GS-DETR: Accurate and Efficient Object Detection in UAV Imagery with Gated Feature Fusion and an Enhanced Pyramid

Junqi Wang[1], Xiangyang Lu[2], Yandan Wang[2], Hengyi Li[1]
{wangjq@zut.edu.cn, 5801@zut.edu.cn, wyd_048024@163.com, lihengyi@zut.edu.cn}

[1]School of Automation and Electrical Engineering, Zhongyuan University of Technology, Zhengzhou, China
[2]School of Information and Communication Engineering, Zhongyuan University of Technology, Zhengzhou, China

Abstract. Object detection in unmanned aerial vehicle (UAV) remote sensing imagery remains a significant challenge due to complex backgrounds, multi-scale objects, and a high prevalence of small targets. To address these issues, a Gated Feature Fusion Net and Small Object Detection Pyramid (GS-DETR) is presented, based on the Real-Time Detection Transformer (RT-DETR) framework. Specifically, a gated feature fusion net is designed to reduce model parameters. It features a lightweight backbone that enhances target feature representation through a unique gating mechanism and anisotropic feature extraction. Furthermore, a small object detection pyramid (SODP) is implemented to preserve high-resolution details and integrate a mini-kernel that utilizes a lossless down-sampling module for deep feature optimization. This design allows the framework to achieve a superior balance between detection accuracy and model efficiency while maintaining the real-time capabilities of its baseline. Extensive experiments on the VisDrone2019 and CoDrone datasets demonstrate that, compared to the baseline model, GS-DETR improves mAP@0.500 by 2.3% and 1.3%, respectively, while reducing the parameter count by 11.3%.

Keywords: Small Object Detection, UAV Remote Sensing, RT-DETR, Feature Fusion

1 Introduction

Unmanned Aerial Vehicle (UAV) remote sensing is a crucial method for Earth observation with wide applications in areas like precision agriculture and urban surveillance, but efficiently and accurately detecting objects from its imagery, especially small targets, remains a significant technical challenge [1]. UAV images are typically characterized by complex backgrounds, extreme variations in object scale, and a high prevalence of small targets. Traditional detection algorithms are often limited when handling these issues, as

IIKI 2025, December 18-21, Chengdu, People's Republic of China

DOI 10.4108/eai.18-12-2025.2365278

critical details are lost during the repeated downsampling in deep networks [2]. In recent years, deep learning has driven the rapid development of object detection, evolving from classic CNN models to novel Transformer architectures. Among these, RT-DETR [3] has emerged as an advanced real-time detector that balances accuracy and efficiency, but a performance gap remains in meeting the stringent demands of UAV remote sensing tasks. To address these challenges, this paper proposes a small object detection algorithm for UAV imagery named GS-DETR. The core innovations of this algorithm are twofold: the Gated Feature Fusion Net (GFF-Net) and a novel Small Object Detection Pyramid (SODP).

2 Related Work

2.1 Object Detection Algorithms

Mainstream deep learning object detection algorithms can be broadly categorized into three types: two-stage, one-stage, and Transformer-based methods.

Two-stage detectors: Represented by the R-CNN series (e.g., Fast R-CNN, Faster R-CNN [4]), these methods follow a "region proposal-classification" pipeline. They typically yield high accuracy but suffer from slow detection speeds, making them unsuitable for real-time applications.

One-stage detectors: Pioneered by the YOLO [5] series and SSD, these methods unify detection into a single framework to achieve superior efficiency. However, their reliance on deep backbone networks for feature extraction often leads to the loss of fine-grained details critical for small object detection.

Transformer-based detectors: DETR [6] pioneered an end-to-end framework by framing object detection as a set prediction problem, eliminating the need for post-processing steps like Non-Maximum Suppression (NMS). Subsequent work like Deformable DETR addressed DETR's slow convergence and high computational cost by introducing a more efficient attention mechanism [7]. To balance accuracy and efficiency for real-world applications, RT-DETR was proposed, becoming a highly competitive baseline model in real-time object detection.

