Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.061363

ARTICLE

DAFPN-YOLO: An Improved UAV-Based Object Detection Algorithm Based on YOLOv8s

Honglin Wang[1], Yaolong Zhang[2,*] and Cheng Zhu[3]

[1]School of Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing, 210044, China
[2]School of Computer Science, Nanjing University of Information Science and Technology, Nanjing, 210044, China
[3]Electrical & Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

*Corresponding Author: Yaolong Zhang. Email: 202312490626@nuist.edu.cn

Received: 22 November 2024; Accepted: 31 January 2025; Published: 16 April 2025

ABSTRACT: UAV-based object detection is rapidly expanding in both civilian and military applications, including security surveillance, disaster assessment, and border patrol. However, challenges such as small objects, occlusions, complex backgrounds, and variable lighting persist due to the unique perspective of UAV imagery. To address these issues, this paper introduces DAFPN-YOLO, an innovative model based on YOLOv8s (You Only Look Once version 8s). The model strikes a balance between detection accuracy and speed while reducing parameters, making it well-suited for multi-object detection tasks from drone perspectives. A key feature of DAFPN-YOLO is the enhanced Drone-AFPN (Adaptive Feature Pyramid Network), which adaptively fuses multi-scale features to optimize feature extraction and enhance spatial and small-object information. To leverage Drone-AFPN's multi-scale capabilities fully, a dedicated 160 × 160 small-object detection head was added, significantly boosting detection accuracy for small targets. In the backbone, the C2f_Dual (Cross Stage Partial with Cross-Stage Feature Fusion Dual) module and SPPELAN (Spatial Pyramid Pooling with Enhanced Local Attention Network) module were integrated. These components improve feature extraction and information aggregation while reducing parameters and computational complexity, enhancing inference efficiency. Additionally, Shape-IoU (Shape Intersection over Union) is used as the loss function for bounding box regression, enabling more precise shape-based object matching. Experimental results on the VisDrone 2019 dataset demonstrate the effectiveness of DAFPN-YOLO. Compared to YOLOv8s, the proposed model achieves a 5.4 percentage point increase in mAP@0.5, a 3.8 percentage point improvement in mAP@0.5:0.95, and a 17.2% reduction in parameter count. These results highlight DAFPN-YOLO's advantages in UAV-based object detection, offering valuable insights for applying deep learning to UAV-specific multi-object detection tasks.

KEYWORDS: YOLOv8; UAV-based object detection; AFPN; small-object detection head; SPPELAN; DualConv; loss function

1 Introduction

In recent years, UAV-based object detection has shown immense application potential and significant advantages [1,2]. UAVs can efficiently cover large areas and capture real-time aerial images, enabling their use in various fields such as agriculture [3], urban planning [4], forest conservation [5], and transportation [2]. Compared to traditional satellite or stationary camera systems, UAVs excel in efficiency, flexibility, and responsiveness, especially in dynamic and complex environments. These features have made the combination of UAV imagery and object detection a critical research area. However, UAV-captured images face challenges [6], including environmental factors like weather changes and wind, which degrade image quality.

Additionally, the prevalence of small objects, occlusions, and complex backgrounds increases the difficulty of detection and recognition.

Object detection methods can be categorized into traditional approaches based on handcrafted features and conventional machine learning algorithms, as well as deep learning-based methods. Traditional techniques, like Haar Cascade [7], struggle with complex scenarios, exhibiting low detection accuracy and speed, making them unsuitable for real-time applications. In contrast, deep learning methods are now the preferred choice due to their robustness and higher accuracy. These methods can be further divided into two-stage and one-stage approaches. Two-stage methods, which first generate region proposals followed by classification and regression, achieve high accuracy but are computationally expensive and slow, rendering them impractical for real-time UAV detection tasks. Among one-stage methods, the YOLO (You Only Look Once) series [8,9] is particularly suitable for UAV detection due to its efficiency, lightweight design, ability to handle multi-scale features, and adaptability to complex environments. The YOLO series has advanced to its latest version, YOLOv11 [10], yet challenges remain in UAV image object detection. These include poor performance in detecting small and occluded objects due to insufficient integration of low-level features, which limits detection capability. Additionally, balancing detection accuracy and computational efficiency is particularly challenging for resource-constrained UAV platforms. Traditional loss functions like CIoU struggle to accurately capture irregular shapes and diverse aspect ratios, leading to reduced localization accuracy of bounding boxes. These issues often result in missed detections or false positives in aerial images. To address these problems, this study introduces DAFPN-YOLO, an enhanced object detection algorithm based on YOLOv8s [11], specifically designed for UAV imagery. This model incorporates multiple innovations to overcome the outlined challenges, aiming to achieve an optimal balance between detection accuracy and speed. It demonstrates outstanding performance on the UAV dataset VisDrone2019 [12]. The primary contributions of this paper are as follows:

1. We propose an enhanced feature fusion network named Drone-AFPN, based on AFPN [13], designed to efficiently integrate multi-scale features and strengthen shallow feature utilization. This network employs a multi-level adaptive spatial feature fusion (ASFF) module [13], which adaptively prioritizes feature maps of varying sizes, enabling effective multi-scale feature blending. This approach significantly improves detection performance, particularly for small objects.
2. Regarding the improvement of the neck network structure, in order to further enhance the detection effect of small targets, we introduce a 160 × 160 pixel small-object detection head. This addition allows the model to focus on finer-grained features, facilitating the detection of smaller and more detailed objects.
3. We design a novel C2f_Dual module by integrating the DualConv module [14] with the C2f module [9], replacing the original C2f module in the backbone network. Additionally, the SPPELAN module [15] replaces the SPPF module [9], ensuring robust feature extraction and information aggregation while reducing model parameters and improving inference speed.
4. We replace the original CIoU loss [16] with Shape-IoU [17] for bounding box regression, enabling the model to more accurately reflect the true shapes of objects. This replacement enhances regression precision, particularly for small targets with detailed and irregular shapes, reducing false negatives and positives and improving the model's robustness in complex scenarios.

2 Related Works

2.1 Object Detection

Recent developments in object detection have seen notable progress. Traditional techniques mainly relied on methods like sliding windows [18] and hand-crafted feature extractors, such as Viola-Jones [18],

HOG (Histograms of Oriented Gradients) [19], and SIFT (Scale-Invariant Feature Transform) [20]. However, with the rapid advancement of deep learning, CNN-based detection algorithms have become predominant, particularly in two-stage and one-stage models. Two-stage detectors, which are region-based methods, include the R-CNN family [20], starting with R-CNN (Region-CNN) [21], and followed by Fast R-CNN [22] and Faster R-CNN [23]. These methods first generate region proposals, followed by classification and regression through deep networks. Enhancements like FPN (Feature Pyramid Networks) [24] and Mask R-CNN [25] improved detection performance by refining their structures. However, these models often face challenges such as high computational complexity, large parameter sizes, and slower processing times, which limit their use in some scenarios. On the other hand, one-stage detectors, including the YOLO series and SSD (Single Shot MultiBox Detector) [26], treat object detection as a single regression task, performing both classification and bounding box regression within one network, leading to significantly faster inference times. Recent models like EfficientDet [27], YOLOv3, YOLOv5, YOLOv8, YOLOv10, and YOLOv11 [8–11] have emerged, offering a good balance between accuracy and speed. These algorithms incorporate innovations such as improved feature fusion, efficient backbones, and attention mechanisms, further enhancing detection performance.

2.2 UAV Object Detection

UAV images present unique challenges for object detection due to characteristics such as extensive backgrounds, low target pixel counts, and frequent interference. These factors exacerbate issues like small object prevalence, occlusion, and information loss, making UAV-based detection more demanding than traditional methods. Researchers have introduced various improvements to address these challenges. Pang et al. [28] proposed a Faster R-CNN-based multi-scale fusion detection method, enhancing UAV detection accuracy but increasing model complexity and slowing down performance. Fang et al. [29] integrated residual blocks into U-Net and employed multi-scale feature fusion for image reconstruction, treating small UAV detection as residual image prediction to boost accuracy, although it relies heavily on precise residual estimation.

UAV detection tasks often face limitations in computational and storage resources, making high-demand methods difficult to deploy. Additionally, real-time detection requirements impose strict constraints, reducing the feasibility of certain approaches. The YOLO series detection networks have alleviated some of these challenges, but their detection accuracy remains relatively low. To address this, researchers have proposed improvements to YOLO-based models. For instance, Deng et al. [30] developed LAI-YOLOv5s, incorporating DFMCPFN for feature fusion and VoVNet for better feature extraction, achieving higher accuracy. Tang et al. [31] introduced HIC-YOLOv5, adding a small object detection head, channel enhancement convolutional blocks, and CBAM to emphasize critical features. Nie et al. [32] extended YOLOv8n with a small object detection layer, an SSFF module, and HPANet, improving accuracy while reducing parameters. Li et al. [33] proposed SOD-YOLO, integrating the RFCBAM module into the backbone to enhance feature extraction and designing the bsi-FPN neck structure to balance spatial and semantic information effectively. Wang et al. [34] developed a YOLOv8-based UAV strawberry detection model, integrating an attention module and the VoV-GSCSP module, which improved both detection speed and accuracy. Li et al. [35] enhanced YOLOv8s by introducing the Bi-PAN-FPN neck structure, replacing some C2f modules with GhostblockV2, and substituting WiseIoU for CIoU. These modifications reduced information loss during feature transmission and increased model robustness.

