electronics

MDPI

Article

GLDS-YOLO: An Improved Lightweight Model for Small Object Detection in UAV Aerial Imagery

Zhiyong Ju, Jiacheng Shui * and Jiameng Huang

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China; juzy@usst.edu.cn (Z.J.); 232260501@st.usst.edu.cn (J.H.)
* Correspondence: 232260509@st.usst.edu.cn

Abstract

To enhance small object detection in UAV aerial imagery suffering from low resolution and complex backgrounds, this paper proposes GLDS-YOLO, an improved lightweight detection model. The model integrates four core modules: Group Shuffle Attention (GSA) to strengthen small-scale feature perception, Large Separable Kernel Attention (LSKA) to capture global semantic context, DCNv4 to enhance feature adaptability with reduced parameters, and further proposes a novel Small-object-enhanced Multi-scale and Structure Detail Enhancement (SMSDE) module, which enhances edge-detail representation of small objects while maintaining lightweight efficiency. Experiments on VisDrone2019 and DOTA1.0 demonstrate that GLDS-YOLO achieves superior detection performance. On VisDrone2019, it improves mAP@0.5 and mAP@0.5:0.95 by 12.1% and 7%, respectively, compared with YOLOv11n, while maintaining competitive results on DOTA. These results confirm the model's effectiveness, robustness, and adaptability for complex small object detection tasks in UAV scenarios.

Keywords: small object detection; YOLOv11; deformable convolution; edge enhancement; spatial pyramid pooling

Academic Editor: Yue Wu

Received: 22 August 2025
Revised: 25 September 2025
Accepted: 25 September 2025
Published: 27 September 2025

Citation: Ju, Z.; Shui, J.; Huang, J. GLDS-YOLO: An Improved Lightweight Model for Small Object Detection in UAV Aerial Imagery. *Electronics* **2025**, *14*, 3831. https://doi.org/10.3390/electronics14193831

1. Introduction

With the rapid development and widespread deployment of unmanned aerial vehicle (UAV) technology in both civil and industrial domains, aerial imaging has become an essential means of acquiring ground information. Owing to their flexible mobility, high-altitude perspectives, and high-resolution imaging capabilities, UAVs demonstrate notable advantages in scenarios such as disaster rescue, environmental monitoring, agricultural management, and public security [1]. However, in many applications the targets are small, sparsely distributed, and weakly featured; detecting such small objects has therefore become a key challenge in UAV aerial imaging. Small-object detection in UAV imagery presents unique complexities: on the one hand, small objects occupy only a few pixels, exhibit limited feature representation, and are easily overwhelmed by cluttered backgrounds; on the other hand, UAV acquisition is frequently affected by illumination variation, dynamic backgrounds, and viewpoint changes, further increasing the difficulty of detection. Consequently, how to efficiently and accurately detect and recognize small objects in complex scenes has become an important research topic in computer vision.

Current object-detection approaches can be broadly grouped into two categories. The first comprises traditional detectors based on manually designed features [2,3], which offer strong interpretability and modest computational demands but have limited representation

capacity and struggle with large-scale data and complex scenes. The second comprises deep learning–based detectors, which possess powerful feature-extraction ability and broad generalization and have become the mainstream research direction. These methods are typically divided into two-stage and one-stage paradigms. Two-stage detectors (e.g., R-CNN, Fast R-CNN and Faster R-CNN [4–6]) first generate region proposals and then perform classification and localization, achieving high accuracy but incurring substantial model complexity and computational cost that hinder efficient deployment in resource-constrained UAV scenarios. One-stage detectors (e.g., YOLO [7] and SSD [8]) omit the proposal stage and directly predict categories and locations, offering high speed and efficiency; they have demonstrated clear advantages for small-object detection in UAV imagery.

2. Related Work

In recent years, advances in deep learning have enabled notable progress in small-object detection based on convolutional neural networks (CNNs). Yang et al. [9] introduced a global spatial attention structure to capture global information; although it alleviates the loss of small-object features, it exhibits a relatively high false-positive rate for visually similar targets. To address mismatches between small-object predictions and ground-truth boxes, Wei et al. [10] replaced CIoU with the Focal-EIoU loss. Building upon Mask R-CNN, Zhou et al. [11] proposed a self-attention feature pyramid network (SA-FPN) that enhances information fusion across pyramid levels, improving detection accuracy at the cost of increased model complexity. To overcome the limitations of independent feature maps and fixed receptive fields in the detection head, Liao et al. [12] designed a Dynamic and RepConv Head (DRHead) that adaptively adjusts the receptive field for the input feature map, thereby integrating multi-scale contextual information and improving robustness and accuracy. Jiang et al. [13] incorporated AKConv into YOLOv8 and devised the C2f-BE module to enhance feature-learning capacity. Liang et al. [14] combined a parallel auxiliary multi-scale feature-enhancement module (MFEM) with RetinaNet and constructed a bidirectional feature pyramid, significantly improving small-object detection; however, in scenes with densely distributed small objects, false positives and missed detections remain issues. Li et al. [15] proposed RemDet, which redesigns the YOLOv8 architecture by introducing ChannelC2f, GatedFFN, and CED modules to efficiently improve both accuracy and speed for small-object detection. Lu et al. [16] presented MASF-YOLO, based on the YOLOv11 architecture, achieving a favorable balance between accuracy and efficiency through multi-scale feature aggregation. Wan et al. [17] proposed DAU-YOLO, which integrates receptive-field attention (RFA) and dynamic upsampling (DAU) into the YOLOv11 backbone to enhance the modeling of dense small-object patterns, yielding notable improvements under occlusion and low-illumination conditions. BPD-YOLO [18] integrates dual-phase and pyramid fusion to strengthen tiny-object perception while curbing parameter growth, while LightUAV-YOLO [19] introduces optimized feature enhancement and local attention tailored for UAV scenes with a compact budget. EMFE-YOLO [20] employs efficient multi-scale feature enhancement and prunes redundant heads to improve throughput, and ELNet [21] simplifies the backbone and detection heads for real-time deployment on edge devices. In addition, [22] incorporates intra-group multi-scale fusion attention and adaptive weighted feature fusion into YOLOv8, significantly improving sensitivity to small targets in UAV remote sensing imagery.

Beyond algorithmic advances, electromagnetic interference (EMI/IEMI) can corrupt or interrupt UAV imaging—e.g., by disturbing CMOS/CCD camera pipelines (readout timing and data integrity), which degrades captured frames and downstream perception [23–25].

Research Gaps

- Existing UAV small-object detectors often lack lightweight designs, making them unsuitable for real-time deployment on aerial platforms.
- Prior studies typically focus on either global semantic context or local detail features, but effective integration of both remains underexplored.
- Robustness under practical UAV conditions, such as electromagnetic interference, image degradation, and onboard hardware constraints, is rarely investigated.
- Detail-preserving modules have received limited attention, and existing approaches often increase parameters significantly, conflicting with lightweight requirements.

