobileNetV2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
54. Raychaudhuri, D.; Seskar, I.; Zussman, G.; Korakis, T.; Kilper, D.; Chen, T.; Kolodziejski, J.; Sherman, M.; Kostic, Z.; Gu, X.; et al. Challenge: COSMOS: A city-scale programmable testbed for experimentation with advanced wireless. In Proceedings of the ACM MobiCom'20, London, UK, 21–25 September 2020.
55. Kostic, Z.; Angus, A.; Yang, Z.; Duan, Z.; Seskar, I.; Zussman, G.; Raychaudhuri, D. Smart city intersections: Intelligence nodes for future metropolises. *IEEE Comp.* **2022**, *55*, 74–85. [CrossRef]
56. COSMOS Project. Hardware: Cameras. 2022. Available online: https://wiki.cosmos-lab.org/wiki/Hardware/Cameras (accessed on 1 May 2024).