

Article

Not peer-reviewed version

SEMA-YOLO: Lightweight Small Object Detection in Remote Sensing Image via Shallow-layer Enhancement and Multi-scale Adaptation

[Zhenchuan Wu](#) , Hang Zhen , [Xiaoxinxi Zhang](#) , [Xuechen Bai](#) , [Xinghua Li](#) *

Posted Date: 14 April 2025

doi: 10.20944/preprints202504.1115.v1

Keywords: small object detection; remote sensing; YOLO; feature fusion



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Article

SEMA-YOLO: Lightweight Small Object Detection in Remote Sensing Image via Shallow-Layer Enhancement and Multi-Scale Adaptation

Zhenchuan Wu ^{1,†} , Hang Zhen ^{1,†} , Xiaoxin Zhang ¹, Xuechen Bai ² and Xinghua Li ^{1,*} 

¹ School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China

² the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China

* Correspondence: lixinghua5540@whu.edu.cn

† These authors contributed equally to this work.

Abstract: Small object detection remains a challenge in remote sensing field due to feature loss during downsampling and interference from complex backgrounds. A novel network, termed SEMA-YOLO, is proposed in this paper as an enhanced YOLOv11-based framework incorporating three technical advancements. By fundamentally reducing information loss and incorporating a cross-scale feature fusion mechanism, the proposed framework significantly enhances small object detection performance. First, the Shallow Layer Enhancement (SLE) strategy reduces backbone depth and introduces small-object detection heads, thereby increasing feature map size and improving small object detection performance. Then, the Global Context Pooling-enhanced Adaptively Spatial Feature Fusion (GCP-ASFF) architecture is designed to optimize cross-scale feature interaction across four detection heads. Finally, the RFA-C3k2 module, which integrates Receptive Field Adaptation (RFA) with the C3k2 structure, is introduced to achieve more refined feature extraction. SEMA-YOLO demonstrates significant advantages in complex urban environments and dense target areas, while its generalization capability meets the detection requirements across diverse scenarios. Experimental results show that SEMA-YOLO achieves mAP₅₀ scores of 72.5% on the RS-STOD dataset and 61.5% on the AI-TOD dataset, surpassing state-of-the-art models.

Keywords: small object detection; remote sensing; YOLO; feature fusion

1. Introduction

Object detection is a crucial research topic in the field of remote sensing and has been widely applied in various domains, including military applications [1], agriculture [2], urban planning [3], environmental monitoring [4], and traffic management [5]. These application scenarios cover a wide range of fields, but involve a variety of object types. According to the definition of small objects in MS-COCO [6], objects with a size smaller than 32×32 pixels are considered as small objects. In particular, some small objects in remote sensing images are often difficult to identify accurately. The enhancement of the detection accuracy of small objects in remote sensing images has become a key research direction in the remote sensing field [7]. In recent years, the resolution of remote sensing images has improved significantly. In the optical domain, the rapid development of high-resolution (HR) satellites and unmanned aerial vehicle (UAV) technology has enabled remote sensing images to achieve meter-level or even sub-meter-level resolution. For example, the WorldView-3 satellite provides a panchromatic resolution of 0.31 m and a multispectral resolution of 1.24 m [8], while UAVs such as the DJI Phantom 4 RTK [9] can achieve centimeter-level resolution when flying at low altitudes. In the synthetic aperture radar (SAR) domain, the Gaofen-3 03 satellite offers a C-band resolution of up to 1 m [10], and the TerraSAR-X satellite can achieve a ground resolution of 1 m in SpotLight mode [11]. The availability of HR remote sensing imagery has laid a solid foundation for the detection and analysis of small objects.

However, small object detection in remote sensing images still encounters numerous challenges. **Firstly, it is the complexity of remote sensing imagery backgrounds.** Due to imaging conditions, remote sensing images often suffer from geometric distortions, radiometric variations, shadows, and noise, which not only interfere with object recognition but also lead to false positives or missed detections [12]. **Secondly, it is a limited number of pixels of small objects.** This makes their feature representation inadequate for traditional feature extraction methods, thereby increasing detection difficulty [13]. **Thirdly, the dense distribution of small objects.** In scenarios where small objects are densely distributed, occlusion and overlap among objects can severely impact detection accuracy, making it challenging to distinguish and accurately locate objects [14]. These challenges hinder the performance of small object detection in HR remote sensing imagery, necessitating the development of more advanced algorithms for effective solutions.



Figure 1. The challenges of small object detection in remote sensing images. Left: the shadows. Middle: limited number of pixels of vehicles. Right: dense distribution of ships.

To address these challenges, this paper implements small object detection in HR remote sensing images based on the YOLOv11 framework. Specifically, the proposed approach introduces the following contributions:

SEMA-YOLO, a lightweight and high-accuracy YOLOv11-based model, is proposed. This model demonstrates strong performance on both RS-STOD dataset and AI-TOD [15] dataset.

Shallow Layer Enhancement (SLE): To improve small object detection in YOLOv11, an extra tiny detection head is added, and the backbone output is adjusted from P5 to P4. This reduces parameters while enhancing localization and classification, especially in dense and complex scenes.

Global Context Pooling and Enhanced Adaptively Spatial Feature Fusion (GCP-ASFF): Traditional FPNs often lose fine details of small objects. To solve this, ASFF [16] dynamically learns spatial weights to optimize multiscale feature fusion and reduce conflicts. However, ASFF lacks global context awareness, so a Global Context Pooling (GCP) module is introduced to enhance semantic consistency and improve feature integration.

Receptive Field Attention Convolution Module (RFA): This lightweight module enhances detection under high IoU thresholds by dynamically generating receptive field attention weights, overcoming fixed kernel limitations [17]. Compared to YOLOv11's C3k2 module, RFA improves fine-grained feature extraction and sensitivity to small objects.

The remainder of this paper is structured as follows. Section 2 reviews related works on small object detection in remote sensing images. Section 3 provides a detailed description of the proposed methodology. Section 4 presents the experimental setup, results, and analysis. Section 5 discusses further aspects of the proposed model. Section 6 concludes the paper.

2. Related Works

This section reviews recent progress in small object detection through two perspectives: task-specific methods designed to tackle inherent challenges of such targets and mainstream detection

frameworks. The core ideas of these approaches will be thoroughly examined, and their strengths and weaknesses in small object detection will be evaluated.

2.1. Task-Specific Methods

2.1.1. Multi-Scale Feature

Small objects, characterized by their limited resolution and low visibility, often exhibit insufficient feature representation in detection tasks. To mitigate this issue, multi-scale feature extraction and fusion have emerged as a critical strategy, enabling models to simultaneously capture fine-grained details from HR shallow layers and rich semantic context from low-resolution (LR) deep layers.

In 2017, Lin et al. proposed FPN [18], which combines high-level semantic information with low-level spatial information to create multi-scale feature representations. This significantly improves the detection of multi-scale objects, especially small ones. FPN is a foundational structure in many modern object detection frameworks, and numerous variants have been proposed to further refine feature processing and improve detection performance. PANet [19] introduces a bottom-up pathway to strengthen feature propagation, improving localization and segmentation. NAS-FPN [20] leverages automated architecture search to optimize feature selection and fusion. BiFPN [21] refines this process by introducing weighted connections for more efficient multi-scale information flow. Recursive-FPN [22] incorporates extra feedback connections from FPN into the bottom-up backbone layers. AFPN [23] first fuses adjacent low-level features and then gradually incorporates higher-level features to reduce the semantic gap between non-adjacent levels. In the YOLO algorithm, YOLOv3 [24] performs multi-scale predictions by introducing parallel branches, where HR features specialize in detecting small objects, improving detection accuracy across different scales. YOLOv4 [25] further enhances this by incorporating Path Aggregation Network (PAN), which then became a standard component in the YOLO series for strengthening feature fusion. In Transformer-based detectors, FPT [26], as one of the earlier implementations, enables full feature interaction across space and scales, enhancing feature pyramids with richer contexts. Similarly, DNTR [27] integrates FPN through its DeNoising FPN (DNFPN) module, which employs contrastive learning to suppress noise in multi-scale feature fusion, improving tiny object detection. Most of these variants, such as PANet, NAS-FPN, and BiFPN, focus on improving multi-scale feature interactions to enhance model performance. However, this inevitably increases the number of parameters, and these methods often lack dedicated optimizations for small object detection.

2.1.2. Super-Resolution

Super-resolution (SR) techniques have been widely explored to enhance image quality by reconstructing HR images from LR counterparts. Given that small objects often occupy only a few pixels in an image, SR has been considered a promising approach to improve small object detection by enhancing fine-grained details. Many SR methods, such as GAN [28] and RCAN [29], have been applied to object detection.