Although these improvements have shown promising results, they are often accompanied by an increase in parameter counts or computational burden. Existing methods still have limitations, such as insufficient integration of low-level features, leading to poor detection performance for small and occluded objects,

and difficulty balancing detection speed and accuracy. Methods that prioritize accuracy often significantly reduce speed. Additionally, traditional bounding box regression methods (e.g., CIoU) struggle to handle the irregular shapes and diverse aspect ratios of objects in UAV detection. To address these gaps, this study proposes DAFPN-YOLO, an improved UAV object detection model based on YOLOv8s, which achieves high detection accuracy and competitive speed on UAV imagery. Meanwhile, Bakirci's [36] research highlights that YOLOv8-Nano, by reducing parameters and computational complexity, delivers high inference speed suitable for edge devices in intelligent transportation systems. However, its lightweight design sacrifices detection accuracy in complex environments, making it ineffective for detecting occluded objects and small targets in UAV imagery with background variations. This study introduces modules like Drone-AFPN and Shape-IoU to prioritize detection accuracy, addressing the unique challenges of UAV detection. These innovations enhance multi-scale feature fusion and improve the localization of irregular objects, delivering exceptional performance in complex UAV scenarios. A comparison reveals that this study achieves a better balance between speed and accuracy, maximizing detection accuracy with minimal sacrifice to detection speed.

3 Methods

To improve object detection accuracy in UAV images, we introduce DAFPN-YOLO, an enhanced detection algorithm based on YOLOv8s. This section outlines the architectures of both YOLOv8 and DAFPN-YOLO, emphasizing the specific advancements made in DAFPN-YOLO over the original model.

3.1 Network Framework of YOLOv8

YOLOv8 is a highly advanced and widely used version in the YOLO series, consisting of three key components: the backbone network, the neck network, and the detection head. These components work collaboratively to perform feature extraction, multi-scale feature fusion, and produce the final output of bounding boxes and class predictions, ensuring efficient and real-time object detection. The backbone network integrates an upgraded CSPNet [37], which enhances feature extraction by leveraging its deep learning capabilities. The neck network incorporates a PAN-FPN [38] structure, improving the fusion of features at different scales and ensuring robust performance, especially for small-object detection. Additionally, the anchor-free detection head simplifies the design by removing the need for anchor boxes, which in turn accelerates detection speed and improves overall efficiency. YOLOv8 strikes an optimal balance between detection accuracy and real-time capability, making it suitable for a wide range of object detection applications across various industries. The architecture of YOLOv8 is shown in Fig. 1.

3.2 Overview of DAFPN-YOLO

Detecting objects in UAV imagery poses significant challenges due to unique image characteristics, including complex backgrounds, a high ratio of small-to-medium targets, and frequent occlusions. These factors complicate accurate and efficient detection, particularly for small objects in dense scenes. To tackle these challenges, we propose an advanced algorithm, DAFPN-YOLO, built upon YOLOv8 and tailored for UAV object detection, as illustrated in Fig. 2. This model achieves a balance between accuracy and speed through four key enhancements. First, the backbone's C2f module is replaced with the C2f_Dual module, improving feature extraction and overall efficiency. Second, the SPPF module is substituted with the SPPELAN module, balancing performance and computational cost to optimize detection. Third, enhancements to the neck and head include integrating the Drone-AFPN structure for improved multi-scale feature fusion and adding a small-object detection head to better capture fine details. Lastly, the Shape-IoU loss function is introduced

for bounding box regression, enhancing localization accuracy and accelerating convergence by accounting for shape and positional variations.

