remote sensing

MDPI

Article

Small Object Detection in UAV Remote Sensing Images Based on Intra-Group Multi-Scale Fusion Attention and Adaptive Weighted Feature Fusion Mechanism

Zhe Yuan, Jianglei Gong, Baolong Guo *, Chao Wang, Nannan Liao, Jiawei Song and Qiming Wu

Institute of Intelligent Control and Image Engineering, Xidian University, Xi'an 710071, China; zheyuan@stu.xidian.edu.cn (Z.Y.); gongjianglei@stu.xidian.edu.cn (J.G.); cwang_93@stu.xidian.edu.cn (C.W.); nnliao@stu.xidian.edu.cn (N.L.); jwsong@stu.xidian.edu.cn (J.S.); 22131214225@stu.xidian.edu.cn (Q.W.)

* Correspondence: blguo@xidian.edu.cn

Abstract: In view of the issues of missed and false detections encountered in small object detection for UAV remote sensing images, and the inadequacy of existing algorithms in terms of complexity and generalization ability, we propose a small object detection model named IA-YOLOv8 in this paper. This model integrates the intra-group multi-scale fusion attention mechanism and the adaptive weighted feature fusion approach. In the feature extraction phase, the model employs a hybrid pooling strategy that combines Avg and Max pooling to replace the single Max pooling operation used in the original SPPF framework. Such modifications enhance the model's ability to capture the minute features of small objects. In addition, an adaptive feature fusion module is introduced, which is capable of automatically adjusting the weights based on the significance and contribution of features at different scales to improve the detection sensitivity for small objects. Simultaneously, a lightweight intra-group multi-scale fusion attention module is implemented, which aims to effectively mitigate background interference and enhance the saliency of small objects. Experimental results indicate that the proposed IA-YOLOv8 model has a parameter quantity of 10.9 MB, attaining an average precision (mAP) value of 42.1% on the Visdrone2019 test set, an mAP value of 82.3% on the DIOR test set, and an mAP value of 39.8% on the AI-TOD test set. All these results outperform the existing detection algorithms, demonstrating the superior performance of the IA-YOLOv8 model in the task of small object detection for UAV remote sensing.

Keywords: UAV remote sensing images; small object detection; feature fusion; attention mechanism; adaptive

check for updates

Citation: Yuan, Z.; Gong, J.; Guo, B.; Wang, C.; Liao, N.; Song, J.; Wu, Q. Small Object Detection in UAV Remote Sensing Images Based on Intra-Group Multi-Scale Fusion Attention and Adaptive Weighted Feature Fusion Mechanism. *Remote Sens.* **2024**, *16*, 4265. https://doi.org/10.3390/rs16224265

Academic Editor: Riccardo Roncella

Received: 30 September 2024
Revised: 3 November 2024
Accepted: 13 November 2024
Published: 15 November 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The rapid evolution of unmanned aerial vehicle (UAV) technology has precipitated its extensive applications across diverse sectors [1–3]. Owing to their compact form factor, exceptional mobility, cost efficiency, and operational adaptability, UAVs have emerged as indispensable instruments in domains such as military reconnaissance, power line inspection, and traffic surveillance. In the sphere of military reconnaissance, UAVs can adeptly infiltrate adversarial territories to efficiently acquire critical intelligence and monitor enemy military installations [4,5]. For power line inspections, UAVs are proficient in conducting systematic evaluations of high-voltage transmission lines, thereby mitigating the hazards associated with manual assessments [6]. In the domain of traffic surveillance, UAVs facilitate the real-time monitoring of vehicular flow and the detection of traffic infractions [7] thus enabling timely interventions and effective accident management.

Despite the considerable potential of UAV applications, they encounter numerous challenges in executing object detection and tracking tasks, particularly when addressing small objects [8,9]. First, small objects occupy a minimal number of pixels in UAV remote sensing images compared to their larger counterparts, resulting in insufficient feature information that complicates their detection. Second, the expansive field of view inherent in UAV remote sensing imagery encompasses substantial background information; consequently, small objects are susceptible to background noise interference, which hinders the detector's ability to distinguish between the object and its surroundings. In addition, UAVs are typically equipped with lightweight embedded chips that limit computational power; however, detecting small objects often requires greater computational resources for processing complex image data. Thus, under real-time operational constraints, achieving the efficient detection of small objects with restricted computational capabilities poses a significant challenge [10].

In recent years, the advent of deep learning technology has instigated transformative changes within the domain of image processing [11–13]. In contrast to conventional image processing techniques, such as edge detection [14,15] and template matching [16,17], deep learning-based object detection algorithms present substantial advantages in terms of accuracy and robustness. In the current domain of deep learning, object detection algorithms are primarily classified into the following two major categories based on whether they depend on the Region Proposal Network (RPN): two-stage object detection algorithms that are RPN-dependent and one-stage object detection algorithms that are RPN-independent. Two-stage object detection algorithms, like Faster R-CNN [18] and Cascade R-CNN [19], usually entail a preprocessing step that generates candidate regions via RPN and subsequently performs target classification and localization on these regions. In contrast, one-stage object detection algorithms, for instance, Single Shot MultiBox Detector (SSD) [20] and RetinaNet [21], directly predict the bounding boxes and classes of targets on the image without the need for an extra candidate region generation process. Furthermore, object detection algorithms can be further categorized according to whether they employ anchor boxes. Anchor-based algorithms, such as YOLOv1-v4 [22–25] and YOLOv7 [26], predict the position and size of targets using predefined anchor boxes. These algorithms enhance the detection accuracy by adjusting the anchor boxes to match the actual bounding boxes. On the other hand, anchor-free algorithms, such as YOLOv6 [27], YOLOv8 [28], YOLOv9 [29], and YOLOv10 [30], do not rely on predefined anchor boxes but directly predict the bounding boxes, which contributes to a simplification of the detection process and potentially enhances model flexibility. These classifications not only mirror the diversity in the design of object detection algorithms but also embody the distinct requirements for speed and accuracy in practical applications. As deep learning technology continuously progress, these algorithms are also being constantly optimized and evolved to accommodate increasingly complex detection tasks.

Despite the significant advances achieved by the aforementioned deep learning techniques in object detection, these algorithms still face specific limitations in detecting small objects in UAV imagery, such as missed detections, false positives, and an excessive number of model parameters, which complicate deployment on resource-constrained mobile devices. To address these challenges, numerous researchers have sought to enhance small object detection accuracy and reduce model size through the integration of multi-scale fusion and attention mechanisms. Incorporating the attention mechanism allows the model to focus on the object region thus improving the detection accuracy, while multi-scale feature fusion effectively mitigates the issue of scale variation issues among small objects. To tackle the challenges posed by background interference and scale variation in small object detection within UAV remote sensing images, Tan et al. [31] proposed the YOLOv4_UAV model which employs an ultra-lightweight subspace attention mechanism (ULSAM) to generate distinct attention maps for each subspace of the feature map, facilitating multi-scale feature representation. However, this attention mechanism does not take the channel dimension into account. Shang et al. [32] introduced an enhanced YOLOv5s algorithm aimed at improving small object detection in UAV aerial images by reinforcing multi-layer feature fusion alongside advanced attention mechanisms. While this approach significantly enhances performance for detecting small objects, it concurrently increases computational complexity. Shen et al. [33] developed a method based on ASFF-YOLOv5s specifically designed for small object detection from UAVs; this method integrates the Convolutional Block Attention Module (CBAM) with an improved adaptive spatial feature fusion (ASFF) module to incorporate shallow feature maps into the network's feature fusion process, thereby augmenting the extraction capabilities for features associated with small objects. Nonetheless, this algorithm is still vulnerable to missed detections and false positives in complex scenarios. Li et al. [34] presented a refined YOLOv5s algorithm tailored for the detection of small objects in UAV aerial photography by reconstructing its feature fusion network while incorporating SPD convolution along with enhancements to EIoU loss function. Although this strategy significantly boosts both accuracy and real-time performance in detecting smaller objects, further validation is necessary regarding its generalization across diverse datasets or varying environmental conditions. Xiong et al. [8] proposed AS-YOLOv5—a specialized algorithm objecting small object detection from UAVs—which improves the capabilities concerning low-resolution objects featuring diminutive characteristics via adaptive feature fusion coupled with an enhanced attention mechanism; however, they pointed out that future research should explore anchor-free detection algorithms as their current methodology may still exhibit inefficiencies or inadequacies within certain application contexts.

In conclusion, while the aforementioned algorithms have demonstrated improvements in small object detection performance within UAV remote sensing images, they suffer from high model complexity and inadequate generalization capabilities. This results in the persistent issue of missed detections and false positives for small objects. Furthermore, in the detection of small objects in UAV images, the detection performance is typically deteriorated due to the following physical factors: (1) Small objects merely occupy a limited number of pixels, making it challenging for the model to extract sufficient discriminative features. (2) Objects in the natural environment are frequently partially occluded by other objects, thereby triggering missed detections and false detection. (3) Changes in weather and lighting conditions lead to a reduction in the contrast between the object and the background, thereby influencing the detection accuracy. To cope with the challenges of the aforementioned algorithmic deficiencies and the decline in detection performance caused by physical factors, this paper proposes a small object detection network for UAV remote sensing images based on YOLOv8. This approach is intended to address these challenges, enhance the accuracy of small object detection, and facilitate its effective deployment on UAVs. The primary contributions are as follows:

(1) We propose a novel adaptive weighted feature fusion (AWFF) module, which dynamically adjusts feature weights to enhance the representation of key features, thereby significantly improving the discriminative power of the model for object recognition. In addition, our module effectively integrates feature information from multiple levels, such that the model can simultaneously capture the details and semantic information of the object;

(2) We designed a Mixed Spatial Pyramid Pooling Fast (Mix-SPPF) module that combines the advantages of average pooling and maximum pooling to improve the accuracy of recognizing small objects;

(3) We introduce a novel lightweight intra-group multi-scale fusion attention module (IGMSFA) that effectively reduces the influence of background noise while ensuring high performance in resource-constrained environments;

(4) In comparison with the current mainstream YOLO series algorithms and classical object detection methods, our proposed IA-YOLOv8 algorithm demonstrates significant advantages. Specifically, IA-YOLOv8 achieves a higher mAP while maintaining a reduced number of parameters.

The structure of this paper is organized as follows: Section 2 presents related work, discusses the choice of the foundation framework, and provides an in-depth analysis of the attention mechanism and feature fusion strategy. Section 3 details the methodology, including a comprehensive overview of the IA-YOLOv8 architecture and its key components. Section 4 focuses on the experiments, outlining the experimental environment, the datasets used for small object detection in UAVs, and the analysis of the experimental results. Finally, Section 5 summarizes the findings of this study and provides insights for the future research directions.

2. Related Work

2.1. Object Detection Algorithms

Currently, small object detection algorithms are primarily developed through the enhancement of regular conventional object detection methods. The two-stage object detection algorithms require the generation of candidate boxes for subsequent analysis, whereas the one-stage algorithm performs the detection directly. This distinction provides one-stage algorithms with a clear advantage in terms of speed and computational efficiency. Consequently, in the domain of UAV object detection, one-stage algorithms exhibit a markedly superior performance [35].

In the domain of one-stage object detection algorithms, the YOLO family of models has gained wide recognition in both the academic and industrial circles due to their exceptional performance, high accuracy, and robust scalability. In contrast with the anchor-free approach, the anchor-based strategy employed by YOLOv1 through YOLOv5, YOLOv7 exhibits certain limitations in terms of computational speed and capacity. Moreover, among the three most recent advances in object detection—YOLOv6, YOLOv8 to YOLOv10—these models demonstrate superior performance compared to their predecessors on large-scale general object detection datasets. Therefore, when selecting a foundation framework for small object detection in UAV remote sensing imagery, it is recommended to prioritize the utilization of YOLOv8 over YOLOv10 models.

To address the requirements for small object detection in UAV remote sensing images, a comprehensive review of the literature [36–39] reveals that YOLOv10 exhibits slightly inferior performance compared to YOLOv8 and YOLOv9 in this specific task. While YOLOv9 offers advantages over YOLOv8 in terms of lightweight architecture, it lacks the same level of task adaptation. Therefore, this study adopts YOLOv8 as the foundational framework and seeks to enhance its performance through objected modifications aimed at improving small object detection capabilities.

2.2. Attention Mechanisms

Recognized as plug-and-play modules, attention mechanisms have found widespread applications in the domain of small object detection due to their straightforward architecture and low computational cost. Currently, attention mechanisms based on deep learning can be categorized into the following four fundamental types: channel attention, spatial attention, fusion attention, and self-attention mechanisms [35,40,41]. In the context of drone detection where objects are submerged by the complex background, Wang et al. [42] enhanced feature extraction by introducing the SeNet channel attention mechanism, thereby suppressing background interference. Nevertheless, the SeNet channel attention mechanism merely focuses on the information of the channel dimension and fails to effectively capture spatial features. To overcome this constraint, Li et al. [43] incorporated the subspace attention mechanism (ULSAM) into the YOLOv4 model, generating distinct attention feature maps for each feature map subspace. However, the ULSAM also only prioritizes the spatial dimension while disregarding the channel information. Although these attention mechanisms have demonstrated a certain degree of effectiveness in practice, they frequently neglect the key elements of the spatial or channel dimensions, resulting in the loss of crucial information. In response to these challenges, Wang et al. [44] proposed a global attention mechanism that integrates the information of both the channel and spatial dimensions, enhancing the accuracy of target detection. However, compared with methods that only focus on a single dimension, this approach demands more computational resources and exhibits higher complexity. The aforementioned CNN-based methods mainly focus on object local feature enhancement; however, recent advances in large-scale models have shifted the focus toward self-attention mechanisms grounded in Transformer architectures.