Contributions

- We propose GLDS-YOLO, a lightweight detection model tailored for UAV small-object detection, integrating four complementary modules: GSA, LSKA-SPPF, DCNv4, and SMSDE.
- We design SMSDE, a novel lightweight edge-detail enhancement module that reduces parameters by nearly one-third compared with MSDE while maintaining detection accuracy.
- We conduct extensive experiments on VisDrone2019 and DOTA1.0, demonstrating consistent improvements over state-of-the-art baselines.
- We provide comprehensive ablation studies, including module-wise contributions and a direct comparison between MSDE and SMSDE.
- We discuss UAV deployment considerations, highlighting both the potential applications and the current limitations of our approach.

The rest of this paper is organized as follows. Section 3 introduces the architecture of GLDS-YOLO and its core modules. Section 4 presents the experimental settings, results, and ablation studies. Section 5 concludes the paper by summarizing the main findings, limitations, and future directions.

3. Method

Building on the above improvements, we present the overall architecture of GLDS-YOLO and the cooperation among its components. Figure 1 illustrates the end-to-end pipeline: GSA [26–29] in the backbone enhances the perception of low-scale targets; SPPF-LSKA [30] embedded in the feature pyramid strengthens global semantics and multi-scale fusion; DCNv4 [31] is introduced into the neck to maintain lightweight design while improving robustness and a lightweight SMSDE [32–35] is placed before the detection head to recover edge details and reduce false positives.

These four modules were deliberately chosen for their complementary mechanisms to address the main challenges of UAV small-object detection. Specifically, GSA improves fine-grained perception of weak local features, LSKA-SPPF provides global semantic context to alleviate cluttered backgrounds, DCNv4 enhances adaptability to irregular target shapes and orientations, and SMSDE restores edge details while maintaining lightweight efficiency. Together, they form a mechanism-driven design that directly targets the problems of insufficient feature representation, limited global context, and blurred boundaries in UAV imagery.

The following subsections detail each module and its integration into the overall framework, followed by visualizations and ablations in Section 4.

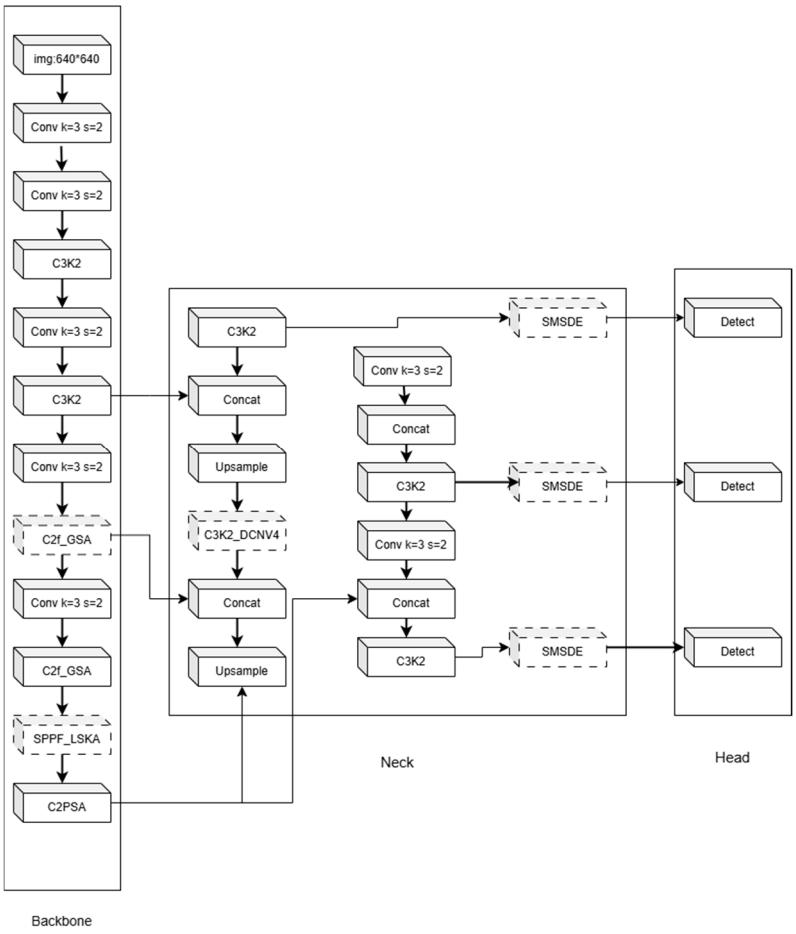


Figure 1. GLDS-YOLO Network Architecture.

3.1. Small-Object Perception Module

In UAV imagery, small objects often appear with weak visual responses, fuzzy boundaries, and limited texture, making them easily overwhelmed by background clutter. To address these challenges, we introduce a lightweight attention mechanism—Group Shuffle Attention (GSA)—into the YOLOv11n backbone to enhance the sensitivity to fine-grained object features while maintaining low computational cost. The overall architecture of the GSA module is shown in Figure 2.

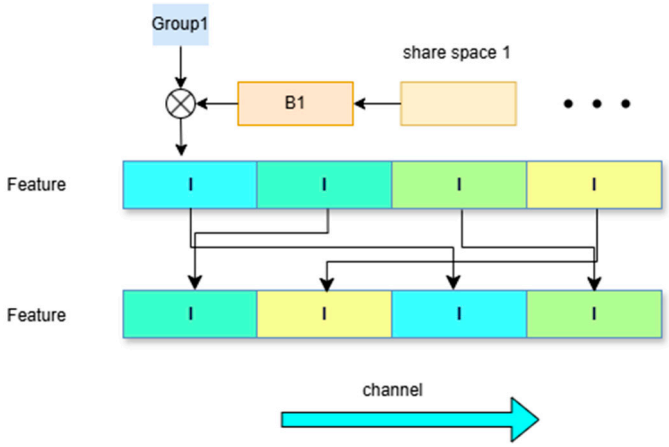


Figure 2. GSA Module Architecture.

"Group1" denotes the grouped feature channels used for partitioned processing to reduce computation, and "B1" refers to the predicted boundary features that assist in enhancing edge precision in the segmentation output.

The GSA module is composed of five sequential components: (1) channel normalization, (2) feature grouping, (3) Hadamard Product Attention (HPA) for group-wise enhancement, (4) group-wise shuffle for inter-group interaction, and (5) depthwise separable convolution for final feature integration.

First, the input feature map $X \in R(B \times C \times H \times W)$ is normalized using Layer Normalization:

$$X' = LayerNorm(X) \tag{1}$$

The normalized features are then equally divided into four groups along the channel dimension:

$$X' = \{X_1, X_2, X_3, X_4\}, X_i \in R^{B \times \frac{C}{4} \times H \times W}, i \in \{1, 2, 3, 4\} \tag{2}$$

For each feature group, the GSA module incorporates a Hadamard Product Attention (HPA) mechanism to enhance feature representation. Within the HPA structure, a learnable global memory tensor $S_i \in R^{(1 \times \frac{C}{4} \times K \times K)}$ is introduced, which serves as a shared global context prior. This memory is upsampled via interpolation to match the spatial resolution of the input features $H \times W$.

$$Resize(S_i) \in R^{(1 \times \frac{C}{4} \times H \times W)} \tag{3}$$

Subsequently, the memory features are integrated into the grouped features via element-wise multiplication (Hadamard product):

$$X_i' = X_i \bigodot Conv_i(Resize(S_i)), i \in \{1, 2, 3, 4\} \tag{4}$$

$\bigodot$ means element-wise multiplication, $Conv_i$ means group-wise convolution operation is applied within each feature group to further refine local representations. By introducing global contextual cues through the HPA mechanism, the discriminative capacity of grouped features is significantly enhanced, especially in capturing fine-grained details and improving local-global feature alignment.

