drones

MDPI

Article

RLRD-YOLO: An Improved YOLOv8 Algorithm for Small Object Detection from an Unmanned Aerial Vehicle (UAV) Perspective

Hanyun Li [1,2], Yi Li [2], Linsong Xiao [3], Yunfeng Zhang [2,*], Lihua Cao [2] and Di Wu [2]

[1] University of Chinese Academy of Sciences, Beijing 100049, China; lihanyun24@mails.ucas.ac.cn
[2] Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China; liyi@ciomp.ac.cn (Y.L.); caolihua@ciomp.ac.cn (L.C.); wudi@ciomp.ac.cn (D.W.)
[3] College of Communication Engineering, Chengdu University of Information Technology, Chengdu 610200, China; xiaolinsong123@outlook.com
* Correspondence: zhangwei_power@126.com

Abstract: In Unmanned Aerial Vehicle (UAV) target detection tasks, issues such as missing and erroneous detections frequently occur owing to the small size of the targets and the complexity of the image background. To improve these issues, an improved target detection algorithm named RLRD-YOLO, based on You Only Look Once version 8 (YOLOv8), is proposed. First, the backbone network initially integrates the Receptive Field Attention Convolution (RFCBAMConv) Module, which combines the Convolutional Block Attention Module (CBAM) and Receptive Field Attention Convolution (RFAConv). This integration improves the issue of shared attention weights in receptive field features. It also combines attention mechanisms across both channel and spatial dimensions, enhancing the capability of feature extraction. Subsequently, Large-Scale Kernel Attention (LSKA) is integrated to further optimize the Spatial Pyramid Pooling Fast (SPPF) layer. This enhancement employs a large-scale convolutional kernel to improve the capture of intricate small target features and minimize background interference. To enhance feature fusion and effectively integrate low-level details with high-level semantic information, the Reparameterized Generalized Feature Pyramid Network (RepGFPN) replaces the original architecture in the neck network. Additionally, a small-target detection layer is added to enhance the model's ability to perceive small targets. Finally, the detecting head is replaced with the Dynamic Head, designed to improve the localization accuracy of small targets in complex scenarios by optimizing for Scale Awareness, Spatial Awareness, and Task Awareness. The experimental results showed that RLRD-YOLO outperformed YOLOv8 on the VisDrone2019 dataset, achieving improvements of 12.2% in mAP@0.5 and 8.4% in mAP@0.5:0.95. It also surpassed other widely used object detection methods. Furthermore, experimental results on the HIT-HAV dataset demonstrate that RLRD-YOLO sustains excellent precision in infrared UAV imagery, validating its generalizability across diverse scenarios. Finally, RLRD-YOLO was deployed and validated on the typical airborne platform, Jetson Nano, providing reliable technical support for the improvement of detection algorithms in aerial scenarios and their practical applications.

Keywords: UAV; small-target detection; YOLOv8n; deep learning

check for updates

Academic Editor: Diego González-Aguilera

Received: 26 February 2025
Revised: 31 March 2025
Accepted: 2 April 2025
Published: 10 April 2025

Citation: Li, H.; Li, Y.; Xiao, L.; Zhang, Y.; Cao, L.; Wu, D. RLRD-YOLO: An Improved YOLOv8 Algorithm for Small Object Detection from an Unmanned Aerial Vehicle (UAV) Perspective. *Drones* **2025**, *9*, 293. https://doi.org/10.3390/drones9040293

Copyright: © 2025 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 deployment of Unmanned Aerial Vehicles (UAVs) for aerial target detection has grown markedly in recent years, propelled by rapid advances in UAV technology. UAVs

are now extensively applied across various domains, such as power line inspection [1], ecological protection [2], and urban traffic monitoring [3]. However, target identification in UAV images is more complex and differs significantly from conventional image processing. Compared to natural scene photos, UAV photographs have a substantially higher percentage of small objects due to differences in flight altitude and the resulting changes in object scale. Additionally, drones operating at low altitudes in complex environments frequently experience target obstruction, resulting in discontinuous and incomplete target representations in imagery. Furthermore, their extensive field of view captures abundant background details, exacerbating noise interference during detection. These factors collectively complicate object detection in drone imagery, contributing to both misdetections and omissions. Thus, creating more accurate target detection algorithms specifically designed for the distinct issues posed by drone aerial imagery is of paramount practical significance.

Currently, object detection algorithms are broadly divided into single-stage or two-stage methods, based on whether they explicitly generate candidate bounding boxes. For instance, two-stage methods, including Fast R-CNN [4], Faster R-CNN [5], and SSD [6], generate candidate regions before performing regression and classification operations on those regions. Nevertheless, due to their relative slowness, two-stage algorithms are not appropriate for continual monitoring in drone applications. In drone-related scenarios necessitating swift inference, single-stage detectors typically provide superior speed performance. However, they frequently exhibit reduced accuracy with smaller objects, needing additional modifications. The You Only Look Once (YOLO) [7] family of designs is renowned due to its remarkable efficacy and has been utilized in diverse detection scenarios. However, optimizing YOLO for accurately identifying small objects in aerial images continues to pose a major challenge in ongoing research.

Targets with a width or height smaller than 32 pixels are typically referred to as small targets. These targets are challenging to detect due to their weak feature representation and tendency to be overwhelmed by the background. This is especially true in aerial long-distance monitoring, where detection algorithms need strong feature extraction and multi-scale perception capabilities. Huang et al. [8] introduced an enhanced Path-Aggregation-Network-based feature pyramid to enhance the model's representational capacity and feature fusion abilities. Nevertheless, the overall performance improvement remained constrained. Tang et al. [9] introduced YOLO-RSFM, optimized for tiny target detection and multi-scale detection. This architecture integrates a Transformer Decoder Head to improve the alignment between actual and predicted boxes. Nonetheless, while this method effectively diminishes mistakes in dense object detection, it leads to a substantial rise in model parameters. Taheer et al. [10] introduced the PVswin-YOLOv8 model, which combines YOLOv8 with the Swin Transformer for boosting detection efficacy in drone imagery. Additionally, this model employs the Soft-NMS technique to more effectively handle intersecting bounding boxes. Nonetheless, identifying small objects remains challenging and requires further refinement.

Despite significant progress in small-target detection, existing methods struggle to optimize information fusion and effectively leverage small-target feature information, particularly in complex scenarios. This limitation increases the likelihood of missed or incorrect detections. To improve the difficulties posed by small-target detection in complex scenarios, this work introduces an enhanced YOLOv8n model. The main contributions of this paper are summarized as follows:

(1) A novel Receptive Field Attention Convolution (RFCBAMConv) module is introduced to replace standard convolutions and refine the C2f module within the backbone. By integrating the strengths of the Convolutional Block Attention Module (CBAM) [11] and Receptive Field Attention Convolution (RFAConv) [12], this com-

ponent significantly boosts the network's feature extraction capacity. Furthermore, the Spatial Pyramid Pooling Fast (SPPF) layer employs Large Separable Kernel Attention (LSKA) [13], which broadens the overall feature-map perception and enhances multi-scale target processing.

(2) The neck component of the model utilizes the Reparameterized Generalized Feature Pyramid Network (RepGFPN) [14] in place of the original architecture. RepGFPN effectively integrates deep semantic information with shallow, detailed features, improving feature complementarity across different levels. Additionally, a small-target detection layer is integrated into RepGFPN to strengthen both classification and localization for smaller objects.

(3) The original detection head is replaced with a Dynamic Head (Dyhead) [15], which improves performance in complex environments and accommodates diverse target features by dynamically adjusting parameters. Through the improvement of scale perception, spatial awareness, and task-specific comprehension, this adjustment increases the model's capacity to recognize small targets.

(4) Experimental results on the VisDrone2019 [16] dataset show that RLRD-YOLO significantly outperforms YOLOv8n in detection accuracy. Compared to other leading algorithms, RLRD-YOLO achieves superior performance on key metrics, validating its effectiveness. Additionally, generalization experiments on an infrared UAV dataset were conducted to validate the reliability of RLRD-YOLO in various scenarios.

2. Related Work

Drones play a crucial role in remote sensing due to their exceptional mobility, rapid deployment, and extensive surveillance capabilities. However, the presence of complex backgrounds and blurred target edges makes object detection in drone imagery particularly challenging [17]. Deep-learning-based targeting algorithms have made impressive strides in UAV small target recognition in recent times. Because of its effective immediate response and high detection accuracy, the YOLO series has drawn plenty of attention. The following section provides a detailed overview of YOLO algorithms and their improvement strategies.

2.1. YOLO Detection Algorithm

YOLOv1, first introduced in 2016, provides a robust framework for target detection. By refining its network architecture and loss function, it achieved notable gains in detection accuracy, enabling reliable target identification even in high-noise environments and complex scenes. YOLOv5 [18], introduced in 2020, has become known as one of the most extensively utilized variations within the YOLO family. This approach employs a characteristic pyramid structure with three predictive characteristic maps and mosaic data enhancement at the input stage to make multi-scale detection more reliable.

Ultralytics introduced YOLOv8 [19] in 2023, integrating several advanced technologies, such as the C2f module, the SilU activation function, and the anchor-free mechanism. These enhancements significantly improve the model's feature fitting capabilities. Furthermore, the incorporation of the Feature Pyramid Network (FPN) [20] and Path Aggregation Network (PAN) [21] strengthens YOLOv8's robustness and accuracy across diverse detection scenarios. The model also incorporates a decoupled head to boost detection adaptability. Consequently, YOLOv8 serves as the baseline for experiments in this research. Figure 1 illustrates its architecture.

