# FESL-YOLO: Improved YOLOv11 Small Object Detection Algorithm for Aerial Images

Wei-Qing Ge[1], Heyrim-Ju[1], Wang-Su Jeon[2*]
Department of IT Convergence Engineering[1], Department of Computer Engineering[2]
University of Kyungnam, South Korea, Changwon City.
1060137524@qq.com[1], gpfla2030@naver.com[1], jws2218@naver.com[2*]

***Abstract*—To address the challenges posed by UAV flight conditions such as long distances from targets, significant variations in object sizes, and occlusions this paper introduces FESL-YOLO (Feature Enhancement and Small-object detection Layer YOLO), a YOLO-based algorithm built upon YOLOv11s and augmented with feature enrichment and a small object detection layer, aiming to advance small object detection efficacy from a UAV perspective. A novel Feature Enhancement Convolution module (FEConv) is designed to replace the original convolutional structures in the backbone network, strengthening feature acquisition and representation capabilities with minimal increase in parameters. Given the typical characteristics of small objects such as small size, sparse information, and dense distribution and the structural limitation of YOLOv11 due to its aggressive down sampling, a dedicated small object detection branch is further introduced. This branch extracts fine-grained target information from shallow feature maps and fuses it with deep features, hence enhancing the model's ability to perceive small objects and considerably improving detection precision. Experimental results on the VisDrone2019 dataset demonstrate that our algorithm achieves 44.6% in mAP@0.5 and 27.1% in mAP@0.5:0.95, representing gains of 6.1%p and 4.3%p, respectively, over the original YOLOv11s model. These results verify the effectiveness and superior performance of our method for small object detection tasks in UAV-based scenarios.**

***Keywords—UAV, Small Object Detection, YOLOV11, Feature Extraction, Lightweight Object Detection***

## I. INTRODUCTION

In recent years, unmanned aerial vehicles (UAVs) have been widely applied across various fields such as military reconnaissance, commercial inspection, and everyday life due to their advantages of small size, low cost, and flexible deployment. With continuous advancements in UAV technology, their application value in scenarios such as remote sensing image detection [1], urban traffic monitoring [2], and aerial patrol [3] has become increasingly prominent. At the same time, deep learning has made breakthrough progress in the field of image processing, exhibiting stronger robustness and accuracy compared to traditional methods, especially in handling complex remote sensing tasks, hence providing new technical support for UAV platforms. The integration of UAVs and deep learning offers efficient and automated solutions for information acquisition and intelligent analysis across multiple domains.

Driven by deep learning, object detection technology has developed rapidly. The introduction of deep convolutional neural network (CNN) has greatly elevated detection efficacy. Current mainstream object detection approaches can be broadly categorized into two-stage and one-stage methods. Two-stage algorithms (e.g., Faster R-CNN [4], Cascade R-CNN [5], and Mask R-CNN [6]) typically generate region proposals followed by classification and regression, achieving high detection precision but often at the cost of heavy computation, making them less suitable for real-time applications. To address this, researchers have introduced more efficient one-stage detection methods such as SSD [7], RetinaNet [8], and the YOLO series [9]. These approaches eliminate the proposal generation step and perform end-to-end prediction directly, substantially boosting inference speed and making them more suitable for deployment on UAV platforms with limited computational resources.

Among them, the YOLO series has gained widespread attention for its simple architecture, high speed, and deployment-friendly characteristics. It reformulates the object detection task as a regression problem, relying on a single neural network for both object localization and classification, which considerably reduces dependence on hardware and computational resources. However, when dealing with small object detection tasks in aerial imagery captured from high altitudes, YOLO still encounters performance bottlenecks. Due to the small size of objects, weak feature representation, and frequent occlusion by challenging backgrounds, traditional detection models tend to miss or misidentify targets, thus reducing overall detection precision.

Compared with conventional image processing tasks, image analysis in UAV aerial scenarios presents greater complexity and challenges. On one hand, variations in flight altitude and viewing angles introduce significant scale inconsistencies, leading to a higher proportion of small objects in the images. On the other hand, small objects inherently carry limited information, often exhibit blurred boundaries, and tend to be closely packed, making them easily confused with the background and difficult to segment and recognize accurately. In addition, the presence of substantial noise and interference in the background further increases the modeling difficulty for small objects. Therefore, designing more robust detection algorithms with strengthened small object perception capability is a critical research direction for refining the intelligent perception of UAV systems.