Several studies have explored different ways to integrate SR with object detection. SuperYOLO [30] enhances small object detection in remote sensing images by combining multimodal fusion and SR-assisted learning while keeping computational cost low. Jin et al. [31] proposed a SRD network combined with an improved Faster R-CNN to address the challenges of pedestrian detection in low-quality images. SRCGAN-RFA-YOLO [32] integrates cyclic generative adversarial networks (CycleGAN) with residual feature aggregation for image SR and combines it with YOLOv3 to enhance small object detection in remote sensing images. Truong et al. [33] demonstrated that image SR significantly improves object detection performance on LR images by training a custom RCAN and evaluating its impact on detection tasks. Despite the remarkable success of detection methods combined with image SR, they often introduce significant model complexity and require additional computational overhead.

2.1.3. Context Based

Context-based methods in small object detection focus on utilizing the surrounding information of an object to improve detection accuracy. These approaches recognize that small objects often become more distinguishable when considering the spatial, semantic, and relational context within an image. By incorporating features from the local and global environment, context-based techniques enhance the recognition of small objects, addressing the limitations of traditional models that typically fail to capture subtle cues that aid in detection. As a result, context-based strategies have shown great promise in improving the robustness and reliability of small object detection systems. Several context-based methods have been proposed to further improve small object detection by utilizing different aspects of contextual information.

Lim et al. [34] proposed a context-based object detection method that improves small object detection by incorporating multi-scale features and an attention mechanism. PyramidBox [35] fully exploits multi-level contextual information to effectively detect blurry and indistinct faces that might otherwise be lost in the background. SCDNet [36] enhances detection in dense scenes by decoupling scene context through a dedicated scene classification subnetwork and incorporating scene context information with a lightweight fusion module and foreground enhancement module. FS-SSD [37] makes use of implicit spatial relationships by analyzing the distances between objects, refining the detection of instances with initially low confidence scores. CAB Net [38] constructs HR semantic feature maps by introducing context-aware blocks with pyramidal dilated convolutions, and attaches them to a shallow backbone to preserve spatial detail while incorporating multi-level contextual information. Tong et al. [39] introduced a context decoupling strategy that separately feeds semantic and spatial features into the classification and localization branches, enabling more precise detection through task-specific contextual information.

2.2. Mainstream Frameworks

2.2.1. YOLO Frameworks

Currently, deep learning-based object detection methods can be categorized into two main types: one-stage and two-stage detection methods. YOLO (You Only Look Once) belongs to the former category. Among various one-stage detection approaches, YOLO maintains high accuracy while achieving real-time object detection.

Since the introduction of YOLOv1 [40], the YOLO algorithm has evolved significantly. YOLOv2 [41] introduced batch normalization and data augmentation. YOLOv3 [24] integrated FPN for better multi-scale detection. YOLOv4 [25] added Spatial Pyramid Pooling (SPP) and PAN to boost speed and accuracy. Subsequent versions, including YOLOv5 [42], adopted the PyTorch framework, making the model more accessible and customizable. YOLOv6 [43] introduced the RepVGG module and a hybrid channel strategy. YOLOv7 [44] leveraged the Extended Efficient Layer Aggregation Network (E-ELAN) to enhance information flow between layers, improving efficiency and effectiveness. YOLOv8 [45], developed by Ultralytics, provided a scalable version to accommodate diverse applications. YOLOv9 [46] further improved performance by optimizing programmable gradient information (PGI) to enhance inter-layer information flow. YOLOv10 [47] eliminated the need for Non-Maximum Suppression (NMS) and introduced lightweight classification heads, spatial-channel decoupled scaling, and rank-based block designs. YOLOv11 replaced the C2F module in YOLOv8 with the more efficient C3K2 module and introduced the C2PSA module. YOLOv12 [48] introduced the attention-centric real-time object detector, incorporating an Attention-Aware (A2) module and a Residual Efficient Layer Aggregation Network (R-ELAN) to address optimization challenges associated with attention mechanisms, especially in large models.

Due to its real-time performance, high accuracy, and adaptability, the YOLO algorithm has been widely applied in the remote sensing field. For example, Qiu et al. [49] proposed ASFF-YOLOv5, an improved YOLOv5 network with multiscale feature fusion, designed for multi-element detection in UAV road traffic imagery. Lin et al. [50] introduced YOLO-DA, an efficient YOLO-based detector tailored to

overcome detection challenges posed by high inter-class similarity and complex backgrounds in optical remote sensing images. Li et al. [51] developed RSI-YOLO, a YOLOv5-based remote sensing object detection model incorporating attention mechanisms, enhanced feature fusion structures, additional small object detection layers, and optimized loss functions, significantly improving detection precision and recall rates. Xie et al. [52] introduced CSPPartial-YOLO, a lightweight YOLO-based model that addresses the computational complexity and large model size issues commonly encountered in deep learning-based object detection.

Given YOLO's outstanding performance in remote sensing object detection, this study selects YOLOv11 as the foundational framework for small object detection in HR remote sensing images.

2.2.2. Other Object Detection Frameworks

Object detection frameworks can be categorized from different perspectives. From the detection stage perspective, they can be broadly divided into one-stage and two-stage methods. One-stage methods directly extract features from images and predict object bounding boxes, prioritizing detection speed. In contrast, two-stage methods first generate candidate regions and then perform classification and regression, typically achieving higher accuracy. From the architectural perspective, object detection models can be classified into CNN-based and Transformer-based approaches. CNN-based models have been widely used in object detection, while Transformer-based models leverage self-attention mechanisms to capture global dependencies and enhance detection performance.

In two-stage methods, Faster R-CNN incorporates a Region Proposal Network (RPN) to improve detection speed, while Cascade R-CNN refines bounding box quality through multi-stage regression, leading to higher precision. Several approaches have also been proposed to enhance the detection of small objects in two-stage frameworks, such as [53–55]. However, two-stage object detectors generally suffer from high computational complexity and slow inference speed. Among one-stage methods, apart from YOLO, RetinaNet introduces focal loss to address class imbalance and improve small object detection. CenterNet reformulates object detection as a keypoint detection problem, reducing redundant bounding boxes and enhancing efficiency. Other one-stage approaches have also been developed to further improve accuracy and robustness, particularly for small object detection. FCOS [56] introduces an anchor-free and proposal-free fully convolutional one-stage object detection framework, simplifying the detection process while achieving superior accuracy. RFLA [57] introduces a Gaussian receptive field-based label assignment strategy to improve tiny object detection, addressing the limitations of anchor-based and anchor-free methods by reducing outlier tiny-sized ground truth samples and achieving more balanced learning. Kang et al. [58] proposed aligned matching to improve small object detection in SSD, achieving better performance without compromising large object detection or increasing model complexity.

The introduction of the Transformer [59] has established a unified framework bridging computer vision and natural language processing. DETR [60] stands as a landmark achievement in applying Transformer to object detection within computer vision. However, compared to CNN-based models, DETR faces two significant challenges: slower convergence and weaker performance on small objects. Consequently, subsequent DETR-based models have focused their efforts on addressing these two critical issues. Deformable DETR [61] replaces the standard Transformer attention with deformable attention modules, achieving better performance (especially on small objects) and 10x faster convergence compared to DETR. DN-DETR [62] accelerates training by reconstructing noised ground-truth boxes, reducing bipartite graph matching difficulty and improving performance. DINO [63] enhances DETR with contrastive denoising training, mixed query selection, and look-forward-twice box prediction, significantly boosting detection accuracy. Liu et al. proposed two models, DNTR [27] and DQ-DETR [64]. DNTR introduces the RCNN structure to DETR, while DQ-DETR optimizes the design of dynamic queries, both achieving significant improvements in tiny object detection. RT-DETR [65] designs an efficient hybrid encoder and proposes uncertainty-minimal query selection, achieving better speed and accuracy than contemporary YOLO and DETR-based models. Although DETR-based models have made strides in addressing real-time performance and small object detection, no existing model has

successfully balanced both aspects. Specifically, RT-DETR exhibits suboptimal performance in small object detection, while other DETR variants still fall short of the real-time efficiency achieved by the YOLO series. Furthermore, the Transformer-based architecture of DETR models inherently results in larger model sizes, posing significant challenges for lightweight optimization.

3. Proposed Method

3.1. Overview

This study proposes SEMA-YOLO, an enhanced network based on YOLOv11, designed to improve small object detection in HR remote sensing images. It employs SLE to preserve low-level features and introduces a tiny detection head for better detail capture. Additionally, GCP-ASFF integrates global context information to reduce false and missed detections. Finally, RFA is applied to emphasize critical features, ensuring strong performance under high IoU thresholds. The overall structure of the proposed model is shown in Figure 2.