(1) Backbone Network: YOLOv8 employs the C2f module, an improvement over the earlier CSPDarkNet53 (C3) [22]. By merging outputs from all bottleneck modules, the C2f module accelerates training and improves the extraction of key features.

(2) Neck Network: The neck network is critical for facilitating feature fusion across different levels. YOLOv8 integrates FPN and PAN in its architecture, enabling the more effective extraction and fusion of information from multi-resolution feature maps.

(3) Head Network: YOLOv8's head shifts from an anchor-based scheme to an anchor-free design, employing a decoupled architecture that separately handles classification and regression, thereby boosting overall performance. In addition, the loss function is refined by applying binary cross-entropy for classification, alongside distribution focal loss [23] and Complete Intersection over Union [24] for regression, ultimately enhancing both accuracy and efficiency.

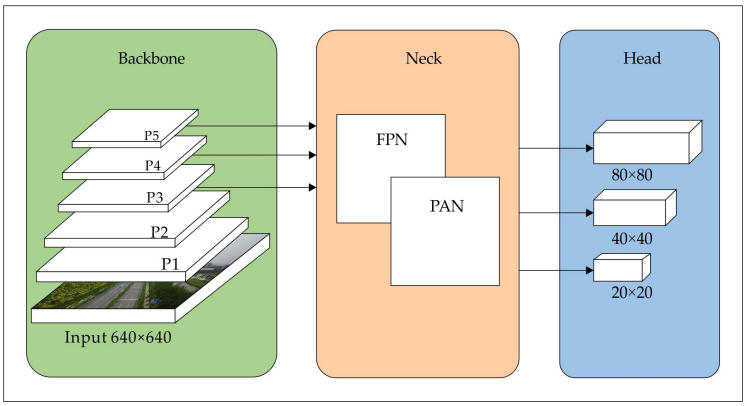


Figure 1. YOLOv8 network architecture.

2.2. Improvement of YOLO Detection Algorithm

The YOLO series has achieved remarkable success in the field of target detection. However, detecting small targets and handling complex scenes remain challenging. To tackle these challenges and further enhance YOLO's accuracy and efficiency, researchers have proposed several advancements across multiple aspects. YOLO optimization is divided into three primary parts: the backbone, neck and head. Enhancing these modules is crucial for augmenting YOLO's overall performance. This section outlines the enhancements implemented in each module and their impacts.

(1) Backbone Network Improvement: Zhong et al. [25] modified the backbone by replacing traditional stride convolutions and pooling layers with SPD-Conv [26] layers. This adjustment significantly reduced fine-grained information loss and improved feature representation. Liu et al. [27] proposed an efficient spatio-temporal interaction module aimed at enhancing spatial feature preservation, augmenting network depth, and proficiently capturing extensive target information. Zhou et al. [28] incorporated the DualC2f module into the backbone. Leveraging group convolution for efficient filter arrangement, this method enhances inter-channel communication and information retention, significantly increasing model accuracy. Xiao et al. [29] presented the Efficient Multi-scale Convolution Module (EMCM) to strengthen the C2f architecture. By incorporating a Multi-scale Attention Module into EMCM's multi-branch structure, C2f's capacity to extract features in complex environments is further improved. Although improvements to the backbone have enhanced feature extraction capabilities, they may also result in increased computational expenses and problems such as overfitting or poor feature discrimination when detecting small targets.

(2) Neck Network Improvement: Li et al. [30] proposed the Bi-PAN-FPN framework to enhance the neck of YOLOv8s, aiming to improve detection performance. Nie et al. [31] introduced the HPANet architecture as an alternative to the original PANet, improving its capacity to amalgamate feature maps across varying scales while decreasing

the number of network parameters. Li et al. [32] introduced an innovative architecture to address the shortcomings of conventional Feature Pyramid Networks when handling dimension variations. Through the enhancement of the feature fusion approach, MPA-FPN mitigates conflicts among nonadjacent features and amplifies interactions between low-level and high-level information. Xiao et al. [33] introduced the Multi-Scale Ghost Generalized Feature Pyramid Network (MSGFPN) to improve the combination of feature map data across various scales while reducing its parameter count. Zhao et al. [34] integrated the Efficient Multi-Scale Attention (EMA) [35] into the neck, improving the model's sensitivity to both spatial and textural information. While neck enhancements improve feature fusion, they may also introduce feature redundancy, potentially increasing confusion between targets and noise in complex backgrounds.

(3) Head Network Improvement: Zhang et al. [36] tackled the issue of the network's insensitivity to aspect ratio similarity between the assumption and reality in aerial imagery by optimizing the detection head. The basic model's CIoU was substituted by the enhanced loss function, inner-MPDIoU, for improving the recognition performance of small target features. Peng et al. [37] replaced the detection head of YOLOv8 with a lightweight, multi-attention Dynamic Head. This modification replaces the original convolution with FasterConv, reducing the detection head's weight and improving its efficiency in leveraging feature maps that integrate both local and global information. Li et al. [38] utilized a head decoupling strategy in conjunction with MAM, which enabled the classification task to focus on semantic information and the localization task to emphasize boundary information. Consequently, the detection performance was enhanced. Lin et al. [39] introduced the normalized Wasserstein distance for training, with the objective of mitigating the susceptibility of IoU-based metrics to small object localization bias. The enhancements to the detection head have improved classification and localization skills for small targets. Nevertheless, intricate backgrounds may still result in erroneous and overlooked detections, and multi-target scenarios may limit the generalization ability.

Although numerous extant methods have enhanced the precision of small-target detection, they continue to be constrained by obstacles such as complex backgrounds and target occlusions. Each method exhibits its own limitations. To overcome these challenges, integrating multiple methodologies and designing a network structure tailored to specific scenarios is essential.

3. Methods

This paper introduces RLRD-YOLO to tackle challenges such as dense target distributions, large-scale variations in targets, and complex background interference in UAV target detection. Figure 2 depicts the architecture of the RLRD-YOLO.

In the backbone network, the RFCBAMConv was initially incorporated to improve its feature extraction capabilities. Subsequently, the LSKA was integrated into the SPPF, allowing the model to capture more information using a large-scale convolutional kernel. In the neck network, the original Feature Pyramid Network was replaced with the RepGFPN module, improving the model's multi-scale feature fusion capabilities. Additionally, a small-object detection layer was introduced to enhance the capture of minor targets. Finally, the Dynamic Head optimizes the detection head by integrating and reinforcing multi-scale information through an attention mechanism.

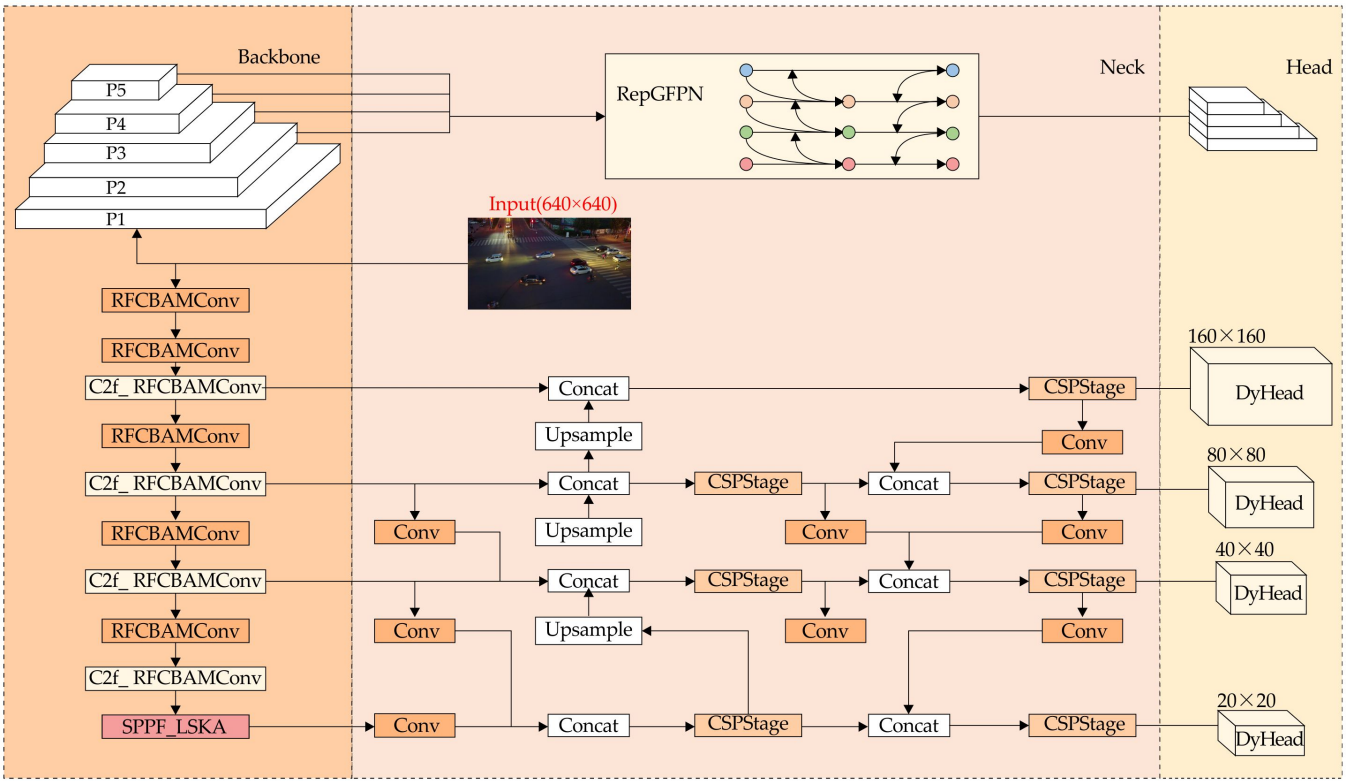


Figure 2. Overall structure of the RLRD-YOLO.