2.2 UAV Small Object Detection

In the field of Unmanned Aerial Vehicle (UAV) object detection, numerous studies have focused on addressing the challenges posed by small objects and complex scenes. However, existing models commonly face a core trade-off: high-accuracy models, such as Drone-DETR [8], can achieve an mAP@0.5 of 53.9%, but their large parameter count (28.7M) and high computational cost (128.3 GFLOPs) limit their practical deployment on resource-constrained UAV platforms. Conversely, lightweight models like YOLO-PEL [9], while compact (2.23M parameters), have detection accuracy (32.5% mAP@0.5) that is often insufficient for practical applications.

Although improved models that attempt to balance performance, such as DV-DETR [10], show promise (achieving 50.2% mAP@0.5 with 19.5M parameters), they still have lim-

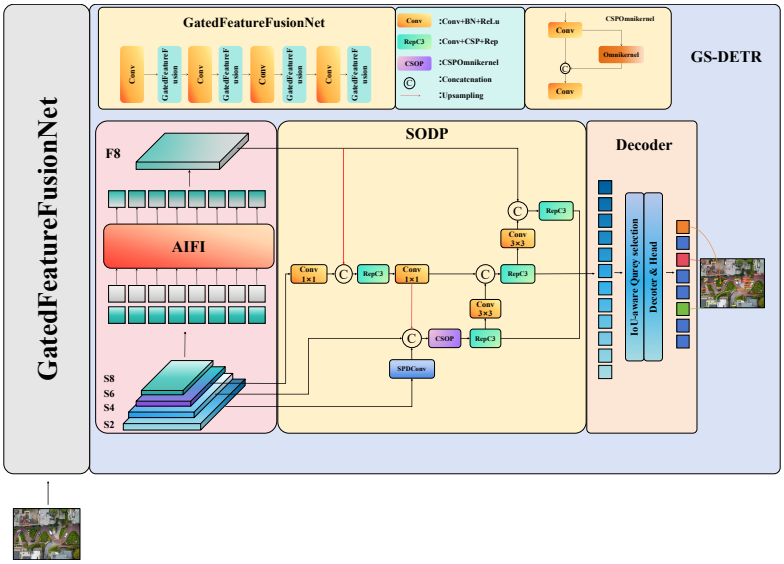

Fig. 1. Structural Diagram of GS-DETR

itations regarding feature alignment and contextual information fusion. This bottleneck inspired our innovations in two key areas: First, to address the issues of parameter redundancy and insufficient feature representation for small objects in traditional backbones, we propose the Gated Feature Fusion Net (GFF-Net). It significantly reduces the parameter count while more accurately capturing small object features. Second, to tackle the challenge of losing high-resolution details in feature pyramids, we designed the Small Object Detection Pyramid (SODP). As an advanced fusion structure, it can losslessly utilize high-resolution details and perform powerful multi-scale feature optimization.

2.3 RT-DETR Model

RT-DETR is a real-time, end-to-end object detector that strikes an optimal balance between high accuracy and computational efficiency through a streamlined three-part architecture. The process begins with a robust backbone that extracts hierarchical, multi-scale feature maps from the input image, providing a rich visual foundation for subsequent processing. These features are then aggregated by the neck, which utilizes a Path Aggregation Feature Pyramid Network (PAFPN) to effectively fuse deep semantic information with shallow spatial details via bidirectional top-down and bottom-up pathways. Finally, the architecture culminates in a detection head comprising an efficient hybrid encoder and decoder. The encoder significantly reduces computational complexity by decoupling the resource-intensive self-attention mechanism from cross-scale feature fusion. Concurrently,

the decoder leverages an IoU-aware query selection mechanism to generate higher-quality queries, thereby enhancing both decoding efficiency and overall detection accuracy.

3 Methodology

For object detection in UAV remote sensing images, we propose a new model that includes two innovations. The overall framework of our GS-DETR is shown in Figure 1. We build upon the robust RT-DETR architecture, but introduce a new backbone network and a specialized feature pyramid neck designed explicitly for the challenges of aerial imaging.

3.1 Reconstructing Backbone via Gated Feature Fusion

The standard ResNet backbone in RT-DETR is suboptimal for detecting small objects in UAV imagery due to its successive downsampling, which erodes fine spatial details, and its isotropic kernels, which are ill-suited for targets with high aspect ratios. To address this, we reconstruct the backbone using our proposed Gated Feature Fusion (GFF) module, designed for enhanced feature preservation and representation. The overall framework of our GFF-Net is shown in Figure 2.