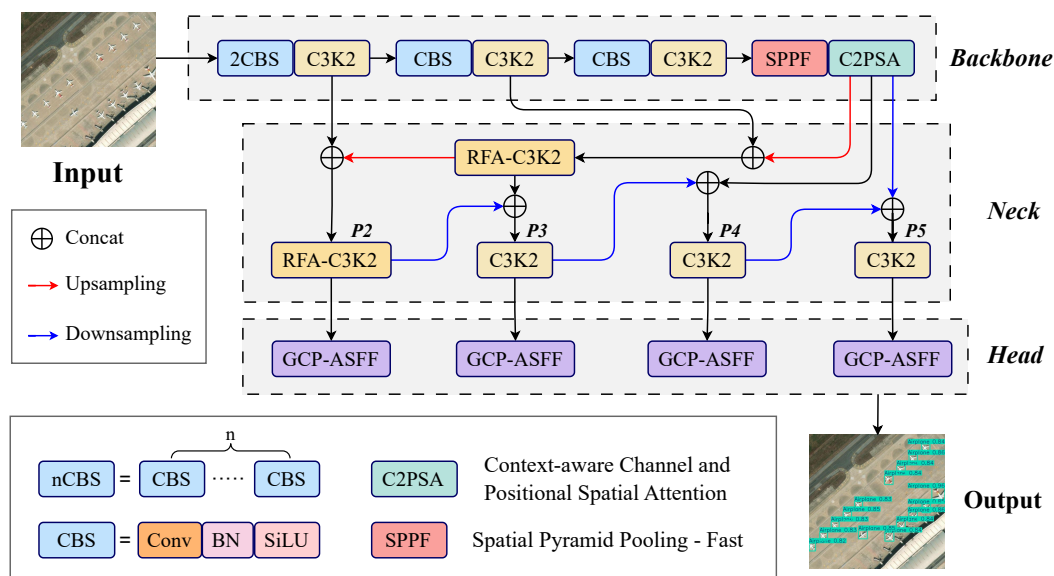


Figure 2. The overall framework of SEMA-YOLO.

3.2. Shallow Layer Enhancement

Increasing the feature map size is an effective method for improving small object detection performance, but it presents two main challenges. Firstly, reducing convolutional layers or downsampling operations to enlarge the feature map may weaken the model's ability to extract semantic information, which can negatively affect detection accuracy. Secondly, increasing the feature map size significantly raises computational costs. To address these challenges, a SLE strategy is proposed, which is specifically designed for networks with a CNN+PAN backbone. This strategy improves detection accuracy with only a slight increase in computational costs and a reduction in the number of parameters, achieving a balance between model size and performance.

The original YOLOv11 adopts a PAN structure with three detection heads corresponding to P3 (1/8 of the input image), P4 (1/16), and P5 (1/32). However, for small object detection tasks, the P5 feature map struggles to capture small object features effectively due to its limited spatial resolution. To enhance the model's perception capability, an additional P2 detection head is introduced, which has a feature map with a side length of 1/4 of the input image. This P2 feature map is obtained by applying two consecutive upsampling operations to the P4 feature map, while maintaining the standard detection branch design in YOLO. By directly utilizing shallow features for detection, this modification improves the recognition of small objects.

To further enhance detection, the backbone network is modified by terminating convolutional operations at the P4 layer and eliminating unnecessary downsampling operations. This adjustment pre-

serves more low-level features and minimizes the loss of semantic information. Despite the reduction in backbone depth, the P5 detection head is retained to ensure multi-scale detection capability. Specifically, the final P4 feature map is processed through a C2PSA module and downsampled separately to match the spatial resolution of the P5-level feature map. The two downsampled feature maps are then fused using a Concat operation to create an enhanced P5 feature map. This modification reduces computational costs while maintaining detection performance across multiple scales. Compared to the original YOLOv11n network, the SLE-enhanced model reduces the number of parameters from 2.583M to 2.075M, achieving a 19.7% reduction. On the RS-STOD dataset, this model also improves mAP_{50:95} by 0.052, demonstrating the effectiveness of the proposed modifications.

3.3. GCP-ASFF Module

In traditional FPN, the direct fusion of feature maps at different levels often results in inconsistent gradient propagation due to differences in spatial features and semantic information. For example, high-level features (e.g., those subject to 32× downsampling) are more suitable for detecting large objects, whereas low-level features (e.g., those subject to 8× downsampling) are more sensitive to small objects. However, when both large and small objects coexist in an image, the fusion of features across different levels may cause interference, leading to inconsistent gradient propagation and subsequently reducing detection accuracy. In other words, small objects tend to lose significant details after multiple downsampling steps (e.g., a 4×4-pixel object is reduced to a 1×1 effective area after 4× downsampling), and the weak signals from low-level features can easily be overwhelmed by high-level features. To address the limitations of multi-scale feature fusion, Adaptively Spatial Feature Fusion (ASFF) is introduced.

Improving the feature fusion capability in FPN enables the network to more effectively handle the challenges posed by intricate environments and objects at varying scales, ultimately enhancing detection accuracy. Let x^1 , x^2 , and x^3 denote the feature vectors corresponding to levels 1, 2, and 3, respectively. The fused feature at hierarchical level k can be expressed through the following parametric combination:

$$z_{ij}^k = \alpha_{ij}^k \cdot x_{ij}^{1 \rightarrow k} + \beta_{ij}^k \cdot x_{ij}^{2 \rightarrow k} + \gamma_{ij}^k \cdot x_{ij}^{3 \rightarrow k} \quad (1)$$

where z_{ij}^k denotes the synthesized feature vector at spatial position (i, j) of the k -th level pyramid. The term $x_{ij}^{n \rightarrow k}$ represents the transformed feature vector from pyramid level n to k at coordinate (i, j) , while α_{ij}^k , β_{ij}^k , and γ_{ij}^k correspond to learnable spatial weights for feature integration. The weight derivation follows a three-stage computational framework. Level-specific transformation is achieved through depth-wise separable convolutions:

Channel projection: Apply 1×1 convolutional kernels to $x^{1 \rightarrow k}$, $x^{2 \rightarrow k}$, and $x^{3 \rightarrow k}$, generating intermediate tensors λ_{α}^k , λ_{β}^k , and λ_{γ}^k respectively. The 1×1 convolution reduces channel dimensions while preserving spatial information, where $\lambda_{\varphi}^k \in \mathbb{R}^{H_k \times W_k \times 1}$ ($\varphi \in \{\alpha, \beta, \gamma\}$) contains unnormalized attention signals for each feature level.

Attention generation: These intermediate results undergo softmax normalization across channels:

$$\hat{\varphi}_{ij}^k = \frac{e^{\lambda_{\varphi ij}^k}}{\sum_{\psi \in \{\alpha, \beta, \gamma\}} e^{\lambda_{\psi ij}^k}}, \quad \varphi \in \{\alpha, \beta, \gamma\} \quad (2)$$

where $\hat{\varphi}_{ij}^k$ represents the normalized attention weight at location (i, j) for level k .

Parameter stabilization: The attention weights are adjusted through linear scaling to maintain numerical stability within $[0, 1]$:

$$\varphi_{ij}^k = \frac{\hat{\varphi}_{ij}^k}{\sum_{\psi \in \{\alpha, \beta, \gamma\}} \hat{\varphi}_{ij}^k} \quad (3)$$

This final normalization step guarantees $\sum \varphi_{ij}^k = 1$ while preventing gradient saturation during backpropagation.

By introducing learnable spatial weight parameters, ASFF dynamically adjusts the contribution of features from different levels during the fusion process. For instance, when detecting small objects, the model assigns higher weights to low-level features while suppressing redundant information in high-level features. This effectively reduces feature conflicts and enhances scale invariance. However, ASFF only fuses multi-scale features at the same spatial location through spatial weights, lacking the utilization of global contextual information. This limitation restricts the receptive field in complex scenarios, such as small object detection. To address this issue, we propose GCP-ASFF, an extension of ASFF. The introduced GCP enhances the semantic consistency of multi-scale features by extracting channel-level global features, thereby improving the quality of feature fusion. The overall structure of GCP-ASFF is illustrated in Figure 3.

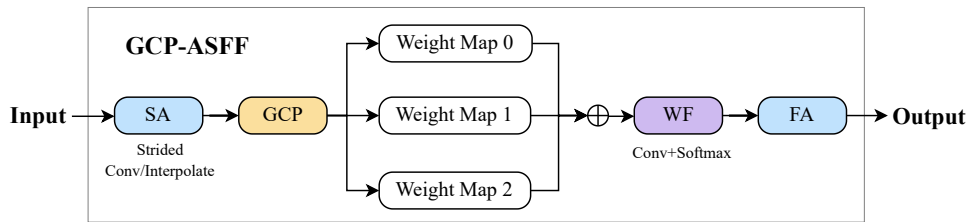


Figure 3. The overview of GCP-ASFF. Before feature fusion, global average pooling is applied to the feature map of each level to generate channel descriptor vectors. These vectors are then concatenated with the original features to enhance the representational capacity of the features.