3.1. Improvements to the Backbone

3.1.1. Improvement of C2F

In UAV detection tasks, the model necessitates enhanced feature extraction abilities to address intricate situations and the difficulties associated with identifying small targets. Traditional convolutions frequently fail to effectively identify essential features in the presence of multi-scale targets and background noise. To enhance detection accuracy and reliability, this paper introduces the RFCBAMConv into the YOLOv8, replacing standard convolutions and optimizing the C2f module.

RFAConv mitigates the limitations of parameter learning in standard convolutional kernels by expanding attention to spatial features within its receptive field. As a result, it notably boosts overall performance when substituted for standard convolutions, all while keeping parameter size and computational cost low. Meanwhile, CBAM introduces attention weighting across both spatial and channel dimensions during convolutions. By embedding CBAM's spatial attention into the receptive field, RFAConv yields RFCBAMConv. This design, which captures extensive information dependencies akin to self-attention, further elevates the module's performance in feature extraction.

The enhanced RFCBAMConv optimizes the functionality of the attention mechanism and reduces computational complexity in comparison to the original CBAM module, as illustrated in Figure 3. Unlike CBAM, which applies channel and spatial attention sequentially, RFCBAMConv processes them simultaneously, enhancing feature fusion efficiency and focusing more precisely on critical features. To improve efficiency, RFCBAMConv integrates Squeeze-and-Excitation (SE) [40], replacing CBAM's channel attention and eliminating the average pooling branch to minimize redundant computations.

The enhanced C2f module, illustrated in Figure 4, includes two convolutional layers. The first uses a standard convolution, while the second replaces the conventional convolution with the RFCBAMConv module. This design enhances the expressiveness of receptive field features, reduces redundancy, and improves the network's overall efficiency.

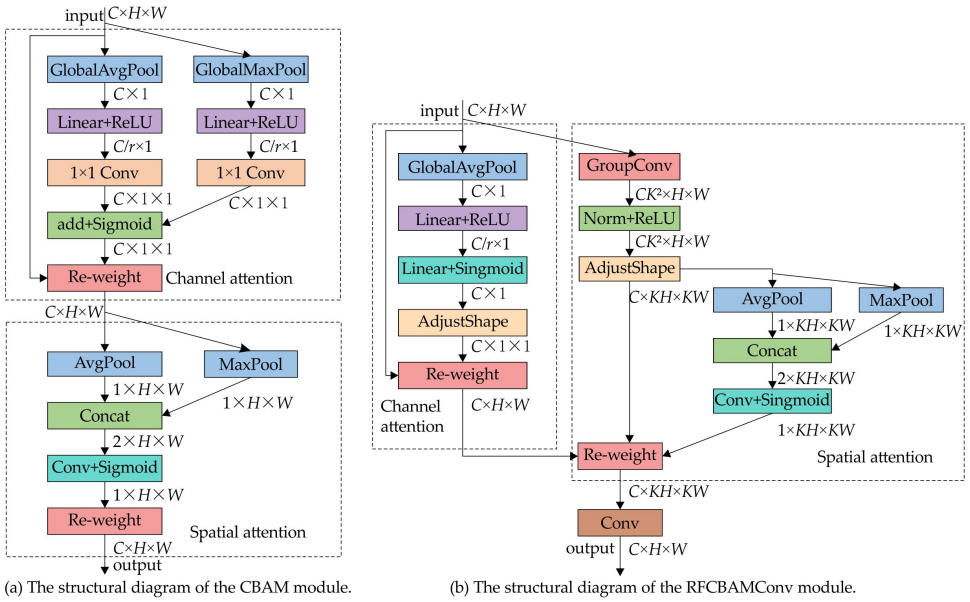


Figure 3. Comparing the architectures of CBAM and RFCBAMConv. (**a**) describes the structural diagram of the CBAM module; (**b**) shows the structural diagram of the RFCBAMConv module, where SE is used to replace the channel attention mechanism of CBAM.

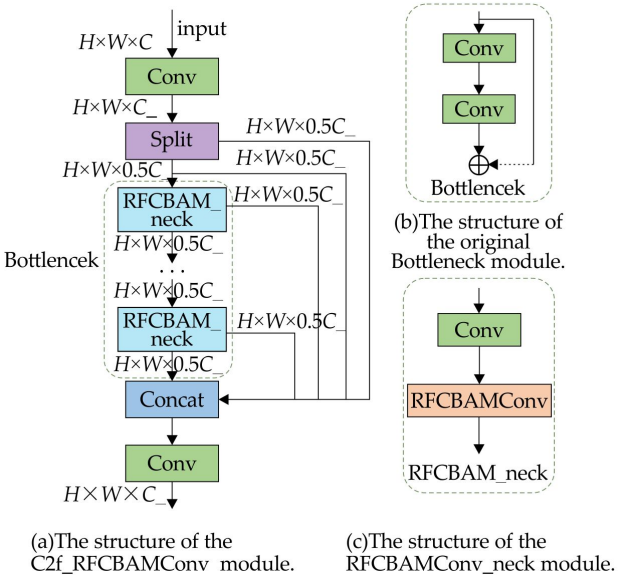


Figure 4. Structure of C2f_RFCBAMConv module.

3.1.2. Improvement of SPPF

In UAV aerial imaging applications, the original model may suffer from partial spatial information loss and misdetections due to complex image features and the presence of small targets. To improve these problems, RLRD-YOLO incorporates the LSKA into the SPPF of the backbone, strengthening the model's focus on critical feature information.

As depicted in Figure 5, LSKA is a separable kernel attention mechanism that captures long-range relationships and enhances adaptability by breaking down large-kernel convolutions. Specifically, the two-dimensional convolution kernel is separated into one-dimensional horizontal and vertical parts in a predefined order. Consequently, standard depthwise convolution (DW-C-Conv) and its dilated counterpart (DW-D-Conv) are each broken down into separate depthwise and dilated depthwise operations, lowering both computational overhead and memory demands.

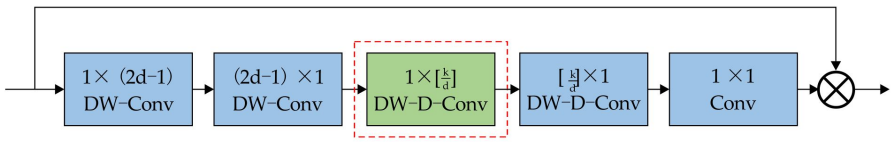


Figure 5. Structure of LSKA module.

Equations (1)–(4) display the LSKA output formula.

$$\bar{Z}^c = \sum_{H,W} W^c_{(2d-1)\times 1} * \left(\sum_{H,W} W^c_{(2d-1)\times 1} * F^c \right), \tag{1}$$

$$Z^c = \sum_{H,W} W^c_{(2d-1)\times 1} * \left(\sum_{H,W} W^c_{\lfloor k/d \rfloor} * \bar{Z}^c \right), \tag{2}$$

$$A^c = W_{1\times 1} * Z^c, \tag{3}$$

$$\bar{F}^c = A^c \otimes F^c, \tag{4}$$

In the equations, F^c denotes the input feature map, where C specifies the number of input channels. $\bar{Z}^c$ refers to the output after depthwise convolution, while Z^c represents the output of the dilated convolution. The summation symbols under H and W signify that the operation is performed across the height and width of the feature map, respectively. The notation $[.]$ denotes a downsampling operation. The parameter d specifies the dilation rate, and k indicates the kernel size. The symbol $*$ denotes the convolution operation. A^c denotes the attention map, while F^c represents the output feature map. The symbol $\otimes$ denotes the Hadamard product. As per the formula, the initial step in the LSKA attention mechanism involves applying a one-dimensional convolution kernel for spatially separable convolution. Next, the resulting feature maps are used to create the attention map, which is subsequently applied to the original feature map.

The LSKA module is placed after all max pooling layers and before the second convolutional layer. The input first moves through a convolutional layer, followed by three consecutive max pooling operations. Their outputs are then merged via concatenation, and the resulting tensor is directed to the LSKA module. Once processed by LSKA, the output proceeds to the final convolutional layer. Figure 6 illustrates the improved SPPF module. Integrating the LSKA attention mechanism into the SPPF layer introduces negligible parameters while significantly enhancing receptive field information.

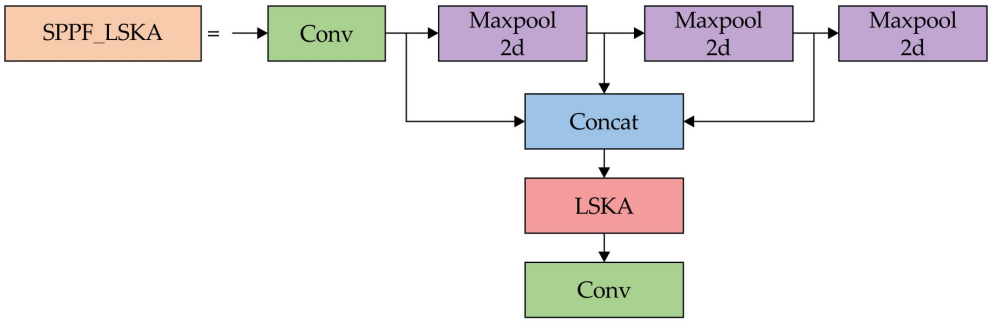


Figure 6. Structure of the SPPF_LSKA module.