In summary, small object detection in aerial imagery remains challenging due to limited feature extraction capability and frequent loss of fine object details during processing. To address these challenges, this paper suggests FESL-YOLO, a YOLO-based framework that has enhanced feature extraction and a specific small-object detection layer. The main contributions of this work are as follows:

**Acknowledgment: This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP)-Innovative Human Resource Development for Local Intellectualization program grant funded by the Korea government (MSIT)(IITP-2025-RS-2024-00436773).**

***Corresponding author**

- To address the shortcomings of standard convolutions in effectively capturing fine-grained local features, we have introduced the FEConv module to compensate for this. Spatial feature extraction and channel-wise feature recalibration within the backbone can be improved by FEConv by combining max-pooling with Squeeze-and-Excitation (SE) attention.

To prevent the loss of information caused by YOLOv11's aggressive downsampling, we introduce a specific small object detection layer. Cross-scale fusion allows for the cross-resolution spatial details essential for detecting densely distributed small objects to be preserved while bridging shallow and deep features.

## II. RELATED WORKS

### A. Small Object Detection in UAV

Small target detection poses significant challenges in the field of object detection due to the small size, weak semantic information, and easy confusion with the background, especially in applications such as remote sensing monitoring and security surveillance. Currently, small objects are mainly defined in two ways: one is absolute small objects, referring to targets with dimensions less than 32×32 the other is relative small objects, referring to targets whose area occupies less than 10% of the total image area. According to the International Society for Optics and Photonics (SPIE), a target is considered small if its area is less than 0.12% of the total image area in a 256×256 image. The main difficulties in small object detection lie in feature loss, insufficient localization accuracy, and lack of contextual information.

To address these issues, researchers propose improvements from multiple perspectives. Yang et al. [9] propose QueryDet, which fully leverages the advantages of high-resolution feature maps while avoiding redundant computation over background regions, thus improving detection capability for small objects. Koyun et al. [10] design a two-stage detection framework called Focus and Detect, specifically optimized for small object detection, markedly improving localization accuracy. To further advance small object detection capability, an increasing number of studies explore integrating attention mechanisms into detection networks. Wang et al. [11] introduce the Convolutional Block Attention Module (CBAM) [13] based on YOLOX [12], enabling the model to focus on key features and suppress irrelevant information; however, there remains room for performance improvement in complex backgrounds. Li et al. [14] further augment the network's sensitivity to targets by incorporating a global scheduling and the GAM attention[15].

In the domain of UAV target detection, Yuan et al. [16] propose an Infrared Small Target Detection Module (IRSTDM) specially designed for small UAV targets, effectively retaining target detail information and improving small target detection capability. Meng et al. [17] develop a thermal infrared moving target detection method called LAGSwin (Locally focused Attention-based Swin-transformer), which encodes spatial transformations and directional information of moving targets to strengthen interaction and fusion of features across different resolutions. However, the computational complexity of this method may exceed the processing capabilities of UAV devices. Sun et al. [18] propose the Multi-YOLOv8 model for infrared moving target detection, which takes the current frame, background difference image, and optical flow image as inputs to fuse original features, target information, and motion information, thus capably improving detection capability.

### B. YOLO model

YOLOv11 is the most recent version of the YOLO series of real-time object detectors, with its architecture depicted in Figure 1. Although the YOLO series has been developed by various teams since YOLOv4, we selected YOLOv11 as our baseline for several reasons: (1) it represents the latest architectural advances from the Ultralytics team who also developed YOLOv5 and YOLOv8, ensuring continuity in optimization strategies; (2) YOLOv11 demonstrates improved small object detection capability compared to YOLOv10, which has been reported to exhibit misclassification issues in dense scenarios; and (3) its C3K2 and C2PSA modules provide a more suitable foundation for our feature enhancement modifications than YOLOv9's GELAN architecture, which is optimized for different use cases. The refined backbone and neck structure in YOLOv11 [18] improves feature extraction capacity and detection accuracy on complex tasks. Compared with YOLOv8, YOLOv11 replaces the CF2 module with C3K2, introduces a new C2PSA module following the SPPF block, and incorporates the head design concept from YOLOv10 by using depthwise separable convolutions to reduce redundant computation and increase computational efficiency.