For instance, Liu et al. [45] presented the Swin Transformer model that improves contextual understanding by employing self-attention over the relevant regions. Although its resource requirements significantly exceed those associated with the traditional CNN-based attentional frameworks, the deployment of rendering on mobile devices is challenging. To mitigate the computational demands while enhancing deployability for mobile applications, Hou et al. [46] proposed Coordinate Attention (CA), a lightweight method built upon a novel CNN structure that simultaneously attends to both channel and spatial information through horizontal and vertical feature aggregation, effectively integrating vital coordinate data within generated representations.

Through an in-depth exploration of the attention mechanisms, we conclude that selecting the appropriate attention mechanisms can significantly enhance model performance across a variety of task requirements and contexts.

2.3. Feature Fusion

Feature fusion, which involves integrating features from different levels to enhance model performance and robustness, has been widely applied in the field of object detection. Currently, feature fusion can be categorized into early fusion and late fusion based on the sequence of integration [47–49].

Early fusion involves an initial fusion of multi-layer features, followed by training on the fused representation. Common early fusion techniques include the Concat and Add feature fusion operations. In their work, Bell et al. [50] introduced the Inside–Outside Net (ION) method, which employs a Concat feature fusion strategy to achieve a more discriminative integration of multi-layer features. Similarly, Kong et al. [51] proposed the HyperNet method that utilizes an Add feature fusion strategy to consolidate the features from various layers. Additionally, Sun et al. [52] extended the concept of a canonical correlation analysis (CCA) to feature fusion, resulting in a CCA-based approach that leverages the correlation between two input feature sets through transformations designed to enhance their inter-correlation compared to the original sets. However, this methodology is primarily limited by its inability to effectively maximize correlations across different feature sets. To address this limitation, Chaib et al. [53] introduced discriminative correlation analysis (DCA), aimed at maximizing correlations between two groups of features while simultaneously enhancing distinctions among different classes. Furthermore, Dai et al. [54] proposed an iterative attention feature fusion (iAFF) method based on a multi-scale channel attention mechanism for a more effective integration of semantically and scale-inconsistent features; this approach generates weights via an attention mechanism and then optimizes the key information extraction process across the different layers.

Late fusion techniques enhance detection performance by integrating the results from different layers. Currently, late fusion can be categorized into feature pyramid fusion and multi-branch fusion based on its integration methodology [35]. The Bidirectional Cross-scale Connections and Weighted Feature Fusion (BiFPN) [55] serves as a representative feature pyramid fusion module that transmits the positional information of low-level features in a bottom–up manner while conveying semantic information of high-level features in a top–down fashion. Moreover, it establishes skip connections between input and output features at the same layer to mitigate information loss during transmission. However, this approach is associated with increased model complexity and computational demands, which can hinder its deployment on mobile devices. Zhou et al. [56] introduced the Small-Scale Feature Enhancement Module (SFEM), which enhances the feature representation of small objects through a parallel multi-branch structure for feature fusion on the input feature map. Nonetheless, this approach has limitations due to the elevated computational cost arising from parallel architecture, as well as the potential low-resolution issues caused by dilated convolutions.

In summary, selecting the appropriate feature fusion strategies tailored to the specific requirements of different scenarios and task types can significantly enhance object detection performance, especially in the context of small object detection on mobile devices.

3. Methods

3.1. Overall Architecture of IA-YOLOv8

To better adapt to different task requirements, the YOLOv8 family offers five distinct scale variants—YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x—to address diverse application requirements. These variants differ in parameter counts and detection accuracy, facilitating a balance between speed and precision. Therefore, this paper presents the development of the IA-YOLOv8 algorithm based on the YOLOv8s variant to enhance the detection capability for small objects. The detailed architecture of this algorithm is illustrated in Figure 1. This network consists of the following four parts: (1) Input: This section is accountable for adjusting the image to the size demanded by the network to guarantee effectiveness during the feature extraction process. Meanwhile, the Mosaic data augmentation technique is adopted to enhance the generalization ability and anti-interference capacity of the network; (2) Backbone: This section employs the CSPDarkNet-53 network to extract features of the input image at diverse scales to accommodate the variations in the size of the target; (3) Neck: This section utilizes the PANet (Path Aggregation Network) structure to enable a better integration of the spatial detail information of the shallow layers with the semantic information of the deep layers thus forming multi-scale features; and (4) Prediction: This section constitutes the final part of the entire detection network and is responsible for decoding and predicting the features from the neck. It is utilized to generate the final category and bounding box regression. By generating predictions on feature maps of different scales, the precise detection of targets of different scales can be achieved. The following are the enhancements made to the algorithmic framework:

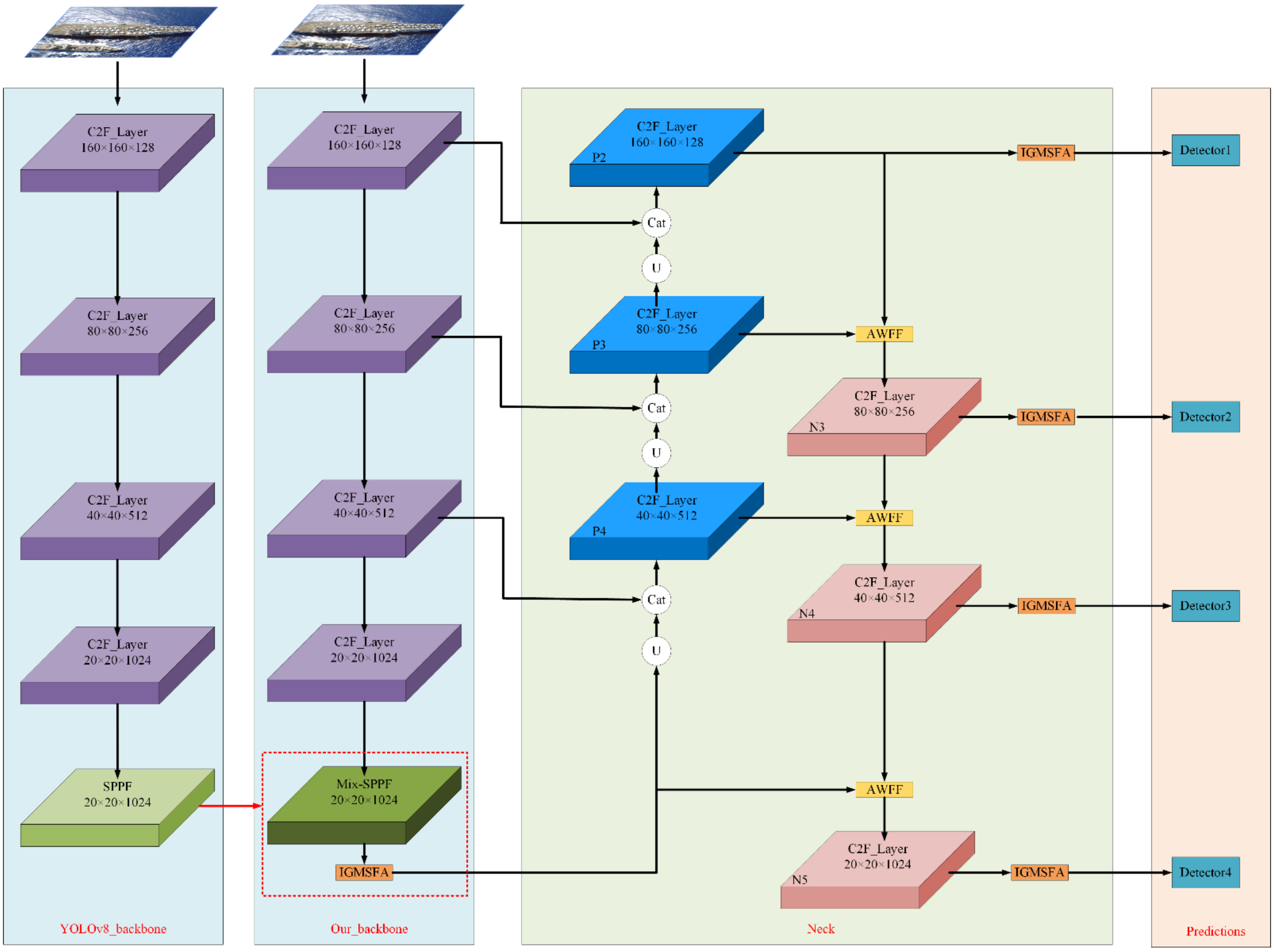

Figure 1. Overall Architecture of IA-YOLOv8.

First, the backbone network of YOLOv8s integrates the Spatial Pyramid Pooling Fast (SPPF) module to effectively capture object details across diverse scales, thereby enhancing the model's capacity for object perception. This paper proposes substituting the SPPF module with a Mixed Spatial Pyramid Pooling Fast (Mix-SPPF) module. In SPPF, using Max pooling for multi-scale feature aggregation can lead to the loss of information, especially when addressing small objects with low resolution. Consequently, Mix-SPPF replaces the singular max pooling operation with a hybrid approach that combines both Max and Avg pooling to mitigate information loss and further enhance detection accuracy for small objects.

Second, within the neck layer, YOLOv8s employs the PANet framework. In PANet, a Concat operation is used to merge high-resolution spatial features from shallow layers with robust semantic features from the deeper layers. However, this operation merely concatenates feature maps without considering their individual significance or relevance; Thus, it fails to fully leverage the complementary information between them. To address this limitation, this paper introduces an adaptive weighted feature fusion (AWFF) module that dynamically assigns weights to features. Compared to the Concat operation, AWFF demonstrates enhanced efficiency in multi-layer feature fusion by reducing redundant information and optimizing computational resources while smoothing the gradient flow and enabling more nuanced feature selection; consequently, both model performance and stability are improved. Furthermore, in order to enhance detection accuracy for small objects while minimizing background interference in UAV remote sensing images, this paper presents an intra-group multi-scale fusion attention (IGMSFA) module. This module amplifies the focus on the critical features while significantly reducing the background noise, thereby improving both the accuracy of small object detection and the overall model performance.

Finally, in the prediction layer, the IA-YOLOv8s network model employs four detection heads for object detection. In contrast to the three detection heads employed in the YOLOv8s network model, this approach allows for a more efficient exploitation of multi-scale features, thereby enabling the capture of more complex detailed features and enhancing the detection accuracy of small objects.

3.2. Mix Spatial Pyramid Soft Pool Fast (Mix-SPPF)

In the contemporary research landscape, the following four dominant pyramid pooling modalities have been delineated: SPP (Spatial Pyramid Pooling); SPPF; SPPCSPC (Spatial Pyramid Pooling—Cross Stage Partial Connections); and SPPFCSPC (Spatial Pyramid Pooling Fast—Cross Stage Partial Connections). Notably, both SPP and SPPCSPC leverage a parallel architecture to perform Max pooling operations, employing three distinct kernel sizes. Conversely, the SPPF and SPPFCSPC modules adopt a cascaded framework that sequentially interconnects three Max pooling layers of identical kernel dimensions for data processing. This architectural paradigm facilitates superior computational efficiency in SPPF and SPPFCSPC compared to their parallel counterparts, as the former capitalizes on an optimized serial processing mechanism that generates expedited processing times. In addition, the computational speeds of SSPFCSPC and SSPCSPC are slower than those of SPPF and SPP due to the higher complexity of the SSPFCSPC and SSPCSPC models compared to the SPPF and SPP models.

To enhance the detail representation capability of small objects, this section introduces a Mix-SPPF module that integrates an average pooling layer into the existing SPPF framework, as illustrated in Figure 2. In contrast with traditional Max pooling methods, which solely preserve max values within a local region, average pooling captures more comprehensive and nuanced feature information by considering all the elements within the pooling region. Moreover, the smoothing effect provided by average pooling effectively mitigates the effect of outliers on the results, thereby reducing the risk of overfitting. This enhancement enables the Mix-SPPF module to extract more refined and high-dimensional features without compromising computational efficiency, significantly enhancing the generalization capability of the model with respect to the input data.

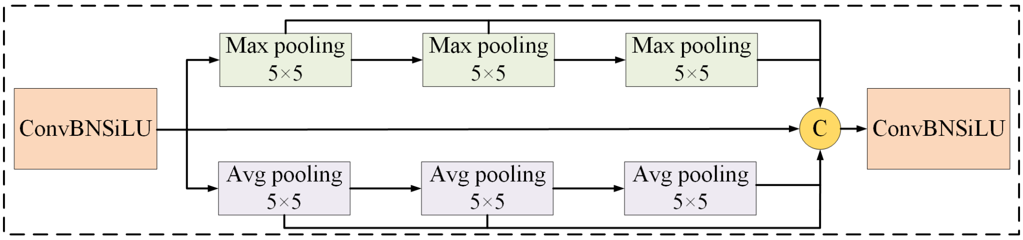

Figure 2. Mix-SPPF Module.

3.3. Adaptive Weighted Feature Fusion (AWFF) Module

Multi-scale feature fusion represents a key strategy for augmenting the detection performance of a model and has been extensively utilized in the domain of object detection. The underlying principle of this methodology is that shallow feature maps exhibit high resolution and encapsulate intricate details, yet they are characterized by limited semantic richness and heightened noise; in contrast, deep feature maps provide substantial semantic information but suffer from lower resolution and reduced detail fidelity. By synergistically integrating shallow and deep feature maps, it becomes possible to leverage their respective strengths, thereby enhancing the overall detection efficiency of the model.

