scientific reports

Check for updates

OPEN

LRDS-YOLO enhances small object detection in UAV aerial images with a lightweight and efficient design

Yuqi Han[1,2], Chengcheng Wang[1,2], Hui Luo[3], Huihua Wang[2], Zaiqing Chen[1,2], Yuelong Xia[1,2] & Lijun Yun[1,2]✉

Small object detection in UAV aerial images is challenging due to low contrast, complex backgrounds, and limited computational resources. Traditional methods struggle with high miss detection rates and poor localization accuracy caused by information loss, weak cross-layer feature interaction, and rigid detection heads. To address these issues, we propose LRDS-YOLO, a lightweight and efficient model tailored for UAV applications. The model incorporates a Light Adaptive-weight Downsampling (LAD) module to retain fine-grained small object features and reduce information loss. A Re-Calibration Feature Pyramid Network (Re-Calibration FPN) enhances multi-scale feature fusion using bidirectional interactions and resolution-aware hybrid attention. The SegNext Attention mechanism improves target focus while suppressing background noise, and the dynamic detection head (DyHead) optimizes multi-dimensional feature weighting for robust detection. Experiments show that LRDS-YOLO achieves 43.6% mAP50 on VisDrone2019, 11.4% higher than the baseline, with only 4.17M parameters and 24.1 GFLOPs, striking a balance between accuracy and efficiency. On the HIT-UAV infrared dataset, it reaches 84.5% mAP50, demonstrating strong generalization. With its lightweight design and high precision, LRDS-YOLO offers an effective real-time solution for UAV-based small object detection.

Keywords UAV (unmanned aerial vehicle), Small object detection, Feature pyramid network, Real time, Attention mechanisms

UAV aerial photography technology has been extensively used in various fields, including smart city development[1], agriculture[2], traffic monitoring[3], and disaster management[4], becoming a critical technological tool. Thanks to the unique aerial perspective of UAVs, they are capable of efficiently and comprehensively collecting data over large areas, providing crucial support for modern applications such as crop growth monitoring, urban traffic management, security surveillance, and emergency rescue operations. In these fields, UAV technology has not only significantly enhanced the efficiency and accuracy of data acquisition but also provided unprecedented capabilities for real-time monitoring and dynamic management.

However, the analysis of UAV-captured images faces unique challenges, particularly when detecting targets from high-altitude shots. These targets are typically small, occupy a very low proportion of the image pixels, and often have low contrast due to lighting conditions and the characteristics of the targets[5]. Furthermore, UAV-captured scenes usually feature complex background elements, with aerial images often containing substantial noise, and target objects may be partially occluded by other objects. In addition, due to the limited computational resources of UAV equipment, achieving real-time object detection requires minimizing the computational burden while ensuring detection accuracy. Traditional object detection algorithms often struggle to effectively address the complex backgrounds and real-time demands presented by high-altitude aerial images. Therefore, improving the real-time performance and accuracy of algorithms in small object detection has become a key challenge that needs to be addressed in the field of computer vision.

The definition of small objects varies across different contexts, with two primary interpretations currently established. The first is an absolute size definition, which categorizes small objects based on their absolute pixel dimensions. In the MS COCO dataset proposed by Microsoft, targets with dimensions below 32×32 pixels are classified as small objects[6]. The second interpretation adopts a relative size definition, determining small objects by their proportional relationship to the overall image dimensions. According to the International Society for Optics and Photonics (SPIE), in a 256×256-pixel image, objects occupying less than 80 pixels in area-

[1]School of Information, Yunnan Normal University, Kunming 650500, Yunnan, China. [2]Department of Education of Yunnan Province, Engineering Research Center of Computer Vision and Intelligent Control Technology, Kunming 650500, Yunnan, China. [3]Kunming Branch of the Third College of PLA Information Engineering University, Kunming, Yunnan, China. ✉email: yunlijun@ynnu.edu.cn

equivalent to less than 0.12% of the original image size-are defined as small targets[7]. Figure 1 shows small targets in different scenarios in the VisDrone dataset.

To address the issues mentioned above, this paper proposes a UAV small target detection method named LRDS-YOLO. This method introduces the Light Adaptive-weight Downsampling (LAD) downsampling approach and the SegNext Attention mechanism[8] within the backbone network to enhance the feature extraction of small targets and improve background suppression capabilities. By adaptively adjusting the downsampling strategy with LAD, the method effectively reduces information loss, while the SegNext Attention mechanism focuses on key regions within the image, further improving the detection accuracy of small targets in complex backgrounds. Furthermore, LRDS-YOLO incorporates a Re-Calibration Feature Pyramid Network (Re-Calibration FPN) at the detection head, coupled with the Dyhead[9] dynamic adjustment mechanism, to optimize the fusion of multi-scale features and small target detection strategies. This results in improved localization accuracy and recall rate, while maintaining a low computational cost.

The main contributions of this paper can be summarized as follows:

Introduction of lightweight adaptive downsampling

To enhance both accuracy and efficiency in small object detection tasks, this paper proposes a novel lightweight downsampling method with adaptive capability, termed Light Adaptive-weight Downsampling (LAD). Unlike conventional downsampling techniques that rely on fixed rules or spatial configurations, LAD introduces an attention-guided adaptive weighting strategy that dynamically emphasizes critical regions during the downsampling process. Specifically, LAD accurately identifies salient areas where small objects are located and assigns greater retention weights to these regions during feature compression. This approach significantly reduces semantic loss caused by downsampling while preserving computational efficiency and improving the model's sensitivity to small objects.

Design of re-calibration feature pyramid network

To enhance the interaction between shallow and deep features, this paper introduces, for the first time, a novel feature fusion architecture termed the Re-Calibration Feature Pyramid Network (Re-Calibration FPN). While shallow layers preserve fine-grained details and deeper layers capture high-level semantic cues, their straightforward fusion often leads to feature redundancy and semantic inconsistency. To tackle this long-standing issue, we further propose a Selective Boundary Aggregation (SBA) module, which selectively incorporates boundary cues into semantic representations. This design uniquely refines object contours and enhances localization precision, offering a new perspective on hierarchical feature fusion for small object detection.

Introduction of the DyHead (dynamic head) mechanism

Traditional detection heads are fixed, leading to performance limitations when handling targets of varying scales and complex backgrounds. To overcome this, DyHead introduces dynamic adjustments to the detection head's structure and parameters, enabling adaptive modification based on the input feature map's content and resolution. This flexibility optimizes feature extraction and detection strategies across diverse scenarios, improving accuracy and enhancing model robustness.

Enhancement of the model's focus on key features

To enhance the model's focus on key features and improve small-target detection, this paper introduces the SegNext Attention mechanism. Traditional attention mechanisms often prioritize global information, risking the neglect of critical features and information loss, especially in complex backgrounds with small targets. SegNext Attention addresses this by emphasizing regions of interest (ROI), enabling precise capture of target details. In complex environments, it effectively suppresses background noise and interference, improving small-target detection accuracy.

The remainder of the paper is organized as follows: Chapter 2 reviews related work. Chapter 3 provides a detailed description of the improved LRDS-YOLO UAV detection model. Chapter 4 outlines the experimental setup and parameter configuration, and conducts ablation and comparative experiments on the open-source VisDrone 2019 dataset[10]. Furthermore, to evaluate the performance of LRDS-YOLO on other datasets, comparative and ablation experiments are also performed on the publicly available infrared UAV small target dataset HIT-UAV[11], with visual interpretations of the experimental results to validate the superiority of the proposed model. Finally, the paper discusses potential future research directions.

Related work

Object detection algorithm

In recent years, deep learning-based object detection algorithms have been mainly categorized into two types: region proposal-based and regression-based methods. Region proposal-based object detection algorithms, also known as two-stage methods, divide the object detection process into two stages: first, generating region

Fig. 1. Images of small targets in different environments.

proposals, and then classifying and refining the positions of these proposals using a classifier. Representative methods in this category include Fast-RCNN[12] and Faster-RCNN[13], which use convolutional neural networks (CNN) to generate candidate bounding boxes during the training phase and then classify them using deep convolutional networks to determine the object categories. Although two-stage detection algorithms achieve high detection accuracy, they are relatively slow and may not meet the requirements for real-time applications.

In contrast, regression-based object detection algorithms adopt a single-stage approach, directly regressing the predicted object locations. Representative methods of this type include the YOLO series[14] and SSD series[15]. Single-stage detection algorithms combine object localization and classification tasks into a unified framework, aiming to rapidly identify object positions while maintaining high detection accuracy. Additionally, these methods typically have smaller model sizes, making them more suitable for hardware deployment in practical scenarios. As a result, such models have been widely adopted in object detection tasks.

Small target detection

Despite significant progress in the detection of medium and large objects, challenges remain when it comes to detecting weak small targets, which are prevalent in various image datasets[16,17]. These small targets typically exhibit the following characteristics: (1) they occupy only a few pixels in the image; (2) high-frequency details, such as texture information, boundary cues, and color, exhibit substantial variations and are highly susceptible to interference from current imaging conditions and complex backgrounds[18]. Additionally, small targets often carry critical information, making their effective detection crucial for enhancing the performance of systems in practical applications.