In UAV aerial imagery with densely distributed small objects that take up a relatively large portion of the scene, the YOLOv11 architecture is still struggling to meet higher detection requirements. To address these limitations, this paper presents the FESL-YOLO algorithm, designed to address common issues like missed and false detections in UAV imaging and significantly improve detection performance, especially for small targets.

## III. METHODOLOGY

### A. Overall model

The YOLOv11 architecture is not capable of capturing fine-grained features in UAV aerial imagery, as demonstrated in Figure 1, as small targets are prone to being affected by background complexity and occlusion, as well as extremely limited pixel representation. The result of these limitations is missed detections and false positives, which can affect overall detection performance.

The propose of this paper is to propose FESL-YOLO, a YOLO-based framework that incorporates feature enhancement and a small-object detection layer to address these limitations. As shown in Figure 2, the proposed architecture effectively addresses these challenges and greatly improves the accuracy of detecting small objects.

To enhance the network's perception capability for small objects while maintaining model lightweight design and improving detection precision and practicality in complex scenarios, we designed a FEConv module to replace standard convolutional structures in the original backbone network. The module combines max-pooling with SE attention. It not only reduces the size of the feature map and emphasizes relevant areas, but also adjusts feature weights based on the importance of channels.

As a result, it enhances the representation of crucial details like textures and edges. Furthermore, this work enhances the Neck structure of the original model by adding a separate

branch that is specifically designed to detect small objects. This branch allows for the use of both shallow and deep-level features to fuse detection, leading to significant improvements in detection of multi-scale objects, especially small targets.

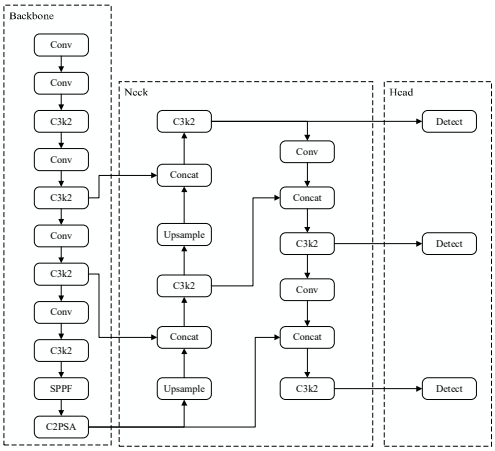



**Figure 1. Overall Architecture of YOLOv11**

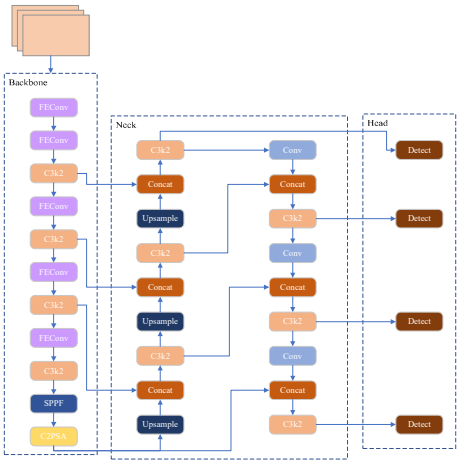



**Figure 2. Overall Architecture of the FESL-YOLO**

## B. *Feature Enhancement Convolution Module*

This paper presents a convolutional module that enhances the backbone network's ability to perceive and model small-object features, which can be seen in Figure 3, to enhance its ability to perceive and model small-object features. The FEConv module consists of two parallel branches: the main branch sequentially applies a convolutional layer, max pooling, and a Squeeze-and-Excitation (SE) attention module, while the residual branch provides a shortcut path to preserve the original feature information. The dual-branch structure can be seen in the second and third stages of Figure 3. Batch normalization (BN), an activation function (SiLU), and a residual path are all included in the module.