In the contemporary landscape of object detection, the prevalent feature fusion strategies predominantly include the Concat and Add operations. The Concat operation merges feature maps across various levels along the channel dimension to facilitate efficient feature integration; however, this approach may introduce redundant information due to partial content overlaps among the feature maps at different levels. In contrast, the Add operation combines feature maps from distinct levels through simple addition to achieve a fusion effect. Nevertheless, this method merely executes basic arithmetic and fails to fully exploit the complementary information inherent in each feature layer, leading to some degree of information loss, which is particularly pronounced when significant scale disparities exist. Figure 3 illustrates a schematic representation of these commonly employed feature fusion strategies.

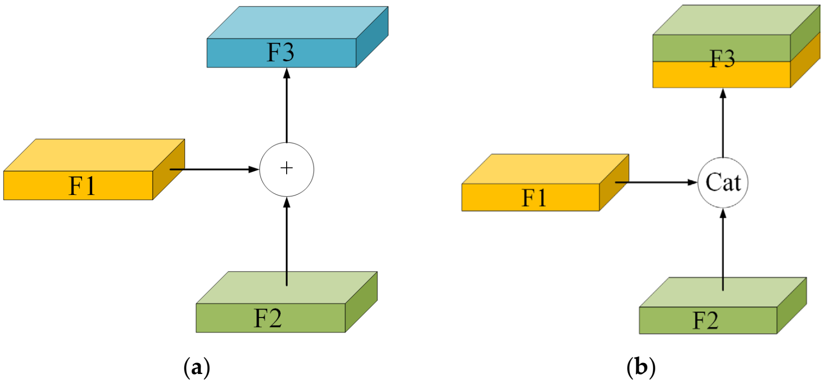

Figure 3. Commonly employed feature fusion strategies. (**a**) Add; (**b**) Concat.

In the contemporary landscape of image processing and computer vision, common operations often face the problem of low feature fusion efficiency, especially for small object detection. To address this issue, we introduced an adaptive weighted feature fusion (AWFF) module, optimized from the Attention Feature Fusion (iAFF) module developed by the Nanjing University of Aeronautics and Astronautics [54]. The primary advantage of the AWFF module is its capacity to dynamically assign weights to various feature layers, thereby enhancing both efficiency and accuracy in feature fusion. The architecture of the AWFF module is depicted in Figure 4.

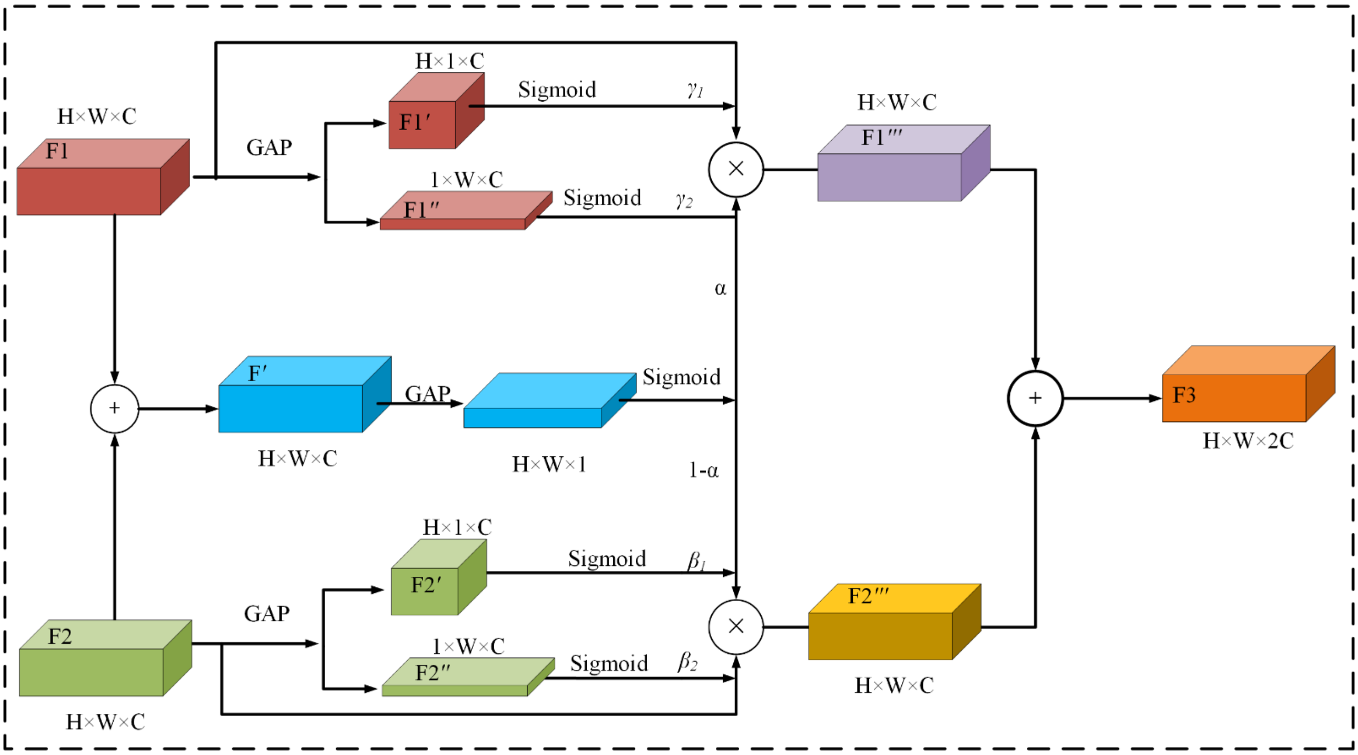

Figure 4. Adaptive Weighted Feature Fusion (AWFF) Module.

Considering that the dimensions of the adjacent feature maps $F1$ and $F2$ are both H × W × C, we initiate an Add feature fusion operation on $F1$ and $F2$ to produce a fused result denoted as F′. Subsequently, we analyze the influence of weight vectors on the feature maps $F1$ and $F2$ by extracting their respective weight vectors across height and width dimensions through global average pooling (GAP). Following this, we normalize these weight vector elements utilizing a Sigmoid function to derive γ_1, γ_2, β_1, and β_2. Furthermore, to ascertain the weight vector α for the fused result F', we implement global average pooling (GAP) along the channel dimension, followed by normalization via a Sigmoid function. Thereafter, we amalgamate γ_1, γ_2, and α with feature map $F1$ to generate a new feature map designated as $F1'''$; Similarly, we integrate β_1, β_2, and $(1-\alpha)$ with feature map $F2$ to yield another new feature map referred to as $F2'''$. Ultimately, an Add feature fusion operation is executed on these newly generated feature maps, $F1'''$ and $F2'''$ thus resulting in the final output feature map labeled as $F3$. The details of the calculation are described as follows:

$$F' = F1 \oplus F2 \tag{1}$$

$$F1''' = \gamma_1 \otimes \gamma_2 \otimes \alpha \otimes F1 \tag{2}$$

$$F2''' = \beta_1 \otimes \beta_2 \otimes (1-\alpha) \otimes F2 \tag{3}$$

$$F3 = F1''' \oplus F2''' \tag{4}$$

In this context, H, W, and C denote the height, width, and number of channels of the feature map, respectively. The dimensions of $F3$, $F1'''$, and $F2'''$ are uniformly H × W × C. '$\oplus$' denotes the Add fusion operation, while '$\otimes$' denotes element-wise multiplication. Importantly, α, β_1, β_2, γ_1, and γ_2 are normalized using the Sigmoid activation function, which constrains their values within a range of 0 to 1.

In the adaptive weighted feature fusion (AWFF) module, weights are determined based on the scale of input features and are generated through a global average pooling operation across multiple dimensions. This design allows the module to optimize its feature fusion strategy in an adaptive manner. Such dynamism enhances both the flexibility and robustness of the model when dealing with multi-scale features. Furthermore, distinct features often encapsulate rich information. Thus, adaptive fusion maximizes the extraction of complementary information, thereby augmenting the discriminative power of the model. Compared to the conventional feature fusion methods, AWFF effectively mitigates the effect of irrelevant or redundant features by dynamically adjusting the weights thus improving the efficacy of feature representations.

3.4. Intra-Group Multi-Scale Fusion Attention Module (IGMSFA)

The core of the attention mechanism resides in concentrating on the key information within an image and inhibiting background interference. Particularly in UAV remote sensing images, due to the complexity of the background, the high proportion of small targets, and the low resolution, problems such as missed detection and false detection are prone to occur when conducting small target detection. To address this challenge and guarantee that the proposed scheme can be effectively deployed on the UAV platform, this paper puts forward a lightweight intra-group multi-scale fusion attention module (IGMSFA) by enhancing the ultra-light subspace attention module (ULSAM) [57]. The IGMSFA module effectively overcomes the limitation of ULSAM that merely focuses on the spatial dimension. The specific improvement scheme is as follows:

To enable the attention module to simultaneously focus on both spatial and channel dimensions, we propose a Fusion Attention (FA) mechanism. The underlying principle of this fusion module is illustrated in Figure 5. This mechanism integrates the coordinate attention mechanism with the spatial attention mechanism while also enhancing the coordinate attention component to reduce parameter count. First, GAP is applied to the input feature map $F \in R^{H\times W\times g}$ along both the X and Y directions to encode the channel dimension. Consequently, the output for the g-th channel at height h and width w can be represented as z_g^h and z_c^w. Furthermore, by performing a 1 × 1 convolution followed by Sigmoid normalization on z_g^h and z_c^w, we obtain corresponding weight vectors σ and τ. These weights are then used to multiply with feature map F, resulting in a new feature map denoted as $F_1' \in R^{H\times W\times g}$. Subsequently, feature map F_1' undergoes both average pooling and max pooling operations to generate two channel descriptions of size H × W × 1. These two descriptions are concatenated along the channel dimension, yielding a combined description of size H × W × 2. Following this step, a 7 × 7 convolution operation coupled with Sigmoid normalization is performed on this description to derive weight vector ζ. Finally, we multiply feature map F_1' by weight vector ζ to produce an updated feature map denoted as F_2. This proposed mechanism effectively addresses the limitation encountered when utilizing only spatial attention mechanisms, specifically their tendency to prioritize spatial information at the expense of neglecting channel information. The mathematical formulation is presented as follows:

$$z_g^h = \frac{1}{W}\sum_{0<i<W} x_g(h,i) \tag{5}$$

$$z_g^w = \frac{1}{H}\sum_{0<j<H} x_g(j,w) \tag{6}$$

$$\sigma = Sigmoid(Conv^{1\times 1}(z_g^h)) \tag{7}$$

$$\tau = Sigmoid(Conv^{1\times 1}(z_g^w)) \tag{8}$$

$$F_1' = F \otimes \sigma \otimes \tau \tag{9}$$

$$F_2 = Sigmoid(Conv^{7\times 7}(\mathrm{Cat}[Avg\mathrm{pool}(\mathrm{F_1}'), Max\mathrm{pool}(\mathrm{F_1}')])) \tag{10}$$

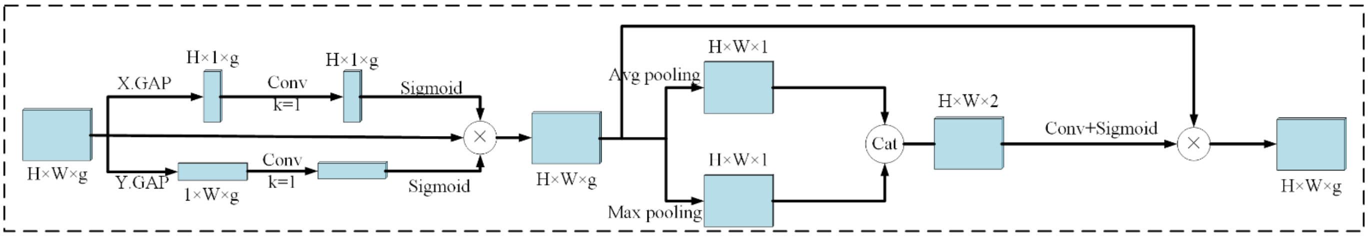

Figure 5. Fusion Attention (FA) Mechanism.