To address the challenges in small target detection, many researchers have been making continuous efforts in this area. Hu et al.[19] were the first to introduce YOLOv3-based algorithms for UAV target detection. They utilized feature maps from the last four scales to predict object bounding boxes and adjusted the number of anchor boxes by calculating the size of the UAV based on the input data. This method not only improved detection accuracy but also yielded more precise UAV bounding boxes. Huang et al.[20], focusing on the Yolov8 model, incorporated shadow convolution at its neck, added EMA attention, and improved the detection head with DCNv2 to enhance the detection accuracy of UAVs for small targets in aerial imagery.Li et al.[21] proposed a novel method called "Perceptual Generative Adversarial Network" (Perceptual GAN), which leverages the generator to extract detailed information from low-level features and incorporates it into the features of small targets. Meanwhile, the discriminator evaluates the enhancement in detection performance brought by the generated features, optimizing small target detection. Cascade R-CNN[22] introduced a cascading detection strategy that progressively refines object localization and classification through multiple stages, improving detection accuracy. Shi et al.[23] proposed a FocusDet method, which effectively utilizes the fusion of shallow and deep features, preserving features of large objects while supplementing the features of small targets. Li et al.[24] proposed a Texture and Boundary Aware Network (TBNet), which enhances the detection of weak small targets by introducing a Texture Aware Enhancement Module (TAEM) and a Boundary Aware Fusion Module (BAFM). Tan et al. introduced the EfficientDet[25] detection model, which integrates EfficientNet as the backbone network and employs BiFPN (bidirectional feature pyramid network)[22] to handle multi-scale features. The model structure is optimized using Neural Architecture Search (NAS)[26], improving both detection efficiency and accuracy. Zhao et al.[27] enhanced the YOLOv5 model by integrating the Transformer encoder module, global attention mechanism, and coordinated attention mechanism into the C3 module, enabling fast and accurate small target detection in complex environments. The robustness and generalization performance were validated on the custom SUAV-DATA dataset.

In recent years, Transformer-based architectures have demonstrated remarkable performance in object detection tasks, owing to their superior global modeling capabilities. Unlike traditional convolutional neural networks (CNNs), Transformers effectively capture long-range dependencies, which is particularly beneficial for detecting small and sparsely distributed objects. This section reviews several representative Transformer-based detection frameworks that are highly relevant to small object detection.

Wang et al. proposed the Pyramid Vision Transformer (PVT)[28] as a general backbone for dense prediction tasks. By introducing a pyramid structure with progressively downsampled feature maps, PVT enables multi-scale feature extraction while preserving the resolution of small object representations. Compared with CNN-based backbones, PVT exhibits stronger performance in high-resolution input scenarios, making it suitable for small object detection in remote sensing and surveillance images.Liu et al.[29] introduced the Swin Transformer, which applies a hierarchical design and shifted window-based self-attention mechanism. This approach significantly reduces the computational complexity of global attention while maintaining local contextual modeling. The Swin Transformer has been widely adopted as a backbone in detection frameworks such as Faster R-CNN and YOLOX, achieving notable improvements in small object recall rates due to its fine-grained feature aggregation and scale-aware representation learning.To address the slow convergence and poor performance on small objects of the original DETR, Zhu et al.[30] proposed Deformable DETR, which incorporates deformable attention modules. These modules attend to a sparse set of key sampling points, allowing the model to focus on relevant regions with less computational overhead. This deformable mechanism enhances the detector's ability to localize small or irregularly distributed objects more accurately, especially in dense scenes.DINO (DETR with Improved Queries and Optimization)[31] builds upon the Deformable DETR architecture by introducing dynamic anchor boxes, query selection strategies, and denoising training. These innovations result in faster convergence and significantly improved detection performance on small objects. DINO represents the current state of the art in Transformer-based object detectors and serves as a strong benchmark for evaluating detection models on challenging datasets involving small and low-contrast targets.

Optimizing small object detection for edge devices

To deploy models on edge computing devices and enhance their accuracy, Ni et al.[32] proposed an enhanced small target detection model based on YOLOv8s. By introducing a Parallel Multi-Scale Feature Extraction (PMSE) module, a Scale Compensation Feature Pyramid Network (SCFPN), and an ultra-small target detection layer, the model significantly improves the detection accuracy of small targets in UAV imagery. Zeng et al. proposed the SCA-YOLO[33] method, which effectively enhances small target detection accuracy in UAV images by incorporating a hybrid attention module, an improved Simple Efficient Bottleneck (SEB) module, and a multi-layer feature fusion structure.Song et al. proposed the multi-scale hybrid attention-based MHA-YOLOv5[34], which enhances small target detection accuracy in UAV imagery by introducing a multi-scale attention module, a foreground enhancement module, and a depthwise separable channel attention module. Experimental results demonstrate significant improvements in mean Average Precision (mAP) across multiple datasets. Sun et al. introduced a real-time small target detection algorithm, RSOD[35], which improves small target detection performance in UAV traffic images by leveraging shallow feature maps for position prediction, fusing shallow and deep features, and enhancing the SE attention mechanism.

In current research, many scholars tend to focus on improving model accuracy or reducing model size, often overlooking the importance of balancing model size and accuracy. Therefore, this study aims to develop a detection model that achieves a balance between high accuracy, compact model size, and excellent FPS performance.

Method

YOLOv11

YOLOv11[36] consists of three main components: the backbone network, the neck structure, and the detection head. YOLOv11 builds upon YOLOv8[37] with improvements that introduce a more efficient architecture, optimizing both model accuracy and inference speed. Several key modifications have been integrated into the architecture, further enhancing object detection performance. The backbone network employs an improved C3k2 module, which serves as an optimized version of the Cross Stage Partial (CSP) Bottleneck. This is achieved by replacing large convolutions with two smaller convolutional layers, significantly improving computational efficiency while preserving rich feature representation capabilities. Additionally, the backbone integrates Spatial Pyramid Pooling-Fast (SPPF) and Cross Stage Partial with Spatial Attention (C2PSA) modules, which enhance the model's spatial attention focusing ability, enabling it to more accurately capture information from key regions.

In the neck structure, YOLOv11 utilizes the improved C3k2 module for efficient multi-scale feature aggregation, while combining the C2PSA module to further optimize detection performance for small targets and objects in complex backgrounds. The detection head employs multi-path C3k2 modules to process feature maps at different scales and incorporates a Convolution-BatchNorm-SiLU (CBS) module before the output layer. This further refines the features, ensuring that the final outputs-bounding boxes, object confidence scores, and classification results are more accurate. A diagram of the YOLOv11 model architecture is shown in Fig. 2.

In this paper, we propose several improvements to YOLOv11 to enhance object detection accuracy while minimizing computational overhead and model parameters, thus achieving more efficient small target detection performance.

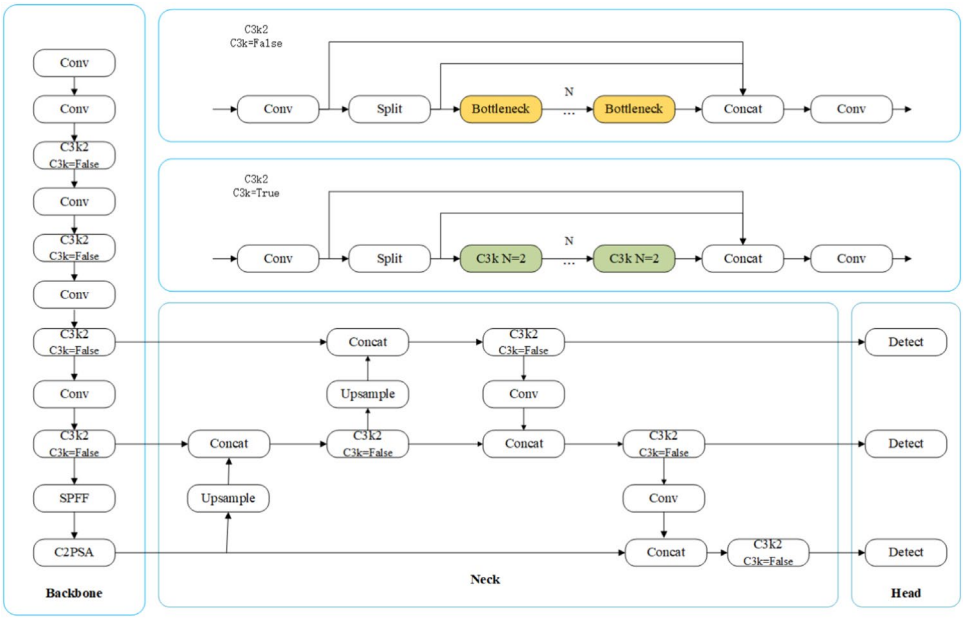

Fig. 2. YOLOv11 model structure.

LRDS-YOLO

To improve UAV target detection accuracy, achieve model lightweighting, and reduce deployment costs, this paper proposes an efficient detection model based on YOLOv11, called LRDS-YOLO. The model structure is illustrated in Fig. 3.

Compared to YOLOv11, we have made the following optimization improvements: First, we introduced the lightweight LAD (Light Adaptive-weight Downsampling) mechanism and the SegNext Attention mechanism into the backbone network to enhance feature extraction efficiency and global representation capability, thereby improving the model's ability to capture small targets. Additionally, we adopted the Re-Calibration Feature Pyramid Network (Re-Calibration FPN), which further enhances the model's adaptability to variations in target location, orientation, and scale, thereby improving detection accuracy and robustness. Finally, the DyHead dynamic adjustment mechanism is integrated into the detection head to dynamically allocate attention to multi-scale features, effectively optimizing the model's detection ability. These improvements not only significantly enhance the model's detection performance but also add minimal computational overhead compared to the original model, better meeting the lightweight and small target detection requirements in UAV scenarios.

LAD (light adaptive-weight downsampling)

Traditional backbone networks often suffer from severe information loss when processing small targets, primarily due to the uniform and indiscriminate nature of conventional downsampling operations. Standard convolutional neural networks (CNNs) typically employ strided convolutions or pooling layers to reduce the spatial resolution of feature maps, thereby enhancing computational efficiency. However, this fixed downsampling strategy fails to differentiate between informative regions and background noise, leading to the loss of fine-grained details that are critical for small-object detection. As small targets often occupy only a few pixels in the image, their features are prone to dilution or elimination during multi-layer feature fusion. Moreover, conventional downsampling methods lack the capability to prioritize salient features, causing target boundaries to become indistinct especially under cluttered backgrounds or low-contrast conditions.