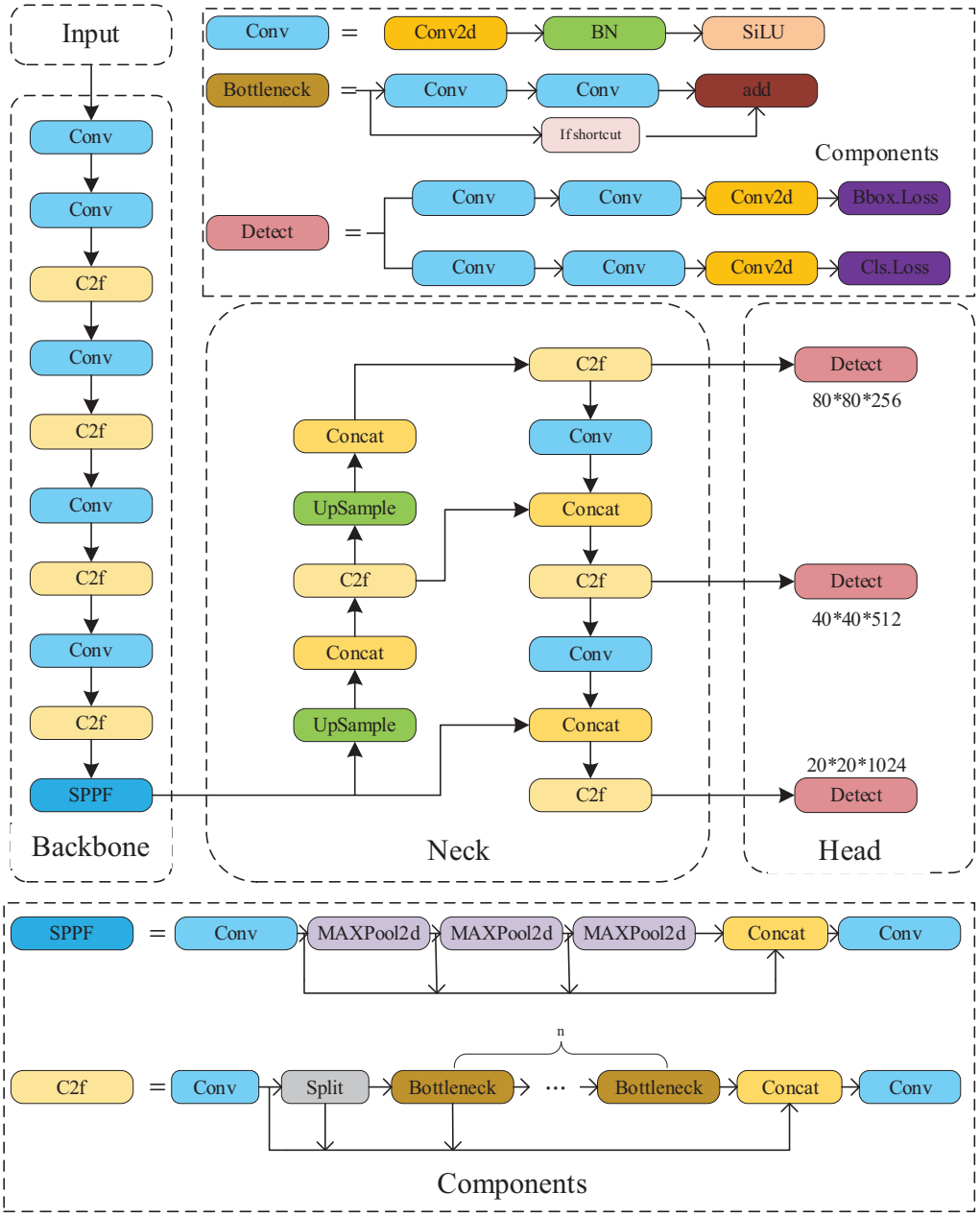

Figure 1: Overall architecture of the YOLOv8

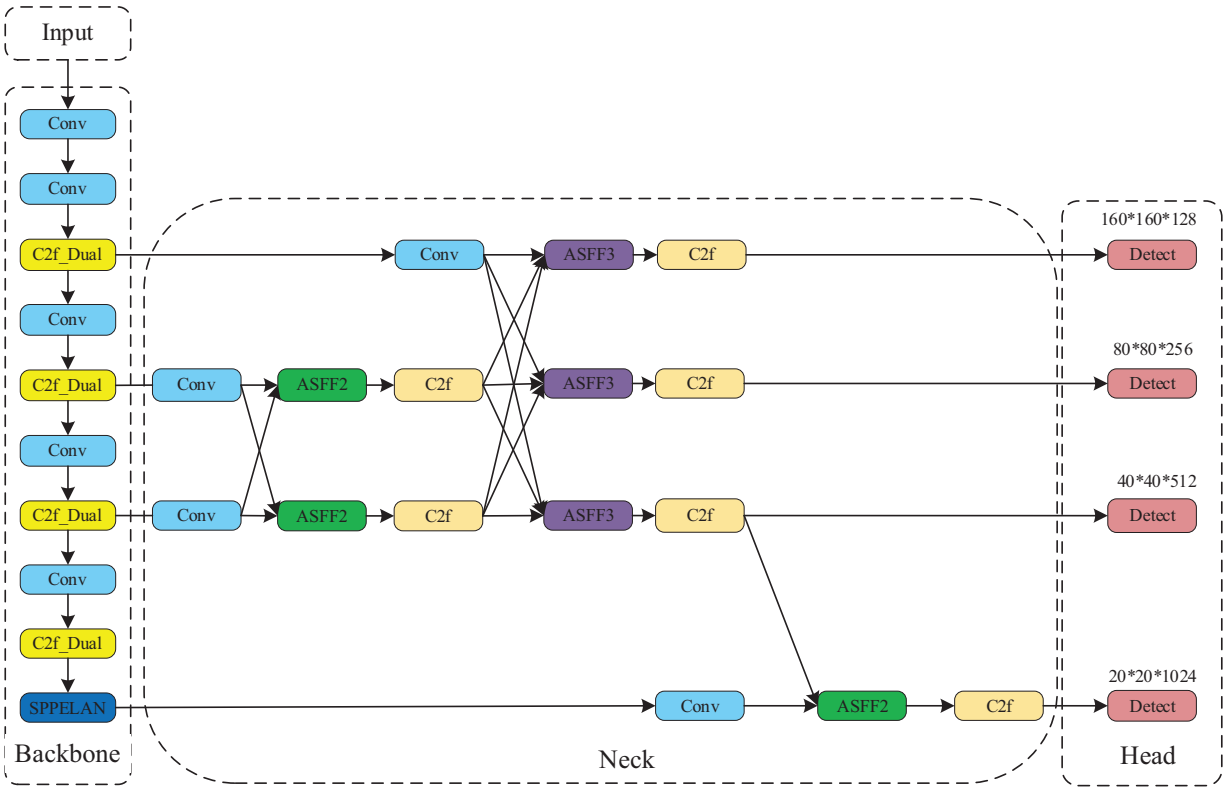

Figure 2: Overall architecture of the DAFPN-YOLO

3.3 DualConv

To improve feature extraction efficiency and overall model performance, we propose the C2f_Dual module, designed as a replacement for the C2f module in the backbone network. This module combines the strengths of DualConv and the traditional C2f module, enabling efficient feature representation and multi-scale information fusion.

DualConv is a novel network component that enhances feature extraction capabilities and computational efficiency. It processes input feature maps using 3 × 3 group convolution and 1 × 1 pointwise convolution simultaneously, optimizing information processing. The 3 × 3 group convolution extracts spatial features, while the 1 × 1 pointwise convolution aggregates these features for effective fusion. Grouped convolution further divides input and output feature maps into separate groups, each processed independently with dedicated kernels, significantly reducing parameters and computational complexity. This design ensures efficient resource use while controlling model complexity by limiting the scope of each group's filters. The structural layout of DualConv is illustrated in Fig. 3, where K denotes the kernel size, M represents the number of input channels, N indicates the depth of the output feature map, and G is the number of convolution groups.

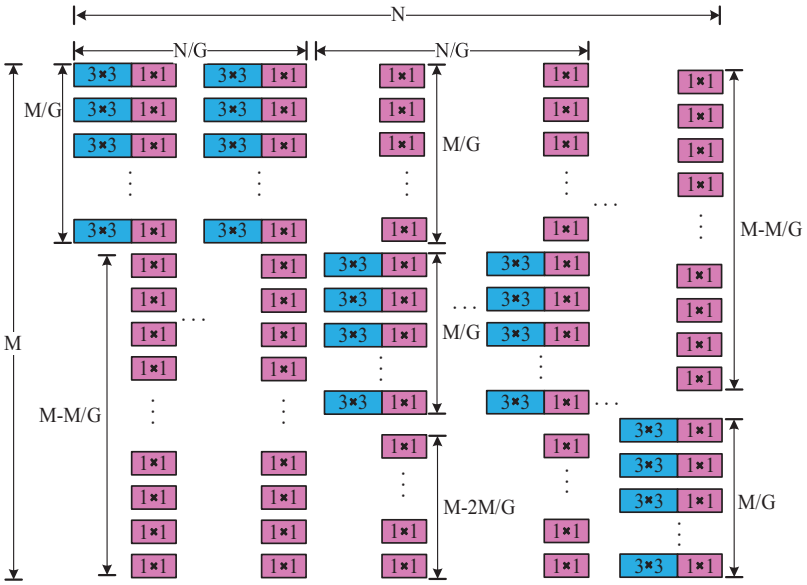

Figure 3: Structural layout of dual convolution

By innovatively combining C2f with DualConv, replacing the residual convolution in the C2f module with DualConv, this design demonstrates significant advantages over the traditional C2f module. First, DualConv employs smaller convolution kernels and depthwise separable convolutions. Traditional convolutions typically use larger kernels (e.g., 5 × 5), while DualConv adopts smaller kernels (e.g., 3 × 3 and 1 × 1), directly reducing the parameters for each layer. Additionally, depthwise separable convolutions decompose traditional convolutions into two smaller operations, significantly lowering computational cost and parameter count. DualConv is further optimized in terms of channel efficiency, and combined with residual connections, it minimizes redundant convolution operations, leading to even greater reductions in parameter count. Moreover, the use of depthwise separable convolutions and smaller kernels allows for more precise extraction of local features, enabling the model to capture richer and finer details across multiple layers and scales, especially in scenarios with intricate image details. As a result, the C2f_Dual module provides more comprehensive and valuable feature representations. This improvement enhances the model's ability to generate high-quality feature representations while reducing computational complexity and parameter overhead, laying the foundation for efficient model applications. The structure of the C2f_Dual module is shown in Fig. 4.

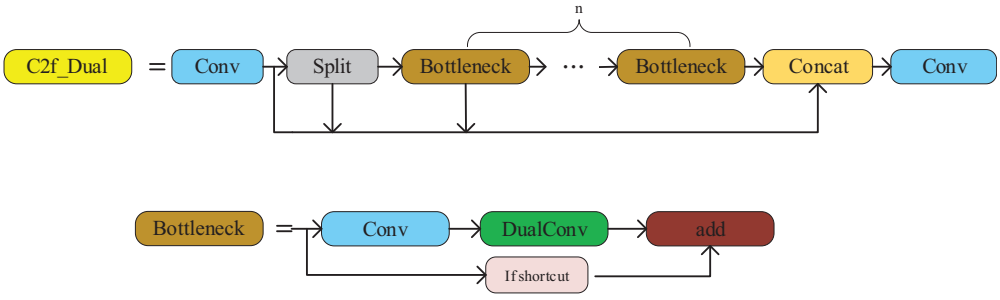

Figure 4: Structure of the C2f_Dual module

3.4 Space Pyramid Pool Improvement

To enhance the model's ability to capture crucial information in complex scenarios while improving inference speed, this study replaces the SPPF module in the backbone network with SPPELAN (Spatial Pyramid Pooling Enhanced with ELAN). SPPELAN, derived from Spatial Pyramid Pooling (SPP) [39], optimizes max-pooling operations by integrating SPP with the Efficient Layer Aggregation Network (ELAN) [40] through a sequential structure of a CBS module and three MaxPool2d modules. This design enables faster and more effective feature extraction. Unlike SPPF, SPPELAN addresses redundant feature extraction in some images, significantly improving recognition accuracy without increasing parameters. Its local attention mechanism dynamically adjusts feature map weights, emphasizing target areas and suppressing irrelevant background details. This refinement enhances feature extraction for small or occluded objects, reducing false positives and missed detections, thereby improving detection performance and robustness in complex scenarios. By integrating the strengths of SPP and ELAN, SPPELAN achieves an optimal balance between efficiency and performance. Fig. 5 illustrates its structure.

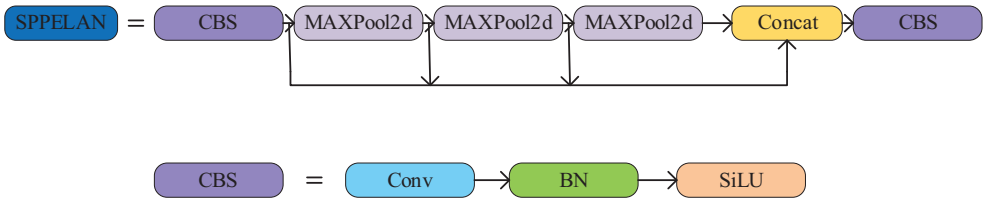

Figure 5: Structure of the SPPELAN module

A study by Jin et al. [41] also optimized the SPPCSPC module with pooling layers by integrating the Bi-level Routing Attention (BRA) mechanism into the original SPPCSPC pyramid pooling network. BRA dynamically weights critical regions and important channels of feature maps through a dual-level routing mechanism, enhancing the interaction of multi-scale features. In comparison, the BRA-integrated SPPCSPC is more suitable for high-precision tasks but comes with significant computational overhead. On the other hand, SPPELAN, designed with a lightweight architecture, performs exceptionally well in embedded environments, balancing performance and efficiency. Despite its limited improvement in accuracy, SPPELAN consumes fewer resources, offers higher real-time performance, and is better suited for UAV detection scenarios.