3.2. Improvement of Neck

The PAN-FPN architecture of YOLOv8 reduces feature loss through the integration of multi-scale features. Nonetheless, PAN-FPN is constrained to feature aggregation among proximate scales during fusion, neglecting interactions between remote scales.

To improve this issue, we introduce RepGFPN to augment cross-scale feature combination and guarantee the effective merging of features across various scales.

The primary objective of RepGFPN is to employ varying channel volumes to depict characteristics at diverse scales while adhering to lightweight computational limitations. RepGFPN improves real-time performance compared to GFPN by removing the extra upsampling process in queen fusion. Meanwhile, Cross-Stage Partial Network (CSPNet) [41] is utilized to substitute the prior 3×3 convolution for feature fusion, as illustrated in Figure 7. After the Concat operation, the input is split into two pathways: one pathway undergoes 1×1 convolution, while the other is processed by the ELAN feature-combining component, comprising RepConv3×3 convolutions and 3×3 convolutions. Finally, the two branches are ultimately combined using Concat and subsequently processed through a convolution to produce the final feature map.

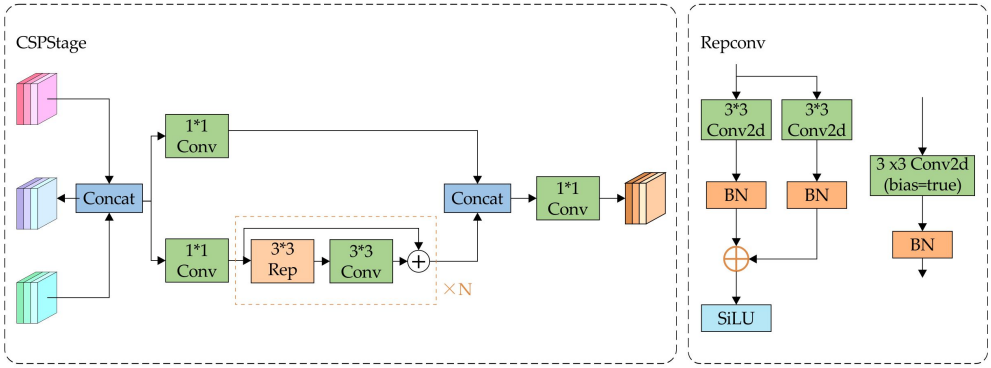


Figure 7. Structure of the CSPNet module. (In the figure, ∗ represents the multiplication sign, and N represents the number of N items).

In UAV-based target detection, the combination of low-altitude operations and long imaging distances results in small targets occupying only a tiny fraction of the available image pixels, often leading to their being overlooked or misclassified. To address this, RLRD-YOLO introduced a specialized small-target detection layer within RepGFPN. By adjusting anchor boxes specifically for smaller targets and leveraging a high-resolution feature map, this layer achieves more precise detection and localization. Figure 8 illustrates the integration of the RepGFPN structure with the small-target detection layer. By fusing feature information across multiple layers, this design boosts detection performance for small targets.

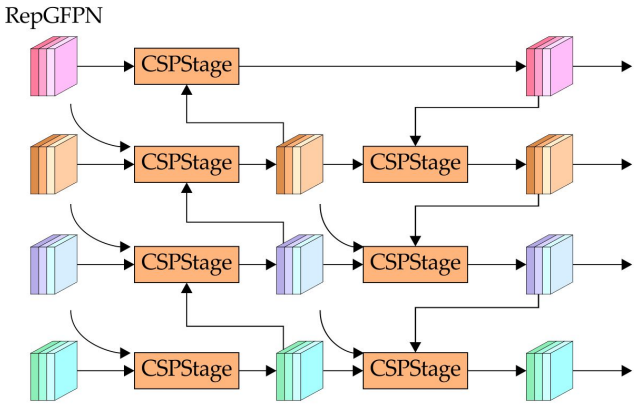


Figure 8. Structure of RepGFPN module.

3.3. Improvement of Head

Although YOLOv8 separates classification and detection tasks through a decoupled head design, accurately identifying and localizing small objects remains problematic because of the broad perceptual scope and limited resolution of aerial images. To address this, we present a Dynamic Head module that enhances the detection head's feature representation. By employing an attention-based mechanism to merge multiple detection head architectures, the Dynamic Head increases the model's flexibility and overall effectiveness. Specifically, it incorporates scale-aware attention for robust multi-scale target detection, spatial-aware attention for more precise localization, and task-aware Attention to further boost performance across diverse tasks.

As illustrated in Figure 9, three perceptual enhancement modules are applied to different locations. By integrating the three attentional features, an optimized output is produced.

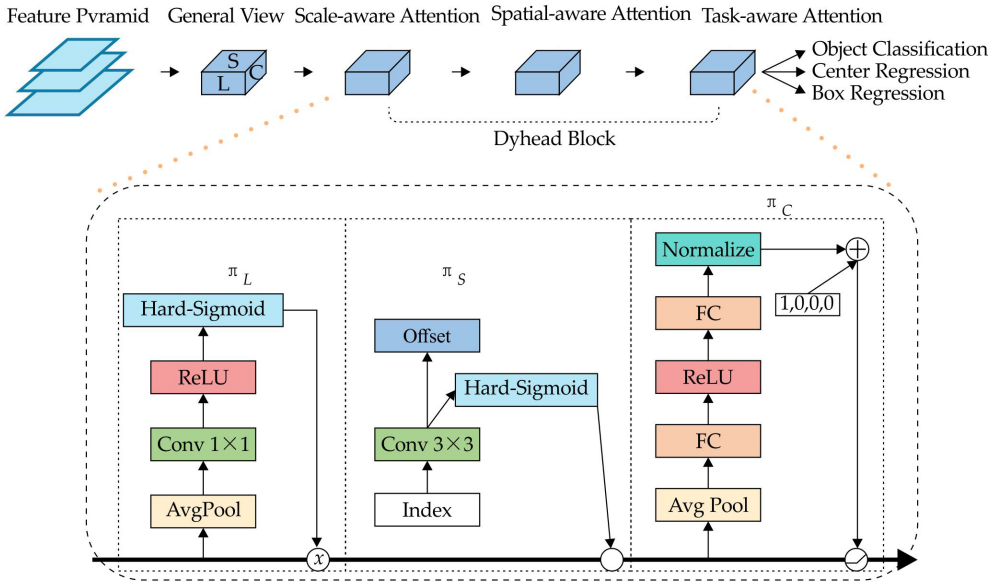


Figure 9. Structure of the dynamic head module.

Enhanced output:

$$W(F) = \pi_C(\pi_S(\pi_L(F) \cdot F) \cdot F) \cdot F, \tag{5}$$

Here, π_L, π_S, and π_C indicate the attention functions acting on the L, S, and C dimensions, respectively. The computation formulas are given as follows:

(1) Scale-aware attention π_L

$$\pi_L(F) \cdot F = \sigma\left(f\left(\frac{1}{SC}\sum_{SC} F\right)\right) \cdot F, \tag{6}$$

Here, $f(\cdot)$ denotes the linear operation approximated by the convolution layer, while $\sigma(x) = \max\left(0, \min\left(1, \frac{x+1}{2}\right)\right)$ represents the hard-sigmoid function.

(2) Spatial-aware attention π_S

$$\pi_S \cdot F = \frac{1}{L}\sum_{l=1}^{L}\sum_{k=1}^{K} w_{l,k} \cdot (l; p_k + \Delta p_k; c) \cdot \Delta mk, \tag{7}$$

Here, K denotes the number of sparse sampling positions, and $p_k + \Delta p_k$ represents the offset position obtained through learnable spatial shifts, which is used to distinguish the attention regions. Δmk represents the learnable importance factor at position p_k. Both parameters are inferred from the input feature map F.

(3) Task-aware attention π_C

$$\pi_C(F) \cdot F = \max\left(\alpha^1(F) \cdot F_C + \beta^1(F), \alpha^2(F) \cdot F + \beta^2(F)\right), \quad (8)$$

Here, F_C represents the feature map of the C-th channel, and $\left[\alpha^1, \alpha^2, \beta^1, \beta^2\right]^T = \theta(\cdot)$ represents the activation control values learned through the function $\theta(\cdot)$. The function $\theta(\cdot)$ begins by applying global average pooling across each dimension, then processes the pooled features through two fully connected layers and a normalization step, and ultimately uses a shifted sigmoid to constrain the output within $[-1, 1]$. The three aforementioned attention mechanisms are each applied in turn, adhering to the same formulation. Furthermore, multiple instances of the formulation in Equation (5) can be iteratively stacked to integrate multiple π_L, π_S, and π_C blocks together.

4. Experimental Design and Analysis of Results

4.1. Experimental Data

In this research, we employ the VisDrone2019 dataset compiled by the AISKYEYE team at Tianjin University under varying lighting and weather conditions using different drone platforms. With over 2.6 million manually annotated bounding boxes and points of interest, it covers ten categories. As illustrated in Figure 10, the distribution of these categories and their aspect ratios reveals an imbalanced dataset, predominantly consisting of small targets. This trait makes VisDrone2019 both particularly valuable and challenging for small-scale object detection research.