The SA (Spatial Alignment), WF (Weight Fusion), and FA (Feature Aggregation) modules in the diagram retain the ASFF structure. The detailed structure of the GCP module is shown in Figure 4, and the specific algorithm flow is as follows:

1. Multi-scale Feature Alignment and Global Context Extraction. Resize multi-level features to a unified resolution and extract channel-wise global descriptors via GAP:

$$F_i^{\text{resized}} = \text{Resize}(F_i), \quad \forall i \in \{1, 2, 3\},$$

$$g_i = \frac{1}{H \times W} \sum_{h=1}^H \sum_{w=1}^W F_i^{\text{resized}}(h, w, c), \quad g_i \in \mathbb{R}^C, \quad (4)$$

where $F_i \in \mathbb{R}^{C \times H_i \times W_i}$ denotes the input feature maps from different pyramid levels.

2. Lightweight Context Integration. Concatenate the global descriptor g_i with the resized features and compress channels using a 1×1 convolution:

$$\hat{F}_i = \text{Conv}_{1 \times 1}(\text{Concat}(F_i^{\text{resized}}, g_i \otimes \mathbf{1}_{H \times W})), \quad \hat{F}_i \in \mathbb{R}^{C \times H \times W}, \quad (5)$$

where $\otimes$ denotes broadcasting and $\mathbf{1}_{H \times W}$ is an all-ones matrix.

3. Adaptive Spatial Weighting and Fusion. Generate spatial attention weights α_i and fuse context-enhanced features

$$\alpha_i = \text{Softmax}(\text{Conv}(\hat{F}_i)), \quad \alpha_i \in \mathbb{R}^{1 \times H \times W},$$

$$F_{\text{fused}} = \sum_{i=1}^3 \alpha_i \odot \hat{F}_i, \quad (6)$$

where $\odot$ denotes element-wise multiplication.

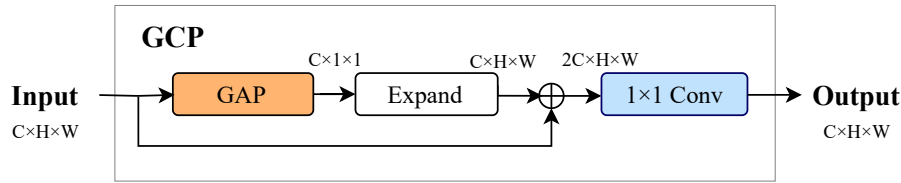


Figure 4. The overview of GCP. The GCP module acts as a global semantic filter, where channel-wise statistics guide local feature enhancement. High global activations amplify target-related patterns, while low activations suppress background noise.

3.4. RFA-C3k2 Module

For the characteristics of remote sensing images, such as small target sizes, dense distributions, and complex backgrounds, we designed the RFA-C3k2 module, with its specific implementation shown in Figure 5. The core improvement lies in the introduction of Receptive Field Attention Convolution (RFAConv), which dynamically adjusts the attention weights of convolutional kernels, effectively enhancing the model's ability to perceive multi-scale features.

In small object detection tasks, the fixed receptive field of traditional convolution is insufficient for effectively capturing the local details of small objects in remote sensing images. RFAConv addresses this issue by employing a multi-branch convolutional structure to generate attention weights, enabling the dynamic fusion of features across different receptive field windows. For example, the large receptive field branch captures the contextual relationships between targets and their surrounding environment, while the small receptive field branch focuses on local texture features. These complementary capabilities enhance the model's sensitivity to small objects. In most cases, the computation of RFA can be reformulated as:

$$\begin{aligned} F &= Softmax(g^{1 \times 1}(AvgPool(X))) \times ReLU(Norm(g^{k \times k}(X))) \\ &= A_{rf} \times F_{rf} \end{aligned} \quad (7)$$

where $g^{i \times i}$ corresponds to a grouped convolution with a kernel size of $i \times i$, k denotes the size of the convolutional kernel, $Norm$ refers to the normalization operation, and X represents the input feature maps. The resulting output F is derived by multiplying the attention map A_{rf} with the spatial features F_{rf} transformed by the receptive field. Compared to the original C3k2 module, which extracts features using fixed-size convolutional kernels, the RFA-C3k2 module dynamically generates receptive field attention weights, breaking the limitation of parameter sharing in convolutional kernels. This enhances the model's ability to capture local features of small targets.

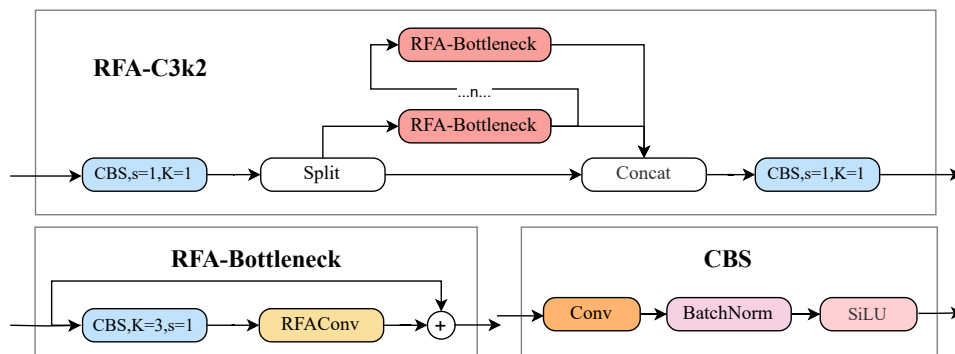


Figure 5. The overview of RFA-C3k2.

In the redesigned bottleneck structure, RFAConv is incorporated to improve the model's capability to sample input feature maps more effectively, thereby facilitating the learning of fine details in small target objects as well as background information. Following the convolutional operations, residual connections are employed to merge the input feature maps with the output feature maps.

4. Experiments

4.1. Datasets

The performance of the SEMA-YOLO model was evaluated on the RS-STOD dataset and the AI-TOD dataset. The descriptions of these two datasets are as follows:

RS-STOD is a HR remote sensing dataset for small object detection, with a training-to-testing ratio of approximately 4:1. This dataset contains a total of 50,854 instances across 2,354 aerial images, covering five categories: small vehicle, large vehicle, ship, airplane, and oil tank. Based on the definition of small objects in the MS-COCO, small objects account for 93% of the RS-STOD dataset. The images are sourced from the New Zealand land use website and Microsoft Bing Maps imagery tiles, with resolutions ranging from 0.4 meters to 2 meters.

AI-TOD is an aerial imagery dataset used for small object detection. It contains 28,036 images and 700,621 object instances, divided into 8 categories. The AI-TOD dataset has been re-partitioned based on the following considerations: Firstly, the original dataset exhibits a highly imbalanced sample distribution, which may affect the final training performance. Secondly, the dataset size has been reduced to accommodate limited computational resources. Additionally, the partitioning process strictly follows a random principle to ensure objectivity and fairness, while guaranteeing that all comparative methods are evaluated under the same conditions. Ultimately, 2,700 images from the AI-TOD dataset are allocated to the training set, while 300 images are assigned to the test set.

4.2. Implementation Details

In this study, all experiments were conducted on a single NVIDIA RTX 4090 GPU, implemented using the PyTorch framework. The Stochastic Gradient Descent (SGD) optimizer was employed for training. Experiments were performed on both the RS-STOD and AI-TOD datasets. For the RS-STOD dataset, the input image size was uniformly resized to 512×512 pixels (consistent with the original resolution of the dataset), with a batch size set to 16. For the AI-TOD dataset, the input image size was set to 640×640 pixels, and the batch size was adjusted to 4. Unless otherwise specified, all experiments followed the aforementioned configurations.

4.3. Evaluation Metrics

A set of standard metrics is adopted to evaluate the detection performance of the model, including Precision, Recall, F₁-score, and mean Average Precision (mAP) at IoU thresholds of 0.5 and 0.5–0.95. In addition, the number of neural network parameters (M) and Giga Floating-point Operations Per Second (GFLOPs) are reported to measure the model's complexity and computational cost. The calculation formulas for some of the evaluation metrics are as follows:

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (10)$$

$$mAP = \frac{\sum_{i=1}^k \sum_{j=1}^{n-1} (R_{j+1} - R_j) P_{inter}(R_{j+1})_i}{k} \quad (11)$$

Here, TP , FP , and FN denote the number of true positives, false positives, and false negatives, respectively. Precision measures the correctness of the predicted positive samples, while recall indicates the proportion of ground-truth objects that are successfully detected. The F₁-score provides a harmonic mean of them. The mAP is computed as the mean of the AP_i over all N object categories.

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

To evaluate the performance of SEMA-YOLO, we conducted comparisons with the RT-DETR series and several lightweight YOLO models such as YOLOv8n and YOLOv9t. The results are shown in Table 1.

Table 1. Comparison experiments for SEMA-YOLO in RS-STOD.