To address these limitations, this paper proposes a lightweight and adaptive downsampling strategy, termed Light Adaptive-weight Downsampling (LAD),as shown in Fig. 4, which introduces a dual-path architecture to dynamically retain task-relevant information during spatial compression. Specifically, LAD integrates a global attention-based weighting mechanism with a localized group convolution path to jointly emphasize discriminative regions and preserve spatial detail.

In the first path, LAD generates pixel-wise adaptive weights that guide the downsampling process toward semantically meaningful areas. Given an input feature map $X \in \mathbb{R}^{B\times C\times H\times W}$, global context is first extracted

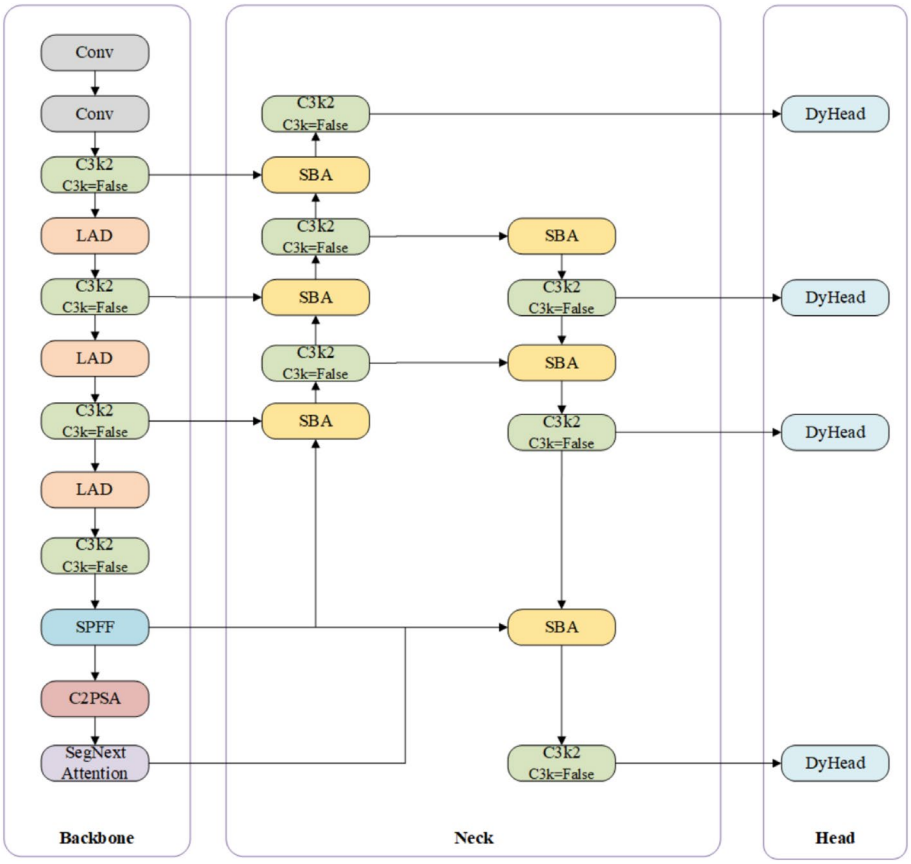

Fig. 3. LRDS-YOLO model structure.

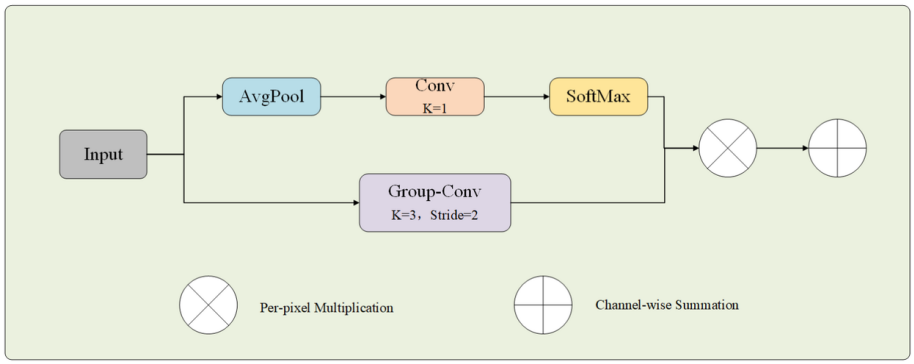

Fig. 4. Model structure of the LAD downsampling module.

using average pooling over a non-overlapping kernel of size $k_H \times k_W$ and stride $s_H \times s_W$, resulting in pooled features:

$$Y_{(b,c,h',w')} = \frac{1}{k_H k_W} \sum_{i=0}^{k_H-1} \sum_{j=0}^{k_W-1} X_{(b,c,h\cdot\frac{s_H}{k_H}+i,w\cdot\frac{s_W}{k_W}+j)} \tag{1}$$

These pooled features are then projected via a 1×1 convolution:

$$Z_{\text{conv},c',h',w'} = \text{Conv}_{1\times1}(Y) \tag{2}$$

followed by softmax normalization to produce the spatial attention map:

$$\text{Weights} = \text{Softmax}(Z). \tag{3}$$

This attention map dynamically assigns higher weights to pixels likely associated with small targets, enhancing their prominence during feature compression.

In the second path, the original features are processed through a 3×3 group convolution with eight groups and a stride of 2 to reduce spatial dimensions while preserving local structure. Simultaneously, the number of output channels is expanded from C to $4C$, increasing feature diversity and alleviating the risk of over-compression. The output features W from this branch are then modulated by the adaptive weights via element-wise multiplication:

$$W_{g,b,c',\frac{h}{2},\frac{w}{2}} = \text{Group-Conv}(X_{b,c,h,w}, K=3, \text{groups}=8, \text{stride}=2) \tag{4}$$

$$F_{\text{out}} = \text{Weights} \odot W. \tag{5}$$

This fusion ensures that critical object information is preserved and amplified, while redundant or irrelevant background features are suppressed.

SegNext attention

To enhance the detection of small objects, this work incorporates a SegNext Attention mechanism between the encoder and decoder,its structure is shown in Fig. 5, aiming to improve the network's capacity for selectively modeling fine-grained features while suppressing irrelevant background noise. Unlike conventional designs that rely on computationally expensive self-attention, this module adopts a lightweight convolutional structure, which is more suitable for preserving the spatial precision required in small object detection. By generating pixel-wise attention maps, the model can focus more effectively on critical regions where small targets appear, thereby mitigating the information loss commonly introduced by downsampling and deep-layer feature aggregation.

The overall architecture consists of an encoder, a convolution-based attention module, a decoder, and a detection head guided by a loss function. The encoder follows a hierarchical design inspired by Vision Transformers[38,39], but replaces self-attention with a Multi-Scale Convolutional Attention (MSCA) module tailored for spatially sparse and small-scale objects. The MSCA module comprises three sequential stages: a depthwise convolution to retain local detail, a multi-branch strip convolution module to capture anisotropic and multi-scale context, and a final 1×1 convolution to model inter-channel interactions and generate a spatial attention map. This attention map serves as a pixel-level weighting mask that reassigns importance across spatial locations in the input feature map.

Given an input feature map F, the MSCA attention weights are computed as:

$$Att = Conv_{1\times1}\left(\sum_{i=0}^{3} Scale_i\left(DW - Conv\left(F\right)\right)\right) \tag{6}$$

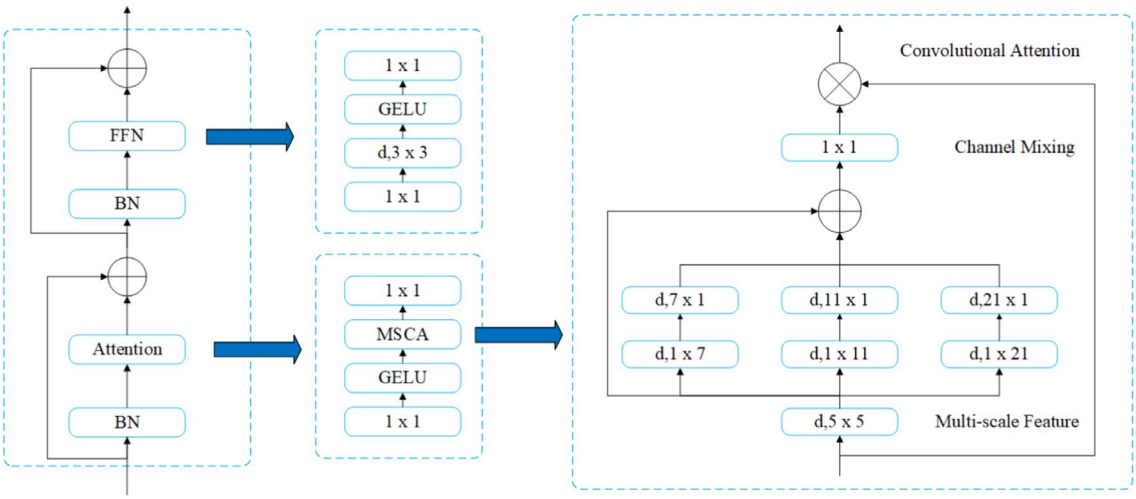

Fig. 5. Illustration of the SegNext Attention. Here, d, $k1 \times k2$ means a depth-wise convolution (d) using a kernel size of $k1 \times k2$. We extract multi-scale features using convolutions and then utilize them as attention weights to reweigh the input of MSCA(Mulit-scale Feature).