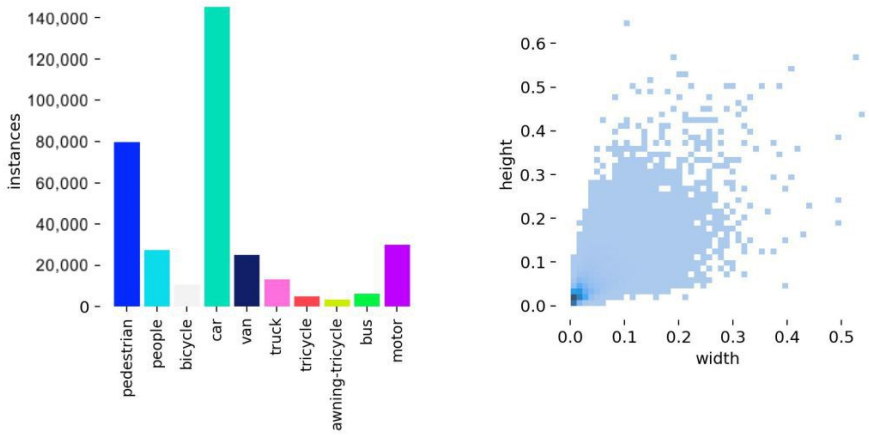


Figure 10. The number of species, instances, and scale sizes in the VisDrone2019 dataset.

4.2. Experimental Equipment

The paper employs the following hyperparameters: an input image size of 640 × 640, 300 epochs for training, and a batch size of 8. Table 1 summarizes additional details about the experimental environment employed for this work.

Table 1. Experimental environment.

Component	Specification
CPU	18 vCPU AMD EPYC 9754 128-Core Processor
GPU	RTX 4090D
PyTorch	PyTorch 1.13.1
CUDA	CUDA 12.1
Python	Python 3.8

4.3. Experimental Evaluative Metrics

In the experiment, Precision (P), Recall (R), Average Precision (AP), and mean Average Precision (mAP) were selected as model evaluation metrics. For object detection performance, Intersection over Union (IoU) is used to calculate mAP at the thresholds of 0.5 and 0.5:0.95, where higher values indicate better detection performance. Equations (9)–(12) show the calculation methods.

$$P = \frac{TP}{TP + FP}, \tag{9}$$

$$R = \frac{TP}{TP + FN}, \tag{10}$$

$$AP = \int_0^1 PR\, dR, \tag{11}$$

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i, \tag{12}$$

In this context, True Positive (TP) represents the count of targets correctly identified by the model, while False Positive (FP) refers to the number of non-targets wrongly classified as targets, and False Negative (FN) indicates the number of targets incorrectly labeled as non-targets.

4.4. Comparison Experiment

Table 2 indicates that the RLRD-YOLO attains a Precision that is 11.9% higher and a mAP@0.5 that is 12.2% greater than YOLOv8n. To visually demonstrate the model's improvements, both the YOLOv8n and the RLRD-YOLO were tested on the same experimental platform, as depicted in Figure 11. During the same training phase, RLRD-YOLO demonstrated significant improvements in all metrics compared to the YOLOv8n. This highlights that RLRD-YOLO is better at learning comprehensive feature information, leading to more effective improvements in detection accuracy.

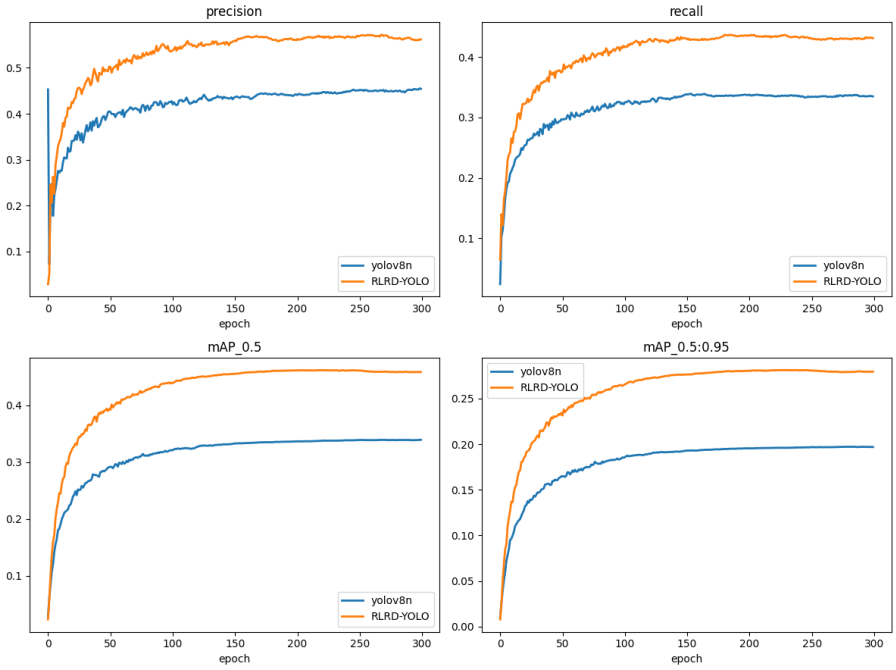


Figure 11. Comparison of the YOLOv8n and RLRD-YOLO metrics for different epochs.

Table 2. Comparison of the YOLOv8n and RLRD-YOLO metrics.

Model	P/%	R/%	mAP@0.5/%	mAP@0.5:0.95/%	Params/M	FLOPs/G
YOLOv8n	44.9	33.7	33.9	19.7	3.0	8.1
Ours	56.8	43.2	46.1	28.1	4.0	27.4

To assess the efficacy of RLRD-YOLO, comparative experiments were performed with the YOLO series, as illustrated in Table 3. The YOLOv3 version contains a more intricate architecture and elevated computational demands, rendering it inappropriate for resource-limited application contexts. YOLOv5 and YOLOv8 have garnered substantial attention as a result of their exceptional performance in the subsequent YOLO series algorithms. YOLOv5 is distinguished by its efficient performance and lightweight network structure, while YOLOv8 enhances detection accuracy and robustness through further model optimizations. The RLRD-YOLO achieves a favorable equilibrium between model effectiveness and precision. Compared to YOLOv5s, the model parameters were reduced by 3.2 M, while mAP@0.5 and mAP@0.5:0.95 increased by 9.4% and 6.5%, respectively. RLRD-YOLO surpasses YOLOv8s in all dimensions, encompassing model parameters, computing efficiency, and detection accuracy. Specifically, RLRD-YOLO improves mAP@0.5 by 1.7% over YOLOv8l, while its model size is only 9.2% of that of YOLOv8l. Furthermore, compared to the newly introduced YOLOv9 [42], YOLOv10 [43] and YOLOV11 series in 2024, RLRD-YOLO achieves superior performance metrics while using fewer parameters.

Table 3. Comparison of experimental metrics of the YOLO series algorithms on the VisDrone2019 dataset.

Model	P/%	R/%	mAP@0.5/%	mAP@0.5:0.95/%	Params/M	FLOPs/G
YOLOv3	56.2	42.3	44.9	27.6	103.7	283.0
YOLOv5n	43.9	31.1	32.5	18.7	1.8	4.2
YOLOv5s	46.3	37.2	36.7	21.6	7.2	15.8
YOLOv8s	50.9	38.1	39.3	23.5	11.2	28.5
YOLOv8m	52.2	41.4	42.5	25.8	25.9	78.7
YOLOv8l	56.0	42.9	44.4	27.3	43.6	164.9
YOLOv9s	50.2	38.9	39.5	23.5	7.2	26.7
YOLOv10s	51.4	38.2	40.0	24.0	8.1	24.6
YOLOv11s	50.9	38.8	40.3	24.2	9.4	21.4
RLRD-YOLO	56.8	43.2	46.1	28.1	4.0	27.4

This research additionally contrasts with contemporary current techniques for small-target detection in aerial photography utilizing the VisDrone 2019 dataset. For a rigorous comparison, we selected both classic model improvement algorithms and the latest advancements in the YOLO series. Among these, ARFP+Faster-RCNN [44] is an enhanced version of the two-stage Faster-RCNN algorithm; GBS-YOLOv5 [27] and DFE-Net [38] are recent improvements to YOLOv5; Improve YOLOv7 [36] represents the latest enhancement to YOLOv7-tiny; and YOLO-RSFM [9], PVswin-YOLOv8s [10], LGFF-YOLO [37], Subtle-YOLOv8 [34], and DDSC-YOLO [28] are recent developments based on YOLOv8. Table 4 summarizes the experimental results of various advanced algorithms. A "-" denotes that the relevant metric is not documented in its original source.

ARFP is an adaptive recursive feature pyramid model incorporated into the two-stage detection framework Faster-RCNN. However, excessive recursion in ARFP compromises the network's overall detection performance, rendering it less effective than RLRD-YOLO. Compared to the improved YOLOv5-based algorithm GBS-YOLOv5, RLRD-YOLO achieves a 10.8% improvement in mAP@0.5 and an 8.1% improvement in mAP@0.5:0.95. Compared to DFE-Net, RLRD-YOLO achieves improvements of 7% in mAP@0.5 and 4.2% in mAP@0.5:0.95 while reducing the parameter count by 15.7M. When compared with the latest improved YOLOv8 series algorithms, RLRD-YOLO demonstrates improvements

across all performance indicators, with precision achieving a maximum improvement of 3.9%, recall achieving a maximum improvement of 6.8%, and mAP@0.5 achieving a maximum improvement of 10%, all while maintaining a minimal parameter count. Although its computational cost surpasses that of certain other methods, its overall efficacy renders it especially appropriate for drone-based target detection tasks.

Table 4. Comparison of different advanced algorithms.