Method	P $\uparrow$	R $\uparrow$	F ₁ -score $\uparrow$	mAP ₅₀ $\uparrow$	mAP _{50:95} $\uparrow$
RT-DETR-L	0.425	0.443	0.434	0.429	0.230
RT-DETR-R50	0.490	0.507	0.498	0.500	0.271
YOLOv8n	0.732	0.630	0.677	0.671	0.416
YOLOv9t	0.707	0.632	0.667	0.661	0.411
YOLOv10n	0.713	0.624	0.666	0.662	0.411
YOLOv11n	0.708	0.643	0.674	0.672	0.412
SEMA-YOLO	0.748	0.699	0.722	0.725	0.468

NOTE: P represents precision, and R represents recall.

SEMA-YOLO achieved mAP₅₀ and mAP_{50:95} scores of 0.725 and 0.468. Compared to RT-DETR-R50, SEMA-YOLO improves mAP₅₀ by 0.225 and mAP_{50:95} by 0.197. Among the YOLO models, SEMA-YOLO outperforms the best-performing variant, YOLOv8n, with an mAP₅₀ improvement of 0.054 and an mAP_{50:95} improvement of 0.053. The strong performance in mAP_{50:95} indicates that the model maintains reliable localization accuracy even at high IoU thresholds, demonstrating its adaptability to varying object scales and complex detection scenarios.

As shown in Table 2, SEMA-YOLO achieved the highest mAP₅₀ across all five object categories. For small objects, it improved mAP₅₀ for small vehicles to 0.722 and large vehicles to 0.196, representing increases of 20.3 % and 39.0 % compared to the second-best model YOLOv11n. For large-scale objects, SEMA-YOLO maintained the highest accuracy for airplane at 0.987 and oil tanks at 0.926, demonstrating the effectiveness of retaining the large-object detection head.

Further analysis of the RS-STOD dataset reveals significant variation in the difficulty of detecting different object categories. Large vehicles are the most challenging due to their low representation, which limits the model's ability to learn their features effectively. Aircraft are the easiest to detect, as they have distinctive visual characteristics and appear against relatively simple backgrounds. Although small vehicles are challenging due to their size, their high frequency in the dataset enables the model to learn them well, leading to reasonable detection performance.

Table 2. Comparison experiments for SEMA-YOLO in RS-STOD. (mAP₅₀.)

Method	SV	LV	SH	AP	OT
RT-DETR-L	0.308	0.025	0.356	0.856	0.602
RT-DETR-R50	0.379	0.041	0.479	0.908	0.691
YOLOv8n	0.600	0.136	0.744	0.983	0.894
YOLOv9t	0.587	0.133	0.716	0.984	0.881
YOLOv10n	0.588	0.148	0.712	0.979	0.884
YOLOv11n	0.600	0.141	0.746	0.984	0.887
SEMA-YOLO	0.722	0.196	0.793	0.987	0.926

NOTE: Table shows the mAP₅₀ results for each object category. SV corresponds to small vehicle, LV to large vehicle, SH to ship, AP to airplane, and OT to oil tank.

Quantitative metrics in Table 2 demonstrate the overall superiority of SEMA-YOLO, while the visual analysis of representative remote sensing scenarios intuitively highlights the detection performance of different models. Four representative scenarios are selected, including small vehicles in complex backgrounds, densely clustered harbor vessels, aircraft in airport scenes, and scale-variant oil tanks. The detection results are presented in Figure 6.

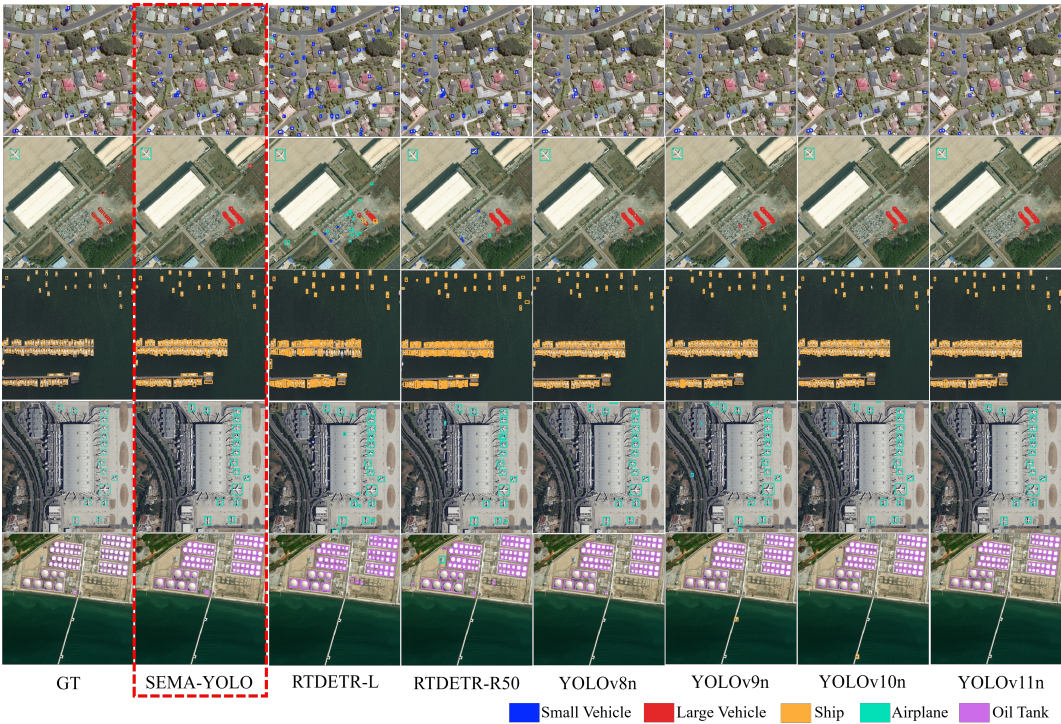


Figure 6. Detection results on RS-STOD datasets. SEMA-YOLO demonstrates more complete bounding boxes and fewer false positives across all scenarios.

To evaluate the detection performance under different data distributions, comparative experiments were conducted on the publicly available AI-TOD dataset. As shown in Table 3, the proposed method demonstrates significant improvements over existing approaches, particularly outperforming the best-performing YOLO model (YOLOv11) in key metrics.

Table 3. Comparison experiments on AI-TOD dataset.

Method	P↑	R↑	F ₁ -score↑	mAP ₅₀ ↑	mAP _{50:95} ↑
RT-DETR-L	0.537	0.199	0.290	0.133	0.043
RT-DETR-R50	0.394	0.308	0.346	0.242	0.080
YOLOv8n	0.662	0.548	0.600	0.557	0.235
YOLOv9t	0.67	0.522	0.587	0.547	0.232
YOLOv10n	0.619	0.514	0.562	0.532	0.229
YOLOv11n	0.703	0.524	0.600	0.563	0.239
SEMA-YOLO	0.740	0.557	0.636	0.615	0.284

NOTE: P represents precision, and R represents recall.

The method achieves an mAP₅₀ of 0.563, representing a 0.430 improvement over the RT-DETR-L baseline. Under the more stringent mAP_{50:95} evaluation metric, it attains a detection accuracy of 0.239 while maintaining superior precision (0.740) and recall (0.557). The comparison of visualization results among different models is shown in Figure 7. These results demonstrate robust performance in tiny object detection and indicate that the proposed approach not only performs well on custom RS-STOD datasets but also exhibits strong competitiveness on standard public benchmarks.

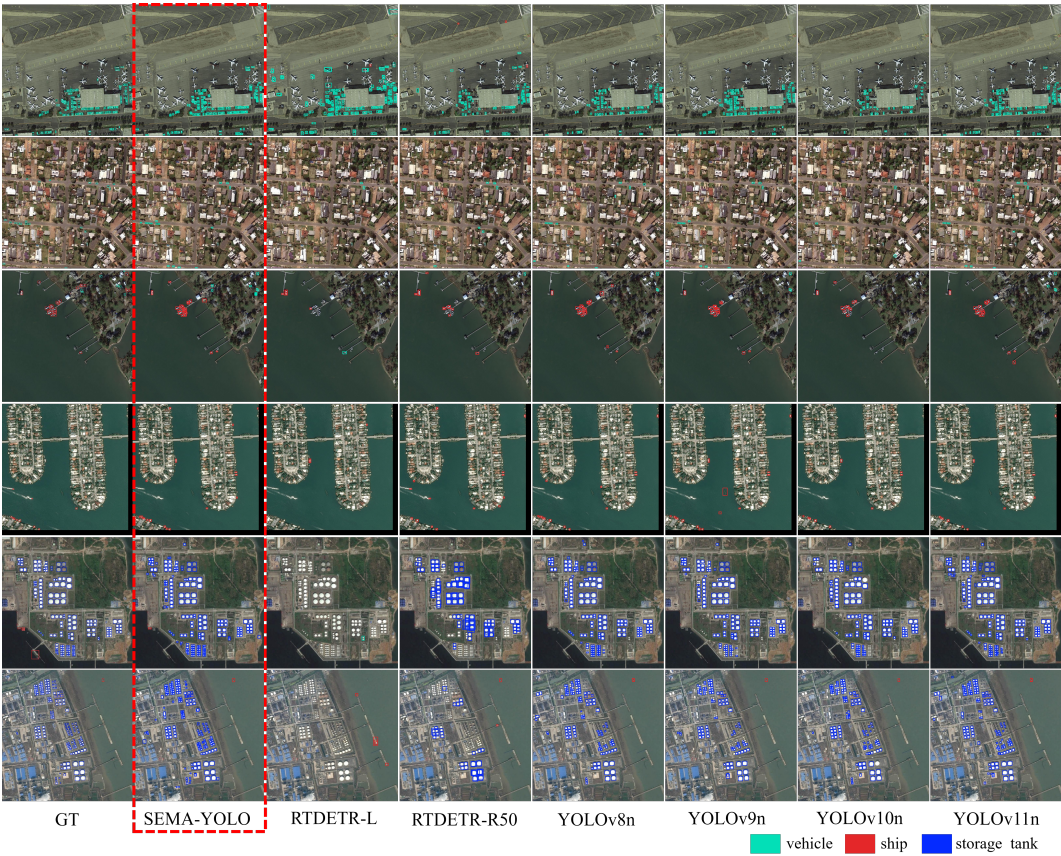


Figure 7. Detection results on AI-TOD datasets.