3.5 Improvement of the Neck and Head

3.5.1 Drone Asymptotic Feature Pyramid Network

For object detection, extracting and fusing multi-scale image features is essential. In YOLOv8, the Path Aggregation Network (PANet) [31] is employed to facilitate multi-scale object detection through effective bottom-up and top-down feature fusion. However, PANet faces challenges in information transmission, often resulting in the loss of low-level feature and positional information during fusion. These limitations hinder classification and localization accuracy in complex scenes. Additionally, PANet's design involves a high parameter count and computational burden.

To address these challenges, the Adaptive Feature Pyramid Network (AFPN) was developed, gaining popularity for its efficiency and lightweight design. Fig. 6 illustrates the AFPN structure in YOLO. Through progressive fusion, AFPN mitigates conflicts in multi-scale information and assigns differentiated spatial weights to features at different levels. This design not only enhances the contributions of critical feature

layers but also reduces semantic gaps between cross-layer feature mappings. However, AFPN in YOLO does not sufficiently integrate low-level features, leading to a lack of positional information. This limitation reduces the model's robustness in complex scenes, as detecting small objects and acquiring positional details heavily rely on the finer details of low-level feature maps. Without adequately fusing low-level features, the model performs poorly in identifying small objects. Since UAV aerial images often contain numerous small and medium-sized objects, the AFPN structure in YOLO generally delivers lower detection accuracy compared to PANet. When integrating AFPN structures from other detection models like Faster R-CNN into YOLOv8, low-level features are sufficiently fused, resulting in improved detection accuracy. However, this approach significantly increases the number of parameters and computational complexity, slowing down inference speed. To balance detection accuracy and inference speed, we propose the Drone-AFPN structure, specifically designed for UAV aerial image detection tasks. Its simple design is shown in Fig. 7. Specifically, Drone-AFPN innovatively builds upon the adaptive feature fusion mechanism of the original AFPN. It retains the advantages of automatically adjusting feature importance for different scenarios, reducing semantic information loss, and preserving spatial details. Additionally, Drone-AFPN streamlines unnecessary multi-scale feature fusion and optimizes the integration of low-level features. By controlling model complexity, it significantly improves small-object detection accuracy and enhances the overall robustness of the model. In the first stage, Drone-AFPN retains the progressive fusion strategy of AFPN in YOLO, initializing feature fusion using shallow features of two different resolutions. In the second stage, the traditional strategy is modified to incorporate the highest-resolution low-level features into the output of the previous stage. This approach not only avoids semantic gaps between cross-layer feature mappings but also alleviates conflicts in multi-object information during spatial feature fusion, resulting in richer positional information. This improvement significantly enhances the detection accuracy of small and medium-sized objects as well as occluded targets. In the final stage, high-level features from the previous stage output are fused with high-level features from the backbone network. This final output feature map thus contains both rich positional and semantic information, achieving high performance in detecting larger objects. Notably, during the second stage of fusion, high-level features are not fully integrated. This decision is based on the fact that UAV aerial images predominantly contain small and medium-sized objects, where fusing high-level features contributes minimally and instead increases model parameters and reduces inference speed.

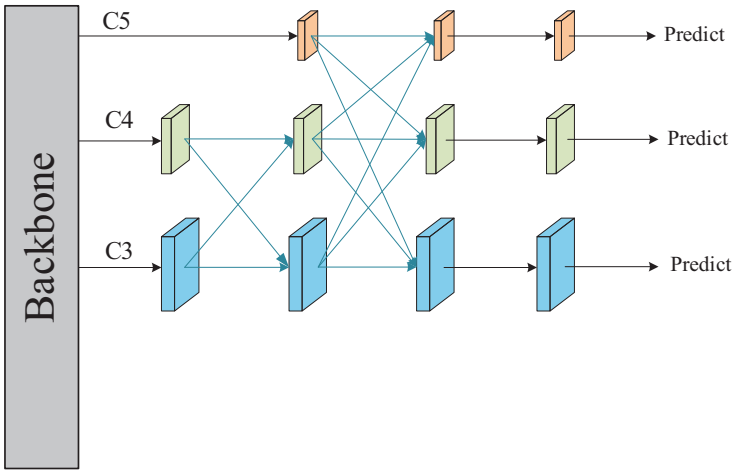

Figure 6: The architecture of the AFPN in YOLO

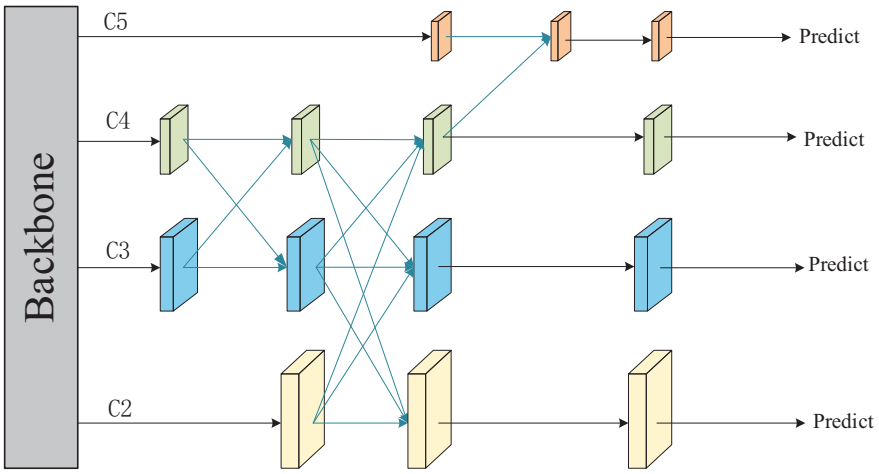

Figure 7: The architecture of the Drone-AFPN

The Adaptive Spatial Feature Fusion (ASFF) Module is crucial in the feature fusion network. Fig. 8 shows an example of adaptive spatial fusion, where features from three different levels are fused. This method can be adapted to incorporate more or fewer feature levels. The main function of ASFF is to perform a weighted linear combination of feature vectors from different levels, creating a fused vector that emphasizes key features while reducing conflicting information.

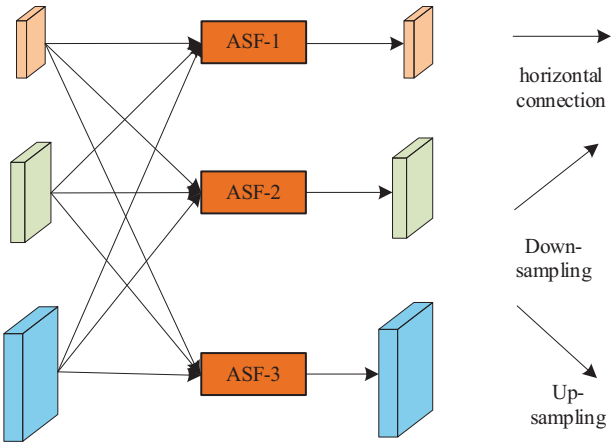

Figure 8: Adaptive spatial fusion operation

Let $x_{ij}^{n\to l}$ represent the feature vector at (i, j) transitioning from level n to level l. The fused feature vector y_{ij}^{l} at level l, obtained through the adaptive spatial fusion of multi-level features, is defined as the linear combination of feature vectors $x_{ij}^{1\to l}$, $x_{ij}^{2\to l}$ and $x_{ij}^{3\to l}$, expressed as:

$$y_{ij}^{l} = \alpha_{ij}^{l} \cdot x_{ij}^{1\to l} + \beta_{ij}^{l} \cdot x_{ij}^{2\to l} + \gamma_{ij}^{l} \cdot x_{ij}^{3\to l} \tag{1}$$

where α_{ij}^{l}, β_{ij}^{l}, and γ_{ij}^{l} represent the spatial weights of the features of the three levels at level l, subject to the constraint that $\alpha_{ij}^{l} + \beta_{ij}^{l} + \gamma_{ij}^{l} = 1$.

3.5.2 Small-Object Detection Head

UAV imagery often contains numerous small objects, making detection particularly challenging. Reflecting these traits, the dataset includes many hard-to-detect small objects. To tackle this, we added a dedicated small-object detection head to the original model. As mentioned earlier, the Drone-AFPN structure was introduced to improve multi-scale feature fusion. Leveraging the positional and small-object information extracted by Drone-AFPN, the small-object detection head minimizes information loss and enhances the model's ability to capture fine details. This addition greatly boosts localization and recognition accuracy, especially for small targets.