(1) In order to effectively capture multi-level information within the input feature map, this section employs convolution kernels of varying sizes within the same convolutional layer to extract local features across different spatial ranges. Furthermore, to mitigate the computational complexity associated with convolution operations and enhance computational efficiency, this study introduces a method that combines depth-wise separable convolutions with dilated convolutions. Figure 6 illustrates the Multi-Scale Attention Fusion (MSAF) module, wherein depth-wise separable convolutions decompose the traditional convolutions into two distinct steps: depth-wise convolution and point-wise convolution for each channel. This decomposition significantly reduces both parameter count and computational load while preserving model performance. Dilated convolutions expand the receptive field by applying zero padding between the convolution kernel and input features without incurring additional parameters or computational overhead, thereby improving the network's capacity to capture extensive contextual information. The mathematical expression for the receptive field of dilated convolutions is presented as follows:

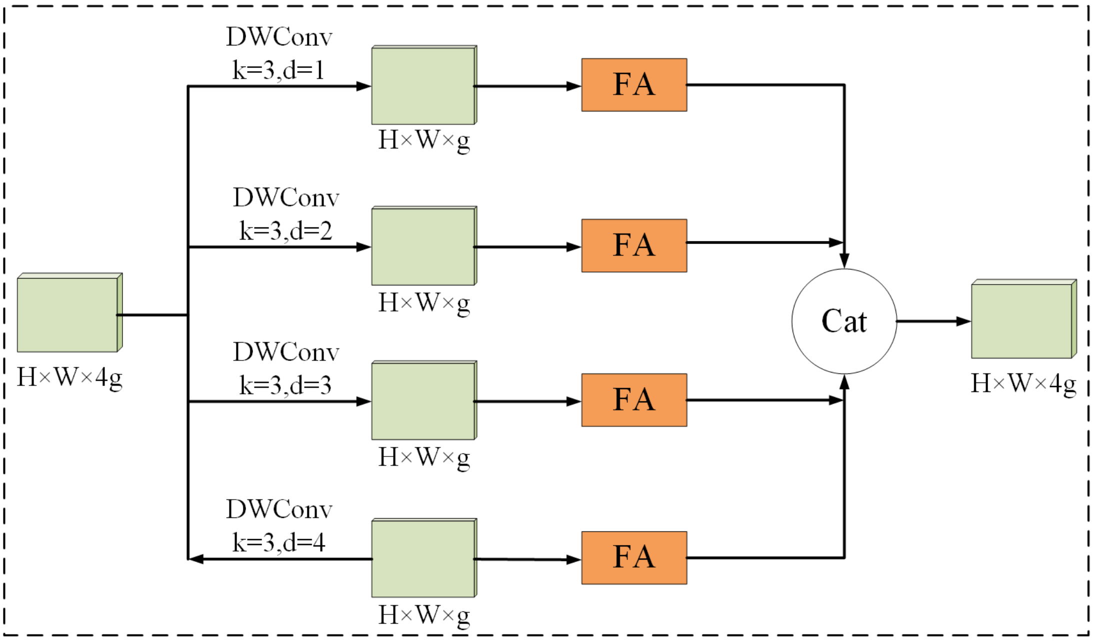

Figure 6. Multi-Scale Attention Fusion (MSAF) Module.

$$RFS = [1 + (k-1) \times (d-1)] \times s \tag{11}$$

In this context, k denotes the size of the convolutional kernel, s indicates the stride, and d represents the dilation rate.

(2) The MSAF module is integrated into the ULSAM module to replace its single spatial attention mechanism. Additionally, to further enhance information flow, a lightweight intra-group multi-scale fusion attention module (IGMSFA) is introduced, as shown

in Figure 7. Firstly, for the input feature map $F \in R^{H \times W \times G}$ (where $G = n \times 4\,g$), it is uniformly divided into n groups along the channel dimension, with the number of channels in each group being 4 g, to obtain the following:

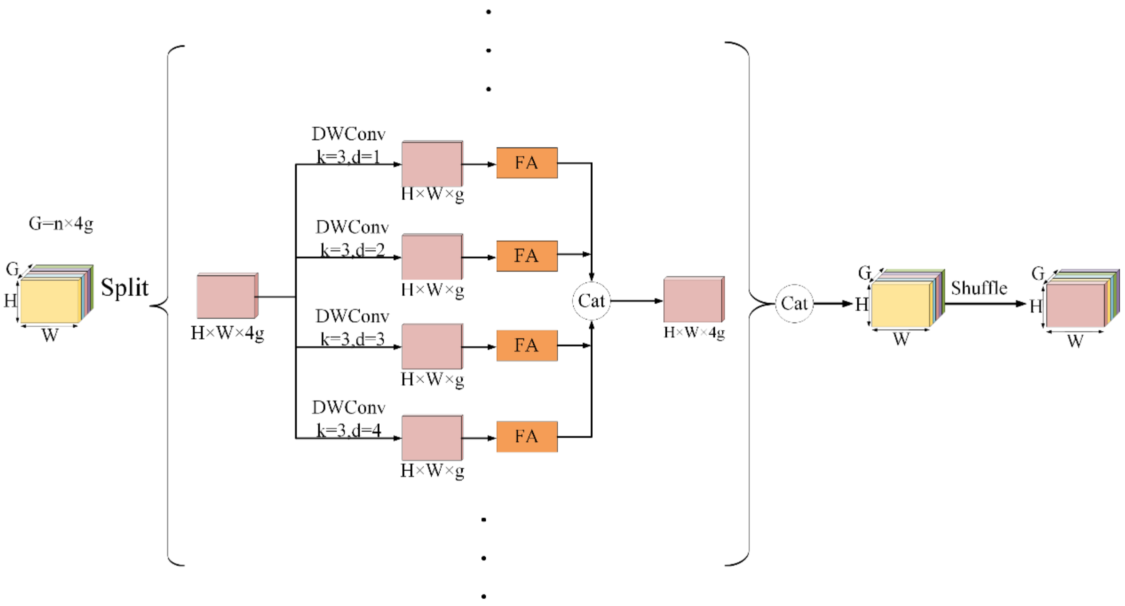

Figure 7. Intra-Group Multi-Scale Fusion Attention (IGMSFA) Module.

$$F \in R^{H \times W \times G} \rightarrow [F_1, F_2, \ldots, F_n] \in R^{H \times W \times 4g} \tag{12}$$

Secondly, a multi-scale fusion attention mechanism is applied to each feature map to obtain the weighted feature map $[\hat{F}_1, \hat{F}_2, \ldots, \hat{F}_n] \in R^{H \times W \times 4g}$.

$$[\hat{F}_1, \hat{F}_2, \ldots, \hat{F}_n] = MSAF([F_1, F_2, \ldots, F_n]) \tag{13}$$

Ultimately, a feature fusion operation based on concatenation is executed for each group of feature maps, followed by a reorganization of channels to yield refined feature representations.

$$F' = Shuffle\big(Concat[\hat{F}_1, \hat{F}_2, \ldots, \hat{F}_n]\big) \tag{14}$$

3.5. Loss Function

The Intersection over Union (*IoU*) [58] is a critical evaluation metric extensively utilized in the context of small object detection tasks for UAVs, serving to assess the performance of models in this domain. *IoU* quantitatively measures the degree of overlap between the predicted bounding box and the ground truth bounding box, with values ranging from 0 to 1. A higher *IoU* value signifies a greater degree of overlap between the predicted and actual boxes, thereby indicating superior detection performance. The formal definition of *IoU* is as follows:

$$IoU = \frac{B_p \cap B_g}{B_p \cup B_g} \tag{15}$$

where B_p denotes the predicted bounding box, while B_g signifies the ground truth bounding box.

To maximize the Intersection over Union (*IoU*) during the training process, researchers have proposed a series of *IoU*-based loss functions. These loss functions are specifically designed to directly optimize *IoU* and its variants, thereby enhancing the model's detection performance. Notably, *IoU_Loss* maximizes the *IoU* value by utilizing $1 - IoU$ as the loss function, which is defined as follows:

$$IoU\_Loss = 1 - IoU \tag{16}$$

Despite the advancements made by *IoU_Loss* in addressing variable independence and scale invariance, it remains ineffective in optimizing scenarios where there is no intersection between the ground truth box and the predicted box. Furthermore, it does not adequately capture the overlapping relationship between these two boxes. To address this limitation, Rezatofighi et al. [59] introduced the concept of the minimum enclosing rectangle based on *IoU* to enhance the matching process between predicted and ground truth boxes. Consequently, they proposed *GIoU* and defined a loss function based on *GIoU*, which is articulated as follows:

$$GIoU = IoU - \frac{\left|C - B_p \cup B_g\right|}{|C|} \tag{17}$$

$$GIoU\_Loss = 1 - GIoU \tag{18}$$

In this context, *C* denotes the smallest enclosing rectangle that encompasses both the predicted bounding box and the ground truth bounding box.

However, due to the slow convergence and insufficient regression accuracy associated with *GIoU_Loss*, Zheng et al. [60] proposed a *DIoU*-based loss function, termed *DIoU_Loss*. This method considers the distance between the centers of the predicted bounding box and the ground truth bounding box. Its definition is as follows:

$$DIoU = IoU - \frac{\left[d\left(B_p, B_g\right)\right]^2}{c^2} \tag{19}$$

$$DIoU\_Loss = 1 - DIoU \tag{20}$$

In this context, *c* denotes the Euclidean distance between the two diagonal vertices of the minimum bounding rectangle, while $d(B_p,\ B_g)$ represents the Euclidean distance between the center points of the predicted and ground truth bounding boxes.

To further optimize the regression loss function, Zheng et al. [60] proposed the *CIoU_Loss* by considering factors such as the overlapping area, center point distance, and aspect ratio. The definition is as follows:

$$CIoU = IoU - \left(\frac{\left[d\left(B_p, B_g\right)\right]^2}{c^2} + \alpha v\right) \tag{21}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^g}{h^g} - \arctan\frac{w^p}{h^p}\right)^2 \tag{22}$$

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{23}$$

$$CIoU = 1 - CIoU \tag{24}$$

In this context, w^g and h^g denote the width and height of the ground truth box, respectively. w^p and h^p represent the width and height of the predicted box. Additionally, α and v signify the correlation terms associated with width and height.

The four mainstream loss functions primarily focus on the overlapping regions between the predicted and ground truth bounding boxes, neglecting the other areas that may exist between them. This oversight can result in biased evaluation outcomes. To address this issue, Tong et al. [61] introduced the following three novel loss functions: *WIoU v1_Loss*, *WIoU v2_Loss*, and *WIoU v3_Loss*.

WIoU v1: A distance-based attention mechanism was developed, resulting in a two-layer attention framework.

$$WIoUv1\_Loss = R_{WIoU} \times IoU\_Loss \tag{25}$$

$$R_{WIoU} = \exp\left(\frac{\left(x_p - x_g\right)^2 + \left(y_p - y_g\right)^2}{\left(W_g^2 + H_g^2\right)^*}\right) \tag{26}$$

In this context, x_p and y_p denote the coordinates of the predicted bounding box, while x_g and y_g represent the coordinates of the ground truth bounding box. Additionally, W_g and H_g indicate the width and height of the minimum enclosing rectangle, respectively. The symbol $*$ signifies a separation operation.

WIoU v2: Based on *WIoU v1*, a monotonic focusing mechanism was introduced to the cross-entropy loss function. This effectively mitigates the influence of simple examples on the overall loss value, thereby enabling the model to concentrate more effectively on challenging instances.

$$WIoUv2\_Loss = \left(\frac{L_{IoU}^*}{\overline{L_{IoU}}}\right)^{\gamma} \times WIoUv1\_Loss, \gamma > 0 \tag{27}$$

In this context, $L_{IoU}^* \in [0,1]$ denotes the monotonic focusing coefficient, while $\overline{L_{IoU}}$ represents the exponential moving average.

WIoU v3: A non-monotonic focusing coefficient r was constructed using the outlier coefficient β and applied to *WIoU v1*, resulting in *WIoU v3* with a dynamic non-monotonic FM.

$$WIoUv3\_Loss = r \times WIoUv1\_Loss \tag{28}$$

$$r = \frac{\beta}{\delta\alpha^{\beta-\delta}} \tag{29}$$

$$\beta = \frac{L_{IoU}^*}{\overline{L_{IoU}}} \tag{30}$$

In this context, α and δ represent hyperparameters.

In the context of small object detection tasks in UAV remote sensing, traditional Intersection over Union (*IoU*) metrics may struggle to effectively differentiate minor variations in bounding boxes due to the typically small size of the objects. In contrast, weighted Intersection over Union (*WIoU*) provides a more precise reflection of these subtle discrepancies. This study employs *WIoUv3_Loss* as the regression loss function, which significantly enhances the model's localization capabilities for small objects through a dynamic non-monotonic mechanism.

4. Experiments and Analysis

In this section, we systematically and comprehensively validate the proposed method across the following four key dimensions: experimental environment setup and dataset introduction, ablation studies, comparative experiments, and visualization analysis. This comprehensive approach aims to ensure the effectiveness, stability, and superiority of the method.

4.1. Experimental Environment Setup and Data Set Introduction

To ensure the reliability and reproducibility of the experimental results, we employed the following configuration for the experimental environment in this study—an NVIDIA GeForce RTX 4080 SUPER with 16 GB of memory running on Windows 10. The CPU used was an Intel(R) Core(TM) i7-14700KF. We implemented CUDA version 11.8, while PyTorch version 2.0.0 and Python version 3.9 were utilized for programming purposes. Additionally, no pre-trained weights were incorporated during the model training phase. Detailed key parameter settings relevant to the model training process are provided in Table 1.

Table 1. Parameter Settings for the Training Phase.

Parameters	Value
Epochs	300
Batch_size	6
Input image size	512 × 512
Regression loss function	WIOU v3
Gradient Optimizer	SGD
Momentum	0.935
Initial learning rate	0.01
Final learning rate	0.00001
Data enhancement	Mosaic
IoU	0.5

In addition, to comprehensively assess the efficacy of the proposed IA-YOLOv8 algorithm, this section will examine the experiments utilizing large-scale datasets such as Visdrone 2019, DIOR, and AI-TOD which are extensively recognized in the domain of small object detection within UAV remote sensing imagery.

The Visdrone 2019 dataset [62] is a large-scale benchmark dataset developed by the AISKYEYE team at the Machine Learning and Data Mining Laboratory of Tianjin University, China, in 2019. This dataset was captured using multiple UAV cameras and encompasses a diverse range of dimensions, including geographic locations (spanning 14 different cities across China that are thousands of kilometers apart), environmental types (urban versus rural settings), object objects (such as pedestrians, vehicles, bicycles, etc.), and scene densities (ranging from sparse to crowded). The dataset comprises a total of 10,209 static images across 10 object categories, with 6471 images designated for training purposes, 548 for validation, and 3190 for testing.