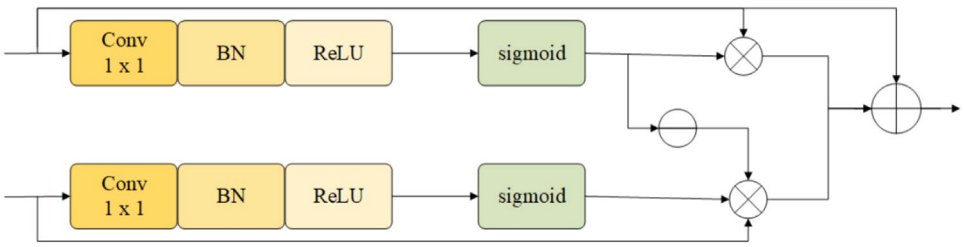

Fig. 6. Re-calibrationFPN model structure.

where DW-Conv denotes depthwise convolution, and Scale_i represents the *i*-th branch in the strip convolution module, with Scale_0 being an identity connection used to preserve low-frequency information. The final output of the module is then obtained via element-wise multiplication:

$$Out = Att \bigotimes F \tag{7}$$

This process adaptively enhances object-relevant features while suppressing background responses. Because small targets often have weak activations that are easily overwhelmed, this spatial reweighting strategy significantly improves the network's focus on subtle yet important cues. Embedding the attention module between the encoder and decoder enables the model to preserve and refine discriminative features throughout the hierarchy, thereby improving localization accuracy and maintaining semantic integrity across scales. This contributes to more reliable small object detection in cluttered or complex visual environments.

Re-calibrationFPN

In the task of small object detection, one of the fundamental challenges lies in the heterogeneity of features extracted at different layers of convolutional neural networks. Specifically, shallow feature maps retain abundant spatial details such as object contours and textures - but lack sufficient semantic abstraction. In contrast, deeper layers capture rich semantic representations, yet suffer from significant loss in spatial resolution due to repeated downsampling operations. This inherent inconsistency between semantic richness and spatial precision severely limits the effectiveness of cross-level feature fusion in conventional Feature Pyramid Networks (FPN). As a consequence, traditional FPN architectures encounter two major issues: semantic-detail mismatch and boundary degradation. The former arises from semantic discrepancies between high-resolution shallow features and low-resolution deep features, often leading to redundant or conflicting information during fusion, which in turn compromises the accurate localization of small objects. The latter results from the loss of boundary information caused by successive downsampling in deeper layers, thereby blurring object contours and degrading overall detection accuracy.

To address these limitations, we propose a novel architecture termed the Re-Calibration Feature Pyramid Network (Re-Calibration FPN), which facilitates more adaptive and resolution-aware multi-scale feature integration. As illustrated in Fig. 6, the core component of Re-Calibration FPN is the Selective Boundary

Aggregation (SBA) module (see Fig. 7), which introduces a bidirectional feature interaction mechanism in conjunction with a dynamic attention modulation strategy.

Unlike conventional fusion approaches that employ static feature aggregation, RC-FPN first enhances the mutual representation of two input feature maps F_s(deep semantic features) and F_b(shallow boundary features) prior to fusion. As shown in Figure 6, the shallow and deep features are processed independently by two separate Re-Calibration (RC) blocks. On one hand, shallow features compensate for the missing boundary details in deeper layers; on the other, semantic cues from the deep layers reinforce the abstract representation of shallow features. The outputs from both RC blocks are subsequently integrated through a 3×3 convolutional operation to form the final fused representation.

Within each RC block, we employ a Progressive Aggregation Unit (PAU) to reconstruct and refine the input features. The detailed formulation of the PAU is as follows:

$$T_1' = W_\theta\left(T_1\right), T_2' = W_\varphi\left(T_2\right) \tag{8}$$

$$PAU\left(T_1, T_2\right) = T_1' \odot T_1 + T_2' \odot T_2 \odot \left(\ominus\left(T_1'\right)\right) + T_1 \tag{9}$$

Here, T_1 and T_2 represent the input features. Two linear mappings and S-shaped functions, denoted as $W_\theta(\cdot)$ and $W_\varphi(\cdot)$, are applied to the input features to reduce the channel dimension to 32, yielding the feature maps T_1' and T_2'. The symbol $\odot$ represents point-wise multiplication, while $\ominus(\cdot)$ refines imprecise and coarse estimations into accurate and complete prediction maps by subtracting the feature T_1'. A 1×1 convolution operation is employed as the linear mapping process.

The SBA module adopts a dual-path fusion mechanism. In the shallow-to-deep path, boundary-rich features $F_b \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times 32}$ are injected into the deeper semantic features $F_s \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times 32}$ to enhance their spatial detail representation. Simultaneously, in the deep-to-shallow path, semantic context from the deep features F_s is propagated back to the shallow features F_b, thereby suppressing noise and improving semantic consistency in the lower layers. The outputs of the two paths are concatenated and then fused via a 3×3 convolution, as defined by:

$$Z = C_{3\times3}(Concat\left(PAU\left(F^s, F^b\right), PAU\left(F^b, F^s\right)\right)) \tag{10}$$

Where, $C_{3\times3}(\cdot)$ represents a 3×3 convolution with batch normalization and a ReLU activation layer. The function $\mathrm{Concat}(\cdot)$ denotes the concatenation operation along the channel dimension. Finally, $Z \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times 32}$ is the output of the SBA module.

DyHead

In small object detection tasks, targets are often characterized by limited scale, weak texture, and sparse distribution, which imposes stricter demands on the feature sensitivity and fine-grained representation capability of the detection head. To address these challenges, the LRDS-YOLO model adopts DyHead (Dynamic Head), the structure of which is shown in Fig. 8, to replace the conventional static detection head, aiming to enhance the model's responsiveness and localization accuracy for small targets.

DyHead introduces dynamic attention mechanisms along the level, spatial, and channel dimensions of the feature tensor. This design guides the network to focus on regions likely to contain small objects, suppresses irrelevant background responses, and adaptively adjusts feature importance based on contextual cues. Implemented using lightweight residual structures, DyHead effectively captures the diverse properties of small targets such as scale variation and irregular spatial distribution.

Given a three-dimensional feature tensor $F \in \mathbb{R}^{L \times S \times C}$ from the backbone, DyHead applies attention functions in sequence along the level, spatial, and channel dimensions. The level-wise attention emphasizes features from different semantic depths, facilitating multi-scale representation learning. Spatial attention enhances the model's ability to localize salient regions, particularly where small objects are present. Channel attention further refines semantic feature selection by emphasizing informative channels and suppressing redundant ones. This process can be formalized as:

$$W\left(F\right) = \pi_C(\pi_S\left(\pi_L\left(F\right)F\right)F) \cdot F \tag{11}$$

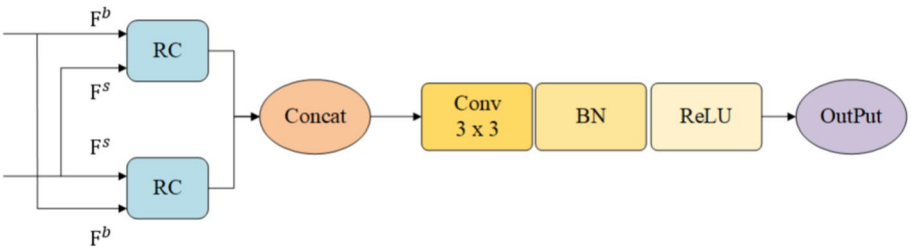

Fig. 7. SBA model structure.

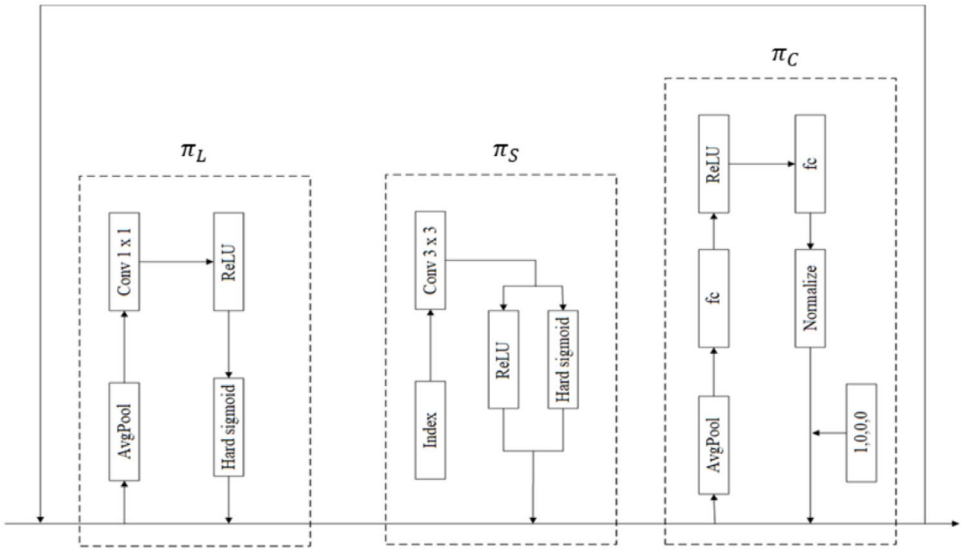

Fig. 8. DyHead model structure.

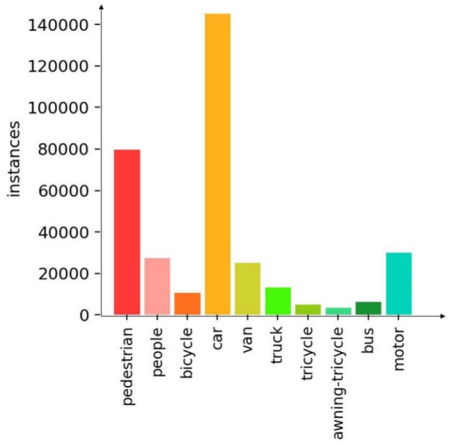

Fig. 9. Distribution of the number of VisDrone2019 training set instances.

where $\pi_L(\cdot)$, $\pi_S(\cdot)$, and $\pi_C(\cdot)$ represent attention operations along the level, spatial, and channel dimensions, respectively. The attention weights are applied via element-wise multiplication, dynamically enhancing target-relevant features and mitigating background interference. This mechanism improves the model's capacity to preserve subtle cues necessary for detecting small objects and compensates for potential information loss caused by feature compression. Experimental results confirm that integrating DyHead significantly improves detection performance under challenging conditions such as dense clutter and low contrast, yielding higher accuracy and better robustness in small-object detection scenarios.