The GFF module dynamically integrates features from two specialized pathways. First, a Feature Correlation Path, realized by our FCM [11] module, processes the input X to produce a contextually-rich feature map X_{enh}. This process begins by splitting the input channels into two groups, X_1 and X_2. These groups are then passed through separate convolutional blocks ($\mathscr{F}_1, \mathscr{F}_2$) and interact via a cross-attention mechanism to model their interdependencies:

$$X_1, X_2 = Split_c(X) \tag{1}$$

$$X_1' = \mathscr{F}_1(X_1), \quad X_2' = \mathscr{F}_2(X_2) \tag{2}$$

$$X_{\text{enh}} = \mathscr{F}_{\text{final}}\left(\mathscr{A}_S(X_2') \odot X_1' + \mathscr{A}_C(X_1') \odot X_2'\right) \tag{3}$$

Here, $\mathscr{A}_S$ and $\mathscr{A}_C$ denote spatial and channel attention modules, respectively. The spatial attention generated from the second feature group modulates the first, while the channel attention from the first group modulates the second. This reciprocal exchange enhances the feature representation, which is then fused by a final convolution, $\mathscr{F}_{final}$.

Concurrently, an Anisotropic Perception Path extracts multi-shape spatial priors using parallel depthwise convolutions—square (Φ_{sq}), horizontal (Φ_h), and vertical (Φ_v)—to produce a spatially-aware map $X_{spatial}$:

$$X_{\text{spatial}} = \mathscr{F}_{1\times1}(Concat[X, \Phi_{sq}(X), \Phi_h(X), \Phi_v(X)]) \tag{4}$$

Finally, these two pathways are integrated via a gating mechanism. The spatial feature map generates a pixel-wise attention gate using a sigmoid function $\sigma(\cdot)$ to adaptively modulate the contextual feature map. The final output Y of the GFF module is derived from their element-wise product ($\odot$):

$$Y = \sigma(X_{\text{spatial}}) \odot X_{\text{enh}} \tag{5}$$

This design enables our backbone to generate a more potent feature representation. By computing a spatial attention map, the gate learns to assign high weights (approaching 1) to regions containing potential small targets while assigning low weights (approaching 0) to background noise. As illustrated by the formulation (5), this multiplicative process selectively preserves and amplifies the weak yet crucial features of small objects.

Furthermore, this mechanism provides a decisive advantage in feature competition. In standard additive fusion, the powerful features of large objects often overwhelm and suppress the subtle features of smaller ones. The gating mechanism preempts this by using its learned spatial weights to amplify the response of small object features before fusion, ensuring they are not "drowned out."

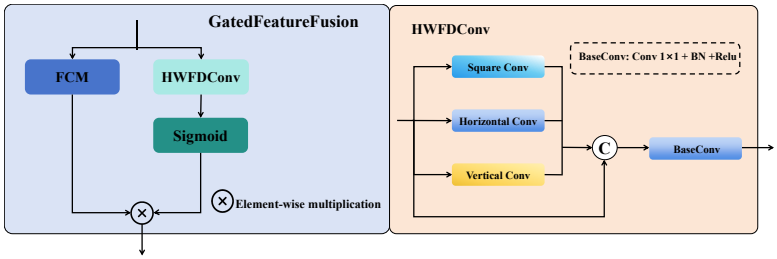

Fig. 2. Structural Diagram of Gated Feature Fusion Net

3.2 Small Object Detection Pyramid (SODP)

Conventional feature pyramids are inefficient for detecting small objects in UAV imagery, and adding a P2 detection head is computationally prohibitive. To overcome this trade-off, we propose the Small Object Detection Pyramid (SODP), a novel architecture that efficiently enhances small object features by strategically enriching the crucial P3 feature level.

The core principle of SODP is to enrich the P3 feature level, which is critical for detecting small-to-medium objects, with fine-grained spatial details from the P2 feature map. This is achieved through a carefully designed feature enhancement and fusion pipeline, replacing the standard top-down pathway in PAFPN. The workflow of our proposed SODP is detailed as follows.