3.6 Shape-IoU Loss Function

Traditional IoU methods focus mainly on the overlapping area, overlooking the shape and aspect ratio of object boundaries. This limitation is especially problematic in UAV imagery, where numerous small, densely packed targets make conventional metrics less effective, leading to reduced detection accuracy. To address this, the Shape-IoU metric was introduced. Unlike traditional IoU, Shape-IoU considers both the overlapping area and the shape and aspect ratio of object boundaries. This enables the model to better distinguish small objects with similar shapes but different sizes, significantly improving detection accuracy. Shape-IoU enhances the detection of small objects by capturing detailed boundary features, reducing false positives and missed detections caused by insufficient overlap. It performs well in complex scenarios with overlapping targets and challenging backgrounds, improving robustness and localization precision. Particularly for UAV images with numerous small objects, Shape-IoU demonstrates superior accuracy. The structure of Shape-IoU is shown in Fig. 9.

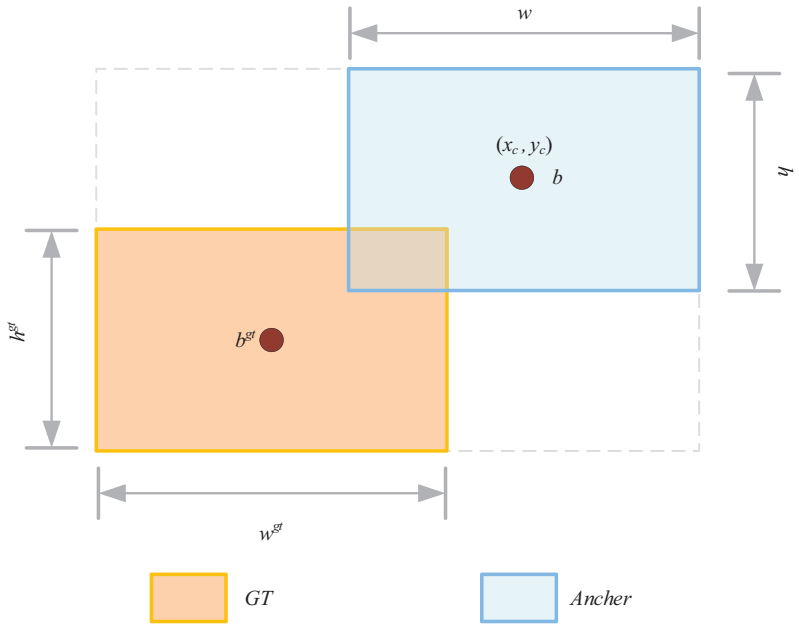

Figure 9: Structure of the Shape-IoU

In comparison, Jin et al. [41] introduced WIoUv3, which uses dynamic weighting to adapt to varying target scales and shapes, excelling in detecting large objects and handling complex backgrounds, albeit with

higher computational cost. While WIoUv3 offers better generalization, Shape-IoU's strength in detecting small objects makes it ideal for drone-based tasks requiring fine detail and robustness.

The calculation formula of Shape-IoU is as follows:

$$IoU = \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} \tag{2}$$

$$ww = \frac{2 \times (w^{gt})^{scale}}{(w^{gt})^{scale} + (h^{gt})^{scale}} \tag{3}$$

$$hh = \frac{2 \times (h^{\mathrm{gt}})^{scale}}{(w^{gt})^{scale} + (h^{gt})^{scale}} \tag{4}$$

$$distance^{shape} = hh \times \left(x_c - x_c^{gt}\right)^2 / c^2 + ww \times \left(y_c - y_c^{gt}\right)^2 / c^2 \tag{5}$$

$$\Omega^{shape} = \sum_{t=w,h} \left(1 - e^{-\omega_t}\right)^{\theta}, \theta = 4 \tag{6}$$

$$\begin{cases} \omega_w = hh \times \dfrac{|w - w^{gt}|}{max\,(w, w^{gt})} \\ \omega_h = ww \times \dfrac{|h - h^{gt}|}{max\,(h, h^{gt})} \end{cases} \tag{7}$$

where *scale* refers to the scale factor, which is linked to the target sizes in the dataset. The parameters *ww* and *hh* represent the weight coefficients for the horizontal and vertical directions, respectively, and are related to the shape of the ground truth (GT) bounding boxes. The bounding box regression loss is then defined as:

$$L_{Shape-IoU} = 1 - IoU + distance^{shape} + 0.5 \times \Omega^{shape} \tag{8}$$

3.7 Overfitting Testing and Mitigation Strategies

To assess generalization, we used cross-validation and training/validation loss comparison to detect overfitting. An increase in validation loss during training signals potential overfitting. To address this, we applied data augmentation (e.g., rotation, flipping, scaling), L2 regularization, and model simplification to reduce parameters. These strategies effectively mitigated overfitting and improved generalization.

4 Experiment

4.1 Datasets

This study utilized the VisDrone 2019 dataset, created by the AISKYEYE team at Tianjin University in collaboration with other research institutions, to advance UAV visual perception tasks such as object detection, tracking, and semantic segmentation. The dataset contains 10,209 annotated UAV images captured from diverse environments across multiple Chinese cities, including urban streets, rural fields, and construction sites. It features multiple perspectives, task-specific data, and 10 object categories, such as pedestrians and vehicles, addressing challenges like detecting small objects in dense, complex backgrounds.

Following the VisDrone 2019 Challenge standard, the dataset is divided into a training set (6471 images), validation set (548 images), and test set (1610 images), ensuring broad coverage of scenarios and conditions. This study selected VisDrone 2019 over VisDrone 2021 due to its maturity and widespread adoption for multi-object and small-object detection tasks. While VisDrone 2021 offers expanded annotations and scenarios, VisDrone 2019 provides stable, diverse imagery that effectively captures complex scenarios, making it an

ideal benchmark for addressing small-object detection and dense multi-object challenges. Its widespread use ensures meaningful comparisons with existing research, meeting the specific needs of this study.

4.2 Experimental Environment

The experiments were conducted on a Windows 10 system using PyCharm as the programming platform. The deep learning framework combined PyTorch 2.3.1, Python 3.9, and CUDA 12.1. The hardware setup included an NVIDIA GeForce RTX 4060 GPU with 8 GB of RAM. Training was performed over 200 epochs with a batch size of 8 and a default image resolution of 640 × 640. The initial learning rate was set to 0.01. Data augmentation utilized the mosaic method [42], which involves randomly cropping four images and combining them into a single composite image for training.

4.3 Evaluation Metrics

This study utilized six evaluation metrics to comprehensively analyze the algorithm's performance: precision (P), recall (R), mAP@0.5, mAP@0.5:0.95, frames per second (FPS), and parameters (Params). Precision (P) quantifies the proportion of correctly identified samples among all predictions, indicating classification accuracy. Recall (R) represents the proportion of correctly detected positive samples among actual positives, reflecting the model's detection capability for all positive categories. Mean Average Precision (mAP), a standard metric in object detection, evaluates accuracy across categories; mAP@0.5 uses an IoU threshold of 0.5, while mAP@0.5:0.95 averages precision over IoU thresholds ranging from 0.5 to 0.95 for a more comprehensive assessment. FPS measures the image processing rate per second, crucial for evaluating real-time detection. Params indicate the number of model parameters, representing its complexity. The formulas for these metrics are as follows:

$$FPS = \frac{1/\left(T_{pre} + T_{in} + T_{post}\right)}{1000} \tag{9}$$

$$P = \frac{TP}{TP + FP} \tag{10}$$

$$R = \frac{TP}{TP + FN} \tag{11}$$

$$AP = \int_0^1 p\left(r\right) dr \tag{12}$$

$$mAP = \frac{1}{k} \sum_{k}^{i} APi \tag{13}$$

where T_{pre}, T_{in} and T_{post} refer to the time taken for preprocessing, inference, and postprocessing, respectively. *TP* (true positives) represents correctly identified positive samples, while *FP* (false positives) indicates negative samples misclassified as positive. *FN* (false negatives) refers to positive samples that were either missed or misclassified as negative. The term p(r) corresponds to the precision-recall curve, *AP* denotes the average precision for the i-th detection category, and k signifies the total number of categories.

4.4 Results and Analysis

4.4.1 Comparison with YOLOv8s

To highlight the detection advantages of our proposed model, we conducted comparative experiments between the DAFPN-YOLO model and the baseline YOLOv8s model on the VisDrone2019 dataset. Fig. 10 presents the PR (Precision-Recall) curves for YOLOv8s and DAFPN-YOLO, showcasing the AP values for individual classes and the overall mAP@0.5 values across all classes. As shown in the figure, our proposed model demonstrates improved detection performance compared to YOLOv8s, achieving a 5.4 percentage point increase in mAP@0.5.