Experiments and results

Datasets

The dataset used in this study is the publicly available VisDrone2019 dataset[10]. This dataset consists of a total of 8599 static images captured from drones at high altitudes, with 6471 images used for training, 548 for validation, and 1580 for testing. The image categories include pedestrian, person, bicycle, car, van, truck, tricycle, canopy-tricycle, bus, and motorcycle, with a total of 2.6 million annotations. The distribution of training set instances is shown in Fig. 9.

The images in the VisDrone2019 dataset have two resolutions: 960×540 and 1360×765. The size distribution of each instance is shown in Fig. 11. As can be seen in Fig. 10, the majority of the target instances have aspect ratios smaller than 0.1 times the dimensions of the entire image, which meets the definition of small target relative size.

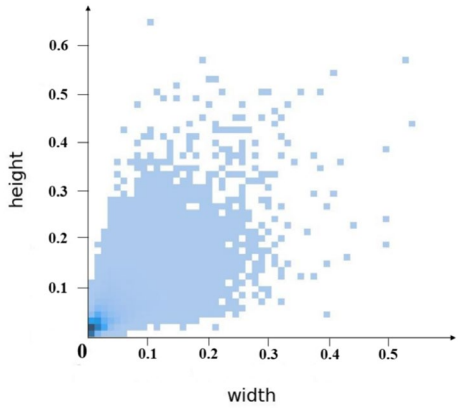

Fig. 10. Target size distribution of the VisDrone2019 dataset.

Evaluation indicators

In this experiment, the performance of the LRDS-YOLOE model is evaluated using the following metrics: Precision (P), Recall (R),F1-score(F1), Mean Average Precision (mAP), Parameters, GFLOPs (Giga-FLOPs per second) and FPS (Frames Per Second).

(1) Precision: Precision refers to the ratio of the number of true targets to the total number of detected targets. The precision formula is:

$$P = \frac{TP}{(TP + FP)} \tag{12}$$

where *TP* is the number of true positives and *FP* is the number of false positives.

(2) Recall: Recall refers to the ratio of the number of detected targets to the total number of true targets. The recall formula is:

$$R = \frac{TP}{(TP + FN)} \tag{13}$$

where *FN* represents false negatives, which are the targets that were not detected among the true targets.

(3) F1-score: F1-score is the harmonic mean of precision and recall. It provides a balanced measure of accuracy by considering both false positives and false negatives. The F1-score formula is:

$$F1 = 2 \times \frac{P \times R}{P + R} \tag{14}$$

where *P* is precision and *R* is recall.

(4) Mean average precision (mAP): mAP@0.5 is the mean detection precision for all classes at an IoU threshold of 0.5; mAP@0.5:0.95 is the mean detection precision for all IoU thresholds ranging from 0.5 to 0.95, with a step size of 0.05. In object detection, a higher mAP value indicates better model performance. The formula is:

$$mAP = \frac{\sum AveragePrecision(c)}{Num(cls)} \tag{15}$$

where *AveragePrecision*(*c*) is the average precision for a specific class c, and *Num*(*cls*) is the number of categories in the dataset.

(5) Number of parameters: This metric evaluates the size and complexity of the model and is obtained by summing the number of weight parameters for each layer.

(6) GFLOPs (Giga-FLOPs): Represents the number of floating-point operations executed per second during inference.

(7)FPS (frames per second): FPS refers to the number of image frames that a detection model can process per second. It reflects the real-time performance of the model. The FPS formula is:

$$FPS = \frac{N}{T} \tag{16}$$

where *N* is the total number of processed frames and *T* is the total processing time in seconds.

Name	Parameter
Operating system	Ubuntu 22.04
CPU	Intel(R) Xeon(R) Platinum 8383C
GPU	NVIDIA RTX 4090
CUDA	11.7
Pytorch	1.13.7

Table 1. Experimental environment.

Name	Parameter
Learning rate	0.01
Momentum	0.937
Optimizer	SGD
Batch size	8
Image size	640

Table 2. Training parameters.

Experimental environment

The experimental environment is based on Ubuntu 22.04, Python 3.8.13, and Pytorch 1.13.7+CUDA11.7. The relevant hardware configurations and model parameters are shown in Tables 1 and 2. The batch size is set to 8, the number of training epochs is 300, and the learning rate is set to 0.01. The experiments use an adaptive image size of 640×640.

Comprehensive comparative experiment

To demonstrate the superiority of the LRDS-YOLO model, it is compared with currently popular object detection models. In the same experimental environment and with identical configurations and parameters, detection experiments are conducted on the VisDrone2019 dataset. The results of the model comparison experiments are shown in Table 3.

Based on the comparative results presented in Table 3, LRDS-YOLO demonstrates superior performance in both detection accuracy and computational efficiency compared to a wide range of YOLO variants, including the most recent models. Specifically, LRDS-YOLO achieves a mean Average Precision at IoU 50% (mAP50) of 43.6% and a mAP50:95 of 26.6%, outperforming the majority of lightweight and even some large-scale models. For example, its mAP50 significantly exceeds that of YOLOv3-tiny (23.6%), YOLOv5n (32.9%), YOLOv8n (33.1%), and even YOLOv5s (39.3%). Compared to newer models such as YOLOv11n (mAP50: 37.7%), YOLOv11s (40.1%), and Drone-YOLO (35.4%), LRDS-YOLO still delivers superior detection accuracy. In terms of mAP50:95, it also outperforms CPDD-YOLOv8 (23.5%) and YOLOv11n (22.5%), indicating its robustness in handling more challenging detection thresholds.

Despite these strong performance metrics, LRDS-YOLO remains highly efficient, requiring only 4.17 million parameters and 24.1 GFLOPs. This is substantially lower than recent models such as CPDD-YOLOv8 (206M parameters, 141.9 GFLOPs) and YOLOv11l (25.28M parameters, 86.6 GFLOPs), both of which are far more computationally intensive yet fail to offer a comparable increase in detection performance.

These results underscore the effectiveness of LRDS-YOLO in achieving an optimal trade-off between accuracy and efficiency. Its lightweight design, combined with its strong detection capabilities particularly for small and difficult targets-makes it a promising candidate for real-time deployment on resource-constrained platforms such as drones, embedded systems, and edge devices.

Ablation experiments

To validate the effectiveness of the proposed improvements, we conducted the following ablation experiments. Based on the YOLOv11 network, the following modifications were introduced: integrating a lightweight adaptive downsampling mechanism, reconstructing the feature extraction pyramid, incorporating the DyHead dynamic adjustment detection head, and finally adding the SegNext Attention mechanism. Each improvement module was added sequentially for experimentation, and the results are presented in Table 4 and Fig. 12.

Based on the results of the ablation experiments, the following conclusions can be drawn: the introduction of LAD and Re-Calibration FPN increased mAP50 from 32.2% to 39.7% and mAP50:95 from 18.6% to 24.3%, demonstrating their crucial role in enhancing model accuracy, particularly in complex scenarios. With the additional integration of DyHead, mAP50 and mAP50:95 further improved to 43.0% and 26.1%, respectively. The incorporation of the SegNext Attention mechanism further optimized detection performance, maintaining mAP50 at 43.6%, enhancing attention to critical features, and improving small-object detection. Although these improvements increased the computational complexity of the model, the trade-off was justified given the significant accuracy gains, ultimately achieving a well-balanced trade-off between precision and efficiency.

Model	Precision (%)	Recall (%)	F1 (%)	mAP50 (%)	mAP50:95 (%)	Parameters(M)	GFlops	FPS
Yolov3-tiny	39.1	24.3	22.5	23.6	13.2	9.52	14.3	17
Yolov5n	44.5	33.2	38.0	32.9	19.1	2.18	5.8	23
Yolov5s	51.1	38.1	43.7	39.3	23.4	7.81	18.8	26
Yolov5m	47.7	36.8	37.0	39.4	23	20.88	48	14
Yolov5l	50.7	38.6	43.9	41.4	24.6	46.15	107.8	8
Yolov5x	52.1	40.4	41	43	26	20.39	86.23	5
Yolov6s	40.3	30.5	30.2	30.2	17.7	4.15	11.5	-
Yolov7tiny	47.6	37.3	41.8	35.8	18.8	6.04	13.3	32
Yolov8n	45	33	33.1	38.1	19.2	2.68	6.8	25
Yolov8s	50.7	37.9	43.3	39.1	23.4	9.83	23.4	18
Yolov8m	53.3	41.1	46.4	42.5	26	23.2	67.5	8
Yolov9s	52	38	39.4	43.9	23.8	61.9	22.1	22
Yolov10n	45.0	34.5	39.1	34.5	19.9	2.26	6.5	36
Yolov10s	52.7	38	44.0	39.8	23.8	7.22	21.4	32
Yolov10m	55.1	42.1	47.4	44.2	26.9	15.31	58.9	30
Yolov11n	42.7	32.7	37.3	32.2	18.6	2.61	6.5	35
Yolov11s	49.9	38.7	43.5	39.4	23.6	9.41	21.3	34
Yolov11m	55.7	42.5	48.2	44.1	27.2	20.03	67.7	28
Yolov11L	55.5	43	48.3	44.4	27.5	25.28	86.6	20
rtdetr-r18	57.2	40	47.1	41.4	25.1	20	57	60
EL-YOLO[40]	48.8	40.3	43	42.9	24.8	6.7	1.08	35
YOLOv8-QSD[41]	44.2	38.6	34.2	34.6	16.8	-	-	-
Drone-YOLO[5]	-	-	40	41.4	25.1	-	5.35	-
CPDD-YOLOv8[42]	51.7	41.7	46.1	41.0	23.5	206	141.9	22
LRDS-YOLO	53.3	41.6	46.0	43.6	26.6	4.17	24.1	31

Table 3. Comparative experiment results for different models.