Let the multi-scale feature maps extracted from our GFF-Net be denoted as $\{P_2, P_3, P_4, P_5\}$, corresponding to stride sizes of $\{4, 8, 16, 32\}$. The enhancement process begins at the P2 level. Instead of using strided convolutions or pooling which discard spatial information, we employ a Space-to-Depth Convolution ($\mathscr{F}_{SPD}$) module. The framework of SPDConv is shown in Figure 3.

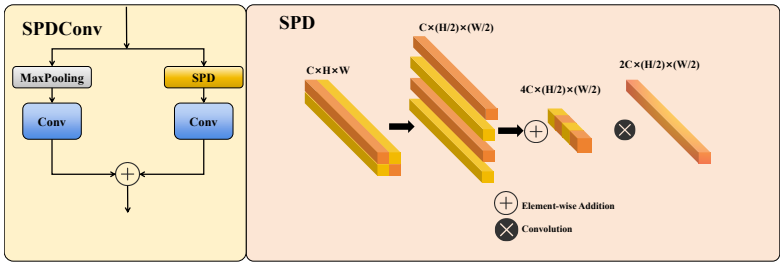

Fig. 3. Structural Diagram of SPDConv

This module first rearranges the high-resolution feature map $P_2 \in \mathbb{R}^{C_2 \times H \times W}$ into a lower-resolution map $P_2' \in \mathbb{R}^{4C_2 \times H/2 \times W/2}$ by re-organizing spatial blocks into channel depth. This operation, denoted as $\mathscr{S}_{S \to D}$, effectively preserves the fine-grained information in the channel dimension. A subsequent convolution then refines these features. The process can be formulated as (6).

$$P_2' = \mathscr{F}_{SPD}(P_2) = \text{Conv}(\mathscr{S}_{S \to D}(P_2)) \tag{6}$$

This lossless down-sampling technique ensures that critical details for identifying small objects are retained and passed to deeper layers.

To efficiently integrate heterogeneous features, we introduce the CSP-OmniKernel ($\mathscr{F}_{CSPOK}$) module, which is based on two key principles:

Cross Stage Partial (CSP) Architecture: Following the CSP design, the input feature map M_3' is split into two pathways. One path (M_{3a}') is processed by the main transformation block, while the other (M_{3b}') acts as a residual connection. This design reduces computational overhead and improves gradient flow.

OmniKernel Feature Enrichment: The powerful OmniKernel ($\mathscr{F}_{OK}$) block processes the M_{3a}' pathway. It uses a multi-branch, large-kernel architecture with a hybrid-domain attention mechanism to learn a comprehensive feature representation, spanning from local textures to global context, thereby enhancing small object features.

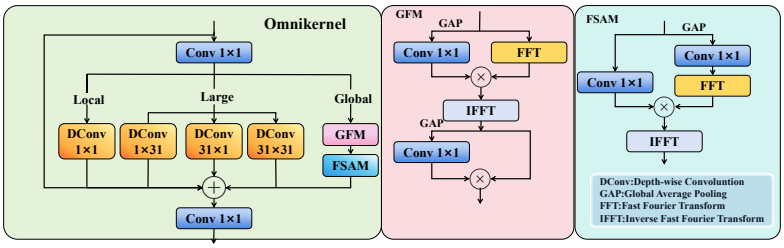

Fig. 4. Structural Diagram of OmniKernel

The outputs of both pathways are concatenated and fused by a convolution to produce the final output N_3:

$$N_3 = \text{Conv}(\text{Concat}[\mathscr{F}_{OK}(M'_{3a}), M'_{3b}]) \tag{7}$$

And the core of OmniKernel is a hybrid-domain attention mechanism consisting of two sequential modules. The framework diagram of OmniKernel is presented in Figure 4 to illustrate this structure clearly.