4.5. Ablation Experiments

To evaluate the effectiveness of the proposed improvements, an ablation experiment was conducted on our custom-built RS-STOD dataset, as shown in Table 4. The goal is to verify the contribution of the SLE strategy, GCP-ASFF module, and RFA module to the overall detection performance. The baseline model, YOLOv11n, serves as a reference, providing a starting point for assessing the impact of these modifications.

Table 4. Ablation experiment for SLE, GCP-ASFF,RFA in RS-STOD.

Method	P↑	R↑	mAP ₅₀ ↑	mAP _{50:95} ↑	Para(M)↓	GFLOPs↓
Baseline	0.709	0.643	0.671	0.412	2.583	6.3
+SLE	0.752	0.681	0.717	0.464	2.075	9.7
+SLE+RFA	0.754	0.680	0.715	0.456	2.078	9.8
+SLE+ASFF	0.765	0.687	0.722	0.465	3.468	12.8
+SLE+ASFF+RFA	0.749	0.698	0.721	0.466	3.471	12.9
SEMA-YOLO	0.746	0.700	0.725	0.468	3.645	14.2

NOTE: P represents precision, and R represents recall.

The results demonstrate that the SLE strategy significantly enhances detection performance. Compared to the baseline, SLE improves precision from 0.709 to 0.752 and recall from 0.643 to 0.681, leading to a notable increase in mAP. This suggests that SLE effectively strengthens feature extraction, optimizing both object localization and classification. Moreover, when combined with ASFF, the performance further improves, highlighting the strong compatibility between these two components.

It is noteworthy that when the RFA module is solely integrated with SLE, the model’s performance metrics experience a slight decline, indicating that RFA alone has limited effectiveness in enhancing feature representation. However, with the introduction of ASFF, the incorporation of RFA significantly improves the model’s recall rate and achieves superior performance under the more stringent mAP_{50:95}

metric. It is hypothesized that this is because ASFF enables the model to learn spatial weights for multi-scale feature map fusion, thereby increasing the contribution of large-scale feature maps. This allows RFA to more effectively capture fine-grained details, leading to enhanced overall performance.

Grad-CAM [66], as an advanced visualization technique, precisely maps the feature map regions that make decisive contributions to prediction results in deep neural networks. By computing gradients of target classes with respect to convolutional feature maps and performing subsequent weighting operations, Grad-CAM effectively localizes critical regions in the network decision-making process, significantly enhancing the transparency and interpretability of the SEMA-YOLO model. The Grad-CAM heatmap visualization results presented in Figure 8 clearly demonstrate the superior performance of SEMA-YOLO in small object detection tasks: the model can effectively distinguish between targets and background, with our proposed GCP-ASFF module successfully suppressing activation responses in non-target regions, while the SLE and RFA modules enhance feature representation for small objects. This high concentration of activation areas and accurate recognition directly validates the effectiveness of our proposed SEMA-YOLO model in addressing small object detection problems under complex backgrounds. The degree of highlighting in target regions in the activation maps is highly consistent with actual detection performance, providing intuitive visual evidence for the rationality of the model design.

These findings confirm the effectiveness of our design choices. By strategically integrating SLE, GCP-ASFF, and RFA, a well-balanced model is achieved that improves detection accuracy while maintaining computational efficiency. The final model demonstrates superior performance, validating our approach and providing valuable insights for future research in object detection.

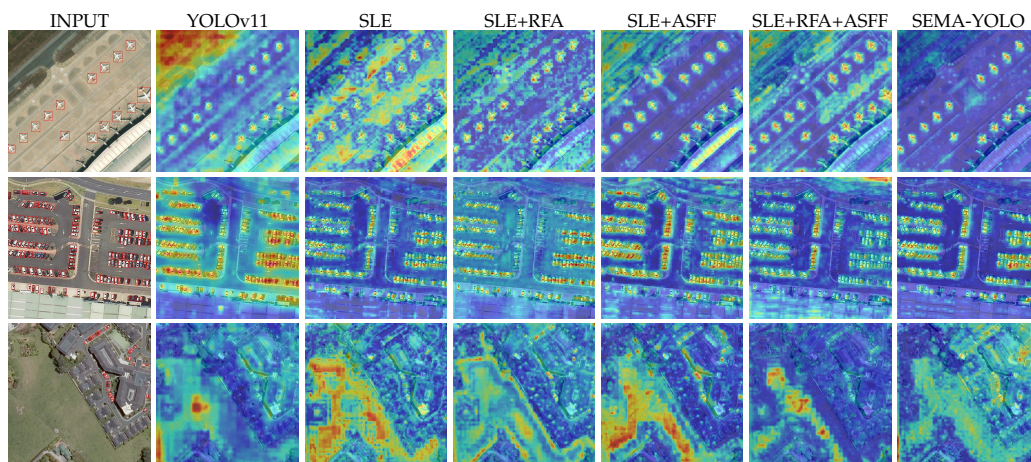


Figure 8. Heatmap of ablation experiment.

5. Discussion

5.1. Model Potential

As summarized in Table 5, the scalability of SEMA-YOLO is further demonstrated by evaluating the s- and m-size variants. When model capacity is increased from n to s, the mAP₅₀ improves from 72.5% to 75.3%, and continuing to m-size yields 76.8%, indicating that our architectural design can effectively leverage additional parameters for more accurate detections. Specifically, the small vehicle and ship categories show particularly strong gains with deeper models, suggesting that the multi-scale feature fusion in GCP-ASFF complements RFA's fine-scale feature extraction.

Table 5. Comprehensive Model Performance Comparison.

Method	mAP ₅₀ ↑	mAP _{50:95} ↑	Para(M)↓	GFLOPs↓
RT-DETR-R50	0.500	0.271	41.9	136
YOLOv11n	0.671	0.412	2.6	6.3
SEMA-YOLOn	0.725	0.468	3.6	14.2
SEMA-YOLOs	0.753	0.499	13.4	43.6
SEMA-YOLOm	0.768	0.521	29.7	138.1

A notable advantage of our approach is its consistently favorable trade-off between accuracy and computational overhead. Although SEMA-YOLO-m approaches the complexity of RT-DETR-R50 with comparable GFLOPs (138.1 vs. 136), it achieves superior detection performance across most categories (e.g., 24.5% vs. 4.1% AP for large vehicle) and a substantial margin (+26.8%) in overall mAP₅₀. These results illustrate that our design priorities—particularly the combination of SLE for preserving spatial granularity and the GCP-ASFF mechanism for robust cross-scale fusion—enable the model to more efficiently utilize its computational budget.

Although larger variants of SEMA-YOLO deliver higher detection performance, the lightweight SEMA-YOLO-n model still presents a compelling balance of accuracy and real-time capabilities (14.2 GFLOPs) for practical deployment on embedded platforms (e.g., Jetson Orin NX). In future work, efforts could focus on refining the GCP module for even more efficient global context extraction and exploring adaptive training strategies for further enhancing small object detection under extreme scale variations or low-contrast environments.

Overall, the marked improvements across diverse object classes, combined with favorable parameter efficiency and robust generalization on challenging remote sensing scenes, affirm the potential of SEMA-YOLO as a state-of-the-art framework for small object detection tasks.

5.2. Computational Efficiency

To further analyze the computational efficiency and deployment potential of SEMA-YOLO, we compared its parameter count and computational cost with those of the RT-DETR series and lightweight YOLO models. The results are shown in Table 6.

Table 6. Model Complexity and Efficiency.