To alleviate the information isolation introduced by group-wise processing and to promote inter-group feature interaction, the GSA module adopts a deterministic group shuffle operation. By reordering the grouped feature channels in a fixed pattern, this operation facilitates cross-group information exchange and enhances global feature integration.

$$X'' = shuffle(X_1', X_2', X_3', X_4') \tag{5}$$

The shuffle operation effectively disrupts the independence between feature groups, facilitating global information fusion. Subsequently, the reordered features are integrated using depthwise separable convolution. Specifically, the depthwise convolution processes each channel independently, thereby reducing computational complexity, while the pointwise (1×1) convolution aggregates cross-channel information. The final output feature map is computed as follows:

$$F_{out} = Conv_{Depthwise}(X'') + Conv_{Pointwise}(X'') \tag{6}$$

The GSA module maps input features to optimized outputs through grouped attention and lightweight convolutional design. It reduces parameters and computation while preserving feature expressiveness. This structure effectively enhances boundary modeling

and fine-grained detail capture, making it well-suited for UAV-based small object detection and real-time edge deployment.

3.2. Large-Kernel Attention for Multi-Scale Fusion Module

The proposed SPPF-LSKA module is specifically designed to enhance both local detail modeling and long-range contextual perception, which are critical for accurate detection of small objects in UAV imagery. As illustrated in Figures 3 and 4, this module integrates the fast Spatial Pyramid Pooling (SPPF) structure with the Large Separable Kernel Attention (LSKA) mechanism. The combined design allows efficient multi-scale feature extraction and receptive field expansion, significantly improving the network's sensitivity to small targets while maintaining low computational overhead.

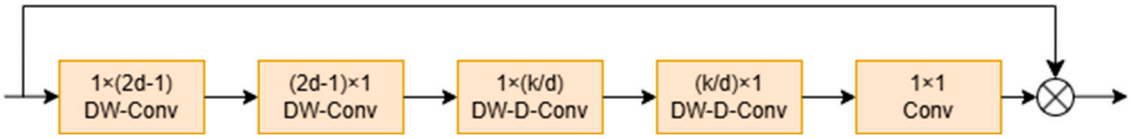


Figure 3. LSKA Architecture.

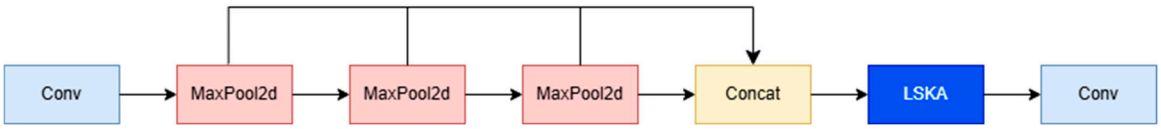


Figure 4. SPPF-LSKA Module.

The SPPF component constructs multi-scale feature responses using hierarchical max-pooling. Let the input feature be $F \in R(C \times H \times W)$, where C denotes the number of channels and H, W are the height and width, respectively. Three sequential max-pooling operations are applied to generate feature maps F_1, F_2, F_3, which are concatenated with the original feature map along the channel dimension:

$$F_{concat} = Concat(F, F_1, F_2, F_3) \tag{7}$$

The pooling kernel size and padding are chosen so that the spatial resolution of all features is preserved, enabling precise alignment. This multi-scale aggregation not only enriches the representation capacity but also injects global contextual cues that benefit small object recognition.

After multi-scale fusion, the LSKA component computes spatial attention on the concatenated feature map. LSKA leverages large-kernel decomposition to approximate a $K \times K$ convolution by factorizing it into horizontal and vertical depthwise 1-D convolutions, reducing computational complexity while expanding the receptive field. Given the input F_{concat}, LSKA first applies horizontal and vertical convolutions:

$$F_h = Conv_{1\times k}(F_{concat}), F_v = Conv_{k\times 1}(F_h) \tag{8}$$

It then incorporates dilated convolutions to capture broader spatial dependencies and long-range semantic context:

$$F_{dh} = Conv_{1\times k}^{dilated}(F_v), F_{dv} = Conv_{k\times 1}^{dilated}(F_{dh}) \tag{9}$$

A final pointwise convolution generates the spatial attention weights A, which are used to reweight the input features through element-wise multiplication:

$$F_{out} = F_{concat} \odot A \tag{10}$$

This kernel decomposition strategy reduces the computational complexity from $O(k^2)$ to $O(2k)$, while the incorporation of dilated convolutions further enlarges the receptive field without incurring significant increases in parameters or computational cost. Consequently, the module preserves the representational advantages of large-kernel modeling while achieving substantially improved efficiency.

3.3. Deformable Convolution v4 Module

In conventional convolutional neural networks (CNNs), the sampling locations of convolution kernels are fixed in a regular grid, which limits their ability to model objects with varying shapes, scales, and orientations. This constraint is particularly detrimental in UAV-based small object detection, where targets often exhibit irregular boundaries and diverse spatial distributions.

To address this limitation, we integrate the Deformable Convolution v4 (DCNv4) module into the neck of the YOLOv11n architecture, aiming to enhance spatial adaptability and representation robustness. The basic structure of the DCNv4 module is illustrated in Figure 5.