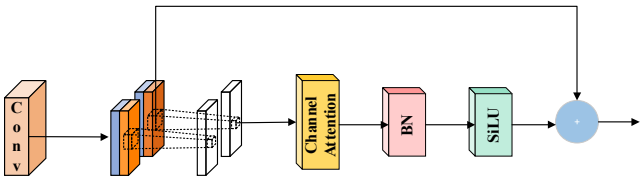



**Figure 3. Structure of FEConv**

Let the input feature map of the module be denoted as $X \in \mathbb{R}^{C\times H\times W}$.First, a standard convolutional layer is applied to extract local spatial features, and the output is given as follows:

$$F_1 = F_{3\times3}^{Conv}(X) \tag{1}$$

In Equation (1), X denotes the input feature map (C×H×W: channels × height × width), $F_{3\times3}^{Conv}$ denotes the 3×3 standard convolution operation, and $F_1$ represents the output feature map after convolution. Subsequently, a max pooling operation is applied to strengthen local response capability and compress spatially redundant information, as formulated in Equation (2):

$$F_2 = max_{u=0}^{f-1} max_{v=0}^{f-1} X\,(i\times s+u, j\times s+v) \tag{2}$$

Here, $max$denotes the maximum operation. In Equation (2), $F_2$ represents the output feature map after max pooling, f denotes the pooling window size, s represents the stride, and $u$ and $v$ are indices ranging within $[0, f-1]$, used to iterate over each element within the pooling window. $s$ $X(i\times s+u, j\times s+v)$ refers to the element in the input feature map $F_1$that corresponds to the local window associated with the output feature map.

The current pooling window represents the receptive field, where max pooling preserves the most prominent features within the local region. This helps highlight object edges and texture information, while also providing a shortcut path for residual connections. To strengthen semantic modeling across channels, the FEConv module incorporates an SE attention mechanism, as illustrated in Figure 4. Here, F_tr denotes the transformation function for initial feature mapping.

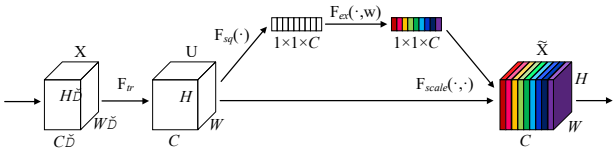



**Figure 4. Structure of SE Attention**

This module performs adaptive channel-wise weighting through a three-stage modeling process. It first applies global average pooling to compress the spatial dimensions, as shown in the following equation:

$$z_c = F_{sq}(u_c) = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} u_c(i,j) \tag{3}$$

$$s = F_{ex}(z,W) = \sigma(g(z,W)) = \sigma(W_2\delta(W_1 z)) \tag{4}$$

Here, $W$ denotes the convolutional kernel used for learning weights, $\sigma$ represents the Sigmoid activation function (for the output gating), and $\delta$ denotes the ReLU activation function (for the hidden layer). In Equation (3), the symbol $z_c$ represents the channel descriptor for the c-th channel, $F_{sq}$ denotes the squeeze function (global average pooling), $u_c$ is the c-th channel of the input feature map, H and W are the spatial height and width of the feature map, and $u_c(i,j)$ indicates the feature value at spatial position (i,j) of the c-th channel. In Equation (4), the symbol s represents the channel attention weight vector, $F_{ex}$ denotes the excitation function, z is the vector of all channel descriptors, W represents the learnable weight parameters, $W_1$ is the weight matrix for

dimensionality reduction, $W_2$ is the weight matrix for dimensionality restoration. $W_1$ performs dimensionality reduction with $W_1 \in R^{\frac{C}{r} \times C}$, and $W_2$ performs restoration, where r is the reduction ratio. Matrix $W_2 \in R^{\frac{C}{r} \times C}$.

Finally, the Scale operation applies the learned attention weights to each channel of the feature map, as expressed in the following formula:

$$\tilde{x}_c = F_{scale}(u_c, s_c) = s_c u_c \quad (5)$$

In Equation (5), $\tilde{x}_c$ is the recalibrated output for channel c, computed by scaling $u_c$ with attention weight $s_c$. $F_{scale}$ denotes the channel-wise scaling function, $u_c$ is the original feature map of the c-th channel before attention weighting, and $s_c$ represents the learned attention weight for the c-th channel computed from Equation (4). The SE attention mechanism successfully enhances the network's focus on key object regions, making it especially suitable for complex scenarios with sparse small object information. After the attention modulation, the feature maps are further normalized and non-linearly transformed by a BN layer and the SiLU activation function, which strengthens training stability and generalization capability. Finally, FEConv employs a residual connection to perform element-wise addition between the pooling branch output and the main branch, effectively mitigating the vanishing gradient problem and promoting feature fusion between shallow structures and deep semantic information. Without substantially increasing the number of parameters, this approach substantially enhances the network's receptive field, feature representation capability, and local responsiveness, and especially elevating detection precision and robustness in low-resolution, densely packed small object scenarios from a UAV perspective.