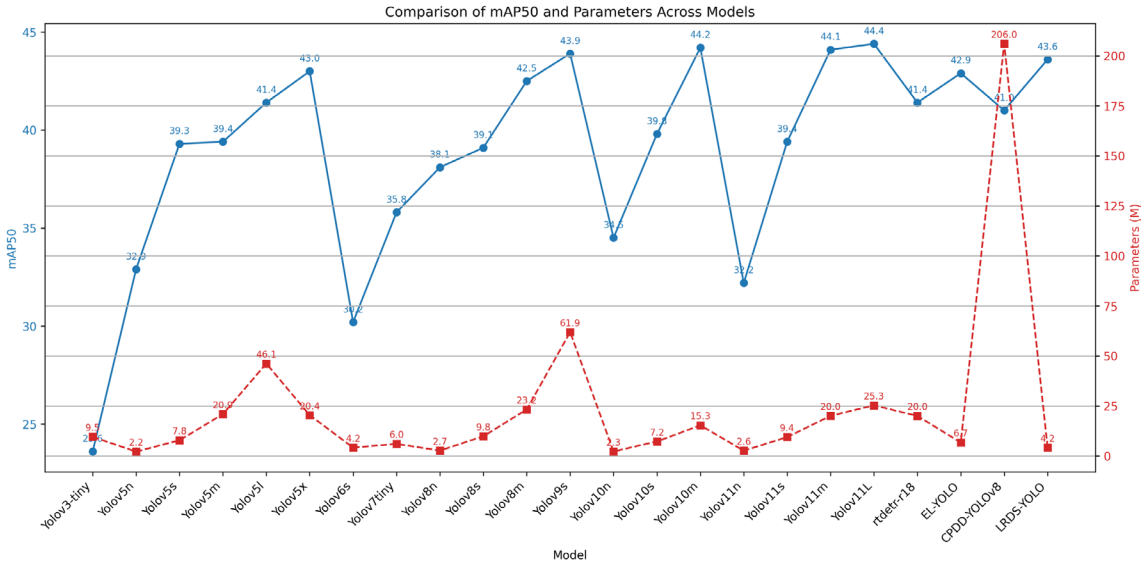

Fig. 11. Comparison of mAP50 of different models.

Attention comparison

According to the heatmap in Fig. 13, the second row presents the performance of the LRDS-YOLO model, while the third row shows the detection results of YOLOv11. The results clearly demonstrate that LRDS-YOLO exhibits significant advantages over YOLOv11 in terms of target localization and attention distribution.

Specifically, in the second-row images, LRDS-YOLO demonstrates more precise target attention, particularly in complex multi-object detection scenarios. The model effectively concentrates high attention on key targets, such as various types of vehicles and pedestrians, with darker regions in the heatmap indicating stronger attention

LAD	Re-calibration FPN	DyHead	SegNext attention	mAP50 (%)	mAP50:95 (%)	mAP_s (%)	Params (M)	GFLOPs (G)
				32.2	18.6	9.1	2.61	6.5
✓				33.2	19.4	9.4	2.27	6.6
	✓			39.5	24.2	13.4	3.85	18.8
		✓		36.2	21.5	11.1	3.13	7.6
			✓	33.3	19.3	9.4	2.71	6.6
✓	✓			39.7	24.3	13.9	3.49	17.9
✓	✓	✓		43.0	26.1	15.6	4.07	23.7
✓	✓	✓	✓	43.6	26.6	16.4	4.17	24.1

Table 4. Results of ablation experiments.

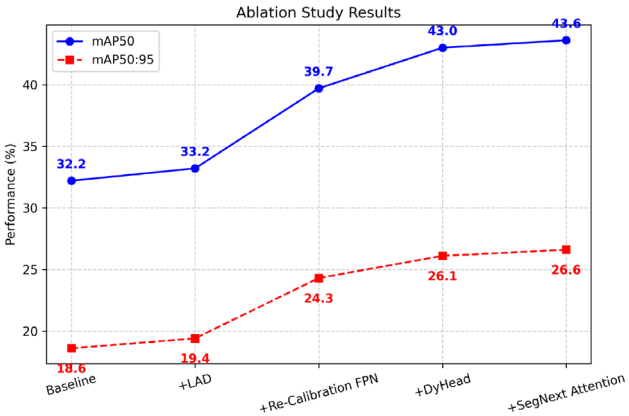

Fig. 12. Impact of ablation study on detection performance.

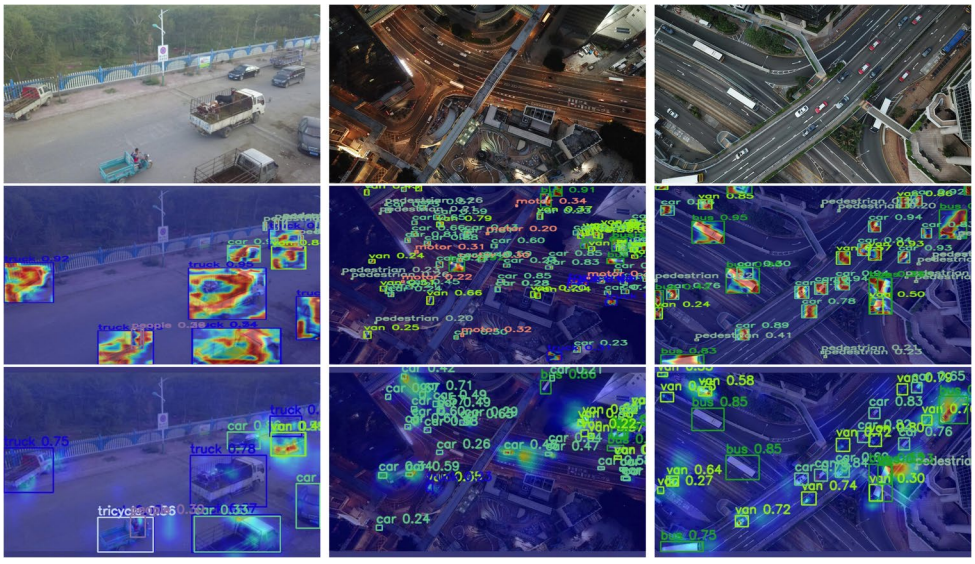

Fig. 13. Heatmap visualization of different models. The second row represents the LRDS-YOLO model, while the third row corresponds to the YOLOv11 model.

to target objects. In contrast, YOLOv11, as shown in the third row, exhibits relatively weaker performance, with its heatmap displaying more dispersed attention and lower focus on small or distant objects. This suggests that LRDS-YOLO is better equipped to handle challenging detection tasks by accurately focusing on target objects, thereby improving detection accuracy.

Moreover, LRDS-YOLO effectively reduces false detections in complex environments, demonstrating strong discriminative capability, particularly when handling multiple objects in dense scenes. In contrast, YOLOv11

exhibits relatively inferior performance, with its heatmap showing broader attention regions, leading to insufficient focus on certain targets and a subsequent decline in detection accuracy.

In summary, LRDS-YOLO outperforms YOLOv11 significantly through its precise attention mechanism and efficient target detection capability. This advantage is particularly evident in complex and densely populated scenes, where LRDS-YOLO excels in both target recognition and detection accuracy. These improvements make LRDS-YOLO more adaptable and effective for real-world applications, especially in detecting small and distant objects.

Comparison of different FPN architectures

To comprehensively evaluate the effectiveness and superiority of the proposed Recalibration FPN, we design a set of comparative experiments involving several representative feature pyramid architectures, including PANet, BiFPN, AFPN, and our Recalibration FPN. The experimental results are summarized in Table 5.

As shown in Table 5, the proposed Recalibration FPN achieves the best performance in terms of mAP50 and mAP50:95, reaching 39.5% and 24.2%, respectively, outperforming other mainstream FPN variants such as PANet, BiFPN, and AFPN. Notably, despite having significantly fewer parameters (3.85M) and lower computational complexity (18.8 GFlops), our model maintains competitive precision and demonstrates superior overall detection accuracy. This result highlights the effectiveness of the recalibration mechanism in enhancing multi-scale feature representation, and confirms that the proposed design achieves a favorable trade-off between accuracy and efficiency, making it particularly suitable for lightweight or real-time detection scenarios.

Visualization of detection results

Through comprehensive visual analysis of detection results under different environmental conditions, UAV flight altitudes, and lighting variations, as shown in Figs. 14, 15, and 16, we observe that the LRDS-YOLO model excels in object detection tasks within complex urban scenes captured by drones. The model demonstrates outstanding performance and robustness. It shows significant adaptability and stability across various scenarios, including daytime and nighttime environments, as well as high-traffic-density road environments. Furthermore, the model exhibits excellent multi-scale object detection capabilities under different UAV altitudes and perspectives, with particularly impressive accuracy in identifying and localizing densely distributed small objects.

Additional experiments

We conducted additional experiments on the HIT-UAV dataset[11] as shown in Fig. 17, to validate the broad applicability of the LRDS-YOLO model. The dataset comprises 2,898 infrared thermal images extracted from 43,470 frames in hundreds of videos captured by UAVs in various scenarios, such as schools, parking lots, roads, and playgrounds. Moreover, the HIT-UAV provides essential flight data for each image, including flight altitude, camera perspective, date, and daylight intensity. Table 6 presents the comparative experimental results against state-of-the-art methods.

As shown in Table 6, the LRDS-YOLO model achieves the highest mAP50 and mAP50:95 values on the HIT-UAV dataset, demonstrating its significant advantage in overall performance. However, for the detection of certain individual objects, some other models exhibit superior performance in specific cases. This phenomenon may be attributed to the design of LRDS-YOLO, which prioritizes balanced detection performance and efficiency across various scenarios, potentially at the expense of optimized performance for specific object types or scenes. Future research could focus on refining these specific aspects to further enhance the model's comprehensive capabilities. Additionally, the experiment highlights the broad applicability of the LRDS-YOLO model across diverse settings.