Method	Para(M)↓	GFLOPs↓	Size(MB)↓	FPS↑
RT-DETR-L	32.0	110	63.1	182
RT-DETR-R50	41.9	136	84.0	156
YOLOv8n	3.0	8.7	5.97	244
YOLOv9t	2.0	7.7	4.44	227
YOLOv10n	2.7	6.7	5.90	277
YOLOv11n	2.6	6.3	5.23	232
SEMA-YOLO	3.6	14.2	7.43	185

SEMA-YOLO has 3.6 million parameters and a computational cost of 14.2 GFLOPs, which is higher than some lightweight models. However, its significant improvements in detection accuracy indicate that the increase in model complexity is justified. While SEMA-YOLO has a considerably lower computational cost than the RT-DETR series, it still achieves superior accuracy, demonstrating a strong balance between efficiency and performance.

6. Conclusions

This study introduces SEMA-YOLO, an effective framework for small object detection in HR remote sensing imagery. By incorporating novel modules like SLE, GCP-ASFF, and RFA on the YOLOv11n baseline, multi-scale feature fusion is enhanced, improving detection accuracy under

complex conditions. Extensive experiments on the RS-STOD and AI-TOD datasets show that SEMA-YOLO delivers competitive accuracy, real-time inference speed, and a compact model size. Future work will focus on exploring Transformer-based architectures, advanced cross-scale fusion techniques, and practical deployment strategies.

References

1. Wenqi, Y.; Gong, C.; Meijun, W.; Yanqing, Y.; Xingxing, X.; Xiwen, Y.; Junwei, H. MAR20: A benchmark for military aircraft recognition in remote sensing images. *National Remote Sensing Bulletin* **2024**, *27*, 2688–2696. <https://doi.org/10.11834/jrs.20222139>.
2. Ariza-Sentís, M.; Vélez, S.; Martínez-Peña, R.; Baja, H.; Valente, J. Object detection and tracking in Precision Farming: A systematic review. *Computers and Electronics in Agriculture* **2024**, *219*, 108757. <https://doi.org/10.1016/j.compag.2024.108757>.
3. Hoalst-Pullen, N.; Patterson, M.W. Applications and Trends of Remote Sensing in Professional Urban Planning: Remote sensing in professional urban planning. *Geography Compass* **2011**, *5*, 249–261. <https://doi.org/10.1111/j.1749-8198.2011.00422.x>.
4. Li, J.; Pei, Y.; Zhao, S.; Xiao, R.; Sang, X.; Zhang, C. A review of remote sensing for environmental monitoring in China. *Remote Sensing* **2020**, *12*, 1130. <https://doi.org/10.3390/rs12071130>.
5. Tan, Q.; Ling, J.; Hu, J.; Qin, X.; Hu, J. Vehicle detection in high resolution satellite remote sensing images based on deep learning. *IEEE Access* **2020**, *8*, 153394–153402. <https://doi.org/10.1109/ACCESS.2020.3017894>.
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 Proceedings of the 13th European Conference on Computer Vision (ECCV), Zurich, Switzerland, 2014, pp. 740–755. https://doi.org/10.1007/978-3-319-10602-1_48.
7. Wang, X.; Wang, A.; Yi, J.; Song, Y.; Chehri, A. Small object detection based on deep learning for remote sensing: A comprehensive review. *Remote Sensing* **2023**, *15*, 3265. <https://doi.org/10.3390/rs15133265>.
8. European Space Agency. WorldView-3 Mission. <https://earth.esa.int/eogateway/missions/worldview-3>, 2025. Accessed: 2025-03-11.
9. DJI. P4RTK System Specifications. <https://www.dji.com/cn/support/product/phantom-4-rtk>, 2019. Accessed: 2025-03-11.
10. Chinese Academy of Sciences. Gaofen-3 03 Satellite SAR Payload Achieves 1-Meter Resolution with World-Leading Performance. http://aircas.ac.cn/dtxw/kydt/202204/t20220407_6420970.html, 2022. Accessed: 11th March 2025.
11. Wikipedia contributors. TerraSAR-X. Available at: <https://en.wikipedia.org/wiki/TerraSAR-X> (Accessed: 11th March 2025), 2025. Accessed: 11th March 2025.
12. Wang, C.; Bai, X.; Wang, S.; Zhou, J.; Ren, P. Multiscale visual attention networks for object detection in VHR remote sensing images. *IEEE Geoscience and Remote Sensing Letters* **2018**, *16*, 310–314. <https://doi.org/10.1109/LGRS.2018.2872355>.
13. Zhang, Y.; Ye, M.; Zhu, G.; Liu, Y.; Guo, P.; Yan, J. FFCA-YOLO for small object detection in remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing* **2024**, *62*, 1–15. <https://doi.org/10.1109/TGRS.2024.3363057>.
14. Deng, Z.; Sun, H.; Zhou, S.; Zhao, J.; Lei, L.; Zou, H. Multi-scale object detection in remote sensing imagery with convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing* **2018**, *145*, 3–22. <https://doi.org/10.1016/j.isprsjprs.2018.04.003>.
15. Wang, J.; Yang, W.; Guo, H.; Zhang, R.; Xia, G.S. Tiny object detection in aerial images. In Proceedings of the 25th International Conference on Pattern Recognition (ICPR), 2021, pp. 3791–3798. <https://doi.org/10.1109/ICPR48806.2021.9413340>.
16. Liu, S.; Huang, D.; Wang, Y. Learning Spatial Fusion for Single-Shot Object Detection. *arXiv e-prints* **2019**, p. arXiv:1911.09516. <https://doi.org/10.48550/arXiv.1911.09516>.
17. Zhang, X.; Liu, C.; Yang, D.; Song, T.; Ye, Y.; Li, K.; Song, Y. RFACnv: Innovating Spatial Attention and Standard Convolutional Operation. *arXiv e-prints* **2023**, p. arXiv:2304.03198. <https://doi.org/10.48550/arXiv.2304.03198>.
18. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. *arXiv e-prints* **2016**, p. arXiv:1612.03144. <https://doi.org/10.48550/arXiv.1612.03144>.

19. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 8759–8768. <https://doi.org/10.1109/CVPR.2018.00913>.
20. Ghiasi, G.; Lin, T.Y.; Le, Q.V. NAS-FPN: Learning Scalable Feature Pyramid Architecture for Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 7029–7038. <https://doi.org/10.1109/CVPR.2019.00720>.
21. Zhu, L.; Deng, Z.; Hu, X.; Fu, C.W.; Xu, X.; Qin, J.; Heng, P.A. Bidirectional Feature Pyramid Network with Recurrent Attention Residual Modules for Shadow Detection. In Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 122–137. https://doi.org/10.1007/978-3-030-01231-1_8.
22. Qiao, S.; Chen, L.C.; Yuille, A. DetectoRS: Detecting Objects With Recursive Feature Pyramid and Switchable Atrous Convolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 10213–10224. <https://doi.org/10.1109/CVPR46437.2021.01008>.
23. Yang, G.; Lei, J.; Zhu, Z.; Cheng, S.; Feng, Z.; Liang, R. AFPN: Asymptotic Feature Pyramid Network for Object Detection. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2023, pp. 2184–2189. <https://doi.org/10.1109/SMC53992.2023.10394415>.
24. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv e-prints* **2018**, p. arXiv:1804.02767. <https://doi.org/10.48550/arXiv.1804.02767>.
25. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv e-prints* **2020**, p. arXiv:2004.10934. <https://doi.org/10.48550/arXiv.2004.10934>.
26. Zhang, D.; Zhang, H.; Tang, J.; Wang, M.; Hua, X.; Sun, Q. Feature Pyramid Transformer. In Proceedings of the European Conference on Computer Vision (ECCV), 2020, pp. 323–339. https://doi.org/10.1007/978-3-030-58604-1_20.
27. Liu, H.I.; Tseng, Y.W.; Chang, K.C.; Wang, P.J.; Shuai, H.H.; Cheng, W.H. A DeNoising FPN With Transformer R-CNN for Tiny Object Detection. *IEEE Transactions on Geoscience and Remote Sensing* **2024**, *62*, 1–15. <https://doi.org/10.1109/TGRS.2024.3396489>.
28. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the Advances in Neural Information Processing Systems, 2014, pp. 73–76. https://doi.org/10.1007/978-3-658-40442-0_9.
29. Zhang, Y.; Li, K.; Li, K.; Wang, L.; Zhong, B.; Fu, Y. Image super-resolution using very deep residual channel attention networks. In Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 286–301. <https://doi.org/10.58837/CHULA.THE.2018.1563>.
30. Zhang, J.; Lei, J.; Xie, W.; Fang, Z.; Li, Y.; Du, Q. SuperYOLO: Super Resolution Assisted Object Detection in Multimodal Remote Sensing Imagery. *IEEE Transactions on Geoscience and Remote Sensing* **2023**, *61*, 1–15. <https://doi.org/10.1109/TGRS.2023.3258666>.
31. Jin, Y.; Zhang, Y.; Cen, Y.; Li, Y.; Mladenovic, V.; Voronin, V. Pedestrian detection with super-resolution reconstruction for low-quality image. *Pattern Recognition* **2021**, *115*, 107846. <https://doi.org/https://doi.org/10.1016/j.patcog.2021.107846>.
32. Bashir, S.M.A.; Wang, Y. Small Object Detection in Remote Sensing Images with Residual Feature Aggregation-Based Super-Resolution and Object Detector Network. *Remote Sensing* **2021**, *13*, 1854. <https://doi.org/10.3390/rs13091854>.
33. Truong, N.T.; Vo, N.D.; Nguyen, K. The effects of super-resolution on object detection performance in an aerial image. In Proceedings of the 7th NAFOSTED Conference on Information and Computer Science (NICS), 2020, pp. 256–260. <https://doi.org/10.1109/NICS51282.2020.9335859>.
34. Lim, J.S.; Astrid, M.; Yoon, H.J.; Lee, S.I. Small Object Detection using Context and Attention. In Proceedings of the International Conference on Artificial Intelligence in Information and Communication (ICAIIIC), 2021, pp. 181–186. <https://doi.org/10.1109/ICAIIIC51459.2021.9415217>.
35. Tang, X.; Du, D.K.; He, Z.; Liu, J. PyramidBox: A Context-assisted Single Shot Face Detector. In Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 797–813. https://doi.org/10.1007/978-3-030-01240-3_49.
36. Zhao, Z.; Du, J.; Li, C.; Fang, X.; Xiao, Y.; Tang, J. Dense Tiny Object Detection: A Scene Context Guided Approach and a Unified Benchmark. *IEEE Transactions on Geoscience and Remote Sensing* **2024**, *62*, 1–13. <https://doi.org/10.1109/TGRS.2024.3357706>.