Model	P/%	R/%	mAP@0.5/%	mAP@0.5:0.95/%	Params/M	FLOPs/G
ARFP+Faster-RCNN	45.2	35.8	26.4	15.0	41.3	207
YOLO-RSFM	53.5	36.4	35.1	20.2	7.1	29.0
PVSwim-YOLOv8	54.5	41.8	43.3	24.6	21.6	-
GBS-YOLOv5	41.4	35.5	35.3	20.0	-	-
DDSC-YOLO	52.9	40.3	42.2	25.5	4.99	-
Subtle-YOLOv8	54.4	43.4	45.1	26.6	11.8	72.9
Improve YOLOv7	51.2	42.1	41.5	23.6	6.39	21.0
LGFF-YOLO	-	-	43.5	22.9	4.15	12.4
DFE-Net	48.4	39.1	39.1	23.9	19.7	19.7
RLRD-YOLO	56.8	43.2	46.1	28.1	4.0	27.4

To further demonstrate the advantages of the RLRD-YOLO, we compares it with various YOLO series algorithms and other enhanced models across different categories. Table 5 summarizes the comparative results. RLRD-YOLO achieved the highest detection performance across eight categories—pedestrian, people, car, van, bicycle, tricycle (Tri), bus, and motor—with mAP@0.5 values of 52.0%, 42.4%, 18.8%, 84.3%, 50.9%, 33.7%, 65.8%, and 53.9%, respectively, and attained the highest average detection accuracy of 46.1%. RLRD-YOLO also performs well in detecting categories with large horizontal and vertical dimensions, such as trucks and buses, achieving mAP@0.5 scores of 40.8% and 65.8%, respectively. Compared to the baseline model, this represents an improvement of 10.8% and 17.3%, respectively. Overall, the RLRD-YOLO algorithm exhibits exceptional proficiency in object detection across diverse scenarios.

Table 5. Comparison of metrics across different categories.

Model	Pedestrian	People	Bicycle	Car	Van	Truck	Tri	Awn-Tri	Bus	Motor	mAP@0.5
YOLOv5s	39.7	31.6	11.5	77.3	41.4	33.0	24.9	13.5	53.1	41.8	36.7
YOLOv8n	35.5	28.1	8.6	76.4	39.2	30.0	22.7	12.2	48.5	38.0	33.9
YOLOv8s	42.5	32.5	12.6	79.2	44.7	36.1	28.8	17.0	55.5	44.5	39.3
YOLOv8m	46.6	36.3	15.5	81.2	47.6	40.8	30.9	17.5	59.9	48.3	42.5
YOLOv9s	43.9	32.5	12.9	79.8	45.0	36.5	28.5	17.0	55.7	44.3	39.5
YOLOv10s	43.6	34.4	14.7	80.3	45.3	37.0	28.7	16.2	54.3	46.0	40.0
YOLO-RSFM	36.3	25.4	12.3	76.1	34.6	34.1	23.7	17.3	52.3	39.2	35.1
PVSwim-YOLOv8s	45.9	35.7	16.4	81.5	49.1	**42.4**	32.8	17.7	62.9	48.2	43.3
GBS-YOLOv5	42.3	32.8	8.8	78.2	39.0	30.9	22.2	11.7	47.7	39.1	35.3
DDSC-YOLO	48.9	40.3	15.1	83.1	47.8	35.0	27.6	16.3	59.1	48.6	42.2
Improve YOLOv7	45.3	41.7	13.1	83.8	45.3	38.5	25.6	**20.7**	53.3	47.7	41.5
LGFF-YOLO	40.9	31.2	12.3	78.7	44.6	35.3	27.1	14.8	55.9	42.1	38.3
DFF-Net	46.4	38.1	14.9	79.6	42.4	35.4	25.9	13.3	51.6	45.0	39.3
RLRD-YOLO	**52.0**	**42.4**	**18.8**	**84.3**	**50.9**	40.8	**33.7**	18.6	**65.8**	**53.9**	**46.1**

In comparison to the YOLOv8n model, RLRD-YOLO enhances the mAP@0.5 metrics for smaller object categories, such as pedestrian, bicycle, and motor, by 16.5%, 10.2%, and 15.9%, respectively. It also shows varying degrees of improvement when compared to larger models, such as YOLOv8m. Compared to recent advanced algorithms in the literature, such as YOLO-RSFM, GBS-YOLOv5, DDSC-YOLO, LGFF-YOLO, and DFE-Net, RLRD-YOLO achieves the highest mAP@0.5 across 10 target categories, with average detection accuracy improvements of 11%, 10.8%, 3.9%, 7.8%, and 6.8%, respectively. Compared to the improved YOLOv7-Tiny, RLRD-YOLO achieves a 4.6% rise in average mAP and

delivers superior mAP@0.5 performance across nine target categories. Furthermore, RLRD-YOLO surpasses the PVswin-YOLOv8s algorithm with a 2.8% improvement in average detection accuracy. It also enhances recognition of tiny objects in categories like pedestrians, people, bicycles, and motor, with improvements of 6.1%, 6.7%, 2.4%, and 5.7%, respectively. The experimental results confirm that the four enhancements introduced by RLRD-YOLO effectively capture the fine details of small targets, yielding the highest overall efficiency.

4.5. Ablation Experiment

To assess the influence of the four proposed improvements on small-scale target detection in UAV aerial imagery, we conducted a series of ablation studies using the YOLOv8 model. The first row in Table 6 shows the detection metrics for YOLOv8n without any enhancements, while a "✓" indicates the application of a specific improvement method.

Table 6. Ablation experiment.

RFCBAMConv	SPPF_LSKA	RepGFPN	Dyhead	mAP@50/%	mAP@50:95/%	Params/M	FLOPs/G
				33.9	19.7	3.0	8.1
✓				35.1	20.6	3.1	8.7
✓	✓			35.5	20.9	3.2	9.2
✓	✓	✓		42.3	25.4	3.2	19.7
✓	✓	✓	✓	46.1	28.1	4.0	27.4

Table 6 demonstrates that integrating the RFCBAMConv module into YOLOv8n increased mAP@0.5 and mAP@0.5:0.95 by 1.2% and 0.9%, respectively, underlining the effectiveness of receptive field-based attention for improving detection precision. After adding the SPPF_LSKA module, mAP@0.5 reached 35.5%, and mAP@0.5:0.95 reached 20.9%. The SPPF_LSKA module employs large-scale convolutional kernels to capture small object details. This improves the performance of finding small objects in backgrounds with a lot of information. Overall, these enhancements to the backbone architecture boost detection performance while adding only a minimal computational overhead.

As a result, the cross-layer feature fusion module RepGFPN replaced the original frameworks by incorporating a specialized small-target detection layer. This improvement enhances the fusion of high-level semantic information with low-level detailed information and improves the localization of small targets. As a result, the mAP@0.5 and mAP@0.5:0.95 values increased by 6.8% and 4.5%, respectively. However, the addition of small object detection layers requires higher-resolution feature maps and cross-scale feature fusion, thereby increasing the computational complexity. After the addition of the Dynamic Head module, it dynamically adjusts the feature extraction process, enabling the model to better adapt to the features of small targets in complex scenes, thereby improving the accuracy of target recognition. However, the introduction of this module requires additional computational resources and parameters to achieve the dynamic adjustment of the feature extraction process. Although these innovations increase the computational complexity of the model, their impact on performance improvement is significant.

4.6. Generalization Experiment

To evaluate the stability and adaptability of the RLRD-YOLO across diverse scenarios and conditions, a generalization comparison test was performed using the HIT-HAV [45] dataset. The dataset consists of over 2800 infrared thermal images captured by drones in diverse environments, such as schools and parking lots and includes targets such as persons, vehicles, and bicycles. Due to insufficient data in the "car" and "other vehicle" categories, the dataset was refined following the methodology outlined in [46]. Specifically, the DontCare category was removed, and "car" was consolidated with other vehicle types

under a single "vehicle" category. As a result, the dataset was restructured into three distinct categories: person, vehicle, and bicycle.

Table 7 demonstrates that the RLRD-YOLO achieves outstanding results for both mAP@0.5 and mAP@0.5:0.95. Compared to the YOLOv8, the RLRD-YOLO improves mAP@0.5 and mAP@0.5:0.95 by 2.4% and 3.8%, respectively.RLRD-YOLO enhances mAP@0.5 by 0.6% relative to the YOLO-ViT method, while possessing just 23% of YOLO-ViT's parameter count. The results demonstrate the superior generalizability of the RLRD-YOLO, consistently delivering high accuracy across diverse scenarios and conditions.

Table 7. Comparison of different algorithms on the HIT-UAV dataset.

Model	mAP@0.5/%	mAP@0.5:0.95/%	Params/M	FLOPs/G
YOLOv3-tiny	86.2	53.1	12.1	19.0
YOLOv5n	93.0	59.0	1.8	4.2
YOLOv5s	93.9	61.0	7.2	15.8
YOLOv8n	92.7	59.0	3.0	8.1
YOLOv8s	94.6	61.5	11.2	28.5
YOLOv9s	93.9	61.3	7.2	26.7
YOLOv10n	93.4	59.6	2.7	8.4
YOLOv10s	94.7	61.6	8.1	24.6
YOLOv11n	93.4	60.2	2.6	6.3
YOLOv11s	94.6	62.1	9.4	21.4
YOLO-ViT [46]	94.5	-	17.3	33.1
RLRD-YOLO	95.1	62.8	4.0	27.4

4.7. Embedded Porting of Models

To verify the adaptability of the proposed RLRD-YOLO model on edge computing devices, this study deployed the trained model on a Jetson Nano platform for testing. Despite its compact size, the Jetson Nano is equipped with 16 tensor cores and 512 NVIDIA Ampere GPU cores, providing high computational power and low power consumption. These features make it well suited for real-time object detection tasks on embedded devices. The detection results for this device are shown in Figure 12.