Global Fourier Module (GFM): This module first performs channel attention in the frequency domain by using channel weights (derived from GAP) to modulate the input's Fourier spectrum. After returning to the spatial domain via Inverse FFT, it applies a standard spatial channel attention to further refine the features. This process can be formulated as:

$$\begin{aligned} X_f &= \left|\mathscr{F}^{-1}\left(\mathscr{F}(X) \odot \text{Conv}_{1\times1}(\text{GAP}(X))\right)\right| \\ Y_{GFM} &= X_f \odot \text{Conv}_{1\times1}(\text{GAP}(X_f)) \end{aligned} \tag{8}$$

Frequency-Spatial Attention Module (FSAM): This module implements a feature gating mechanism. It projects the input X_{in} into two representations, which interact in the frequency and spatial domains. The result is additively fused back to the original input using learnable parameters α and β, enabling an adaptive residual connection. Its formulation is:

$$\begin{aligned} Y_{FSAM} = \beta \odot X_{in} + \alpha \odot \\ \left|\mathscr{F}^{-1}(\text{Conv}_1(X_{in}) \odot \mathscr{F}(\text{Conv}_2(X_{in})))\right| \end{aligned} \tag{9}$$

In summary, the proposed Small Object Detection Pyramid provides three distinct advantages:

Lossless Feature Down-sampling: It uses SPDConv to transfer fine-grained features from the P2 level without information loss, directly enhancing the representation of small objects.

Efficient Feature Enhancement: It boosts performance by enriching the existing P3 level, avoiding the significant computational and latency overhead of adding a dedicated P2 detection head.

Powerful Multi-Scale Feature Integration: The CSP-OmniKernel module powerfully fuses multi-scale features, enabling the model to better distinguish small objects from complex, cluttered backgrounds.

4 Experiments

To systematically evaluate the performance of our proposed GS-DETR model, this chapter will present a series of detailed experiments mainly on VisDrone2019.

4.1 Datasets

VisDrone2019 [12]: It is a large-scale UAV benchmark dataset constructed by a team from Tianjin University. Its image object detection task includes 8629 aerial images and 10

representative object categories. The dataset is highly challenging due to containing a large number of small objects with inconspicuous features, making it very suitable for researching small object detection algorithms in UAV aerial photography scenarios.

CoDrone [13]: It is a rotating object detection dataset from a UAV perspective released by Xiamen University in 2025. It aims to meet future practical application needs by providing higher-quality and more challenging data, including annotations for 12 categories. For use with our model, we processed its rotating annotations into horizontal annotations.

4.2 Experimental Configuration

The experiments were conducted on a Windows 11 system with PyTorch 2.5.1, Python 3.9, and CUDA 12.1. The hardware included an Intel Core i9-14900KF CPU and an NVIDIA GeForce RTX 4090 GPU with 24GB of VRAM. Key hyperparameters are summarized in Table 1.

Table 1: Hyperparameter configuration.

Name	Value	Name	Value
Optimizer	AdamW	Training Epochs:	250
ImageSize	640*640	Workers	8
InitialLearningRate	0.0001	LearningRateDecay	1.0
WeightDecay	0.0001	BatchSize	8
MomentumFactor	0.9	WarmupEpochs	2000

4.3 Evaluation Metrics

To evaluate the small object detection performance of GS-DETR, we use Precision (P), Recall (R), mean Average Precision (mAP), and Parameter count (Par) as evaluation metrics.

Precision (P) represents the ratio of correctly detected objects to the total number of detections. True Positive (TP) refers to the number of correctly detected targets; False Positive (FP) denotes the count of incorrect detections (false alarms); and False Negative (FN) represents the number of missed targets that the model failed to detect. It reflects the model's accuracy. The formula is:

$$P = \frac{TP}{TP + FP} \tag{10}$$

Recall (R) represents the ratio of correctly detected objects to the total number of actual objects. It reflects the model's detection coverage. The formula is:

$$R = \frac{TP}{TP + FN} \tag{11}$$

AP (Average Precision) is calculated by taking the average of the precision values on the PR curve. Mean Average Precision (mAP) is obtained by calculating the weighted average of the AP values for all object classes. Its formula is:

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

Parameter count (Par): The total number of trainable parameters in the model during training.

4.4 Comparative and Ablation Studies

To validate our design choices, we conducted a series of experiments on the VisDrone2019 dataset.