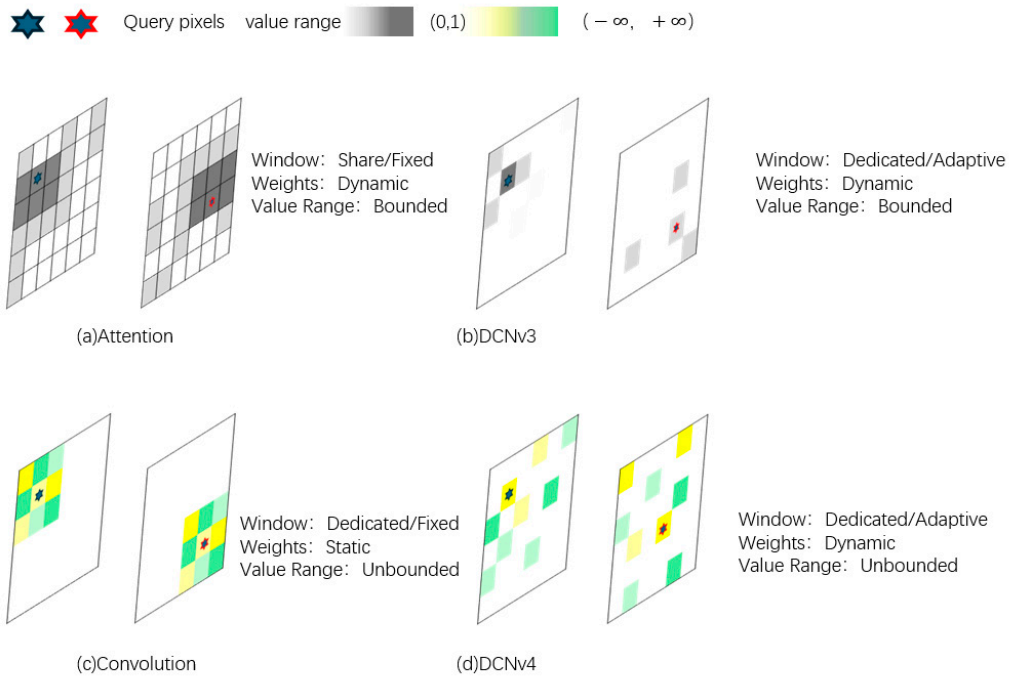


Figure 5. Architectural Comparison of DCNv3 and DCNv4.

The core mechanism of DCNv4 allows the convolution kernel to dynamically adjust its sampling positions according to the geometry of the input features, enabling more flexible and accurate capture of object shapes and spatial layouts.

The basic deformable convolution operation can be formulated as

$$y(p_0) = \sum_{K=1}^{K} w_k * x\left(p_o + p_k + \Delta_{pk}\right) \tag{11}$$

where $y(p_0)$ denotes the output feature value at position p_0, $x\left(p_o + p_k + \Delta_{pk}\right)$ represents the input feature at the dynamically sampled position, K is the number of kernel sampling points, p_k are the predefined offsets, Δ_{pk} are the learnable offsets predicted by the network, and w_k denotes the dynamically computed convolution weights.

This formulation enables adaptive adjustment of sampling positions based on the target's geometric structure, significantly improving spatial representation capabilities.

Although earlier versions of DCN improved small object detection performance, they suffered from computational inefficiencies and slow inference, particularly on high-resolution feature maps. DCNv4 introduces two key enhancements over DCNv3:

1. Dynamic Weight Enhancement—The traditional softmax normalization is removed, allowing the convolution weights w_k to take values in $(-\infty, +\infty)$ This broader dynamic range improves the expressiveness of the convolution operation.
2. Memory Access and Thread Allocation Optimization—DCNv4 reduces redundant memory operations by optimizing access patterns for deformable sampling and introduces an adaptive thread allocation strategy. The number of threads T for parallel computation is determined as

$$T = \frac{m \cdot G \cdot C}{d_{stride}} \tag{12}$$

where m is the product of batch size and the number of query points, G is the number of groups, C is the number of channels, and d_{stride} is the stride. This design improves GPU utilization, reduces idle cycles, and increases throughput.

Meanwhile, both the sampling locations and the convolution weights in DCNv4 are dynamically learned through a neural network, enabling the module to better adapt to complex target shapes and feature distributions. The learning process for the offsets can be expressed as

$$\Delta_{p_k} = h(x), m_k = g(x) \tag{13}$$

where $h(x)$ and $g(x)$ denote the prediction functions for the offsets and the convolution weights, respectively.

These improvements allow DCNv4 to flexibly adapt to the actual morphology and spatial distribution of small objects, delivering significant performance gains in small object detection tasks.

3.4. Small-Object-Enhanced Multi-Scale and Structure Detail Enhancement Module

To address the challenges of blurred edges, insufficient fine-grained details, and high computational cost in UAV-based small object detection, we propose a lightweight edge-aware enhancement module, termed the Small-object-enhanced Multi-scale and Structure Detail Enhancement (SMSDE). As shown in Figure 6, the SMSDE consists of four sequential components: input feature compression, multi-scale edge enhancement, structure-aware feature fusion, and efficient upsampling.

Let F_m denote the intermediate decoder feature and F_d is upsampled detail feature © represents channel concatenation AP denotes average pooling F_x^e is the feature extracted from the x-th layer, F_x^{ee} is the edge-enhanced output of the x-th layer, $\ominus$ denotes element—wise subtraction for edge extraction, and $\bigoplus$ denotes element—wise addition for edge fusion.

(1) Lightweight input feature compression

To reduce the computational burden of high-resolution feature maps, the SMSDE module first applies a channel compression strategy to the input feature map I. By introducing a

channel compression factor γ, the dimensionality is reduced to obtain the low-dimensional initial feature F_{in}, formulated as

$$F_{in} = Conv_{1\times1}(I, \gamma) \tag{14}$$

where the value of γ effectively controls the number of channels, thereby substantially reducing model parameters and computational cost while preserving the effective extraction of critical low-level features.

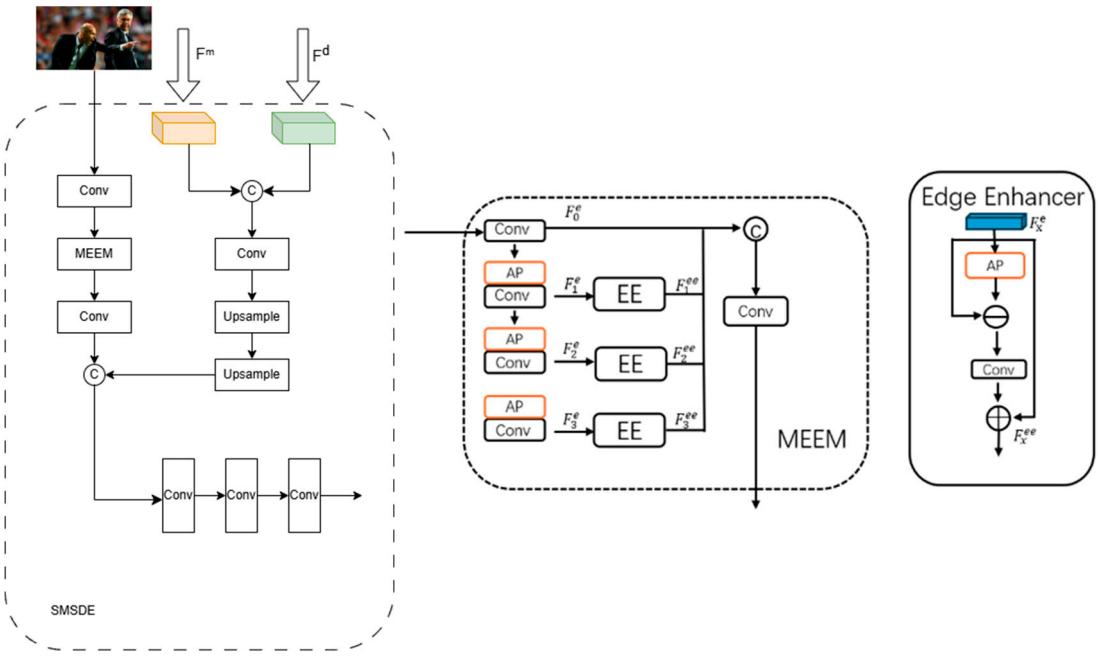


Figure 6. SMSDE Module Architecture.