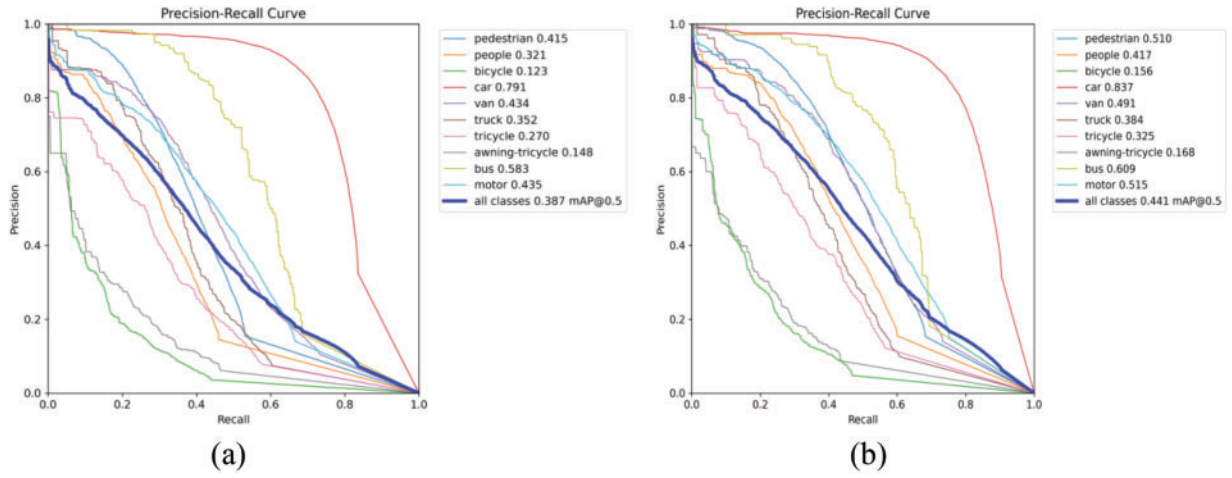

Figure 10: PR curve comparison chart (a) YOLOv8s PR curve; (b) DAFPN-YOLO PR curve

To visually compare the detection performance of the two models for individual target categories, we created bar charts illustrating the detection precision of YOLOv8s and DAFPN-YOLO for various targets, as shown in Fig. 11. In this context, walking or standing individuals are labeled as "pedestrians," while those sitting on vehicles or the ground are categorized as "people.". The chart clearly demonstrates that DAFPN-YOLO significantly improves detection performance across all categories, with the most notable increases observed for "people" and "pedestrians," which show improvements of 9.6 percentage points and 9.5 percentage points, respectively. These results indicate that DAFPN-YOLO enhances detection performance for targets of various scales, particularly for small objects.

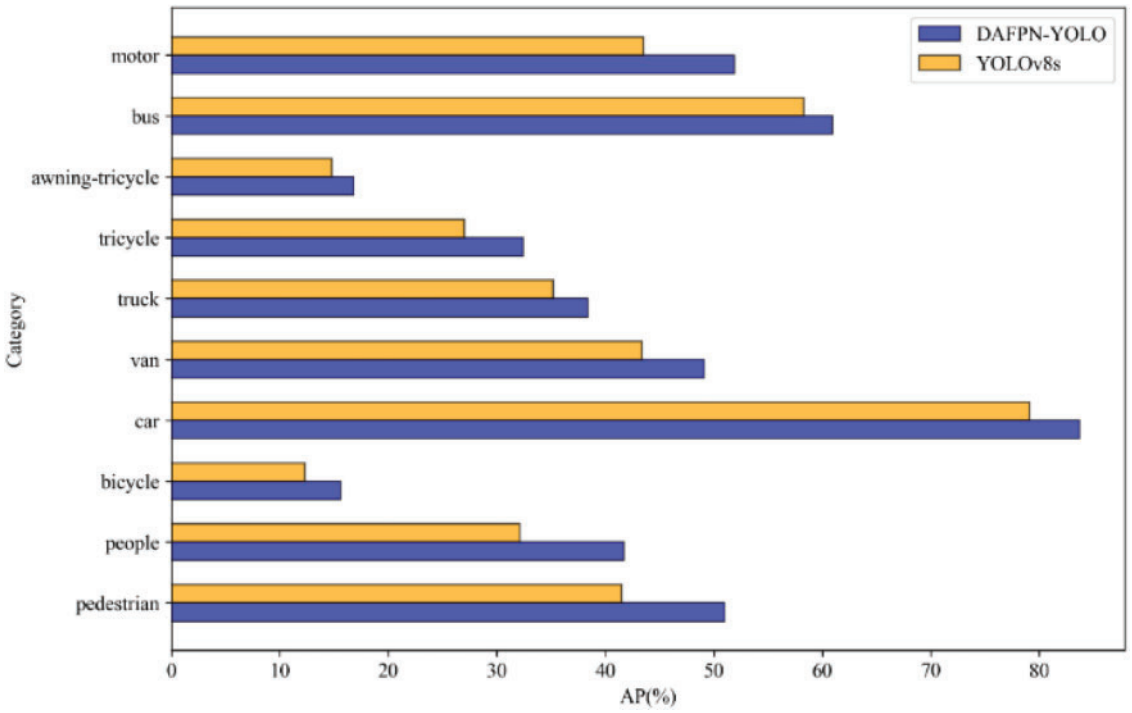

Figure 11: Histogram of different object detection accuracy of YOLOv8s and DAFPN-YOLO

Using visualized images for comparison provides a clearer understanding of the differences in detection performance between the two models across various scenes and environments. This approach is particularly valuable for evaluating the models' ability to detect occluded objects and assess robustness. Fig. 12 presents the visualization results of DAFPN-YOLO and YOLOv8s under different backgrounds and lighting conditions. The comparison reveals that DAFPN-YOLO not only significantly reduces false positives and missed detections but also demonstrates improved detection performance under varying environmental and lighting conditions compared to the baseline model.

4.4.2 Comparison with Current Advanced Algorithms

This study compares one-stage and two-stage detection algorithms. The one-stage algorithms include RetinaNet, YOLOv3-tiny, YOLOv5s, YOLOv6s, YOLOv8n, YOLOv8s, YOLOv8m, YOLOv10s, and the latest YOLOv11s. The two-stage algorithms comprise Faster R-CNN and Cascade R-CNN [43]. Table 1 summarizes the comparison results on the VisDrone2019 dataset. As shown, DAFPN-YOLO exhibits clear advantages over two-stage algorithms in detection accuracy, model size, and speed. Compared to RetinaNet, DAFPN-YOLO outperforms it in detection accuracy, speed, and model size, with an FPS improvement of 46.6%. When compared to the YOLO series, a one-stage algorithm, DAFPN-YOLO also shows strong performance in detection accuracy and model size. It achieves a 5.8% improvement in mAP@0.5 over the latest YOLOv11s algorithm, while reducing the number of parameters by 2.1%. In terms of detection speed, although the detection speed is slightly reduced at the same model size (Small), it outperforms the next model size (Medium). Additionally, by comparing DAFPN-YOLO with YOLOv8n, it can be seen that YOLOv8n is more advantageous for resource-constrained tasks with high real-time requirements.