37. Liang, X.; Zhang, J.; Zhuo, L.; Li, Y.; Tian, Q. Small Object Detection in Unmanned Aerial Vehicle Images Using Feature Fusion and Scaling-Based Single Shot Detector With Spatial Context Analysis. *IEEE Transactions on Circuits and Systems for Video Technology* **2020**, *30*, 1758–1770. <https://doi.org/10.1109/TCSVT.2019.2905881>.
38. Cui, L.; Lv, P.; Jiang, X.; Gao, Z.; Zhou, B.; Zhang, L.; Shao, L.; Xu, M. Context-Aware Block Net for Small Object Detection. *IEEE Transactions on Cybernetics* **2022**, *52*, 2300–2313. <https://doi.org/10.1109/TCYB.2020.3004636>.
39. Tong, K.; Wu, Y. Small object detection using hybrid evaluation metric with context decoupling. *Multimedia Systems* **2025**, *31*, 1–14. <https://doi.org/10.1007/s00530-025-01738-0>.
40. 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, 2016, pp. 779–788. <https://doi.org/10.1109/cvpr.2016.91>.
41. Redmon, J.; Farhadi, A. YOLO9000: better, faster, stronger. In Proceedings of the IEEE on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 7263–7271. <https://doi.org/10.1109/CVPR.2017.690>.
42. Jocher, G.; Chaurasia, A.; Stoken, A.; Borovec, J.; NanoCode012.; Kwon, Y.; Michael, K.; TaoXie.; Fang, J.; Imyhxy.; et al. ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation, 2022. <https://doi.org/10.5281/zenodo.3908559>.
43. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv e-prints* **2022**, p. arXiv:2209.02976. <https://doi.org/10.48550/arXiv.2209.02976>.
44. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 7464–7475. <https://doi.org/10.1109/CVPR52729.2023.00721>.
45. Jocher, G.; Qiu, J.; Chaurasia, A. Ultralytics YOLO (Version 8.0.0) [Computer software]. <https://github.com/ultralytics/ultralytics>, 2023. Accessed: 2025-03-02.
46. Wang, C.Y.; Yeh, I.H.; Mark Liao, H.Y. Yolov9: Learning what you want to learn using programmable gradient information. In Proceedings of the European Conference on Computer Vision (ECCV), 2024, pp. 1–21. https://doi.org/10.1007/978-3-031-72751-1_1.
47. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. *arXiv e-prints* **2024**, p. arXiv:2405.14458. <https://doi.org/10.48550/arXiv.2405.14458>.
48. Tian, Y.; Ye, Q.; Doermann, D. YOLOv12: Attention-Centric Real-Time Object Detectors. *arXiv e-prints* **2025**, p. arXiv:2502.12524. <https://doi.org/10.48550/arXiv.2502.12524>.
49. Qiu, M.; Huang, L.; Tang, B.H. ASFF-YOLOv5: Multielement detection method for road traffic in UAV images based on multiscale feature fusion. *Remote Sensing* **2022**, *14*, 3498. <https://doi.org/10.3390/rs14143498>.
50. Lin, J.; Zhao, Y.; Wang, S.; Tang, Y. YOLO-DA: An efficient YOLO-based detector for remote sensing object detection. *IEEE Geoscience and Remote Sensing Letters* **2023**, *20*, 1–5. <https://doi.org/10.1109/LGRS.2023.3303896>.
51. Li, Z.; Yuan, J.; Li, G.; Wang, H.; Li, X.; Li, D.; Wang, X. RSI-YOLO: Object detection method for remote sensing images based on improved YOLO. *Sensors* **2023**, *23*, 6414. <https://doi.org/10.3390/s23146414>.
52. Xie, S.; Zhou, M.; Wang, C.; Huang, S. CSPPartial-YOLO: A Lightweight YOLO-Based Method for Typical Objects Detection in Remote Sensing Images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **2024**, *17*, 388–399. <https://doi.org/10.1109/JSTARS.2023.3329235>.
53. Li, Y.; Chen, Y.; Wang, N.; Zhang, Z.X. Scale-Aware Trident Networks for Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 6053–6062. <https://doi.org/10.1109/ICCV.2019.00615>.
54. Cao, C.; Wang, B.; Zhang, W.; Zeng, X.; Yan, X.; Feng, Z.; Liu, Y.; Wu, Z. An Improved Faster R-CNN for Small Object Detection. *IEEE Access* **2019**, *7*, 106838–106846. <https://doi.org/10.1109/ACCESS.2019.2932731>.
55. Ren, Y.; Zhu, C.; Xiao, S. Small object detection in optical remote sensing images via modified faster R-CNN. *Applied Sciences* **2018**, *8*, 813. <https://doi.org/10.3390/app8050813>.
56. Tian, Z.; Shen, C.; Chen, H.; He, T. FCOS: Fully Convolutional One-Stage Object Detection. *arXiv e-prints* **2019**, p. arXiv:1904.01355. <https://doi.org/10.48550/arXiv.1904.01355>.
57. Xu, C.; Wang, J.; Yang, W.; Yu, H.; Yu, L.; Xia, G.S. RFLA: Gaussian receptive field based label assignment for tiny object detection. In Proceedings of the European Conference on Computer Vision(ECCV), 2022, pp. 526–543. https://doi.org/10.1007/978-3-031-20077-9_31.

58. Kang, S.H.; Park, J.S. Aligned Matching: Improving Small Object Detection in SSD. *Sensors* **2023**, *23*, 2589. <https://doi.org/10.3390/s23052589>.
59. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv e-prints* **2017**, p. arXiv:1706.03762. <https://doi.org/10.48550/arXiv.1706.03762>.
60. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In Proceedings of the European Conference on Computer Vision (ECCV), 2020, pp. 213–229. https://doi.org/10.1007/978-3-030-58452-8_13.
61. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable DETR: Deformable Transformers for End-to-End Object Detection. *arXiv e-prints* **2020**, p. arXiv:2010.04159. <https://doi.org/10.48550/arXiv.2010.04159>.
62. Li, F.; Zhang, H.; Liu, S.; Guo, J.; Ni, L.M.; Zhang, L. DN-DETR: Accelerate DETR Training by Introducing Query DeNoising. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 13609–13617. <https://doi.org/10.1109/CVPR52688.2022.01325>.
63. 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 **2022**. p. arXiv:2203.03605. <https://doi.org/10.48550/arXiv.2203.03605>.
64. Huang, Y.X.; Liu, H.I.; Shuai, H.H.; Cheng, W.H. DQ-DETR: DETR with Dynamic Query for Tiny Object Detection. In Proceedings of the European Conference on Computer Vision (ECCV); Leonardis, A.; Ricci, E.; Roth, S.; Russakovsky, O.; Sattler, T.; Varol, G., Eds., 2025, pp. 290–305. https://doi.org/10.1007/978-3-031-73116-7_17.
65. Zhao, Y.; Lv, W.; Xu, S.; Wei, J.; Wang, G.; Dang, Q.; Liu, Y.; Chen, J. DETRs Beat YOLOs on Real-time Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024, pp. 16965–16974. <https://doi.org/10.1109/CVPR52733.2024.01605>.
66. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 618–626. <https://doi.org/10.1109/ICCV.2017.74>.

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