Discussion

The LRDS-YOLO model proposed in this paper demonstrates exceptional performance in UAV small object detection tasks, particularly in terms of detection accuracy and computational efficiency. By incorporating modules such as LAD lightweight adaptive downsampling, Re-Calibration FPN, SegNext Attention, and DyHead, LRDS-YOLO effectively enhances the accuracy of small object detection while maintaining a low computational overhead, making it suitable for real-time operation in resource-constrained environments.

However, despite LRDS-YOLO outperforming current mainstream algorithms in overall performance, other models still demonstrate stronger performance in detecting certain specific objects under particular conditions. This phenomenon may be attributed to LRDS-YOLO's design philosophy, which prioritizes balanced performance across various scenarios rather than deeply optimizing for specific object categories or scenes. Therefore, future research could focus on optimizations tailored to specific target types or environments to further enhance the model's performance in these specialized applications.

Model	Precision (%)	Recall (%)	mAP50 (%)	mAP50:95 (%)	Parameters (M)	GFlops
PANet	46.3	35.3	35.7	21	26.7	10.4
BiFPN	48.2	36.9	37.9	22.4	21	19.5
AFPN	47.9	37.7	38.3	22.9	29.0	19.6
Re-Calibration FPN	49.9	34.0	39.5	24.2	3.85	18.8

Table 5. Comparison of different FPN architectures.

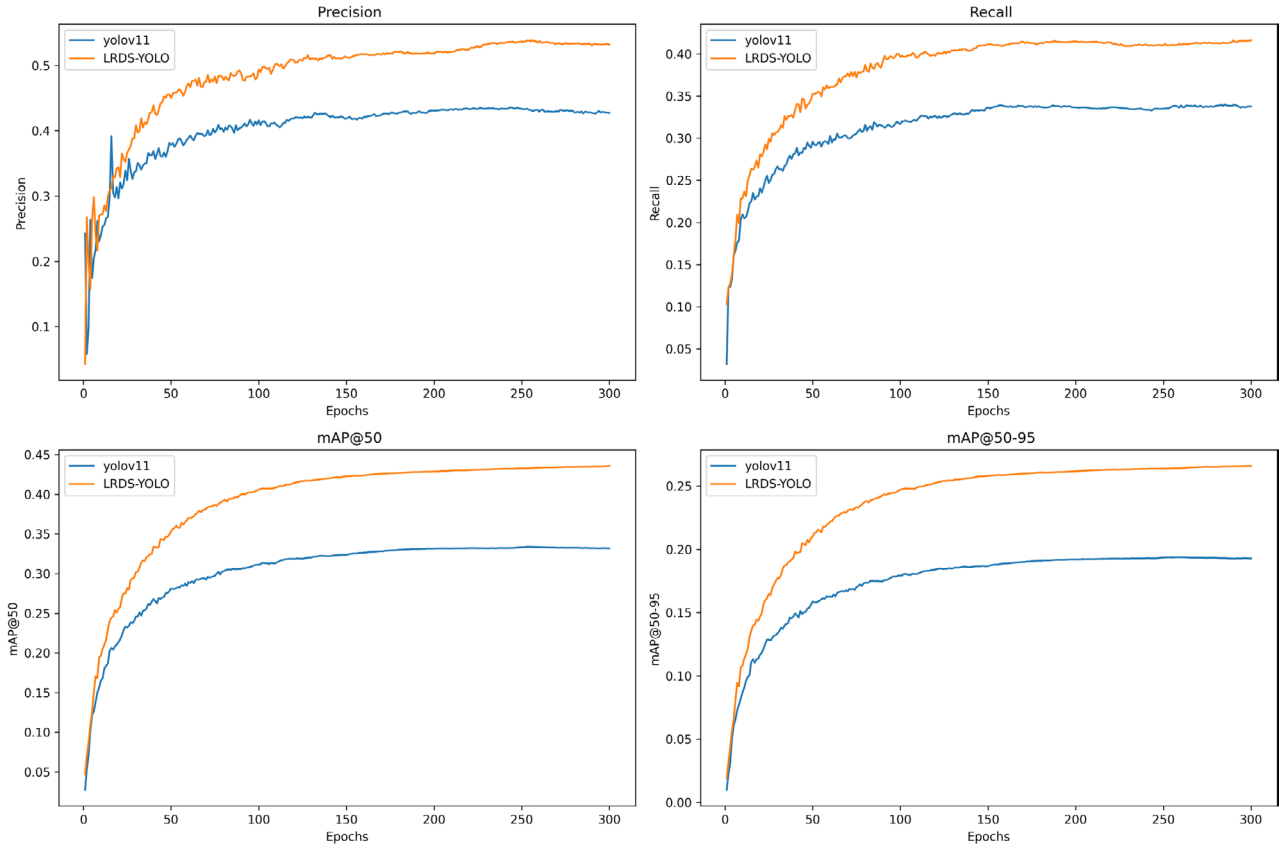

Fig. 14. Comparison of the accuracy of YOLOv11 and LRDS-YOLO.

In addition, LRDS-YOLO demonstrates exceptional capabilities in handling complex scenes, particularly in the detection of small and low-contrast targets. This advantage enables it to perform well in intricate urban environments and dynamic settings, offering significant potential for applications in fields such as drone-based aerial surveillance, traffic monitoring, and security. The model's robustness in these challenging contexts highlights its strong practical value in real-world use cases.

Summary

The LRDS-YOLO model has made significant strides in small object detection through several innovative designs, particularly achieving a good balance between accuracy and efficiency. Compared to other object detection models, LRDS-YOLO's robustness and efficiency in complex scenarios give it a substantial advantage in real-world applications. By incorporating modules such as LAD, Re-Calibration FPN, SegNext Attention, and DyHead, the model not only enhances its small object detection capabilities but also effectively reduces computational burden, meeting the requirements for real-time performance and efficiency. Overall, LRDS-YOLO, as an efficient small-object detection model, demonstrates wide application prospects, especially in the drone field, with considerable practical value and potential.

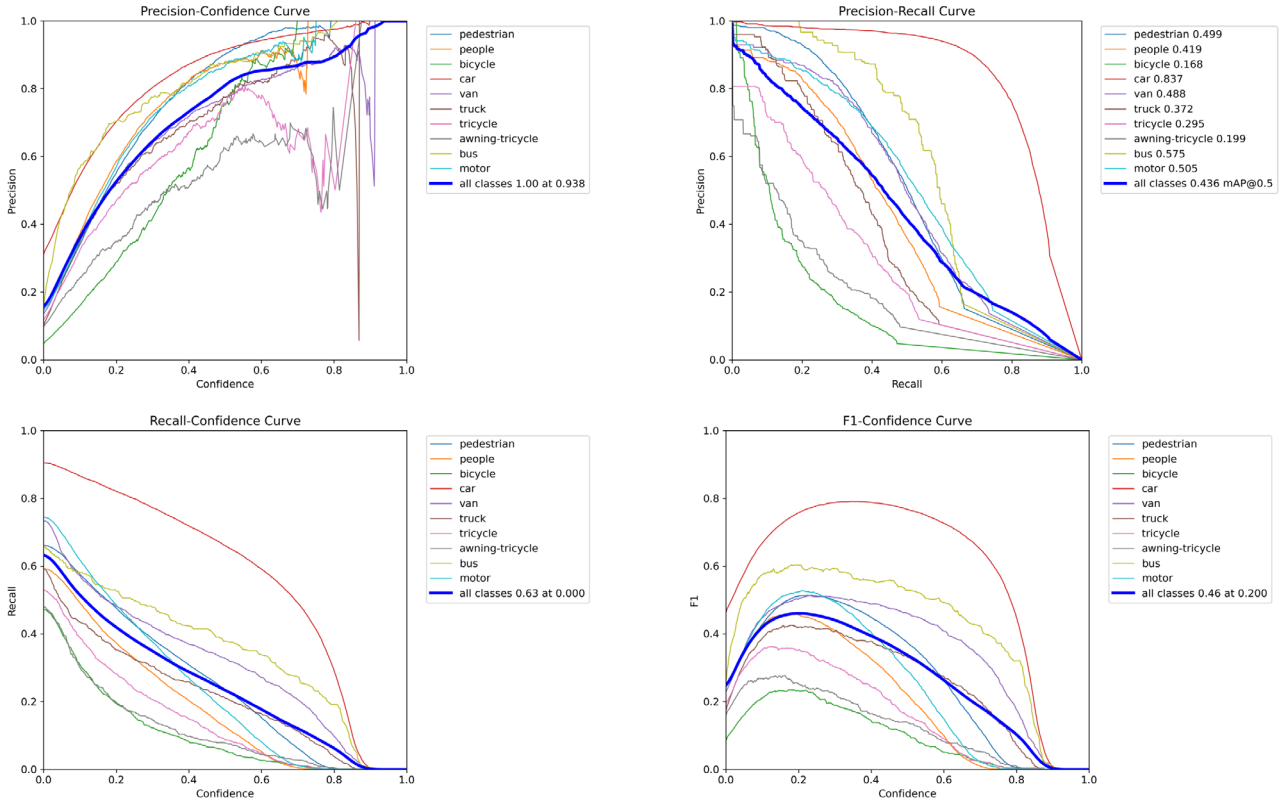

Fig. 15. The precision, mAP50, recall and F1 comparison curves of the LRDS-YOLO model.