The DIOR dataset [63] is a large-scale benchmark dataset for optical remote sensing object detection, which was released by Northwestern Polytechnical University in 2019. This dataset utilizes Google Earth satellite imagery as its data source and encompasses 20 object classes, comprising a total of 23,463 images and 192,472 instances. Each image has a pixel resolution of 800 × 800 pixels, with spatial resolutions ranging from 0.5 m to 30 m. The dataset is partitioned into a training and validation set (11,725 images) and a test set (11,738 images). To ensure an effective distinction between the training and validation sets, we evenly allocated the 11,725 images into 5862 training samples and 5863 validation samples.

The AI-TOD dataset [64] is a large-scale dataset for the detection of micro-targets in aerial images, which was released by Wuhan University in 2021. This dataset was constructed by extracting some images and object instances from datasets such as DOTAv1.5, xView, VisDrone2018, Airbus Ship, and DIOR. It encompasses eight categories, with a total of 28,036 aerial images and 700,621 instances. The pixel resolution of each image is 800 × 800 pixels, and the average size of the targets in the images is 12.8 pixels. Additionally, for a better evaluation of the algorithms, 11,214 images were utilized for training, 2804 for validation, and 14,018 for testing.

4.2. Ablation Experiment

To comprehensively assess the specific contributions of each key component in the proposed IA-YOLOv8 algorithm to model performance, this section meticulously designs a series of ablation experiments and conducts an in-depth analysis on the challenging Visdrone2019 dataset. The experimental baseline is established using the YOLOv8s model, with core evaluation metrics including parameter count, precision (P), recall (R), mean average precision (mAP), and floating-point operations per second (GFLOPs) employed to quantitatively analyze the effectiveness of each component. The experimental results presented in Table 2 indicate that each component of the IA-YOLOv8 model plays a critical role in enhancing the detection performance for small objects. The four detection heads are denoted as “4 × Head”, while the five IGMSFA modules are labeled as “5 × IGMSFA”, and the three AWFF modules are identified as “3 × AWFF”. √ indicates the use of this module.

Table 2. Ablation Study on Visdrone 2019.

Baseline	4 × Head	Mix-SPPF	5 × GMSFA	3 × AWFF	R (%)	P (%)	mAP (%)	Parameter (MB)	GFLOPs
YOLOv8s					32.9	49.9	34.2	10.6	28.5
	√				39.0	52.9	40.8	10.1	36.7
	√	√			39.9	53.2	40.9	10.2	36.7
	√	√	√		40.3	54.2	41.2	10.4	37.2
	√	√	√	√	41.2	54.6	42.1	10.9	37.3

The experimental results presented in Table 2 illustrate that the integration of an additional detection branch into the YOLOv8s benchmark model not only significantly enhances the model’s detection capabilities but also effectively increases its architectural depth. Specifically, the model depth was augmented from the original 168 layers to 207 layers, accompanied by a moderate rise in computational complexity to 36.7 GFLOPs, representing an approximate increase of 22.3% compared to the baseline model. Notably, despite this escalation in computational demands, the number of parameters was reduced from 10.6 MB to 10.1 MB, achieving a significant decrease of 4.7%. In terms of the detection performance metrics, this modification yielded the following substantial improvements: precision (R) rose from 32.9% to 39.0%, recall (P) increased from 49.9% to 52.9%, and the mean average precision (mAP) escalated from 34.2% to 40.8%. These findings clearly indicate that augmenting the detection branch constitutes an effective strategy for enhancing model performance in complex scenarios. Thus, its significance should not be underestimated. These data robustly support both the high efficacy and necessity of this approach for improving detection performance within intricate environments.

To further enhance the performance of the model, we incorporated the designed IGMSFA module into our framework. The experimental results presented in Table 2 demonstrate that the introduction of the IGMSFA module resulted in an increase in both the parameter count and computational load by 0.2 MB and 0.5 GFLOPs, respectively. Furthermore, in small object detection tasks, improvements were observed in precision (R), recall (P), and average precision, with increases of 0.4%, 1.0%, and 0.3%, respectively. These findings indicate that the IGMSFA module plays a significant role in augmenting the model’s capacity to learn and represent complex scene features, thereby serving as an effective strategy for enhancing detection accuracy for small objects.

To further enhance the performance of the model, we incorporated the designed IGMSFA module into our framework. The experimental results presented in Table 2 demonstrate that, with a minimal increase of 0.2 MB in parameter size and 0.5 GFLOPs in computational complexity, the introduction of the IGMSFA module significantly improved detection performance: precision (R), recall (P), and mean average precision (mAP) achieved increases of 0.4%, 1.0%, and 0.3%, respectively. These findings indicate that the IGMSFA module plays a crucial role in augmenting the model’s capacity to learn and represent complex scene features, thereby serving as an effective approach to enhancing detection accuracy.

We incorporated the AWFF module into specific Concat operations within the neck layer to enhance the feature fusion process. The experimental results presented in Table 2 demonstrate that, following the introduction of the AWFF module, precision (R), recall (P), and mean average precision (mAP) improved by 0.9%, 0.4%, and 0.8%, respectively, with a slight increase in the parameters by 0.5 MB and computational load by 0.1 GFLOPs. These findings indicate that the AWFF module significantly enhances feature fusion strategies and improves model detection accuracy, making it an essential tool for optimizing model performance in practical applications.

Furthermore, to facilitate a more comprehensive comparison of the performance between YOLOv8s and IA-YOLOv8 methods before and after ablation, we constructed confusion matrices for both approaches, as illustrated in Figure 8. Figure 8a presents the confusion matrix for YOLOv8s, revealing that this method exhibits notably low detection rates for small objects such as people, bicycles, and awning-tricycles. Additionally, it frequently misclassifies vans as cars, bicycles as motorcycles, and awning-tricycles as cars. In contrast, the confusion matrix for IA-YOLOv8 depicted in Figure 8b demonstrates that this approach effectively reduces the false positive rate associated with small objects while significantly enhancing overall detection accuracy. It can be observed from Figure 8 that when comparing the classification results of YOLOv8 and AS-YOLOv8, the background category exerts an influence on the classification outcome. This is attributed to the fact that the image scenes in the VisDrone2019 dataset are complex, with a majority of the samples belonging to the background category, while the samples of other target categories are relatively scarce compared to the background category. In cases where the features of the background category and the target categories are similar or the boundaries are ambiguous, the model may erroneously classify an object as the background.

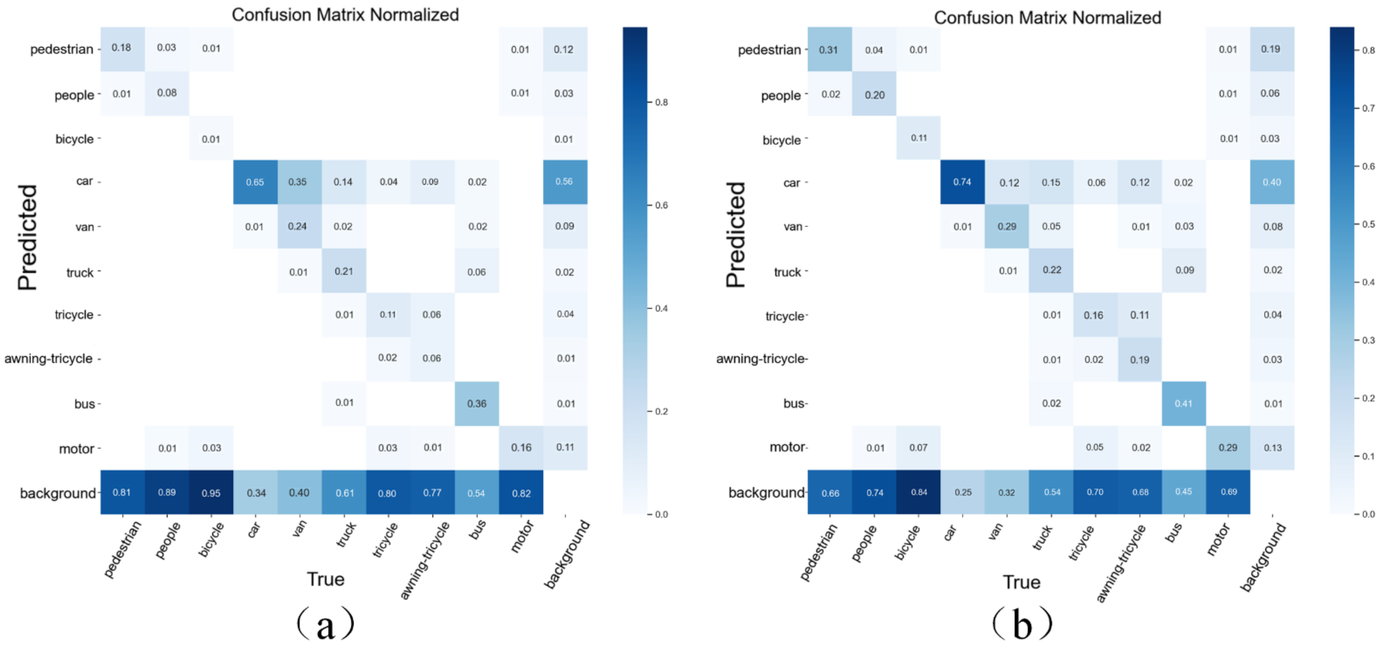

Figure 8. Comparison of confusion matrices for the YOLOv8s and IA-YOLOv8 algorithms on the Visdrone2019 dataset at an IoU threshold of 0.5. Panel (**a**) illustrates the confusion matrix produced by the YOLOv8s algorithm, while panel (**b**) presents the confusion matrix generated by the IA-YOLOv8 algorithm.

4.3. Comparative Experiment

To comprehensively validate the superiority of the proposed IA-YOLOv8 algorithm, this study conducted a comparative analysis with several mainstream methods currently prevalent in the field. Under consistent experimental settings and evaluation metrics, we assessed the performance of each method. Tables 3–5 present the experimental results obtained on the Visdrone2019, DIOR, and AI-TOD validation sets. Moreover, for a more in-depth assessment of the superiority of the IA-YOLOv8 algorithm proposed in this paper, we conducted a comparison between it and the two algorithms currently dedicated for the small object detection of unmanned aerial vehicles [65,66] on various datasets.

Table 3. Comparative Analysis of Different Algorithms on the Visdrone2019 Dataset.

Method	Backbone Network	Input_Size	Parameter (MB)	Model Size (MB)	mAP (%)
Faster RCNN [18]	ResNet50 + FPN	600 × 600	41.5	83.3	28.6
Cascade-RCNN [19]	ResNet50 + FPN	600 × 600	64.3	128.9	32.7
SSD [26]	VGG16	300 × 300	23.7	47.7	15.3
RetinaNet [27]	ResNet50 + FPN	800 × 800	36.8	73.9	24.9
YOLOv5s	CSPDarkNet-53	512 × 512	8.7	17.6	33.4
YOLOv8s [28]	CSPDarkNet-53	512 × 512	10.6	21.5	34.2
YOLOv9s [29]	CSPDarkNet-53	512 × 512	6.8	13.8	34.7
YOLOv10s [30]	CSPDarkNet-53	512 × 512	7.7	15.6	32.5
M-YOLOv8s [65]	CSPDarkNet-53	300 × 300	5.9	11.8	41.2
Imporved_YOLOv8 [66]	CSPDarkNet-53	300 × 300	5.6	11.2	37.4
IA-YOLOv8 (ours)	CSPDarkNet-53	512 × 512	10.9	22.1	42.1

Table 4. Comparative Analysis of Different Algorithms on the DIOR Dataset.

Method	Backbone Network	Input_Size	Parameter (MB)	Model Size (MB)	mAP (%)
Faster RCNN [18]	ResNet50 + FPN	600 × 600	41.5	83.3	65.3
Cascade-RCNN [19]	ResNet50 + FPN	600 × 600	64.3	128.9	72.7
SSD [26]	VGG16	300 × 300	23.7	47.7	55.4
RetinaNet [27]	ResNet50 + FPN	800 × 800	36.8	73.9	68.8
YOLOv5s	CSPDarkNet-53	512 × 512	8.7	17.6	79.1
YOLOv8s [28]	CSPDarkNet-53	512 × 512	10.6	21.5	79.3
YOLOv9s [29]	CSPDarkNet-53	512 × 512	6.8	13.8	78.9
YOLOv10s [30]	CSPDarkNet-53	512 × 512	7.7	15.6	77.1
M-YOLOv8s [65]	CSPDarkNet-53	300 × 300	5.9	11.8	80.7
Imporved_YOLOv8 [66]	CSPDarkNet-53	300 × 300	5.6	11.2	80.2
IA-YOLOv8 (ours)	CSPDarkNet-53	512 × 512	10.9	22.1	82.3

Table 5. Comparative Analysis of Different Algorithms on the AI-TOD Dataset.