### C. Small Object Detection Layer

In UAV applications, target objects in images often appear significantly scaled down, with blurred edges and densely packed distributions. Deep feature maps constructed through large-scale downsampling can only represent a limited amount of data, especially in architectures such as YOLOv11. The degradation or loss of texture, contour, and boundary information of small targets during propagation is caused by feature maps being repeatedly downsampled, resulting in a drastic reduction in their spatial resolution. To overcome this structural bottleneck, we introduce a dedicated small-object-aware detection branch based on the original backbone network, creating a high-resolution detail restoration path for improved detection of small objects.

This branch operates primarily in the early to mid-stages of the network, directly extracting spatial structural information from shallow feature maps where such details have remained intact. To address the lack of semantic richness in shallow layers, we incorporate a progressive upsampling strategy combined with the fusion of supplementary features from higher semantic layers in the backbone.

We develop a mutually advantageous relationship between superficial spatial representations and deep semantic features within the feature map through alignment mechanisms. The fusion method takes into consideration hierarchical differences in features and inter-channel dependencies, enabling the model to maintain an extensive receptive field while also being more sensitive to micro-scale target regions. In order to enhance fusion accuracy and contextual consistency, we utilize feature alignment, channel balancing, and non-linear transformation to features at various levels. By doing this, the detection process can achieve more robust fine-grained modeling.

This approach, unlike conventional pyramid structures or simple skip connections, emphasizes structural guidance and semantic reconstruction, which enhances the model's ability to locate and recognize low-saliency targets in complex viewpoints.

## IV. EXPERIMENTS AND ENVIRONMENT

The experiments are conducted on a Windows 11 host system, with model training and evaluation performed in an Ubuntu 20.04 environment. Computations are accelerated using an NVIDIA RTX 3090 GPU with CUDA 11.8, and the implementation on PyTorch 2.0.0.

Training is performed for 200 epochs with an input resolution of 640 by 640 and a batch size of 16. Stochastic Gradient Descent (SGD) is adopted with an initial learning rate of 0.01, momentum of 0.937, and weight decay of 0.0005. A fixed learning rate is maintained throughout training to ensure stable convergence. For fair comparison, all models are trained and evaluated under identical experimental settings.

### A. Dataset and Evaluation Metrics

To validate the effectiveness and advancement of our method, experiments are conducted on the publicly available VisDrone2019 [19] dataset. This dataset contains a total of 10,209 static images, including 6,471 for training, 548 for validation, and 3,190 for testing. The images are captured by various UAV mounted cameras, covered a wide range of scenarios, and are annotated with 10 different object categories. We adopted Precision (P), Recall (R), mAP@0.5, and mAP@0.5:0.95 as evaluation metrics. Specifically, mAP@0.5 refers to the mean Average Precision calculated across all classes at an Intersection over Union (IoU) threshold of 0.5, while mAP@0.5:0.95 denotes the average precision computed over multiple IoU thresholds ranging from 0.5 to 0.95 with a step size of 0.05.

The calculation formulas are as follows:

$$P = \frac{TP}{TP + FP} \quad (6)$$

$$R = \frac{TP}{TP + FN} \quad (7)$$

$$mAP = \frac{1}{n} \sum_{i=1}^{n} AP_i \quad (8)$$

**TABLE I. RESULT OF COMPARE EXPERIMENT**

| Method | P | R | mAP50 | mAP 50:95 | Params | GFLOPs | FPS |
|---|---|---|---|---|---|---|---|
| Faster RCNN | | 33.8 | 33.2 | - | 136.75 | 369.8 | 31.6 |
| SSD | 21 | 35.5 | 23.9 | - | 24.01 | 61.06 | 167.4 |
| YOLO v3-tiny | 27.8 | 18.5 | 15.8 | 6.9 | 8.67 | 13.0 | 429.5 |
| YOLO v5s | 43.2 | 32.8 | 32.0 | 17.1 | 7.82 | 18.7 | 168.8 |
| YOLO v7-tiny | 47.1 | 35.4 | 33.9 | 17.5 | 6.02 | 13.2 | 126.6 |
| YOLO v8s | 49.1 | 37.5 | 38.3 | 22.8 | 11.13 | 28.4 | 178.8 |
| YOLO v11s | 49.3 | 37.6 | 38.5 | 22.8 | 9.42 | 21.3 | 141.7 |
| **Our** | **54.9** | **42.8** | **44.6** | **27.1** | 9.66 | 47.4 | 96.9 |