- Backbone Network Comparison: In our performance evaluation of lightweight backbones, we compared GFF-Net against the baseline, MobileNetV4 [14], and EfficientViT [15]. The empirical data confirms that GFF-Net successfully resolves the common accuracy-efficiency trade-off inherent in model lightweighting. While other architectures achieve more aggressive optimization, they do so at the cost of a significant decrease in accuracy. In contrast, GFF-Net reduces parameters by 29.36% and GFLOPS by 8.4% while simultaneously increasing the mAP@.50 to 0.480, making it the only design to improve efficiency while surpassing baseline accuracy and showcasing its superior architectural advantages.

Table 2: Comparison of Different Backbones

Model	P	R	mAP50	mAP50-95	GFLOPS	Params(M)
Base	0.625	0.461	0.477	0.292	57.0	19.89
MobileNetV4	0.572	0.420	0.427	0.254	40.9	11.62
EfficientViT	0.581	0.406	0.419	0.253	27.3	10.71
GFF-Net	0.614	0.469	0.480	0.291	52.2	14.05

- Feature Pyramid Comparison: To evaluate the contribution of our proposed neck architecture, we integrated the Small Object Detection Pyramid (SODP) into the baseline model and benchmarked it against the standard PAFPN (Base) as well as two other advanced feature fusion networks, UCTransNet [16] and DCMPNet [17]. The experimental results clearly indicate that competing architectures failed to yield effective improvements over the baseline. In stark contrast, our SODP exhibited a significant performance advantage, not only increasing the mAP@.50 by 1.3% to 0.490 but also substantially boosting the Recall by 1.8 percentage points. This decisive increase in Recall strongly confirms SODP's unique efficacy in capturing and identifying small targets, and its superior accuracy gain fully demonstrates the advanced nature of the design.

Table 3: Comparison of Different Neck-FPNs

Model	P	R	mAP50	mAP50-95	GFLOPS	Params(M)
Base	0.625	0.461	0.477	0.292	57.0	19.89
UCTransNet	0.601	0.457	0.468	0.285	57.4	29.53
DCMPNet	0.616	0.460	0.476	0.289	55.7	19.72
SODP	0.622	0.479	0.490	0.298	65.2	20.50

- Ablation Study: We conducted an ablation study to analyze the individual and synergistic effects of our two main contributions: GFF-Net (Module A) and SODP (Module B). The results in Table 4 strongly suggest their effectiveness. Either module alone improves upon the baseline. When combined (A+B), they achieve a synergistic enhancement, pushing mAP@.50 to 0.500—a 2.3% improvement over the baseline. Crucially, the final model achieves this superior performance with only 17.64M parameters, which is 11.3% lower than the baseline model's parameter count.

Table 4: Ablation experiments based on the RT-DETR baseline on the VisDrone2019 validation dataset.

Model	GFF-Net	SODP	mAP@.50	mAP@.50:.95	Params(M)	GFLOPS
Base			0.477	0.292	19.89	57.0
A	✓		0.480	0.291	14.05	52.2
B		✓	0.490	0.298	20.50	65.2
A + B	✓	✓	0.500	0.305	17.64	70.9

- Training Process Analysis: As shown by the training convergence curves in Figure 5, GS-DETR consistently outperforms the baseline RT-DETR on the VisDrone2019 dataset. Our model not only achieves a higher final mAP but also demonstrates a clear performance advantage throughout the training process, indicating that the proposed GFF-Net and SODP modules contribute to a more effective and efficient learning process.

4.5 Generalization on Different Datasets

To verify the generalization ability of the proposed method, the final model (Ours) was compared against the baseline RT-DETR model on two distinct datasets: VisDrone2019 and CoDrone. The results, presented in Table 5, demonstrate that the proposed model consistently outperforms the baseline across all core metrics on both datasets, providing strong evidence that the architectural improvements are not over-specialized to a single dataset but offer robust generalization for diverse UAV remote sensing scenarios.