Method	Backbone Network	Input_Size	Parameter (MB)	Model Size (MB)	mAP (%)
Faster RCNN [18]	ResNet50 + FPN	600 × 600	41.5	83.3	25.6
Cascade-RCNN [19]	ResNet50 + FPN	600 × 600	64.3	128.9	27.4
SSD [26]	VGG16	300 × 300	23.7	47.7	14.7
RetinaNet [27]	ResNet50 + FPN	800 × 800	36.8	73.9	26.3
YOLOv5s	CSPDarkNet-53	512 × 512	8.7	17.6	30.7
YOLOv8s [28]	CSPDarkNet-53	512 × 512	10.6	21.5	35.9
YOLOv9s [29]	CSPDarkNet-53	512 × 512	6.8	13.8	30.9
YOLOv10s [30]	CSPDarkNet-53	512 × 512	7.7	15.6	29.5
M-YOLOv8s [65]	CSPDarkNet-53	300 × 300	5.9	11.8	38.6
Imporved_YOLOv8 [66]	CSPDarkNet-53	300 × 300	5.6	11.2	37.4
IA-YOLOv8 (ours)	CSPDarkNet-53	512 × 512	10.9	22.1	**39.8**

As presented in Table 3, the Faster R-CNN algorithm, a classical two-stage detection approach, possesses a parameter volume of 41.5 MB and a model size of 83.3 MB, featuring a relatively high complexity. Nevertheless, its detection accuracy is relatively low, with a mean average precision (mAP) merely reaching 28.6%. This implies that, in scenarios with limited resources or stringent real-time requirements, it might not be the optimal option. The Cascade-RCNN algorithm, by virtue of its cascading structure, significantly enhanced the detection accuracy, raising the mAP by 4.1% compared to Faster R-CNN. However, this improvement was accompanied by an additional 22.8 MB of parameters and a 45.6 MB increment in the model size. Conversely, the SSD algorithm, an early one-stage detection method, despite having a parameter size of 23.7 MB and a model size of 47.7 MB, exhibited suboptimal performance (with an mAP of only 15.3%) in complex environments, highlighting its limitations in such contexts. After the introduction of the focal loss function, RetinaNet achieved a significant improvement based on SSD, increasing the mAP by 9.6%, although this also led to a parameter increase of 13.1 MB and a model size increase of 26.2 MB compared to the SSD algorithm. The YOLO series represents a new generation of one-stage detection methodologies that integrate lightweight models with high detection precision and has received wide acclaim in real-time applications. Notably, YOLOv5s achieved an mAP of 33.4% while maintaining a parameter size of only 8.7 MB and a model size of 17.6 MB, significantly outperforming the previously mentioned algorithms. Compared to YOLOv5s, YOLOv8s realized an mAP improvement of 0.8%, while only increasing the parameter size by 1.9 MB and the model size by 3.9 MB. Meanwhile, compared to the YOLOv8s algorithm, YOLOv9s achieved an mAP growth of 0.5% while reducing the parameter size and model size by 3.8 MB and 7.7 MB, respectively. Despite the fact that the parameter size of YOLOv10s was approximately 0.9 MB larger than that of YOLOv9s while the model size was 1.8 MB larger, unfortunately, its mAP decreased by approximately 2.2%. Compared to YOLOv9s, the IA-YOLOv8 algorithm proposed in this paper witnessed a significant mAP improvement of 7.4% while only increasing the parameter size and model size by 4.1 MB and 8.3 MB, respectively, thereby demonstrating remarkable performance enhancement in this context. Additionally, when comparing M-YOLOv8s with IA-YOLOv8, although the parameter size and model size decreased by 5.0 MB and 10.3 MB, respectively, the mAP decreased by 0.9%. When comparing Improved_YOLOv8 with IA-YOLOv8, the parameter size and model size decreased by 5.3 MB and 10.9 MB, respectively, and the mAP decreased by 4.7%. It is evident that although the mAP values of M-YOLOv8s and Improved_YOLOv8 algorithms are slightly lower than the algorithm proposed in this paper, their parameter sizes and model sizes are much smaller than those proposed herein. This is because both the M-YOLOv8s and Improved_YOLOv8 algorithms eliminated the feature mapping layers responsible for detecting large targets, resulting in a significant reduction in model size and parameter volume. However, their drawback lies in that, during the experiments, it was discovered that the computational complexity of the M-YOLOv8s and Improved_YOLOv8 algorithms is far greater than that of the algorithm proposed in this paper, leading to higher requirements for the equipment.

The experimental results presented in Tables 4 and 5 indicate that the IA-YOLOv8 algorithm proposed herein attains average precisions (mAP) of 82.3% and 39.8% on the DIOR and AI-TOD datasets, respectively. This showcases a superior performance in comparison to other algorithms and accentuates its robust generalization capability. Notably, contrary to the trend witnessed for YOLOv9s on the VisDrone2019 dataset, as well as on the DIOR and AI-TOD datasets, the reduction in the quantity of parameters and model size does not align with an enhancement in performance. The mAP of YOLOv9s declines as both the number of parameters and the model size decrease. This phenomenon underlines the inadequate generalization ability of YOLOv9s within the context of optical remote sensing image processing.

4.4. Visual Analysis

In order to evaluate the performance of the IA-YOLOv8 algorithm proposed in this study for small target detection more intuitively and effectively, we selected detection samples from the three test sets of Visdrone2019, DIOR, and AI-TOD. Incorporating the outcomes of the aforementioned comparative experiments, a visualization is presented. Specifically, Figure 9 presents the comparison results of IA-YOLOv8 and YOLOv9s on the Visdrone2019 dataset; Figure 10 showcases the comparison status of IA-YOLOv8 and YOLOv8s on the DIOR dataset; and Figure 11 exhibits the comparative analysis of IA-YOLOv8 and YOLOv8 on the AI-TOD dataset. The achievements of these visualization experiments offer direct evidence for the performance of the IA-YOLOv8 algorithm.

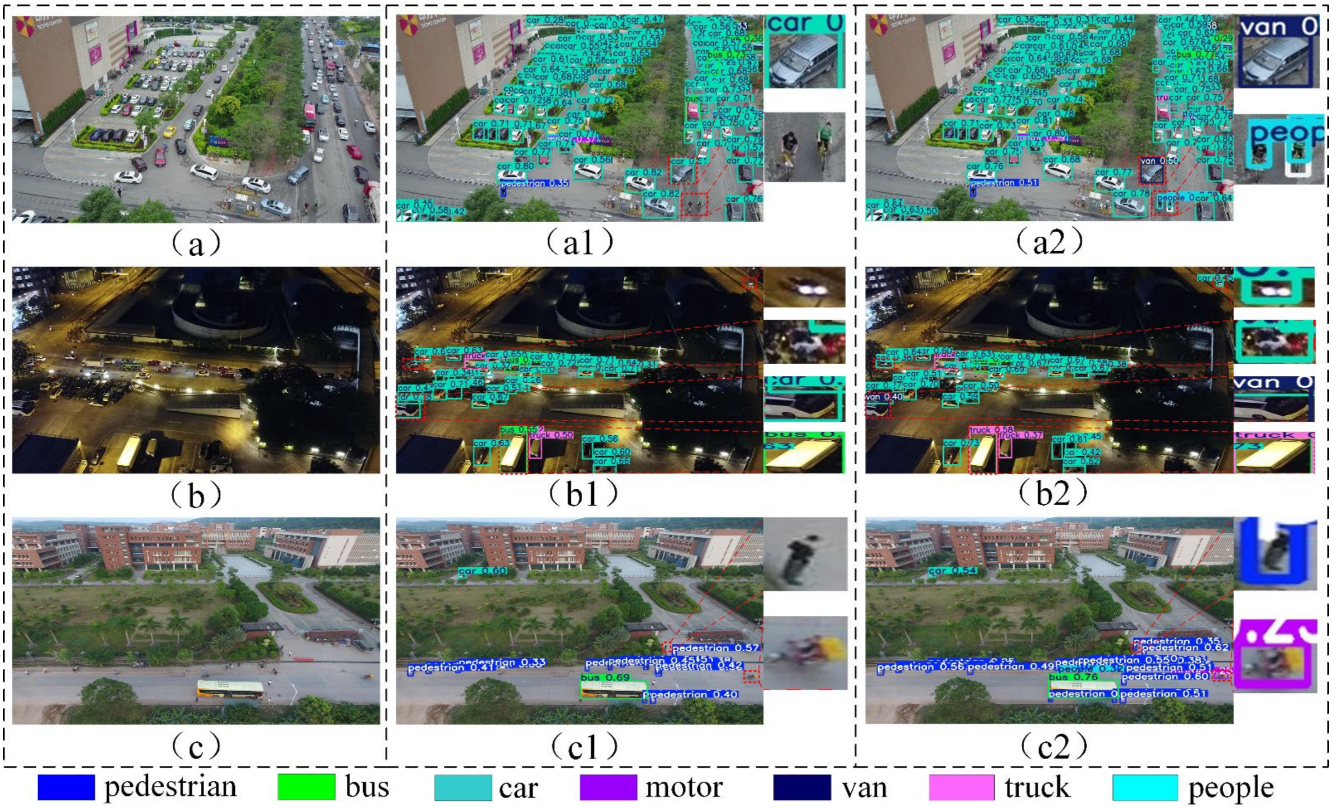

Figure 9. Detection results of YOLOv9s and IA-YOLOv8 on the Visdrone2019 test dataset. The figures in (**a**–**c**) illustrate the input images. The figures in (**a1**,**b1**,**c1**) present the detection results obtained using YOLOv9s. The figures in (**a2**,**b2**,**c2**) present the detection results obtained using IA-YOLOv8.

According to Figure 9, the red dashed boxes delineate the areas of comparison for the two algorithms. A comparison between Figure 9(a1,a2) reveals that the YOLOv9s algorithm misclassifies a van as a car and fails to detect low-resolution individuals, whereas the IA-YOLOv8 algorithm effectively mitigates both false positives and missed detections. In comparing Figure 9(b1,b2), it is evident that the IA-YOLOv8 algorithm successfully identifies cars at greater distances with lower resolution, while also demonstrating robust performance in suppressing instances where YOLOv9s misidentifies vans as cars and trucks as buses. Similar trends are observed in the comparison between Figure 9(c1,c2). These findings indicate that, on the Visdrone2019 test set, IA-YOLOv8 exhibits superior capabilities in small object detection compared to YOLOv9s, significantly reducing both the false positive rates and missed detections.

According to Figure 10, the blue dashed lines delineate the areas of comparison for the two algorithms. A comparative analysis of Figure 10(a1,a2) reveals that the YOLOv8s algorithm fails to detect low-resolution storage tanks and red vehicles, as well as exhibiting missed detections for vehicles whose colors closely match those of their surrounding environment. In contrast, the IA-YOLOv8 algorithm successfully identifies these objects. In examining Figure 10(b1,b2), it is evident that in scenes featuring storage tanks, the YOLOv8s algorithm struggles to recognize tanks with similar background colors and lower resolutions, whereas the IA-YOLOv8 algorithm demonstrates accurate detection capabilities. Furthermore, a comparison between Figure 10(c1,c2) indicates that for ships characterized by low resolution and sparse pixel density, the YOLOv8s algorithm is prone to missed detections. Conversely, the IA-YOLOv8 algorithm exhibits robust detection performance. In summary, on the DIOR test set, the IA-YOLOv8 algorithm shows significant advantages over its YOLOv8s counterpart.

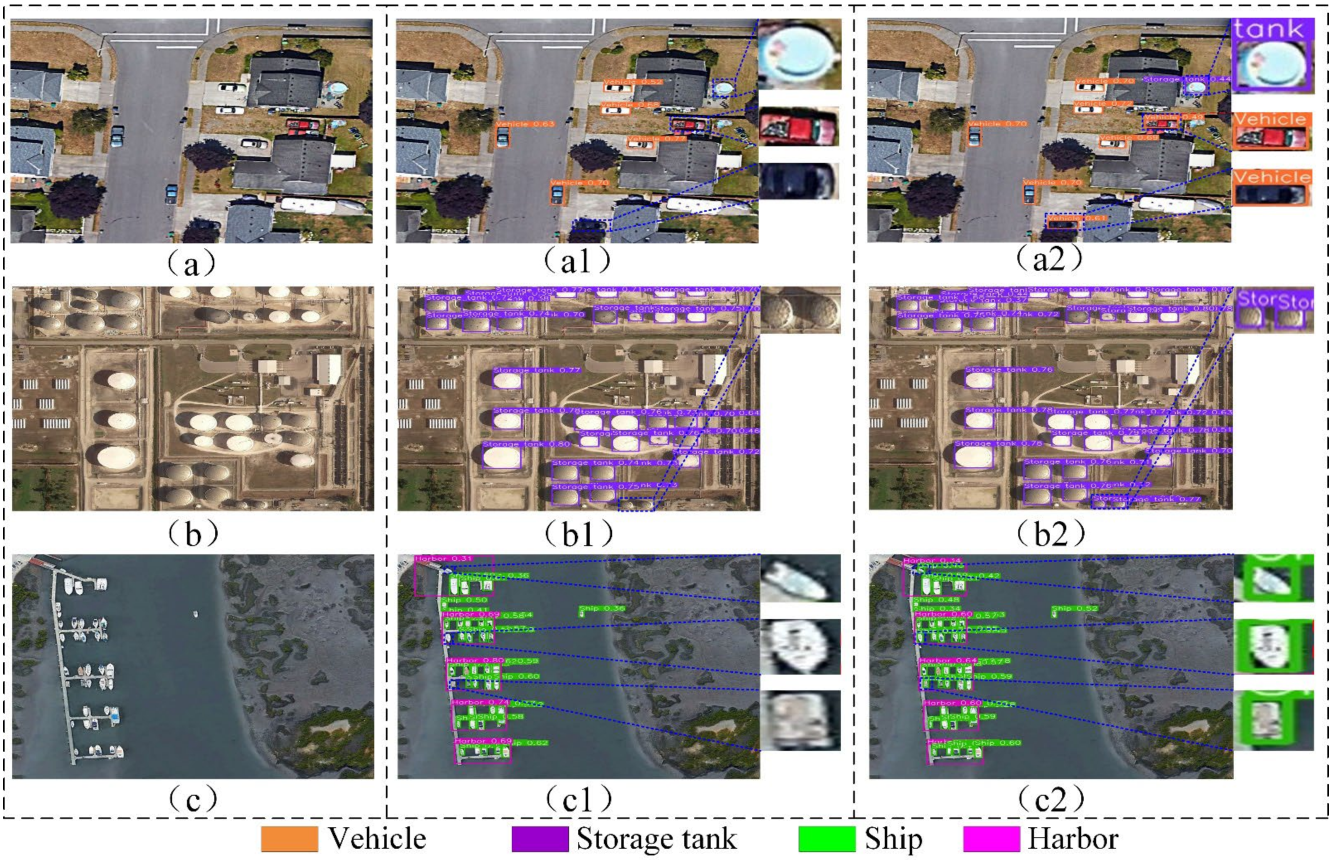

Figure 10. Detection results of YOLOv8s and IA-YOLOv8 on the DIOR test dataset. The figures in (**a**–**c**) illustrate the input images. The figures in (**a1**,**b1**,**c1**) present the detection results obtained using YOLOv8s. The figures in (**a2**,**b2**,**c2**) present the detection results obtained using IA-YOLOv8.