(2) Multi-scale Edge Detail Enhancement

To effectively capture and reinforce edge details of small objects, the SMSDE module incorporates a multi-scale edge enhancement mechanism. The input feature map is first processed by average pooling (AP) operations at multiple scales, followed by a residual subtraction strategy to highlight edge responses. Convolutional operations are then applied to produce multi-scale edge-enhanced features F_{edge}:

$$F_{edge} = \sum_{i=1}^{n} \omega_i \cdot [Conv_{1\times1}(F_i - AP(F_i)) + F_i - AP(F_i)] \tag{15}$$

where F_i denotes the pooled feature at the i-th scale ω_i is a learnable adaptive scale weight that satisfies as 1 This design significantly improves the sensitivity to weak edge features and enhances fine-grained detail characterization.

(3) Feature Fusion and Residual Optimization Strategy

To avoid redundancy from direct feature concatenation SMSDE adopts a two-step fusion and residual connection method. First, the backbone feature F_{main} is concatenated with the enhanced edge feature F_{edge} along the channel dimension followed by 3×3 and 1×1 convolutions to produce the fused feature F_{fuse}:

$$F_{fuse} = Conv_{1\times1}\left(Conv_{3\times3}\left(F_{main} \oplus F_{edge}\right)\right) \tag{16}$$

A residual connection is then applied to stabilize the information flow and enhance fusion effectiveness, producing the final fused feature F_{out}:

$$F_{out} = F_{fuse} + \lambda \cdot F_{main} \tag{17}$$

where λ is a residual balancing coefficient controlling fusion strength.

(4) Efficient Feature Upsampling Strategy

In the edge reconstruction stage, SMSDE combines bilinear interpolation with lightweight convolution to balance spatial detail recovery and computational efficiency. The high-resolution output feature F_{up} is as follows:

$$F_{up} = \alpha \cdot Bilinear(F_{out}) + (1 - \alpha) \cdot Conv_{3\times3}(F_{out}) \tag{18}$$

where α is a learnable fusion coefficient that adaptively balances the benefits of interpolation and convolution, enabling optimal detail restoration with minimal computational cost.

Through the coordinated design of these four substructures, the SMSDE module enhances detection accuracy and edge structure representation for small objects while reducing parameter count and improving robustness in complex UAV scenarios. Compared with the original MSDE structure, SMSDE adopts a simplified design that reduces redundant operations in edge-detail fusion. This modification decreases the parameter count by nearly one-third while preserving detection accuracy. A detailed comparison with MSDE is reported in Section 4.3.

4. Experimental Results

4.1. Datasets and Experimental Setup

This study evaluates the proposed GLDS-YOLO model on two representative public small object detection datasets: VisDrone2019 [36] and DOTA [37]. VisDrone2019, collected by the AISKYEYE team at Tianjin University, contains 10,209 images captured by various UAV platforms under diverse geographic, lighting, and weather conditions. It is split into 6471 training images, 548 validation images, and 3190 testing images, with annotations for 10 object categories and approximately 2.6 million instances. The dataset presents challenges such as small object sizes, occlusion, overlapping objects, and complex backgrounds.

DOTA (Dataset for Object Detection in Aerial Images) contains 21,046 high-resolution aerial images covering 15 object categories, including aircraft, vehicles, and ships. It is split into training and validation sets with a 3:1 ratio. The dataset's high annotation quality, diversity, and extensive small object coverage make it ideal for evaluating detection algorithms' generalization ability in complex multi-scene scenarios.

The experiments were conducted on a Windows 11 system with Python 3.10.15, PyTorch 2.4.0, and CUDA 12.6. Training, validation, and inference were performed on an NVIDIA RTX 6000 GPU (24 GB). The optimizer was stochastic gradient descent (SGD) with momentum 0.937 and weight decay 5×10^{-4}, and a cosine learning rate scheduler with 3 warmup epochs was applied. Automatic mixed precision (AMP) was disabled, and Exponential Moving Average (EMA) was enabled for stability. Data augmentation included Mosaic (1.0), random horizontal flipping (0.5), and HSV color jitter (h = 0.015, s = 0.7, v = 0.4). Focal-EIoU was used for bounding-box regression and binary cross-entropy (BCE) for classification and objectness. Evaluation followed the COCO mAP protocol (mAP@0.5 and mAP@0.5:0.95), with VisDrone2019 results additionally verified using the official server. The key training parameters are summarized in Table 1.

Table 1. Training hyperparameters of GLDS-YOLO.

Parameter	Value
Epochs	200
Batch size	16
Image size	640 × 640
learning rate	0.01
Optimizer	SGD

4.2. Training Process and Convergence Analysis

The convergence behavior of GLDS-YOLO was evaluated by training the model for 200 epochs on the VisDrone2019 dataset (Figure 7). The training and validation loss curves exhibit similar downward trends and converge smoothly without overfitting, indicating good generalization ability. Precision, Recall, and mAP metrics increase steadily and stabilize in the later training stages, which is consistent with the performance reported in the following ablation and comparison experiments.

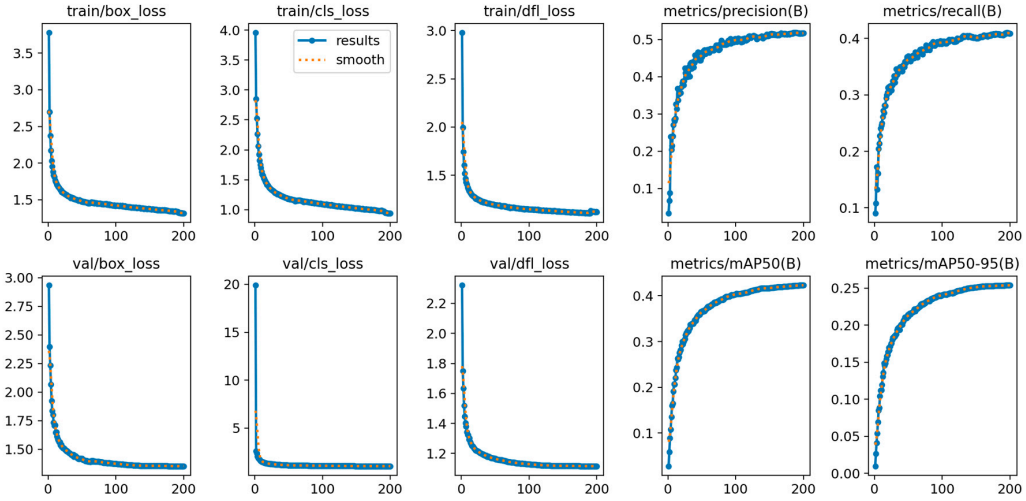


Figure 7. Training convergence curves of GLDS-YOLO on the VisDrone2019 dataset.