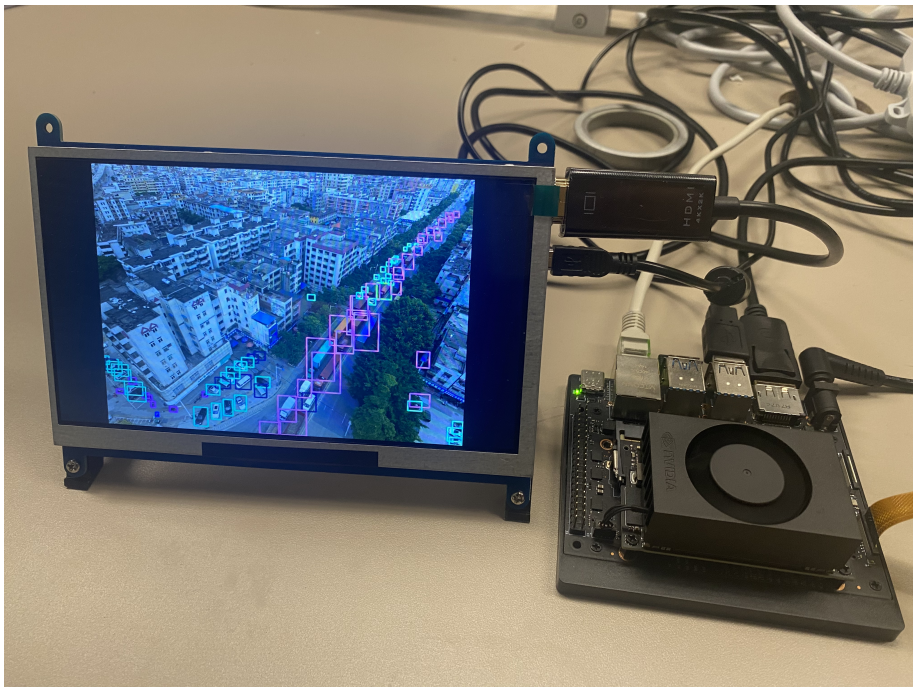


Figure 12. Schematic Diagram of the RLRD-YOLO Detection on Jetson Nano.

As shown in Table 8, compared to the baseline model YOLOv8-n, the proposed RLRD-YOLO achieves higher detection accuracy and more effectively captures target details and

features in complex scenes. Although its inference speed decreases with improved accuracy, the frame rate of RLRD-YOLO remains high enough for real-time detection, making it suitable for most embedded scenarios. Overall, RLRD-YOLO demonstrates a favorable trade-off between accuracy and computational overhead, underscoring its balance between practicality and performance.

Table 8. Quantitative comparison of algorithms on Jetson Nano.

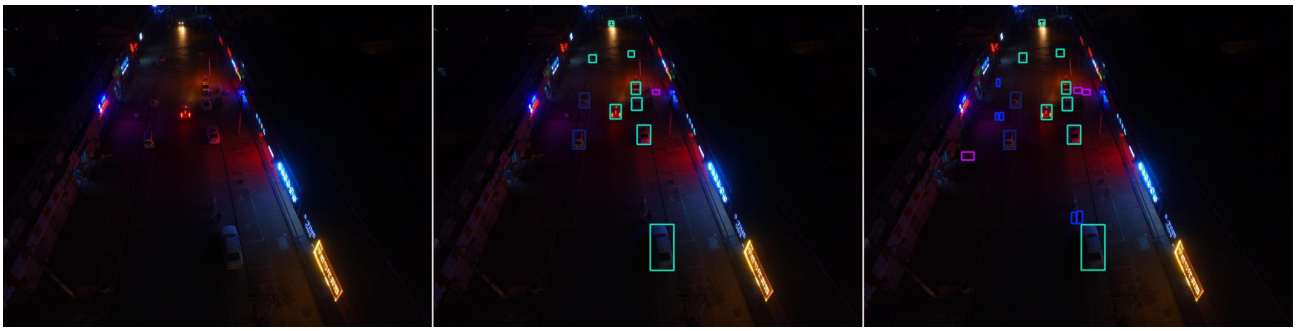
Method	mAP@50/%	mAP@50:95/%	Params (M)	FPS
YOLOv8-n	33.9	19.7	3.0	39.2
YOLOv8-s	39.3	23.5	11.2	19.3
YOLOv11-n	34.4	19.9	2.6	40.9
RLRD-YOLO	46.1	28.1	4.0	21.7

4.8. Visual Analysis

To assess the efficacy of the RLRD-YOLO in practical circumstances, images from various times and conditions within the VisDrone 2019 test set were selected for evaluation. The test scenarios consist of four representative settings: dimly lit streets at night, traffic roads at night, traffic scenes surrounded by complex buildings, and pedestrian roads in densely built environments. Additionally, two representative scenarios, a traffic road and a school playground, were selected from the HIT-HAV dataset for testing. The detection outcomes are presented in Figure 13. The left panel displays the original image, the middle panel shows YOLOv8n's detection results, and the right panel provides the outcomes produced by RLRD-YOLO.

In the nocturnal small-target scenario depicted in Figure 13a, the YOLOv8n exhibits difficulty in reliably detecting targets in low-light areas. In contrast, the RLRD-YOLO detects targets in low-light environments with superior accuracy and efficiency. In the nocturnal traffic scenario shown in Figure 13b, the baseline algorithm is susceptible to noise interference and has difficulty accurately detecting obstructed or blurred vehicles. In contrast, RLRD-YOLO exhibits superior robustness and achieves higher accuracy in detecting these targets. In Figure 13c, the baseline algorithm misclassifies numerous trucks as cars and exhibits multiple instances of missed target detections. In Figure 13d, the baseline incorrectly categorizes pedestrians as motorcycles and exhibits numerous issues with missed pedestrian detections. The RLRD-YOLO algorithm enhances feature extraction and effectively suppresses complex background noise, leading to improved detection accuracy. In Figure 13c, this improvement corrects previous misclassification errors, such as mistaking trucks for cars, and significantly reduces missed detections. Additionally, in Figure 13d, the enhanced algorithm corrects the misclassification of pedestrians as motorcycles and substantially improves pedestrian detection ability.

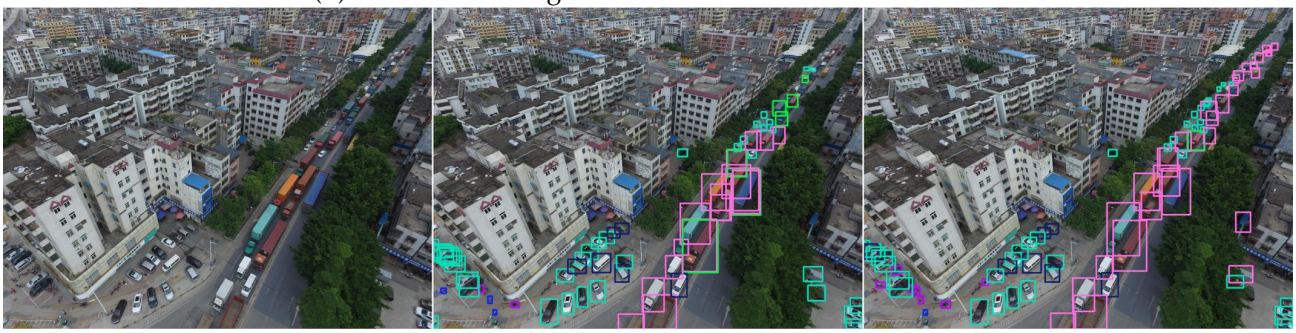
Figure 13e,f present the results of infrared thermal imaging detection captured through drone aerial photography. In Figure 13e, it is evident that the baseline algorithm suffers from detection leakage with vehicles, while Figure 13f displays a similar issue with pedestrians. In contrast, RLRD-YOLO accurately identifies these targets. Overall, the RLRD-YOLO algorithm demonstrates superior recognition capabilities when processing images of small objects and complex backgrounds from a drone's perspective, markedly decreasing erroneous results and overlooked detections.



(**a**) Dimly lit streets at night



(**b**) Traffic road at night



(**c**) Traffic scenes amidst complex buildings



(**d**) Pedestrian roads under complex building environments



(**e**) Traffic road

Figure 13. *Cont*.



(**f**) School playground

Figure 13. Detection effect comparison chart.

5. Conclusions

Aerial image target detection from UAV perspectives faces challenges such as varying target scales and complex backgrounds. This study proposes a modified YOLOv8-based target detection algorithm, RLRD-YOLO, to increase detection performance. The model's feature extraction capability is enhanced by incorporating RFCBAMConv, which optimizes the Conv and C2f components. Furthermore, integrating Large Separable Convolutional Kernels into the SPPF structure improves the model's capacity to extract multi-scale information. In the neck network, RepGFPN was implemented to enable the effective integration of low-level features with high-level semantic information. To overcome challenges related to small object scales and positioning difficulties, an additional detection layer was incorporated. In the head network, the DyHead was used to further enhance the model's performance. In experiments utilizing the VisDrone 2019 dataset, RLRD-YOLO exhibited a distinct superiority over alternative models. Furthermore, on the HIT-HAV infrared drone aerial imagery dataset, RLRD-YOLO maintained high detection performance, further confirming its generalization ability. To assess the algorithm's adaptability in real-world applications, we deployed it on the Jetson Nano for testing. The model demonstrated a strong balance between accuracy and computational efficiency. Subsequent studies will concentrate on diminishing computing complexity while enhancing the precision of small target detection.