According to Figure 11, the green dashed lines delineate the comparison regions for the two algorithms. Through the comparison of Figure 11(a1,a2), it can be ascertained that the YOLOv8s algorithm fails to detect low-resolution "persons" and "vehicles". By contrast, the IA-YOLOv8 algorithm can successfully identify these objects. When observing Figure 11(b1,b2), it is shown that in scenarios containing "vehicles", the YOLOv8s algorithm has a considerable miss detection rate for low-resolution "vehicles", while the IA-YOLOv8 algorithm not only exhibits precise detection capabilities but also reduces the miss detection rate to a certain extent. Likewise, by contrasting Figure 11(c1,c2), it can be observed that for "vehicles" with a low resolution, the YOLOv8s algorithm is prone to miss detections. Conversely, the IA-YOLOv8 algorithm presents robust detection performance. In conclusion, on the AI-TOD test set, the IA-YOLOv8 algorithm demonstrates significant superiority over the YOLOv8s algorithm.

To further validate the effectiveness of IA-YOLOv8 from a visual perspective, we employed gradient-weighted class activation mapping (Grad-CAM) for an interpretability analysis of the IA-YOLOv8 method. By generating heatmaps for both YOLOv9s and IA-YOLOv8 using Grad-CAM, we obtained a clear representation of the extent to which each network model focuses on different regions within the image. In these heatmaps, red indicates areas of highest attention, while blue signifies areas with minimal focus. As illustrated in Figure 12, the experimental results indicate that YOLOv8s exhibits suboptimal performance in object aggregation and demonstrates limited capability in capturing distant small objects. Conversely, IA-YOLOv8 not only effectively concentrates on objects intended for detection and recognition but also significantly outperforms YOLOv8s in its ability to capture distant small objects. Thus, the IA-YOLOv8 method proposed in this study shows marked advantages in small object detection tasks.

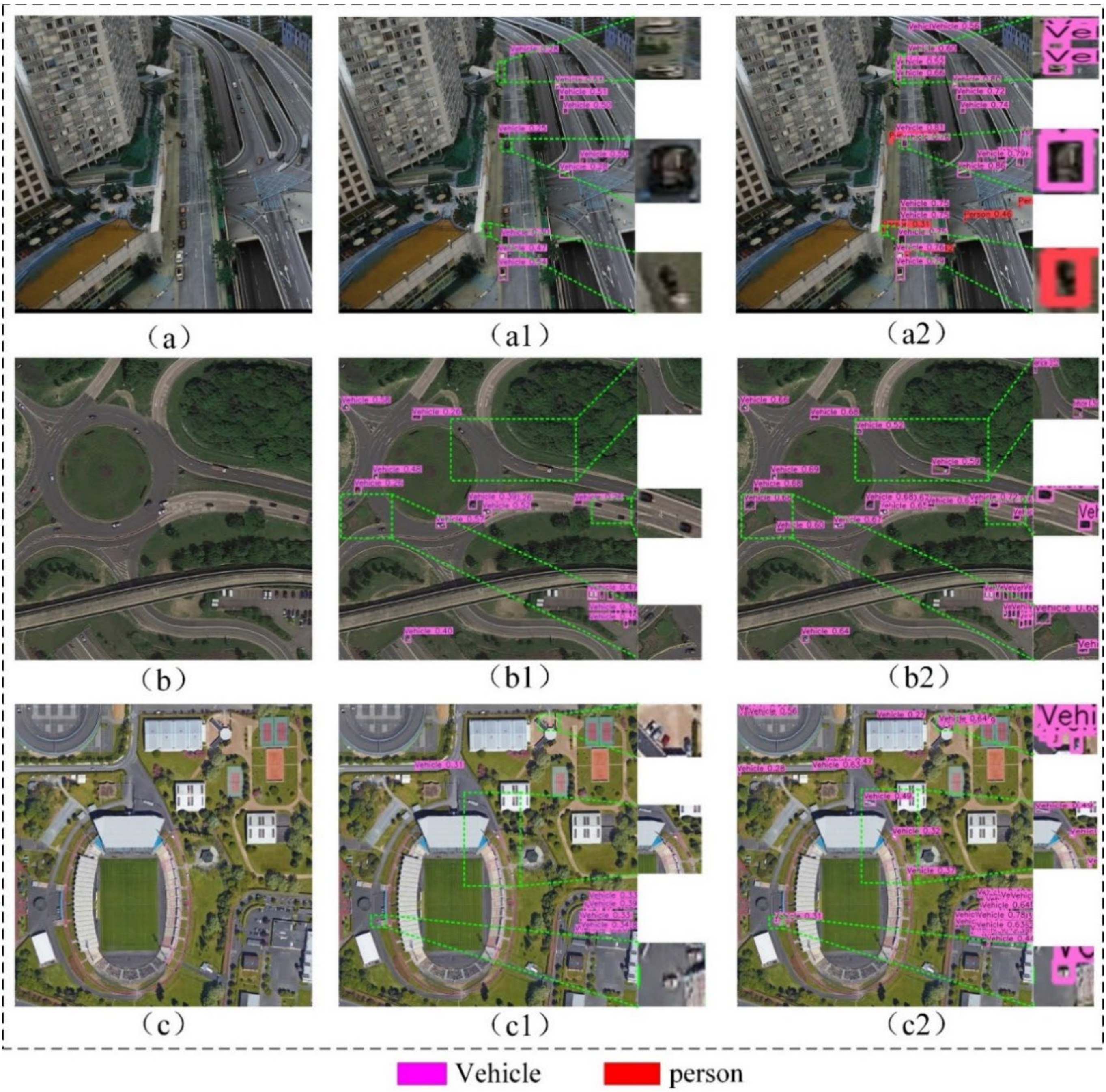

Figure 11. Detection results of YOLOv8s and IA-YOLOv8 on the AI-TOD test dataset. The figures in (**a**–**c**) illustrate the input images. The figures in (**a1**,**b1**,**c1**) present the detection results obtained using YOLOv8s. The figures in (**a2**,**b2**,**c2**) present the detection results obtained using IA-YOLOv8.

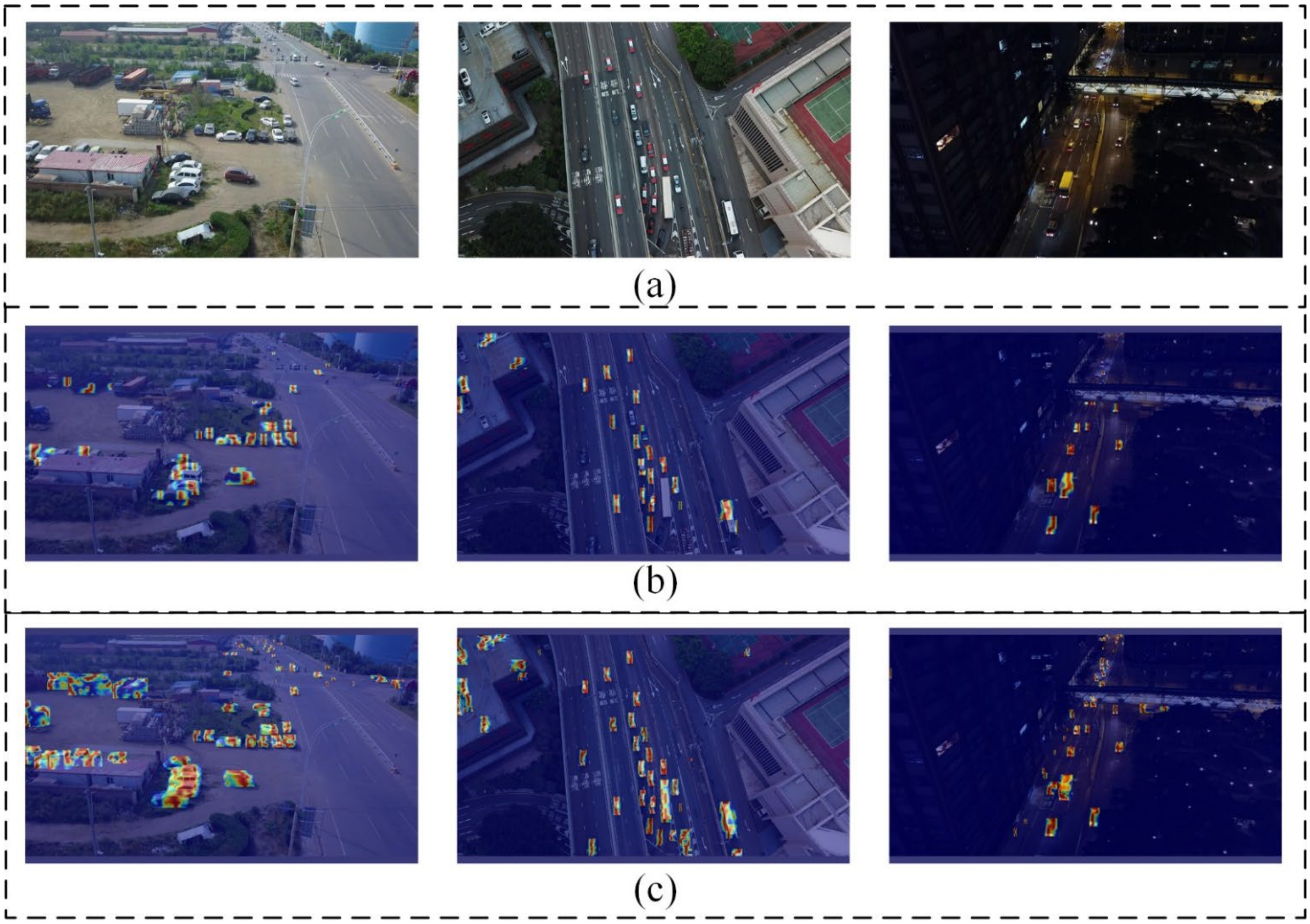

Figure 12. Comparative heatmap analysis of YOLOv9s and IA-YOLOv8 on the Visdrone2019 test set. (**a**) Input images. (**b**) Heatmaps of YOLOv9s. (**c**) Heatmaps of IA-YOLOv8.

5. Conclusions

In this paper, we proposed IA-YOLOv8, a UAV small object detection algorithm that addresses the challenges of identifying low-resolution and indistinct small objects in UAV remote sensing images by incorporating an intra-group multi-scale fusion attention mechanism and adaptive weighted feature fusion mechanism. The algorithm effectively mitigates detail loss associated with the original SPPF's reliance on Max pooling alone by integrating Avg pooling and Max pooling, thereby significantly enhancing the model's capacity to capture and represent small object features. Furthermore, an adaptive feature fusion module combines deep semantic features with shallow detail features to enable a more comprehensive capture of small object characteristics. Additionally, we introduced a lightweight intra-group multi-scale fusion attention module to improve small object feature information while reducing background interference. Experimental results on the Visdrone2019, DIOR, and AI-TOD datasets demonstrate that our IA-YOLOv8 algorithm achieves mAP values of 42.1%, 82.3%, and 39.8%, respectively, requiring only 10.9 MB of parameters, thereby showcasing substantial improvements over the existing object detection algorithms.

Future research will explore the integration of super-resolution techniques to further enhance detection capabilities for such minuscule objects, thereby improving existing models' efficacy in extreme small object detection. Moreover, challenges associated with sample annotation represent another critical factor influencing small object detection performance. The infrequent occurrence of small objects in remote sensing imagery combined with a complex annotation process renders the current datasets insufficiently comprehensive across diverse scenarios. Consequently, we aim to investigate few-shot learning methodologies in future studies to address the issues related to inadequate labeled data and enhance both model generalization and detection performance.

Author Contributions: Conceptualization and methodology, Z.Y.; software and validation, C.W. and J.G.; formal analysis and investigation, N.L. and J.G.; resources, J.S.; data curation, B.G. and Q.W.; writing—original draft preparation, Z.Y.; writing—review and editing, Z.Y.; visualization, N.L. supervision, B.G. and C.W.; project administration and funding acquisition, B.G. and N.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported financially by National Natural Science Foundation of China (Grant No. 62171341).