The evaluation metrics used in this study include Precision (P), Recall (R), and Mean Average Precision (mAP) at two thresholds: *mAP@0.5* and *mAP@0.5:0.95*.

Precision and Recall are computed based on the confusion matrix parameters—True Positives (TP), False Positives (FP), and False Negatives (FN) defined as follows:

- **TP**: The number of correctly detected targets.
- **FP**: The number of incorrectly detected targets.
- **FN**: The number of missed targets.

Precision (P) and Recall (R) are calculated as

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN} \tag{19}$$

The Average Precision (AP) is obtained by integrating the Precision–Recall (PR) curve, as shown in Equation:

$$AP = \int_0^1 P(R)dR \tag{20}$$

The Mean Average Precision (mAP) is the mean of the AP values over all classes, defined as

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{21}$$

where N denotes the number of object categories in the dataset. Specifically,

mAP@0.5 represents the average detection precision when the Intersection over Union (IoU) threshold is fixed at 0.5;
mAP@0.5:0.95 is calculated by averaging AP values over IoU thresholds ranging from 0.5 to 0.95 with a step size of 0.05.

4.3. Ablation Studies

Ablation experiments were conducted to quantify the contributions of each proposed module (GSA, LSKA-SPPF, DCNv4, and SMSDE) to overall performance (Table 2). Starting from the baseline YOLOv11n, modules were incrementally added, and results were evaluated on the VisDrone2019 dataset. The results demonstrate that each module individually improves detection accuracy, while their combination achieves the best performance. Furthermore, since SMSDE is redesigned from the original MSDE, we also provide a direct comparison between the two modules under identical settings. This additional comparison demonstrates that SMSDE maintains accuracy while significantly reducing the parameter count, confirming its effectiveness as a lightweight improvement.

Table 2. Ablation results of GLDS-YOLO on the VisDrone2019 dataset.

Module	P	R	mAP@0.5	mAP@0.5:0.95	Para (M)
YOLOv11n	0.427	0.318	0.315	0.182	2.5
YOLOv11n + GSA	0.452	0.401	0.366	0.206	2.9
YOLOv11n + GSA + LSKA	0.476	0.409	0.394	0.217	3.2
YOLOv11n + GSA + LSKA + DCNv4	0.503	0.417	0.411	0.235	3.5
YOLOv11n + LSKA + DCNv4 + SMSDE	0.521	0.468	0.421	0.238	3.7
YOLOv11n + GSA + LSKA + DCNv4 + SMSDE	0.557	0.421	0.436	0.252	3.9
YOLOv11n + GSA + LSKA + DCNv4 + MSDE	0.546	0.420	0.434	0.251	5.8

In addition to the overall ablation results presented in Table 2, we further report detailed per-scale and per-class performance to better evaluate the effectiveness of the proposed modules. Table 3 presents results across different object scales (APs, APm, APl and ARs, ARm, ARl), while Table 4 summarizes per-class AP values on the VisDrone2019 dataset. These supplementary results demonstrate that GLDS-YOLO achieves particularly notable improvements for small objects, while maintaining competitive performance on medium and large targets.

Table 3. Per-scale results on the VisDrone2019 dataset.

Module	APs	APm	APl	ARs	ARm	ARl
YOLOv11n	0.106	0.271	0.344	0.201	0.447	0.499
Ours	0.194	0.349	0.387	0.295	0.518	0.514

Table 4. Per-class AP@0.5 results on the VisDrone2019 dataset.

Model	Pedestrian	People	Car	Van	Truck	Tricycle	Awning-Tricycle	Bus	Motor
YOLOv11n	33.9	26.6	75.2	37.7	26.5	18.6	11.6	43.4	34.7
Ours	51.3	41.7	85.1	50.9	37.9	29.0	17.7	59.6	49.9

Overall, the per-scale and per-class results confirm that GLDS-YOLO delivers consistent gains across all categories and scales, with particularly notable improvements on small objects, which are the most challenging in UAV scenarios.

4.4. Comparison with State-of-the-Art Methods

To comprehensively evaluate the detection performance and deployment efficiency of the proposed GLDS-YOLO in small object detection tasks, we conducted systematic comparisons under identical datasets, training parameters, and hardware settings. The comparison includes mainstream small-object detection models such as the YOLO series (YOLOv5S, YOLOv6, YOLOv7-tiny, YOLOv8n), structure-enhanced models (GBS-YOLOv5S, MDH-YOLOv8, PS-YOLOM, YOLOv10, YOLOv10S), end-to-end architectures (RT-DETR), recent lightweight improved models (MASF-YOLO, RemDet series), and the YOLOv11 baseline series (YOLOv11n). Most YOLO-based models were re-trained under the same environment, datasets, and hyperparameter settings as described in Section 4.1, ensuring consistent experimental conditions. The results of the other methods were directly extracted from their original publications [38–42].

All models were evaluated on the VisDrone2019 dataset, and five core metrics were reported: Precision (P), Recall (R), mAP@0.5, mAP@0.5:0.95, and model size (Table 5). These metrics collectively reflect detection accuracy, robustness, and computational efficiency, enabling a fair and comprehensive performance comparison.

Table 5. Performance Comparison of Different Algorithms on the VisDrone2019 Dataset.

Module	P/%	R/%	mAP@0.5/%	mAP@0.5:0.95/%	Model Size/MB
GBS-YOLOv5S	49.7	36.8	35.3	20.1	14.5
YOLOv5S	47.3	35.1	34.8	19.1	13.4
YOLOv6	45.2	32.4	30.8	17.8	9.94
YOLOv7-tiny	42.0	30.7	32.8	16.7	12.3
YOLOv8n	50.6	33.2	33.3	19.3	6.3
MDH-YOLOv8	54.9	34.1	37.5	22.7	6.0
YOLOv10	52.8	36.0	31.7	23.3	9.2
RT-DETR	58.7	41.6	45.3	27.5	40
PS-YOLO-M	50.2	38.9	37.6	22.3	7.9
YOLOv10-S	50.5	38.3	39.1	23.5	9.8
MASF-YOLO-n	56.3	40.7	43.2	28.2	13.2
RemDet-Tiny	57.9	39.7	37.1	21.8	12.8
RemDet-M	62.7	37	46.1	28.2	93.2
YOLOv11n	50.3	41.7	31.5	18.2	5.4
GLDS-YOLO	**55.7**	**42.1**	**43.6**	**25.2**	**7.2**

As shown in Table 5, GLDS-YOLO attains mAP@0.5 and mAP@0.5:0.95 of 43.6% and 25.2%, respectively. Relative to RT-DETR (45.3%/27.5%), GLDS-YOLO provides comparable accuracy with a markedly smaller model size (7.2 MB vs. 40 MB; ≈18%). Among models ≤10 MB, it ranks first on both metrics. Compared with MDH-YOLOv8, GLDS-YOLO improves mAP@0.5 by 6.1 pp, precision by 0.8 pp, and recall by 8.0 pp, with a modest size increase (+1.2 MB). Relative to the YOLOv11n baseline, gains are +12.1 pp (mAP@0.5) and +7.0 pp (mAP@0.5:0.95) at +1.8 MB. These results indicate a favorable accuracy–efficiency trade-off within the lightweight regime.