In Equations 6 to 8, TP, FP, and FN represent true positives, false positives, and false negatives, respectively. P

and R denote precision and recall, n is the number of classes, and AP for each class is averaged to compute mAP.

### *B. Comparative Experimental Results and Analysis*

To validate the effectiveness of the proposed approach for small object detection, representative detectors including Faster R-CNN, SSD, YOLOv3-tiny, YOLOv5s, YOLOv7-tiny, YOLOv8s, and the baseline YOLOv11s are selected for comparison. All models are evaluated using a unified training strategy with an input resolution of 640 by 640. The quantitative results are summarized in Table 1.

Traditional detectors exhibit limited performance in small object scenarios. Faster R-CNN and SSD achieve mAP at IoU 0.50 values of 33.2% and 23.9%, respectively. Owing to its shallow architecture, YOLOv3-tiny records precision of 27.8%, recall of 18.5%, mAP at IoU 0.50 of 15.8%, and mAP at IoU 0.50 to 0.95 of only 6.9%. With the evolution of detection architectures, YOLOv5s, YOLOv7-tiny, YOLOv8s, and YOLOv11s demonstrate progressively improved performance. Among them, YOLOv11s achieves 49.3% precision, 37.6% recall, 38.5% mAP at IoU 0.50, and 22.8% mAP at IoU 0.50 to 0.95, representing one of the strongest lightweight baselines.

In contrast, the proposed method delivers consistent improvements across all evaluation metrics. Precision and recall reach 54.9% and 42.8%, improving by 5.6%p and 5.2%p over YOLOv11s. The mAP at IoU 0.50 and mAP at IoU 0.50 to 0.95 increase to 44.6% and 27.1%, corresponding to gains of 6.1%p and 4.3%p. Compared with other lightweight models, the proposed approach improves mAP at IoU 0.50 to 0.95 by 10.0%p over YOLOv5s and by 9.6%p over YOLOv7-tiny.

In terms of model complexity, FESL-YOLO contains 9.66 million parameters, which is only a 2.5% increase over the 9.42 million parameters of YOLOv11s. Despite this marginal increase, the proposed model achieves performance gains of 6.1%p in mAP at IoU 0.50 and 4.3%p in mAP at IoU 0.50 to 0.95. These results demonstrate that FESL-YOLO significantly enhances small object detection performance while maintaining computational efficiency suitable for resource constrained UAV platforms.

### *C. Ablation study*

To verify the effectiveness of the proposed modifications for UAV aerial image object detection, ablation experiments are conducted on the VisDrone2019 dataset using YOLOv11s as the baseline. The proposed components are introduced incrementally: FEConv is first integrated into the baseline model, followed by the addition of the small object detection layer. The corresponding experimental results are presented in Table 2.

TABLE II. ABLATION EXPERIMENT RESULT

| | P | R | mAP50 | mAP50 :95 | Params | GFLOPs | FPS |
|---|---|---|---|---|---|---|---|
| YOLO v11s | 49.3 | 37.6 | 38.5 | 22.8 | 9.42 | 21.3 | 141.7 |
| +FEConv | 51.0 | 38.5 | 39.6 | 23.7 | 9.46 | 38.9 | 116.9 |
| +FEConv +layer | 54.9 | 42.8 | 44.6 | 27.1 | 9.66 | 47.4 | 96.9 |