Data Availability Statement: The Visdrone2019 dataset and the reference codes in this work are available at https://github.com/VisDrone/VisDrone-Dataset (accessed on 28 October 2019). The DIOR dataset and the reference codes in this work are available at http://www.escience.cn/people/gongcheng/DIOR.html (accessed on 1 January 2020).

Acknowledgments: We thank the editor and anonymous reviewers for their suggestions and comments, which helped us to improve the quality of our work.

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

References

1. Li, Z.; Zhang, Y.; Wu, H.; Suzuki, S.; Namiki, A.; Wang, W. Design and application of a UAV autonomous inspection system for high-voltage power transmission lines. *Remote Sens.* **2023**, *15*, 865. [CrossRef]
2. Mohsan, S.A.H.; Othman, N.Q.H.; Li, Y.; Alsharif, M.H.; Khan, M.A.J. Unmanned aerial vehicles (UAVs): Practical aspects, applications, open challenges, security issues, and future trends. *Intell. Serv. Robot.* **2023**, *16*, 109–137. [CrossRef]
3. Yuan, S.; Li, Y.; Bao, F.; Xu, H.; Yang, Y.; Yan, Q.; Zhong, S.; Yin, H.; Xu, J.; Huang, Z. Marine environmental monitoring with unmanned vehicle platforms: Present applications and future prospects. *Sci. Total Environ.* **2023**, *858*, 159741. [CrossRef]
4. Li, X.; Lu, X.; Chen, W.; Ge, D.; Zhu, J. Research on UAVs Reconnaissance Task Allocation Method Based on Communication Preservation. *IEEE Trans. Consum. Electron.* **2024**, *70*, 684–695. [CrossRef]
5. Mahmud, I.; Cho, Y.Z. Detection avoidance and priority-aware object tracking for UAV group reconnaissance operations. *J. Intell. Robot. Syst.* **2018**, *92*, 381–392. [CrossRef]
6. Li, Y.; Zhang, W.; Li, P.; Ning, Y.; Suo, C. A method for autonomous navigation and positioning of UAV based on electric field array detection. *Sensors* **2021**, *21*, 1146. [CrossRef]
7. Iftikhar, S.; Asim, M.; Zhang, Z.; Muthanna, A.; Chen, J.; El-Affendi, M.; Sedik, A.; Abd El-Latif, A. Object detection and recognition for traffic congestion in smart cities using deep learning-enabled UAVs: A review and analysis. *Appl. Sci.* **2023**, *13*, 3995. [CrossRef]
8. Xiong, X.; He, M.; Li, T.; Zheng, G.; Xu, W.; Fan, X.; Zhang, Y. Adaptive Feature Fusion and Improved Attention Mechanism Based Small Object Detection for UAV Object Tracking. *IEEE Internet Things J.* **2024**, *11*, 21239–21249. [CrossRef]
9. Wang, C.; Zhao, R.; Yang, X.; Wu, Q. Research of UAV object detection and flight control based on deep learning. In Proceedings of the 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD), Chengdu, China, 26–28 May 2018; pp. 170–174.
10. Adoni, W.Y.H.; Lorenz, S.; Fareedh, J.S.; Gloaguen, R.; Bussmann, M.J.D. Investigation of autonomous multi-UAV systems for object detection in distributed environment: Current developments and open challenges. *UAVs* **2023**, *7*, 263.
11. Yang, Y.; Guo, B.; Li, C.; Zhi, Y. An improved yolov3 algorithm for pedestrian detection on uav imagery. In Proceedings of the Genetic and Evolutionary Computing: Proceedings of the Thirteenth International Conference on Genetic and Evolutionary Computing, Qingdao, China, 1–3 November 2019; pp. 253–261.
12. Zhang, C.; Zheng, Y.; Guo, B.; Li, C.; Liao, N. SCN: A novel shape classification algorithm based on convolutional neural network. *Symmetry* **2021**, *13*, 499. [CrossRef]
13. Zhang, C.; Li, C.; Guo, B.; Liao, N. Neural Network Compression via Low Frequency Preference. *Remote Sens.* **2023**, *15*, 3144. [CrossRef]
14. Li, C.; Duan, H.J.A.S. Object detection approach for UAVs via improved pigeon-inspired optimization and edge potential function. *Aerosp. Sci. Technol.* **2014**, *39*, 352–360. [CrossRef]
15. Wang, X.; Deng, Y.; Duan, H. Edge-based object detection for unmanned aerial vehicles using competitive Bird Swarm Algorithm. *Aerosp. Sci. Technol.* **2018**, *78*, 708–720. [CrossRef]
16. Sahani, S.K.; Adhikari, G.; Das, B. A fast template matching algorithm for aerial object tracking. In Proceedings of the 2011 International Conference on Image Information Processing, Shimla, India, 3–5 November 2011; pp. 1–6.

17. Zhang, Q.; Duan, H. Chaotic biogeography-based optimization approach to object detection in UAV surveillance. *Optik* **2014**, *125*, 7100–7105. [CrossRef]
18. 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.* **2016**, *39*, 1137–1149. [CrossRef]
19. 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 (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
20. Berg, A.C.; Fu, C.Y.; Szegedy, C.; Anguelov, D.; Erhan, D.; Reed, S.; Liu, W. SSD: Single Shot MultiBox Detector. *arXiv* **2015**, arXiv:1512.02325.
21. Lin, T.-Y.; Goyal, P.; Girshick, R.B.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell* **2017**, *42*, 318–327. [CrossRef]
22. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
23. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
24. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
25. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
26. 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 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.
27. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A single-stage object detection framework for industrial applications. *arXiv* **2022**, arXiv:2209.02976.
28. Reis, D.; Kupec, J.; Hong, J.; Daoudi, A. Real-time flying object detection with YOLOv8. *arXiv* **2023**, arXiv:2305.09972.
29. Wang, C.-Y.; Yeh, I.-H.; Liao, H.-Y.M. Yolov9: Learning what you want to learn using programmable gradient information. *arXiv* **2024**, arXiv:2402.13616.
30. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. Yolov10: Real-time end-to-end object detection. *arXiv* **2024**, arXiv:2405.14458.
31. Tan, L.; Lv, X.; Lian, X.; Wang, G. YOLOv4_UAV: UAV image object detection based on an improved YOLOv4 algorithm. *Comput. Electr. Eng.* **2021**, *93*, 107261. [CrossRef]
32. Shang, J.; Wang, J.; Liu, S.; Wang, C.; Zheng, B. Small object detection algorithm for UAV aerial photography based on improved YOLOv5s. *Electronics* **2023**, *12*, 2434. [CrossRef]
33. Shen, S.; Zhang, X.; Yan, W.; Xie, S.; Yu, B.; Wang, S. An improved UAV object detection algorithm based on ASFF-YOLOv5s. *Math. Biosci. Eng. MBE* **2023**, *20*, 10773–10789. [CrossRef] [PubMed]
34. Li, S.; Liu, C.; Tang, K.; Meng, F.; Zhu, Z.; Zhou, L.; Chen, F. Improved YOLOv5s algorithm for small object detection in UAV aerial photography. *IEEE Access* **2024**, *12*, 9784–9791. [CrossRef]
35. Wang, X.; Wang, A.; Yi, J.; Song, Y.; Chehri, A. Small Object Detection Based on Deep Learning for Remote Sensing: A Comprehensive Review. *Remote Sens.* **2023**, *15*, 3265. [CrossRef]
36. Sapkota, R.; Meng, Z.; Ahmed, D.; Churuvija, M.; Du, X.; Ma, Z.; Karkee, M. Comprehensive Performance Evaluation of YOLOv10, YOLOv9 and YOLOv8 on Detecting and Counting Fruitlet in Complex Orchard Environments. *arXiv* **2024**, arXiv:2407.12040.
37. Sundaresan Geetha, A.; Alif, M.A.R.; Hussain, M.; Allen, P. Comparative Analysis of YOLOv8 and YOLOv10 in Vehicle Detection: Performance Metrics and Model Efficacy. *Vehicles* **2024**, *6*, 1364–1382. [CrossRef]
38. Qiu, X.; Chen, Y.; Cai, W.; Niu, M.; Li, J. LD-YOLOv10: A Lightweight Object Detection Algorithm for UAV Scenarios Based on YOLOv10. *Electronics.* **2024**, *13*, 3269. [CrossRef]
39. Luan, T.; Zhou, S.; Liu, L.; Pan, W. Tiny-Object Detection Based on Optimized YOLO-CSQ for Accurate UAV Detection in Wildfire Scenarios. *Drones* **2024**, *8*, 454. [CrossRef]
40. Wen, L.; Cheng, Y.; Fang, Y.; Li, X. A comprehensive survey of oriented object detection in remote sensing images. *Expert Syst. Appl.* **2023**, *224*, 119960. [CrossRef]
41. Zhang, X.; Zhang, T.; Wang, G.; Zhu, P.; Tang, X.; Jia, X.; Jiao, L. Remote sensing object detection meets deep learning: A metareview of challenges and advances. *IEEE Geosci. Remote Sens. Mag.* **2023**, *11*, 8–44. [CrossRef]
42. Wang, C.; Shi, Z.; Meng, L.; Wang, J.; Wang, T.; Gao, Q.; Wang, E. Anti-occlusion UAV tracking algorithm with a low-altitude complex background by integrating attention mechanism. *Drones* **2022**, *6*, 149. [CrossRef]
43. Tan, L.; Lv, X.; Lian, X.; Wang, G.J.C. YOLOv4_Drone: UAV image target detection based on an improved YOLOv4 algorithm. *Comput. Electr. Eng.* **2021**, *93*, 107261. [CrossRef]
44. Wang, F.; Wang, H.; Qin, Z.; Tang, J. UAV target detection algorithm based on improved YOLOv8. *IEEE Access* **2023**, *11*, 116534–116544. [CrossRef]
45. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv* **2021**, arXiv:2103.14030.
46. Hou, Q.; Zhou, D.; Feng, J. Coordinate attention for efficient mobile network design. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.

47. Snoek, C.G.; Worring, M.; Smeulders, A.W. Early versus late fusion in semantic video analysis. In Proceedings of the 13th Annual ACM International Conference on Multimedia, New York, NY, USA, 6 November 2005; pp. 399–402.
48. Pereira, L.M.; Salazar, A.; Vergara, L. A comparative analysis of early and late fusion for the multimodal two-class problem. *IEEE Access* **2023**, *11*, 84283–84300. [CrossRef]
49. Gadzicki, K.; Khamsehashari, R.; Zetzsche, C. Early vs late fusion in multimodal convolutional neural networks. In Proceedings of the 2020 IEEE 23rd International Conference on Information Fusion (FUSION), Rustenburg, South Africa, 6–9 July 2020; pp. 1–6.
50. Bell, S.; Zitnick, C.L.; Bala, K.; Girshick, R. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2874–2883.
51. Kong, T.; Yao, A.; Chen, Y.; Sun, F. Hypernet: Towards accurate region proposal generation and joint object detection. In Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 845–853.
52. Chaib, S.; Liu, H.; Gu, Y.; Yao, H. Deep feature fusion for VHR remote sensing scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 4775–4784. [CrossRef]
53. Sun, Q.-S.; Zeng, S.-G.; Liu, Y.; Heng, P.-A.; Xia, D.-S. A new method of feature fusion and its application in image recognition. *Pattern Recognit.* **2005**, *38*, 2437–2448. [CrossRef]
54. Dai, Y.; Gieseke, F.; Oehmcke, S.; Wu, Y.; Barnard, K. Attentional feature fusion. In Proceedings of the 2021 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 3–8 January 2021; pp. 3560–3569.
55. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
56. Zhou, L.; Li, Y.; Rao, X.; Liu, C.; Zuo, X.; Liu, Y. Ship object detection in optical remote sensing images based on multiscale feature enhancement. *Comput. Intell. Neurosci.* **2022**, *2022*, 2605140. [CrossRef] [PubMed]
57. Saini, R.; Jha, N.K.; Das, B.; Mittal, S.; Mohan, C.K. Ulsam: Ultra-lightweight subspace attention module for compact convolutional neural networks. In Proceedings of the 2020 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Snowmass, CO, USA, 1–5 March 2020; pp. 1627–1636.
58. 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, New York, NY, USA, 1 October 2016; pp. 516–520.
59. Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 658–666.
60. 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, 2 June 2020; pp. 12993–13000.
61. Tong, Z.; Chen, Y.; Xu, Z.; Yu, R. Wise-IoU: Bounding box regression loss with dynamic focusing mechanism. *arXiv* **2023**, arXiv:2301.10051.
62. Du, D.; Zhu, P.; Wen, L.; Bian, X.; Ling, H.; Hu, Q.; Zheng, J.; Peng, T.; Wang, X.; Zhang, Y. VisDrone-SOT2019: The vision meets UAV single object tracking challenge results. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Seoul, Republic of Korea, 27–28 October 2019; pp. 199–212.
63. Li, K.; Wan, G.; Cheng, G.; Meng, L.; Han, J. Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS J. Photogramm. Remote Sens.* **2020**, *159*, 296–307. [CrossRef]
64. Wang, J.; Yang, W.; Guo, H.; Zhang, R.; Xia, G.-S. Tiny object detection in aerial images. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 3791–3798.
65. Duan, S.; Wang, T.; Li, T.; Yang, W. M-YOLOv8s: An improved small target detection algorithm for UAV aerial photography. *J. Vis. Commun. Image Represent.* **2024**, *104*, 104289. [CrossRef]
66. Ning, T.; Wu, W.; Zhang, J. Small object detection based on YOLOv8 in UAV perspective. *Pattern Anal. Appl.* **2024**, *27*, 103. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.