Overall, by integrating GSA, LSKA-SPPF, DCNv4, and SMSDE modules, GLDS-YOLO achieves superior accuracy–efficiency trade-offs. Notably, even in lightweight configurations, its performance approaches or surpasses that of larger models, highlighting its strong applicability to real-time UAV-based small object detection scenarios.

To further verify the generalization capability of the proposed model under different data distributions, we conducted additional evaluations on the DOTA-v1.0 dataset, comparing it with several mainstream detection models. The results are shown in Table 6.

Table 6. Experimental Results of Different Models on the DOTA-v1.0 Dataset.

Module	mAP50/%	mAP50:95/%
YOLOv8n	60.4	38.1
TPH-YOLO	69.1	44.7
Drone-DETR	67.9	45.2
Oriented R-CNN	75.8	—
GRA	77.6	—
GLDS-YOLO	**74.5**	**49.7**

As shown in Table 6, GLDS-YOLO achieves 74.5% mAP@0.5 and 49.7% mAP@0.5:0.95 on DOTA-v1.0, outperforming TPH-YOLO, Drone-DETR, and YOLOv8n, and reaching accuracy levels comparable to GRA maintaining high efficiency.

4.5. Visualization Analysis

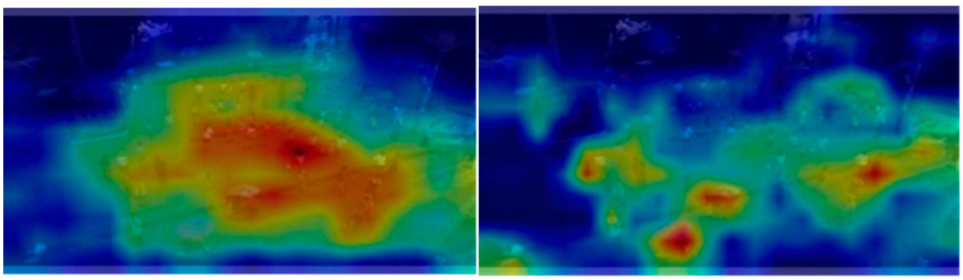
To further validate the effectiveness of the proposed GLDS-YOLO in real-world small object detection scenarios, representative test images were randomly selected from the VisDrone2019 dataset for visual comparison. These include challenging cases with dense targets, low illumination, and severe occlusion. Detection results are compared with the baseline YOLOv11n, as shown in Figures 8–10.

(**a**) GLDS-YOLO (**b**) YOLOv11n

Figure 8. Detection results in a dense crowd scenario.

(a) GLDS-YOLO **(b)** YOLOv11n

Figure 9. Detection results in a low-light nighttime scenario.



(**a**) GLDS-YOLO (**b**) YOLOv11n

Figure 10. Visualization of attention heatmaps.

In dense pedestrian scenes (Figure 8), YOLOv11n exhibits notable missed detections, particularly at image edges and in occluded regions. In contrast, GLDS-YOLO successfully detects a greater number of densely distributed pedestrians, improving recall for small targets. This demonstrates the advantages of the GSA and SMSDE modules in enhancing local detail and edge feature representation.

Under low-light night time conditions (Figure 9), where poor illumination and partial occlusions between objects typically degrade detection accuracy, YOLOv11n produces more false negatives and false positives. The proposed GLDS-YOLO accurately identifies multiple small targets (e.g., vehicles) and maintains high responsiveness in low-texture regions, indicating improved adaptability to low-light environments and robustness to occlusion.

The attention heatmaps (Figure 10) further visualize the models' focus on different regions of the input images. GLDS-YOLO shows more concentrated responses in small-object-dense areas, effectively highlighting semantically relevant regions. This provides additional evidence that the integrated attention mechanism improves both interpretability and detection accuracy.

5. Conclusions

This paper presented GLDS-YOLO, a lightweight detector tailored for UAV-based small object detection. By integrating GSA, LSKA-SPPF, DCNv4, and the optimized SMSDE module, the model enhances perception, context modeling, spatial adaptability, and edge detail representation. Experiments on VisDrone2019 and DOTA demonstrated consistent gains over YOLOv11n and other baselines, particularly for small objects, while maintaining a compact model size suitable for real deployment.

Nevertheless, limitations remain. In complex scenarios with dense occlusion and overlapping objects, missed detections still occur. In addition, current evaluations focus mainly on accuracy and parameter count, whereas broader aspects of deployment efficiency—such as inference latency, memory footprint, and energy consumption—have not yet been systematically examined.

Future work will therefore focus on the following: (1) investigating model robustness under electromagnetic disturbances and image degradations (e.g., noise, blur); (2) extending lightweight evaluation to multiple dimensions, including latency, memory usage, and energy efficiency; and (3) validating real-time performance on embedded GPUs and edge accelerators. These efforts will provide a more comprehensive assessment of deployment efficiency and further enhance the practicality of GLDS-YOLO for UAV applications.

Author Contributions: Conceptualization, Z.J.; Data curation, J.H.; Formal analysis, J.S.; Investigation, J.H.; Methodology, J.S.; Project administration, Z.J.; Software, J.S.; Writing—original draft, J.S.; Writing—review & editing, J.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 81101116.

Data Availability Statement: The VisDrone2019 dataset used in this study is publicly available at the official repository: https://github.com/VisDrone/VisDrone-Dataset (accessed on 19 August 2025).

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

References

1. Abdellatif, T.; Sedrine, M.A.; Gacha, Y. DroMOD: A drone-based multi-scope object detection system. *IEEE Access* **2023**, *11*, 26652–26666. [CrossRef]
2. Konstantinidis, D.; Stathaki, T.; Argyriou, V.; Grammalidis, N. Building detection using enhanced HOG–LBP features and region refinement processes. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, *10*, 888–905. [CrossRef]