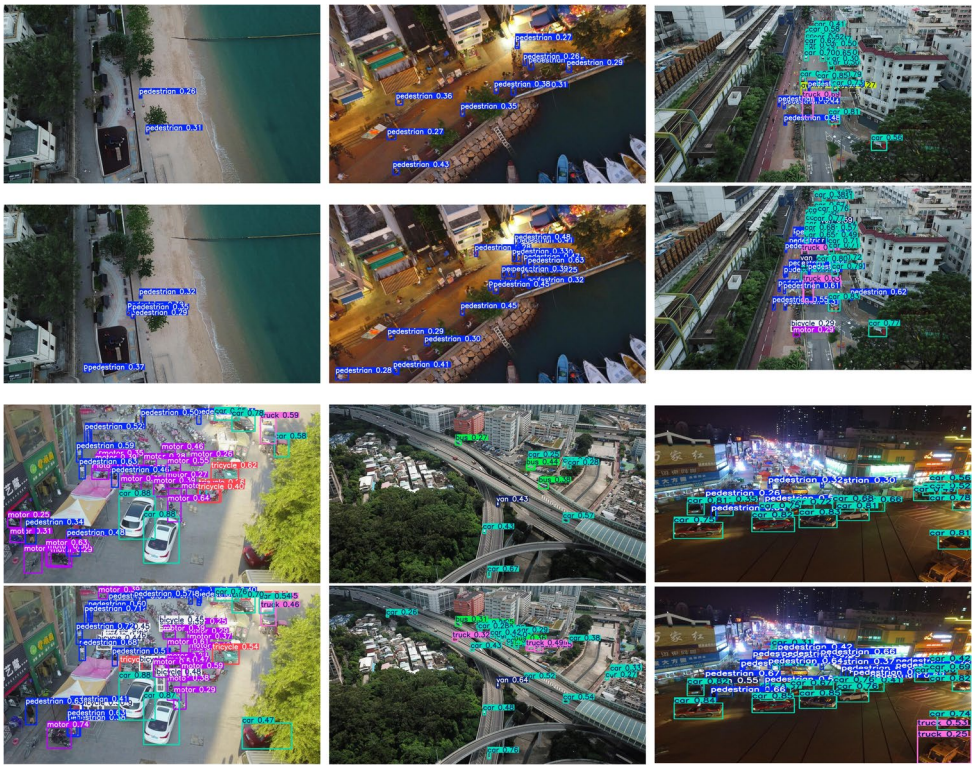

Fig. 16. Detection results of different models. The top of each group is YOLOv11 and the bottom is LRDS-YOLO.

Fig. 17. The partial image of the HIT-UAV dataset.

Model	Person (%)	Car (%)	Bicycle (%)	OtherVehicle (%)	DontCare (%)	mAP50 (%)	mAP50:95 (%)
YOLOv5	92.5	98.0	90.1	73.3	23.3	75.4	48.1
YOLOv6	94.1	96.4	91.3	52.6	57.1	78.3	49.7
YOLOv8	94.4	96.5	91.4	57.7	59.3	79.9	51.0
YOLOv9	92.2	98.8	92.8	77.2	43.1	80.8	52.4
YOLOv10	88.0	96.8	84.4	66.1	52.4	77.7	47.3
YOLOv11	91.9	98.6	90.1	65.8	65.0	82.3	52.4
RT-DETR	93.24	98.52	89.21	44.43	68.66	78.82	52.22
LRDS-YOLO	92.2	98.4	89.8	68.5	73.8	84.5	54.2

Table 6. Comparative experiment results for different models.

Data availability

The datasets used in this study are publicly available. The VisDrone dataset is available on the official website: https://github.com/VisDrone/VisDrone-Dataset.

The HIT-UAV dataset is available on the official website: https://github.com/suojiashun/HIT-UAV-Infrared-Thermal-Dataset.

Received: 25 March 2025; Accepted: 12 June 2025
Published online: 02 July 2025

References

1. Vattapparamban, E., Güvenç, I., Yurekli, A. I., Akkaya, K., & Uluağaç, S. Drones for smart cities: Issues in cybersecurity, privacy, and public safety. In *2016 International Wireless Communications and Mobile Computing Conference (IWCMC)*. 216–221 (IEEE, 2016).
2. Kwon, S.-H. et al. Enhancing citrus fruit yield investigations through flight height optimization with UAV imaging. *Sci. Rep.* **14**(1), 322 (2024).
3. Bisio, I., Garibotto, C., Haleem, H., Lavagetto, F. & Sciarrone, A. A systematic review of drone based road traffic monitoring system. *IEEE Access* **10**, 101537–101555 (2022).
4. Erdelj, M., Natalizio, E., Chowdhury, K. R. & Akyildiz, I. F. Help from the sky: Leveraging UAVs for disaster management. *IEEE Pervasive Comput.* **16**(1), 24–32 (2017).
5. Zhang, Z. Drone-yolo: An efficient neural network method for target detection in drone images. *Drones* **7**(8), 526 (2023).
6. Lin, T. -Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. & Zitnick, C. L. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European Conference*, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13. 740–755 (Springer, 2014).
7. Xu, Y. & Wang, W. A method for single frame detection of infrared dim small target in complex background. *J. Phys. Conf. Ser.* **1634**, 012063 (IOP Publishing, 2020).
8. Guo, M.-H. et al. Segnext: Rethinking convolutional attention design for semantic segmentation. *Adv. Neural Inf. Process. Syst.* **35**, 1140–1156 (2022).
9. 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*. 7373–7382 (2021).
10. Du, D. et al. Visdrone-det2019: The vision meets drone object detection in image challenge results. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops* (2019)
11. Suo, J. et al. Hit-UAV: A high-altitude infrared thermal dataset for unmanned aerial vehicle-based object detection. *Sci. Data* **10**(1), 227 (2023).
12. Girshick, R. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*. 1440–1448 (2015).
13. 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.* **28** (2015).
14. Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 779–788 (2016).

15. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. -Y. & Berg, A. C. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14*, 2016, Proceedings, Part I 14. 21–37 (Springer, 2016).
16. Han, W. et al. Methods for small, weak object detection in optical high-resolution remote sensing images: A survey of advances and challenges. *IEEE Geosci. Remote Sens. Mag.* **9**(4), 8–34 (2021).
17. Li, Z. et al. Context feature integration and balanced sampling strategy for small weak object detection in remote sensing imagery. *IEEE Geosci. Remote Sens. Lett.* **21**, 1–5 (2024).
18. Zhao, Z.-Q., Zheng, P., Xu, S.-T. & Wu, X. Object detection with deep learning: A review. *IEEE Trans. Neural Netw. Learn. Syst.* **30**(11), 3212–3232 (2019).
19. Hu, Y., Wu, X., Zheng, G. & Liu, X. Object detection of UAV for anti-UAV based on improved yolo v3. In *2019 Chinese Control Conference (CCC)*. 8386–8390 (IEEE, 2019).
20. Huang, M., Mi, W. & Wang, Y. Edgs-yolov8: An improved yolov8 lightweight UAV detection model. *Drones* **8**(7), 337 (2024).
21. Li, J., Liang, X., Wei, Y., Xu, T., Feng, J., & Yan, S. Perceptual generative adversarial networks for small object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1222–1230 (2017).
22. Cai, Z. & Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6154–6162 (2018).
23. Shi, Y., Jia, Y. & Zhang, X. Focusdet: An efficient object detector for small object. *Sci. Rep.* **14**(1), 10697 (2024).
24. Li, Z., Wang, Y., Xu, D., Gao, Y. & Zhao, T. Tbnet: A texture and boundary-aware network for small weak object detection in remote-sensing imagery. *Pattern Recognit.* **158**, 110976 (2025).
25. Tan, M., Pang, R. & Le, Q. V. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10781–10790 (2020).
26. Tan, M. & Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*. 6105–6114 (PMLR, 2019).
27. Zoph, B. & Le, Q. V. Neural architecture search with reinforcement learning. *arXiv preprint* arXiv:1611.01578 (2016).
28. Wang, W., Xie, E., Li, X., Fan, D. -P., Song, K., Liang, D., Lu, T., Luo, P. & Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 568–578 (2021).
29. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S. & Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 10012–10022 (2021).
30. Zhu, X., Su, W., Lu, L., Li, B., Wang, X. & Dai, J. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint* arXiv:2010.04159 (2020).
31. Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L. M. & Shum, H. -Y. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint* arXiv:2203.03605 (2022).
32. Ni, J., Zhu, S., Tang, G., Ke, C. & Wang, T. A small-object detection model based on improved yolov8s for UAV image scenarios. *Remote Sens.* **16**(13), 2465 (2024).
33. Zeng, S., Yang, W., Jiao, Y., Geng, L. & Chen, X. Sca-yolo: A new small object detection model for UAV images. *Vis. Comput.* **40**(3), 1787–1803 (2024).
34. Song, G., Du, H., Zhang, X., Bao, F. & Zhang, Y. Small object detection in unmanned aerial vehicle images using multi-scale hybrid attention. *Eng. Appl. Artif. Intell.* **128**, 107455 (2024).
35. Sun, W., Dai, L., Zhang, X., Chang, P. & He, X. Rsod: Real-time small object detection algorithm in UAV-based traffic monitoring. *Appl. Intell.* 1–16 (2022).
36. Khanam, R. & Hussain, M. Yolov11: An overview of the key architectural enhancements. *arXiv preprint* arXiv:2410.17725 (2024).
37. Jocher, G., Chaurasia, A. & Qiu, J. YOLO by Ultralytics (2023).
38. Dosovitskiy, A. et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint* arXiv:2010.11929 (2020).
39. Xie, E. et al. Segformer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inf. Process. Syst.* **34**, 12077–12090 (2021).
40. Xue, C. et al. El-yolo: An efficient and lightweight low-altitude aerial objects detector for onboard applications. *Expert Syst. Appl.* **256**, 124848 (2024).
41. Wang, H., Liu, C., Cai, Y., Chen, L. & Li, Y. Yolov8-qsd: An improved small object detection algorithm for autonomous vehicles based on yolov8. *IEEE Trans. Instrum. Meas.* (2024).
42. Wang, J., Gao, J. & Zhang, B. A small object detection model in aerial images based on cpdd-yolov8. *Sci. Rep.* **15**(1), 770 (2025).

Author contributions

Y.H. wrote the main manuscript text and prepared all figures and tables. All authors reviewed the manuscript.

Funding

This study was supported by Yunnan Fundamental Research Projects (grant number 202401AS070034) and the Yunnan Provincial Forestry and Grass Science and Technology Innovation Joint Project (grant number 202404CB090002).

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to L.Y.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.