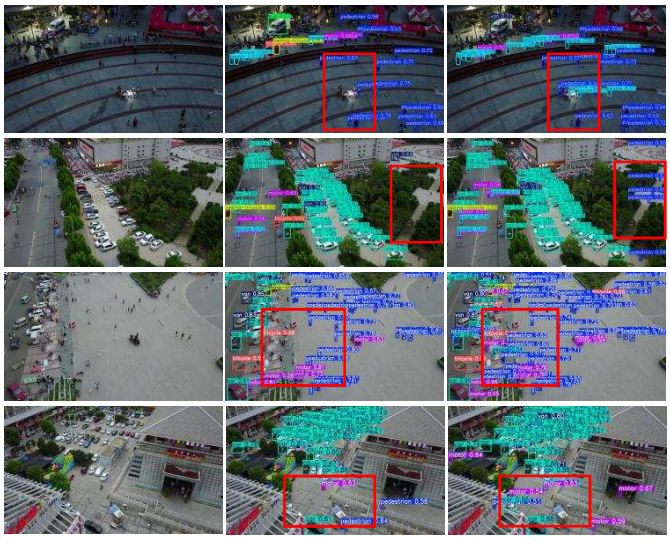
The experimental results show that integrating the FEConv module into YOLOv11s leads to consistent performance improvements. Precision increases by 1.7%p from 49.3% to 51.0%, recall improves by 0.9%p from 37.6% to 38.5%, mAP at IoU 0.50 rises by 1.1%p from 38.5% to 39.6%, and mAP at IoU 0.50 to 0.95 increases by 0.9%p to 23.7%. These gains indicate that FEConv effectively enhances feature representation by strengthening attention to key object regions, particularly improving local structure modeling for small objects while suppressing redundant information.

When a dedicated small object detection layer based on shallow features is further introduced, the performance improvement becomes more pronounced. Precision increases by 3.9%p from 51.0% to 54.9%, recall improves by 4.3%p from 38.5% to 42.8%, mAP at IoU 0.50 rises by 5.0%p from 39.6% to 44.6%, and mAP at IoU 0.50 to 0.95 increases by 3.4%p from 23.7% to 27.1%. This result demonstrates that the integration of shallow and deep features effectively compensates for the loss of small object information in deeper layers, significantly enhancing the detection of small and low contrast objects.

Overall, the proposed lightweight modifications substantially improve small object detection performance, confirming the robustness and practical applicability of the proposed architecture in dense and complex background scenarios.

### *D. Qualitative result*

To demonstrate the effectiveness of our method in complex real-world scenarios with ease and convenience, we select representative UAV images from the VisDrone2019 dataset for visualization-based comparative experiments. In Figure 5, we present the original images (a), detection results of YOLOv11 (b), and detection results of our method (c). In the red rectangular regions, the main areas of comparison are visually highlighted and demonstrate the practical benefits of our method for small-object recognition, occlusion handling, and dense-region detection.



**(a) (b) (c)**

**Figure 5. Detection performance comparison result on VisDrone2019 dataset. (a) Original image, (b) YOLOv11s, (c) FESL-YOLO.**

Under poor lighting conditions, the first sample contains targets that are extremely small. The proposed method is robust to low-resolution inputs and successfully detects these micro objects despite noise interference, demonstrating its robustness to low-resolution inputs. YOLOv11 is unable to reliably detect these micro objects.

Partially obscured targets are present in a cluttered background in the second sample. Uncertain predictions cause YOLOv11 to produce false positives and miss detections. On the other hand, the proposed method effectively eliminates background interference, separates overlapping objects, and produces more precise bounding boxes for occluded or blurred targets.

The third sample displays multiple small vehicles and pedestrians densely distributed throughout the scene. Missed detections and localization errors are common in YOLOv11, with many objects being either undetectable or poorly aligned. By detecting small objects with tighter and more precise bounding boxes, the proposed approach shows stronger perception in dense small-object scenarios.

Targets with similar visual appearances are heavily occluded and densely packed in the fourth sample. YOLOv11 has a problem with missed detections and class confusion, especially when it comes to motorcycles and vehicles. Improved stability is demonstrated by the proposed method's ability to accurately distinguish object categories and produce bounding boxes that closely match object boundaries, even in low-contrast conditions.

The proposed method consistently enhances the detection accuracy of small objects that are obscured or distributed in a dense way while also maintaining reliable performance in low-light and complex background scenarios.