3. Mistry, D.; Banerjee, A. Comparison of feature detection and matching approaches: SIFT and SURF. *Glob. Res. Dev. J. Eng.* **2017**, *2*, 7–13. [CrossRef]
4. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
5. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
6. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Adv. Neural Inf. Process. Syst.* **2015**, 28. [CrossRef]
7. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
8. 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 European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer International Publishing: Cham, Switzerland, 2016; pp. 21–37.
9. Yang, S.; Li, F.; Du, Y.; Gao, W.; Sun, T. GS-YOLOv8: An Improved UAV Target Detection Algorithm Based on YOLOv8. In Proceedings of the 2024 IEEE 4th International Conference on Electronic Technology, Communication and Information (ICETCI), Changchun, China, 24–26 May 2024; pp. 643–647.
10. Wei, L.; Luo, X.; Kang, J. A Small Object Detection Algorithm for UAV Aerial Images Based on Improved YOLOv8. *Comput. Eng. Sci.* **2024**, *46*, 112–118.
11. Zhou, X.; Zhang, L. SA-FPN: An Effective Feature Pyramid Network for Crowded Human Detection. *Appl. Intell.* **2022**, *52*, 12556–12568. [CrossRef]
12. Liao, N.; Cao, T.; Liu, K.; Xu, M.; Zhu, M.; Gu, Y.; Wang, P. UAV Small Object Detection Algorithm Based on Composite Features and Multi-Scale Fusion. *Comput. Eng. Appl.* **2023**, *59*, 145–151.
13. Jiang, W.; Wang, W.; Yang, J. AEM-YOLOv8s: Small Object Detection in UAV Aerial Images. *Comput. Eng. Appl.* **2024**, *60*, 191–202.
14. Liang, H.; Yang, J.; Shao, M. FE-RetinaNet: Small Target Detection with Parallel Multi-Scale Feature Enhancement. *Symmetry* **2021**, *13*, 950. [CrossRef]
15. Li, C.; Zhao, R.; Wang, Z.; Xu, H.; Zhu, X. RemDet: Rethinking Efficient Model Design for UAV Object Detection. *Proc. AAAI Conf. Artif. Intell.* **2025**, *39*, 4643–4651. [CrossRef]
16. Lu, L.; He, D.; Liu, C.; Deng, Z. MASF-YOLO: An Improved YOLOv11 Network for Small Object Detection on Drone View. *arXiv* **2025**, arXiv:2504.18136. [CrossRef]
17. Wan, Z.; Lan, Y.; Xu, Z.; Shang, K.; Zhang, F. DAU-YOLO: A Lightweight and Effective Method for Small Object Detection in UAV Images. *Remote Sens.* **2025**, *17*, 1768. [CrossRef]
18. Zhang, Y.; Chen, L.; Wang, X.; Liu, J. BPD-YOLO: A Lightweight Small Object Detection Model for UAV Images Based on Deep Semantic Integration. *Sci. Rep.* **2025**, *15*, 16878.
19. Zhao, H.; Wu, Q.; Li, F.; Zhang, M. LightUAV-YOLO: A Lightweight YOLOv8n-Based UAV Object Detection Algorithm with Optimized Feature Fusion and Local Attention. *J. Supercomput.* **2025**, *81*, 105.
20. Liu, K.; Sun, Y.; Hu, J. EMFE-YOLO: A Lightweight Small Object Detection Model for UAVs Based on Efficient Multi-Scale Feature Enhancement. *Sensors* **2025**, *25*, 5200.
21. Wang, P.; Li, R.; Zhou, X. ELNet: An Efficient and Lightweight Network for Small Object Detection in UAV Imagery. *Remote Sens.* **2025**, *17*, 2096.
22. Liu, Y.; Chen, X.; Zhao, W. 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.
23. Yang, Z.; Wen, L.; Li, Y.; Zhou, D.; Wang, X.; Ding, R.; Zhong, M.; Meng, C.; Fang, W.; Guo, Q. Analysis of the Interference Effects in CMOS Image Sensors Caused by Strong Electromagnetic Pulses. *J. Electromagn. Eng. Sci.* **2024**, *24*, 151–160. [CrossRef]
24. Kim, S.-G.; Lee, E.; Hong, I.-P.; Yook, J.-G. Review of Intentional Electromagnetic Interference on UAV Sensor Modules and Experimental Study. *Sensors* **2022**, *22*, 2384. [CrossRef]
25. Singh, R.K.; Mishra, S.; Pavan Kumar, Y. Undermining Live Feed ML Object Detection Accuracy with EMI on Vehicular Camera Sensors. In Proceedings of the 2024 IEEE 99th Vehicular Technology Conference (VTC2024-Spring), Singapore, 24–27 June 2024; IEEE: Piscataway, NJ, USA, 2024.
26. Xu, J.; Tong, L. Lb-unet: A lightweight boundary-assisted unet for skin lesion segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Marrakesh, Morocco, 6–10 October 2024; Springer Nature: Cham, Switzerland, 2024; pp. 361–371.
27. Cao, H.; Wang, Y.; Chen, J.; Jiang, D.; Zhang, X.; Tian, Q.; Wang, M. Swin-unet: Unet-like pure transformer for medical image segmentation. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 25–27 October 2022; Springer Nature: Cham, Switzerland, 2022; pp. 205–218.

28. Valanarasu, J.M.J.; Patel, V.M. Unext: Mlp-based rapid medical image segmentation network. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Singapore, 18–22 September 2022; Springer Nature: Cham, Switzerland, 2022; pp. 23–33.
29. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6848–6856.
30. 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]
31. Xiong, Y.; Li, Z.; Chen, Y.; Wang, F.; Zhu, X.; Luo, J.; Wang, W.; Lu, T.; Li, H.; Qiao, Y.; et al. Efficient Deformable ConvNets: Rethinking Dynamic and Sparse Operator for Vision Applications. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 17–21 June 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 5652–5661.
32. Ruan, J.; Xie, M.; Gao, J.; Liu, T.; Fu, Y. Ege-unet: An efficient group enhanced unet for skin lesion segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Vancouver, BC, Canada, 8–12 October 2023; Springer Nature: Cham, Switzerland, 2023; pp. 481–490.
33. Gao, S.; Zhang, P.; Yan, T.; Lu, H. Multi-scale and detail-enhanced segment anything model for salient object detection. In Proceedings of the 32nd ACM International Conference on Multimedia, Melbourne, VIC, Australia, 28 October–1 November 2024; pp. 9894–9903.
34. Chen, T.; Zhu, L.; Ding, C.; Cao, R.; Wang, Y.; Li, Z.; Sun, L.; Mao, P.; Zang, Y. Sam-adapter: Adapting segment anything in underperformed scenes. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–3 October 2023; pp. 3367–3375.
35. He, C.; Li, K.; Zhang, Y.; Tang, L.; Zhang, Y.; Guo, Z.; Li, X. Shenzhen International Graduate School; Tsinghua University, NEC Laboratories America; 3ETH Zürich; Tianyi Traffic Technology. Camouflaged object detection with feature decomposition and edge reconstruction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 22046–22055.
36. 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. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Republic of Korea, 28–29 October 2019.
37. Xia, G.-S.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A Large-Scale Dataset for Object Detection in Aerial Images. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3974–3983.
38. Peng, S.; Fan, X.; Yu, L. PS-YOLO: A Small Object Detector Based on Efficient Convolution and Multi-Scale Feature Fusion. *Multimedia Syst.* **2024**, *30*, 1–16. [CrossRef]
39. Zhao, Y.; Lv, W.; Xu, S.; Wei, J.; Wang, G.; Dang, Q.; Liu, Y.; Chen, J. DETRs Beat YOLOs on Real-Time Object Detection (RT-DETR). *arXiv* **2023**, arXiv:2304.08069.
40. Li, C.; Li, L.; Zhang, B.; Ouyang, W.; Wang, L. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976. [CrossRef]
41. 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. *arXiv* **2022**, arXiv:2207.02696.
42. 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.