Author Contributions: Conceptualization: H.L.; Methodology: L.C.; Software: H.L. and L.X.; Validation: H.L. and D.W.; Formal Analysis: L.X.; Investigation: H.L.; Resources: Y.L.; Data Curation: D.W.; Writing—Original Draft Preparation: H.L.; Writing—Review and Editing: L.C. and Y.L.; Visualization: L.X.; Supervision: Y.Z.; Project Administration: Y.Z. All authors have read and approved the final version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

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

References

1. Shi, W.; Lyu, X.; Han, L. SONet: A Small Object Detection Network for Power Line Inspection Based on YOLOv8. *IEEE Trans. Power Deliv.* **2024**, *39*, 2973–2984. [CrossRef]
2. He, Y.; Li, J. TSRes-YOLO: An Accurate and Fast Cascaded Detector for Waste Collection and Transportation Supervision. *Eng. Appl. Artif. Intell.* **2023**, *126*, 106997. [CrossRef]
3. Bakirci, M. Enhancing Vehicle Detection in Intelligent Transportation Systems via Autonomous UAV Platform and YOLOv8 Integration. *Appl. Soft Comput.* **2024**, *164*, 112015. [CrossRef]

4. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448. [CrossRef]
5. 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**, *39*, 1137–1149. [CrossRef] [PubMed]
6. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the Computer Vision—ECCV 2016, Amsterdam, The Netherlands, 11–14 October 2016; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer: Cham, Switzerland, 2016; pp. 21–37.
7. Redmon, J. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016.
8. Huang, T.; Zhu, J.; Liu, Y.; Tan, Y. UAV Aerial Image Target Detection Based on BLUR-YOLO. *Remote Sens. Lett.* **2023**, *14*, 186–196. [CrossRef]
9. Tang, P.; Ding, Z.; Lv, M.; Jiang, M.; Xu, W. YOLO-RSFM: An Efficient Road Small Object Detection Method. *IET Image Process.* **2024**, *18*, 4263–4274. [CrossRef]
10. Tahir, N.U.A.; 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**, *8*, 84. [CrossRef]
11. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
12. Zhang, X.; Liu, C.; Yang, D.; Song, T.; Ye, Y.; Li, K.; Song, Y. RFAConv: Innovating Spatial Attention and Standard Convolutional Operation. *arXiv* **2023**, arXiv:2304.03198.
13. Lau, K.W.; Po, L.M.; Rehman, Y.A.U. Large Separable Kernel Attention: Rethinking the Large Kernel Attention Design in CNN. *Expert Syst. Appl.* **2024**, *236*, 121352. [CrossRef]
14. Xu, X.; Jiang, Y.; Chen, W.; Huang, Y.; Zhang, Y.; Sun, X. Damo-YOLO: A Report on Real-Time Object Detection Design. *arXiv* **2022**, arXiv:2211.15444.
15. Dai, X.; Chen, Y.; Xiao, B.; Chen, D.; Liu, M.; Yuan, L.; Zhang, L. Dynamic Head: Unifying Object Detection Heads with Attentions. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 7373–7382.
16. Du, D.; Zhu, P.; Wen, L.; Bian, X.; Lin, H.; Hu, Q.; Peng, T.; Zheng, J.; Wang, X.; Zhang, Y.; 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 (ICCV) Workshops, Seoul, Republic of Korea, 27 October–2 November 2019.
17. Wang, Y.; Li, H.; Li, X.; Wang, Z.; Zhang, B. UAV Image Target Localization Method Based on Outlier Filter and Frame Buffer. *Chin. J. Aeronaut.* **2024**, *37*, 375–390. [CrossRef]
18. Jocher, G. YOLOv5 by Ultralytics. 2020. Available online: https://github.com/ultralytics/yolov5 (accessed on 9 June 2020).
19. Jocher, G.; Chaurasia, A.; Qiu, J. YOLOv5 by Ultralytics. 2023. Available online: https://github.com/ultralytics/ultralytics/yolov8 (accessed on 10 January 2023).
20. Lin, T.Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
21. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
22. Redmon, J. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
23. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12993–13000.
24. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Keypoint Triplets for Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6569–6578.
25. Zhong, R.; Peng, E.; Li, Z.; Ai, Q.; Han, T.; Tang, Y. SPD-YOLOv8: A Small-Size Object Detection Model of UAV Imagery in Complex Scene. *J. Supercomput.* **2024**, *80*, 1–21. [CrossRef]
26. Sunkara, R.; Luo, T. No More Strided Convolutions or Pooling: A New CNN Building Block for Low-Resolution Images and Small Objects. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Grenoble, France, 19–23 September 2022; Springer: Cham, Switzerland, 2022; pp. 443–459.
27. Liu, H.; Duan, X.; Lou, H.; Gu, J.; Chen, H.; Bi, L. Improved GBS-YOLOv5 Algorithm Based on YOLOv5 Applied to UAV Intelligent Traffic. *Sci. Rep.* **2023**, *13*, 9577. [CrossRef]
28. Zhou, S.; Zhou, H. Detection Based on Semantics and a Detail Infusion Feature Pyramid Network and a Coordinate Adaptive Spatial Feature Fusion Mechanism Remote Sensing Small Object Detector. *Remote Sens.* **2024**, *16*, 2416. [CrossRef]

29. Xiao, L.; Li, W.; Zhang, X.; Jiang, H.; Wan, B.; Ren, D. EMG-YOLO: An Efficient Fire Detection Model for Embedded Devices. *Digit. Signal Process.* **2025**, *156*, 104824. [CrossRef]
30. Li, Y.; Fan, Q.; Huang, H.; Han, Z.; Gu, Q. A Modified YOLOv8 Detection Network for UAV Aerial Image Recognition. *Drones* **2023**, *7*, 304. [CrossRef]
31. Nie, H.; Pang, H.; Ma, M.; Zheng, R. A Lightweight Remote Sensing Small Target Image Detection Algorithm Based on Improved YOLOv8. *Sensors* **2024**, *24*, 2952. [CrossRef]
32. Li, Z.; He, Q.; Zhao, H.; Yang, W. Doublem-net: Multi-scale spatial pyramid pooling-fast and multi-path adaptive Feature Pyramid Network for UAV detection. *Int. J. Mach. Learn. Cybern.* **2024**, *15*, 5781–5805.
33. Xiao, L.; Li, W.; Yao, S.; Liu, H.; Ren, D. High-Precision and Lightweight Small-Target Detection Algorithm for Low-Cost Edge Intelligence. *Sci. Rep.* **2024**, *14*, 23542. [CrossRef]
34. Zhao, S.; Chen, J.; Ma, L. Subtle-YOLOv8: A Detection Algorithm for Tiny and Complex Targets in UAV Aerial Imagery. *Signal Image Video Process.* **2024**, *18*, 8949–8964. [CrossRef]
35. Ouyang, D.; He, S.; Zhang, G.; Luo, M.; Guo, H.; Zhan, J.; Huang, Z. Efficient Multi-Scale Attention Module with Cross-Spatial Learning. In Proceedings of the ICASSP 2023—IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes, Greece, 4–9 June 2023; pp. 1–5. [CrossRef]
36. Zhang, Z.; Xie, X.; Guo, Q.; Xu, J. Improved YOLOv7-Tiny for Object Detection Based on UAV Aerial Images. *Electronics* **2024**, *13*, 2969. [CrossRef]
37. Peng, H.; Xie, H.; Liu, H.; Guan, X. LGFF-YOLO: Small Object Detection Method of UAV Images Based on Efficient Local-Global Feature Fusion. *J. Real-Time Image Process.* **2024**, *21*, 167. [CrossRef]
38. Li, H.; Ling, L.; Li, Y.; Zhang, W. DFE-Net: Detail Feature Extraction Network for Small Object Detection. *Vis. Comput.* **2024**, *40*, 8853–8866. [CrossRef]
39. Lin, Y.; Li, J.; Shen, S.; Wang, H.; Zhou, H. GDRS-YOLO: More Efficient Multiscale Features Fusion Object Detector for Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* **2024**, *21*, 6008505. [CrossRef]
40. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
41. Wang, C.Y.; Liao, H.Y.M.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A New Backbone That Can Enhance Learning Capability of CNN. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 390–391.
42. Wang, C.Y.; Yeh, I.H.; Liao, H.Y.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. In Proceedings of the Computer Vision—ECCV 2024, Cham, Switzerland, 7–13 October 2024; Leonardis, A., Ricci, E., Roth, S., Russakovsky, O., Sattler, T., Varol, G., Eds.; Springer Nature Switzerland: Cham, Switzerland, 2025; pp. 1–21.
43. 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.
44. Wang, J.; Yu, J.; He, Z. ARFP: A Novel Adaptive Recursive Feature Pyramid for Object Detection in Aerial Images. *Appl. Intell.* **2022**, *52*, 12844–12859. [CrossRef]
45. Suo, J.; Wang, T.; Zhang, X.; Chen, H.; Zhou, W.; Shi, W. HIT-UAV: A High-Altitude Infrared Thermal Dataset for Unmanned Aerial Vehicle-Based Object Detection. *Sci. Data* **2023**, *10*, 227. [CrossRef]
46. Zhao, X.; Xia, Y.; Zhang, W.; Zheng, C.; Zhang, Z. YOLO-ViT-Based Method for Unmanned Aerial Vehicle Infrared Vehicle Target Detection. *Remote Sens.* **2023**, *15*, 3778. [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.