## V. CONCLUSION

In this paper, we proposed FESL-YOLO, a lightweight small object detection framework for UAV aerial imagery. Comparative experiments show that the proposed method achieves the highest overall detection performance, with a mAP@50 of 44.6 and a mAP@50:95 of 27.1, clearly outperforming recent YOLO-based detectors. In particular, the improved precision of 54.9 and recall of 42.8 indicate a substantial reduction in both false and missed detections, which are critical issues in small object recognition.

Despite the accuracy improvement, FESL-YOLO maintains a compact model size with 9.66 million parameters and achieves real-time inference at 96.9 frames per second. Although the computational cost is higher than some lightweight baselines, the accuracy gain demonstrates an effective trade-off for practical UAV deployment.

Future work will focus on reducing computational complexity through network optimization and lightweight design strategies, while extending the proposed approach to broader aerial surveillance scenarios.

## REFERENCES


[1] X. Zhang, T. Zhang, G. Wang, P. Zhu, X. Tang, X. Jia, and L. Jiao, "Remote sensing object detection meets deep learning: a meta-review of challenges and advances," *arXiv preprint arXiv:2309.06751*, 2023.

[2] F. Heintz, P. Rudol, and P. Doherty, "From images to traffic behavior—a UAV tracking and monitoring application," *Proc. 10th Int. Conf. Information Fusion*, Quebec, Canada, pp. 1–8, 2007.

[3] G. Tang, J. Ni, Y. Zhao, Y. Gu, and W. Cao, "A survey of object detection for UAVs based on deep learning," *Remote Sensing*, vol. 16, no. 1, article 149, 2024.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, pp. 1137–1149, 2017.

[5] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, pp. 6154–6162, 2018.

[6] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, pp. 2961–2969, 2017.

[7] W. Liu *et al.*, "SSD: Single shot multibox detector," *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The Netherlands, pp. 21–37, 2016.

[8] T.-Y. Lin *et al.*, "Focal loss for dense object detection," *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Honolulu, HI, USA, pp. 2980–2988, 2017.

[9] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A review of YOLO algorithm developments," *Procedia Comput. Sci.*, vol. 199, pp. 1066–1073, 2022.

[10] C. Yang, Z. Huang, and N. Wang, "QueryDet: Cascaded sparse query for accelerating high-resolution small object detection," *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, pp. 13658–13667, 2022.

[11] O. C. Koyun, R. K. Keser, İ. B. Akkaya, and B. U. Töreyin, "Focus-and-Detect: A small object detection framework for aerial images," *Signal Process. Image Commun.*, vol. 104, p. 116675, 2022.

[12] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO series in 2021," *arXiv preprint arXiv:2107.08430*, 2021.

[13] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, pp. 3–19, 2018.

[14] C. Li and J. Wang, "Remote sensing image location based on elevated YOLOv7 target detection," *Pattern Anal. Appl.*, vol. 27, p. 50, 2024.

[15] S. Yuan et al., "IRSDD-YOLOv5: Focusing on the infrared detection of small drones," *Drones*, vol. 7, p. 393, 2023.

[16] H. Meng, S. Si, B. Mao, J. Zhao, and L. Wu, "LAGSwin: Local attention guided Swin-transformer for thermal infrared sports object detection," *PLoS ONE*, vol. 19, e0297068, 2024.

[17] S. Sun et al., "Multi-YOLOv8: An infrared moving small object detection model based on YOLOv8 for air vehicle," *Neurocomputing*, vol. 588, p. 127685, 2024.

[18] R. Khanam and M. Hussain, "YOLOv11: An overview of the key architectural enhancements," *arXiv preprint arXiv:2410.17725*, 2024.

[19] D. Du et al., "VisDrone-DET2019: The vision meets drone object detection in image challenge results," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*, Seoul, South Korea, Oct. 2019.

[20] J. Redmon and A. Farhadi, "YOLOv3: An incremental advancement," *arXiv preprint arXiv:1804.02767*, 2018.

[21] R. Khanam and M. Hussain, "What is YOLOv5: A deep look into the internal features of the popular object detector," *arXiv preprint arXiv:2407.20892*, 2024.

[22] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, Canada, pp. 7464–7475, 2023.

[23] R. Varghese and M. Sambath, "YOLOv8: A novel object detection algorithm with strengthened performance and robustness," *Proc. Int. Conf. Advances in Data Engineering and Intelligent Computing Systems (ADICS)*, Chennai, India, pp. 1–6, 2024.