On the VisDrone dataset, the proposed model achieved comprehensive outperformance. Specifically, it recorded a precision of 0.628, recall of 0.489, a mAP50 of 0.500, and a mAP50-

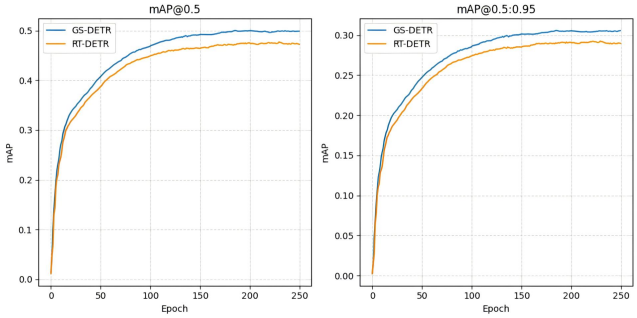

Fig. 5. Performance Comparison of GS-DETR and RT-DETR

Table 5: Generalization On Different Datasets

Dataset	Model	P	R	mAP_{50}	mAP_{50-95}
Visdrone	Rt-detr	0.625	0.461	0.477	0.292
	Ours	0.628	0.489	0.500	0.305
Codrone	Rt-detr	0.471	0.371	0.347	0.179
	Ours	0.486	0.391	0.36	0.188

95 of 0.305. This represents an improvement over the RT-DETR baseline in all categories, including a notable 2.3% increase in mAP50 from 0.477 to 0.500 and a 1.3% increase in mAP50-95 from 0.292 to 0.305.

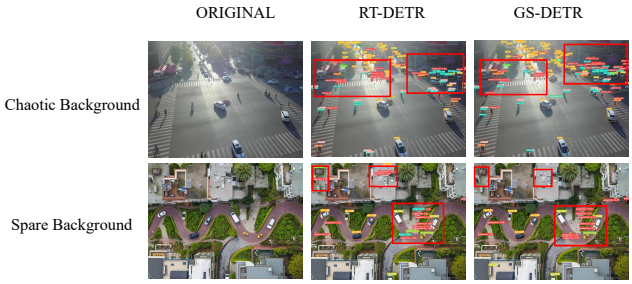

Fig. 6. Visdrone2019 dataset comparison shows GS-DETR outperforms RT-DETR in diverse backgrounds, with red boxes highlighting RT-DETR's false negatives/positives that GS-DETR avoids

The model's superiority was further validated on the CoDrone dataset, achieving a mAP50 of 0.36 (vs. 0.347) and a mAP50-95 of 0.188 (vs. 0.179). Such consistent improvements across diverse datasets underscore the model's robust generalization capabilities for real-world scenarios. This quantitative evidence is visually corroborated by the VisDrone2019 results, where GS-DETR successfully avoids the detection errors that affect the

baseline RT-DETR.

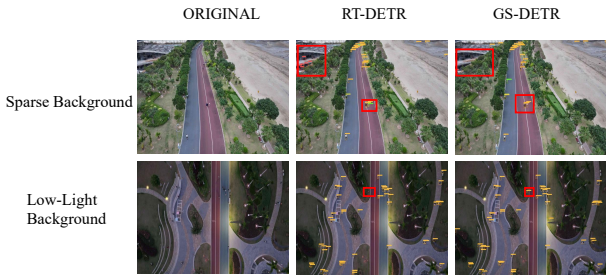

Fig. 7. Codrone dataset comparison shows GS-DETR outperforms RT-DETR in diverse backgrounds, with red boxes highlighting RT-DETR's false negatives/positives that GS-DETR avoids

4.6 Visual Analysis

Qualitative results on different datasets, as shown in Figures 6 and 7, further highlight the superiority of GS-DETR. The model demonstrates robust detection performance in the complex environmental backgrounds characteristic of both datasets.

In direct comparison with RT-DETR, GS-DETR significantly reduces both false negatives (missed objects) and false positives (incorrect detections), particularly in cluttered scenes and for objects at oblique angles. The red boxes in the figures indicate areas where RT-DETR failed, but GS-DETR performed correctly, showcasing its enhanced precision and recall.

5 Conclusion

This paper presents GS-DETR, an enhanced real-time detection framework that effectively addresses small object detection challenges in UAV remote sensing imagery. Our approach introduces two key innovations: the Gated Feature Fusion Network (GFF-Net) and the Small Object Detection Pyramid (SODP).

GFF-Net employs adaptive gating to prioritize informative features amidst background noise, successfully reducing backbone parameters by 29.36%. Meanwhile, SODP mitigates spatial detail loss via lossless Space-to-Depth convolution and strengthens multi-scale representation using the CSP-OmniKernel module. Extensive experiments on VisDrone2019 and CoDrone datasets demonstrate mAP@0.500 improvements of 2.3% and 1.3% respectively, with an 11.3% overall parameter reduction.