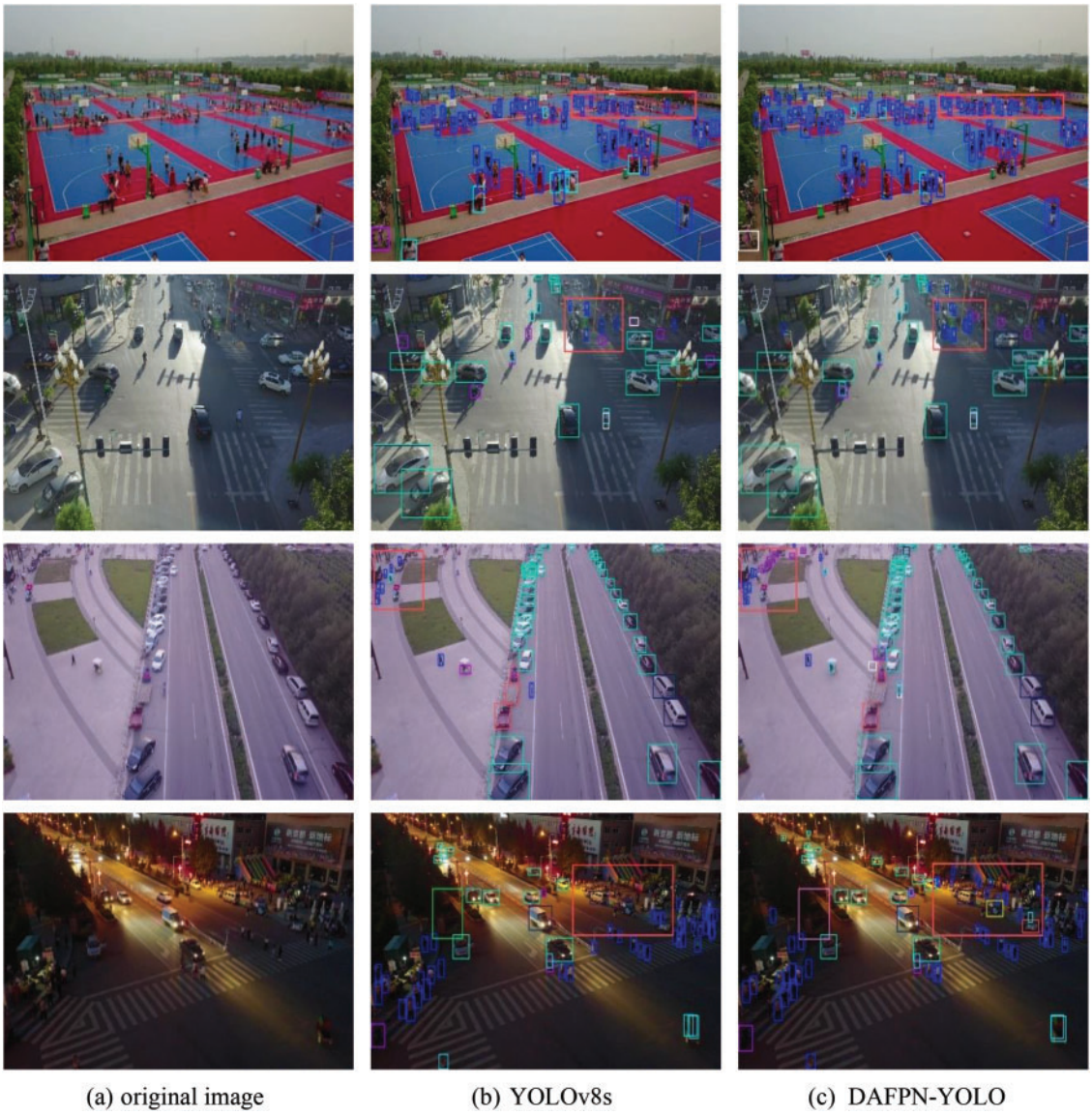

Figure 12: Visualization of the detection results of YOLOv8s and DAFPN-YOLO. The red boxes highlight the areas with the most noticeable differences in detection. (a) Original image to be detected. (b) Detection results of YOLOv8s. (c) Detection results of DAFPN-YOLO

Table 1: Comparison results of different detection algorithms on the VisDrone2019 dataset

Models	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Params (M)	FPS
Faster-RCNN	37.4	22.0	41.2	36.2
Cascade-RCNN	39.0	24.2	69.1	31.6
RetinaNet	28.1	16.1	19.8	62.3
YOLOv3-tiny	23.5	13.1	12.1	**207.0**
YOLOv5s	38.0	22.6	9.1	148.6
YOLOv6s	36.6	22.0	16.3	151.9

(Continued)

Table 1 (continued)

Models	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Params (M)	FPS
YOLOv8n	32.7	19.1	**3.1**	181.2
YOLOv8s	38.7	23.0	11.1	149.1
YOLOv8m	42.5	25.8	25.8	101.1
YOLOv10s	38.5	22.9	8.0	145.1
YOLOv11s	38.3	22.7	9.4	162.3
DAFPN-YOLO (ours)	**44.1**	**26.8**	9.2	117.9

Note: Bold indicates the highest ranking for this metric.

4.4.3 Ablation Experiment

To further verify the effectiveness of the proposed improvements in the DAFPN-YOLO network, ablation experiments were conducted using the YOLOv8s model, with the proposed enhancements progressively integrated. The design of the ablation experiments is as follows:

(1) A: Replace C2f with C2f_Dual in the backbone network.
(2) B: Replace SPPF with SPPELAN in the backbone network.
(3) C: Modify the neck and head by introducing Drone-AFPN and a small-object detection head.
(4) D: Integrate the Shape-IoU loss function.

The experimental results are presented in Table 2. Models M1, M2, and M4 show improvements over M0 across all metrics, indicating that the individual introduction of C2f_Dual, SPPELAN, or their combined integration can enhance detection accuracy, reduce model size, and improve detection speed. Compared to M0, M3 achieves a 4.8% increase in mAP@0.5 (%), demonstrating that the improvements made to the neck and head contribute to the most significant accuracy gains. However, we observe that the improvements made to the neck and head of the model, while significantly increasing detection accuracy and reducing model parameters, result in a decrease in inference speed, as indicated by a 39.5 drop in FPS. Let's discuss this in detail. First, the number of model parameters is not directly related to inference speed, as speed is more influenced by factors such as computational complexity, feature map size, and hardware utilization efficiency. Even if the parameter count is reduced, if new modules increase computational complexity (such as processing larger feature maps or implementing complex feature fusion mechanisms), inference speed may still decrease. In our model, the Drone-AFPN structure, while optimizing the number of parameters, integrates low-level features (i.e., higher-resolution maps), which increases computational complexity and slows down inference speed. However, this also leads to a significant accuracy improvement. In conclusion, we sacrificed a slight decrease in detection speed for a substantial accuracy boost. Finally, the comparison between M6 and M5 shows that the introduction of Shape-IoU also enhances detection accuracy.

4.4.4 Drone-AFPN Improvement Experiments

To evaluate the effectiveness of Drone-AFPN in UAV aerial image object detection, various feature pyramid networks (FPN) were integrated into the baseline YOLOv8s model for comparative testing, with results shown in Table 3. While Drone-AFPN does not achieve the best results in parameter count or detection speed, it shows notable advantages in precision, recall, mAP@0.5, and mAP@0.5:0.95. Overall, Drone-AFPN effectively extracts multi-scale feature information and demonstrates excellent detection performance for UAV aerial.

Table 2: Results of ablation experiments

Models	A	B	C	D	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Params (M)	FPS
M0					38.7	23.0	11.1	149.1
M1	√				38.8	23.0	10.5	152.6
M2		√			39.0	23.1	10.5	151.2
M3			√		43.5	26.3	10.4	109.6
M4	√	√			39.2	23.2	9.9	**153.3**
M5	√	√	√		43.6	26.4	**9.2**	118.2
M6	√	√	√	√	**44.1**	**26.8**	**9.2**	117.9

Note: Bold indicates the highest ranking for this metric.

Table 3: Comparative experiment of feature fusion network on the VisDrone2019 dataset

Method	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Params (M)	FPS
PAFPN	50.6	37.7	38.7	23.0	11.1	149.1
BIFPN	50.6	38.4	39.4	23.8	10.3	148.6
AFPN	49.2	37.1	37.8	22.5	**8.6**	**149.7**
Drone-AFPN	**52.6**	**41.5**	**43.5**	**26.3**	10.4	109.6

Note: Bold indicates the highest ranking for this metric.

5 Conclusion

This paper presents DAFPN-YOLO, an enhanced model based on YOLOv8s, designed to tackle challenges in UAV object detection, such as small objects, occlusions, and complex backgrounds, while maintaining a balance between accuracy and speed. The model integrates the C2f_Dual and SPPELAN modules into the backbone, enhancing feature extraction and information aggregation, thereby improving inference efficiency. The inclusion of Drone-AFPN significantly enhances feature fusion by fully integrating low-level features through an adaptive fusion mechanism. This approach minimizes semantic information loss and spatial detail blurring, while strengthening multi-scale feature fusion capabilities. Additionally, a 160 × 160 small-object detection layer is added, enabling a finer receptive field to improve the detection of smaller objects and enhancing accuracy for small targets. Finally, Shape-IoU is employed as the bounding box regression loss, improving localization accuracy for irregularly shaped and occluded objects. Experiments conducted on the VisDrone2019 dataset demonstrate that DAFPN-YOLO outperforms the YOLOv8s baseline, achieving a 5.4 percentage point increase in mAP@0.5, a 3.8 percentage point increase in mAP@0.5:0.95, and a 17.2% reduction in parameters. These results highlight the model's superior detection performance and efficiency. DAFPN-YOLO not only provides an innovative solution for drone-based object detection tasks but also offers valuable insights for addressing small object detection, real-time optimization, and target recognition in complex environments. Future work will focus on further optimizing the Drone-AFPN structure to enhance its lightweight design, expanding its applicability for real-time UAV detection tasks.

Acknowledgement: We deeply appreciate the valuable suggestions provided by the reviewers and editors, as well as the support from the National Natural Science Foundation.

Funding Statement: This work was supported by the National Natural Science Foundation of China (Grant Nos. 62101275 and 62101274).