The consistent improvements across diverse datasets confirm the robustness and generalization capability of our approach. GS-DETR establishes a new benchmark for efficient

small object detection in UAV remote sensing, providing a practical solution that maintains real-time performance while achieving superior detection accuracy for small targets in complex aerial environments.

Acknowledgments

We thank the Henan Province Key Technologies Research and Development Project (Grants 252102211106, 252102320281), the National Natural Science Foundation of China (Grants 61975015, 62375017), and the Key R&D Project of Henan Province (251111220900) for their support. We also appreciate contributions from all team members.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] Zhang H, Liu K, Gan Z, Zhu GN. UAV-DETR: Efficient End-to-End Object Detection for Unmanned Aerial Vehicle Imagery. arXiv preprint arXiv:250101855. 2025.

[2] Song X, Fan B, Liu H, Wang L, Niu J. HPRT-DETR: A High-Precision Real-Time Object Detection Algorithm for Intelligent Driving Vehicles. Sensors. 2025;25(6):1778.

[3] Zhao Y, Lv W, Xu S, Wei J, Wang G, Dang Q, et al. DETRs Beat YOLOs on Real-Time Object Detection. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR); 2024. p. 16965-74.

[4] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans Pattern Anal Mach Intell. 2016 Jun;39(6):1137-49.

[5] Redmon J, Divvala S, Girshick R, Farhadi A. You Only Look Once: Unified, Real-Time Object Detection. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR); 2016. p. 779-88.

[6] Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. End-to-End Object Detection with Transformers. In: Proc. Eur. Conf. Comput. Vis. (ECCV). Springer; 2020. p. 213-29.

[7] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention Is All You Need. In: Adv. Neural Inf. Process. Syst. (NeurIPS). vol. 30; 2017. .

[8] Kong Y, Shang X, Jia S. Drone-DETR: Efficient Small Object Detection for Remote Sensing Image Using Enhanced RT-DETR Model. Sensors. 2024;24(17):5496.

[9] Wang Z, Zhang K, Wu F, Lv H. YOLO-PEL: The Efficient and Lightweight Vehicle Detection Method Based on YOLO Algorithm. Sensors. 2025;25(7):1959.

[10] Wei X, Yin L, Zhang L, Wu F. DV-DETR: Improved UAV Aerial Small Target Detection Algorithm Based on RT-DETR. Sensors. 2024;24(22):7376.

[11] Xiao Y, Xu T, Xin Y, Li J. FBRT-YOLO: Faster and Better for Real-Time Aerial Image Detection. In: Proc. AAAI Conf. Artif. Intell.. vol. 39; 2025. p. 8673-81.

[12] Du D, Zhu P, Wen L, Bian X, Lin H, Hu Q, et al. VisDrone-DET2019: The Vision Meets Drone Object Detection in Image Challenge Results. In: Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW). Seoul, South Korea; 2019. p. 213-26.

[13] Ye K, Tang H, Liu B, Dai P, Cao L, Ji R. More Clear, More Flexible, More Precise: A Comprehensive Oriented Object Detection Benchmark for UAV. arXiv preprint arXiv:250420032. 2025.

[14] Qin D, Leichner C, Delakis M, Fornoni M, Luo S, Yang F, et al. MobileNetV4: Universal Models for the Mobile Ecosystem. In: Proc. Eur. Conf. Comput. Vis. (ECCV). Springer; 2024. p. 78-96.

[15] Liu X, Peng H, Zheng N, Yang Y, Hu H, Yuan Y. EfficientViT: Memory Efficient Vision Transformer with Cascaded Group Attention. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR); 2023. p. 14420-30.

[16] Wang H, Cao P, Wang J, Zaiane OR. UCTransNet: Rethinking the Skip Connections in U-Net from a Channel-Wise Perspective with Transformer. In: Proc. AAAI Conf. Artif. Intell.. vol. 36; 2022. p. 2441-9.

[17] Zhang Y, Zhou S, Li H. Depth Information Assisted Collaborative Mutual Promotion Network for Single Image Dehazing. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR); 2024. p. 2846-55.