Author Contributions: Honglin Wang contributed to the introduction, related work, and experimental support. Yaolong Zhang led the study's conception, design, module improvements, and initial manuscript drafting. Cheng Zhu provided guidance and critically evaluated the manuscript. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data underlying this study's findings can be obtained from the corresponding author upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Tang G, Ni J, Zhao Y, Gu Y, Cao W. A survey of object detection for UAVs based on deep learning. Remote Sens. 2023 Dec;16(1):149. doi:10.3390/rs16010149.
2. Bouguettaya A, Zarzour H, Kechida A, Taberkit AM. Vehicle detection from UAV imagery with deep learning: a review. IEEE Trans Neural Netw Learning Syst. 2022 Nov;33(11):6047–67. doi:10.1109/TNNLS.2021.3080276.
3. Zhu J, Yang G, Feng X, Li X, Fang H, Zhang J, et al. Detecting wheat heads from UAV low-altitude remote sensing images using deep learning based on transformer. Remote Sens. 2022 Oct;14(20):5141. doi:10.3390/rs14205141.
4. Tahir NUA, Long Z, Zhang Z, Asim M, ELAffendi M. PVswin-YOLOv8s: UAV-based pedestrian and vehicle detection for traffic management in smart cities using improved YOLOv8. Drones. 2024 Feb;8(3):84. doi:10.3390/drones8030084.
5. Zhao M, Li W, Li L, Wang A, Hu J, Tao R. Infrared small UAV target detection via isolation forest. IEEE Trans Geosci Remote Sensing. 2023;61:1–16. doi:10.1109/TGRS.2023.3321723.
6. Mohsan SAH, Othman NQH, Li Y, Alsharif MH, Khan MA. Unmanned aerial vehicles (UAVs): practical aspects, applications, open challenges, security issues, and future trends. Intel Serv Robotics. 2023 Jan;6(1):17. doi:10.1007/s11370-022-00452-4.
7. Li CM, Qi ZL, Jia N, Wu JH. Human face detection algorithm via Haar cascade classifier combined with three additional classifiers. In: 2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI); 2017 Oct; Yangzhou, China: IEEE. p. 483–7. doi:10.1109/ICEMI.2017.8265863
8. Terven J, Córdova-Esparza D-M, Romero-González J-A. A comprehensive review of YOLO architectures in computer vision: from YOLOv1 to YOLOv8 and YOLO-NAS. Mach Learn Knowl Extr. 2023 Nov;5(4):1680–716. doi:10.3390/make5040083.
9. Jiang P, Ergu D, Liu F, Cai Y, Ma B. A review of Yolo algorithm developments. Procedia Comput Sci. 2022;199(11):1066–73. doi:10.1016/j.procs.2022.01.135.
10. Khanam R, Hussain M. YOLOv11: an overview of the key architectural enhancements. arXiv:2410.17725. 2024.
11. Sohan M, Sai Ram T, Rami Reddy CV. A review on YOLOv8 and its advancements. In: Jacob IJ, Piramuthu S, Falkowski-Gilski P, editors. Data intelligence and cognitive informatics. Singapore: Springer Nature Singapore. 2024. p. 529–45. doi:10.1007/978-981-99-7962-2_39.
12. Du D, Zhu P, Wen L, Bian X, Lin H, Hu Q, et al. VisDrone-DET2019: the vision meets drone object detection in image challenge results. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops; 2019; Los Alamitos, CA, USA.
13. Yang G, Lei J, Zhu Z, Cheng S, Feng Z, Liang R. AFPN: asymptotic feature pyramid network for object detection. In: 2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC); 2023 Oct; Honolulu, Oahu, HI, USA: IEEE. p. 2184–9. doi:10.1109/SMC53992.2023.10394415.
14. Zhong J, Chen J, Mian A. DualConv: dual convolutional kernels for lightweight deep neural networks. IEEE Trans Neural Netw Learning Syst. 2023 Nov;34(11):9528–35. doi:10.1109/TNNLS.2022.3151138.
15. Wang C-Y, Yeh I-H, Liao H-YM. YOLOv9: learning what you want to learn using programmable gradient information. arXiv:2402.13616. 2024.

16. Zheng Z, Wang P, Liu W, Li J, Ye R, Ren D. Distance-IoU Loss: faster and better learning for bounding box regression. AAAI. 2020 Apr;34(7):12993–3000. doi:10.1609/aaai.v34i07.6999.
17. Zhang H, Zhang S. Shape-IoU: more accurate metric considering bounding box shape and scale. arXiv:2312.17663. 2023.
18. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2001; 2001; Kauai, HI, USA: IEEE. p. I-511–I-518. doi:10.1109/CVPR.2001.990517.
19. Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05); 2005; San Diego, CA, USA: IEEE. p. 886–93. doi:10.1109/CVPR.2005.177.
20. Lowe DG. Distinctive image features from scale-invariant keypoints. Int J Comput Vis. 2004 Nov;60(2):91–110. doi:10.1023/B:VISI.0000029664.99615.94.
21. 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; 2014; Columbus, OH, USA. p. 580–7.
22. Girshick R. Fast R-CNN. arXiv:1504.08083. 2015.
23. Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell. 2017 Jun;39(6):1137–49. doi:10.1109/TPAMI.2016.2577031.
24. Lin T-Y, Dollar P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017 Jul; Honolulu, HI, USA: IEEE. p. 936–44. doi:10.1109/CVPR.2017.106.
25. He KM, Gkioxari G, Dollár P. Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision; 2017; Los Alamitos, CA, USA. p. 2961–9.
26. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Cheng Y, et al. SSD: single shot multibox detector. In: Computer Vision–ECCV 2016: 14th European Conference; 2016; Amsterdam, The Netherlands. p. 21–37. doi:10.1007/978-3-319-46448-0_2.
27. Tan M, Pang R, Le QV. EfficientDet: scalable and efficient object detection. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun; Seattle, WA, USA: IEEE. p. 10778–87. doi:10.1109/CVPR42600.2020.01079.
28. Pang J, Chen K, Shi J, Feng H, Ouyang W, Lin D. Libra R-CNN: towards balanced learning for object detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2019 Jun; Long Beach, CA, USA: IEEE. p. 821–30. doi:10.1109/CVPR.2019.00091
29. Fang H, Xia M, Zhou G, Chang Y, Yan L. Infrared small UAV target detection based on residual image prediction via global and local dilated residual networks. IEEE Geosci Remote Sensing Lett. 2022;19:1–5. doi:10.1109/LGRS.2021.3085495.
30. Deng L, Bi L, Li H, Chen H, Duan X, Lou H, et al. Lightweight aerial image object detection algorithm based on improved YOLOv5s. Sci Rep. 2023 May;13(1):7817. doi:10.1038/s41598-023-34892-4.
31. Tang S, Zhang S, Fang Y. HIC-YOLOv5: improved YOLOv5 for small object detection. arXiv:2309.16393. 2024.
32. Nie H, Pang H, Ma M, Zheng R. A lightweight remote sensing small target image detection algorithm based on improved YOLOv8. Sensors. 2024 May;24(9):2952. doi:10.3390/s24092952.
33. Li Y, Li Q, Pan J, Zhou Y, Zhu H, Wei H, et al. SOD-YOLO: small-object-detection algorithm based on improved YOLOv8 for UAV images. Remote Sens. 2024 Aug;16(16):3057. doi:10.3390/rs16163057.
34. Wang C, Han Q, Li C, Li J, Kong D, Wang F, et al. Assisting the planning of harvesting plans for large strawberry fields through image-processing method based on deep learning. Agriculture. 2024 Apr;14(4):560. doi:10.3390/agriculture14040560.
35. Li Y, Fan Q, Huang H, Han Z, Gu Q. A modified YOLOv8 detection network for UAV aerial image recognition. Drones. 2023 May;7(5):304. doi:10.3390/drones7050304.
36. Bakirci M. Real-time vehicle detection using YOLOv8-nano for intelligent transportation systems. Trait Signal. 2024 Aug;41(4):1727–40. doi:10.18280/ts.410407.

37. Wang C-Y, Mark Liao H-Y, Wu Y-H, Chen P-Y, Hsieh J-W, Yeh I-H. CSPNet: a new backbone that can enhance learning capability of CNN. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW); 2020 Jun; Seattle, WA, USA: IEEE. p. 1571–80. doi:10.1109/CVPRW50498.2020.00203
38. Liu, Shu, Qi L, Qin H, Shi J. Path aggregation network for instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2018; Los Alamitos, CA, USA. p. 8759–68.
39. 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 Sep;37(9):1904–16. doi:10.1109/TPAMI.2015.2389824.
40. Wang C-Y, Bochkovskiy A, Liao H-YM. YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023 Jun; Vancouver, BC, Canada: IEEE. p. 7464–75. doi:10.1109/CVPR52729.2023.00721
41. Jin X, Tong A, Ge X, Ma H, Li J, Fu H, et al. YOLOv7-bw: a dense small object efficient detector based on remote sensing image. IECE Transact Intell Systemat. 2024 May;1(1):30–9. doi:10.62762/TIS.2024.137321.
42. Bochkovskiy A, Wang C-Y, Liao H-YM. YOLOv4: optimal speed and accuracy of object detection. arXiv:2004.10934. 2020 Apr.
43. Cai Z, Vasconcelos N. Cascade R-CNN: delving into high quality object detection. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun; Salt Lake City, UT, USA: IEEE. p. 6154–62. doi:10.1109/CVPR.2018.00644