FBRT-YOLO: Faster and Better for Real-Time Aerial Image Detection

Yao Xiao, Tingfa Xu*, Yu Xin, Jianan Li*

Beijing Institute of Technology
{3220220586, ciom_xtf1,3220220587,lijianan}@bit.edu.cn

Abstract

Embedded flight devices with visual capabilities have become essential for a wide range of applications. In aerial image detection, while many existing methods have partially addressed the issue of small target detection, challenges remain in optimizing small target detection and balancing detection accuracy with efficiency. These issues are key obstacles to the advancement of real-time aerial image detection. In this paper, we propose a new family of real-time detectors for aerial image detection, named FBRT-YOLO, to address the imbalance between detection accuracy and efficiency. Our method comprises two lightweight modules: Feature Complementary Mapping Module (FCM) and Multi-Kernel Perception Unit (MKP), designed to enhance object perception for small targets in aerial images. FCM focuses on alleviating the problem of information imbalance caused by the loss of small target information in deep networks. It aims to integrate spatial positional information of targets more deeply into the network, better aligning with semantic information in the deeper layers to improve the localization of small targets. We introduce MKP, which leverages convolutions with kernels of different sizes to enhance the relationships between targets of various scales and improve the perception of targets at different scales. Extensive experimental results on three major aerial image datasets, including Visdrone, UAVDT, and AI-TOD, demonstrate that FBRT-YOLO outperforms various real-time detectors in terms of performance and speed. Code is will be avaliable at https://github.com/galaxy-oss/FCM.

Introduction

Recent advancements in deep neural networks have significantly improved object detection in low-resolution natural images (2022; 2023). However, these methods struggle with efficiency and accuracy on high-resolution aerial images, especially in resource-constrained flight equipment. Key challenges include: i) detecting objects that are small or obscured by backgrounds in aerial images, and ii) balancing accuracy with real-time detection requirements on devices with limited computational resources.

To improve small object detection, increasing image resolution (2017; 2020) is common but adds computational burden, hampering real-time performance. A key challenge
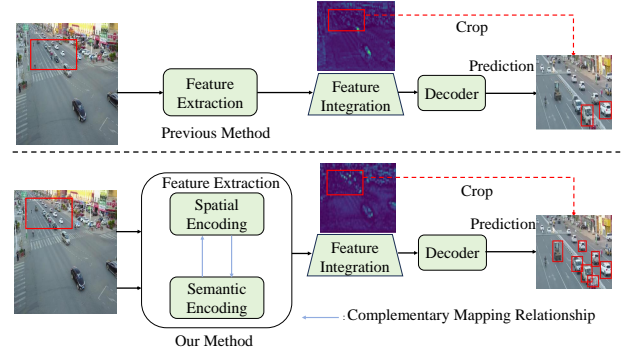
---

*Corresponding authors.

Figure 1: The previous method overlooked the embedding of spatial information in deeper layers of the backbone network during feature extraction, leading to spatial semantic inconsistencies. Our method aims to transfer shallow spatial location information into deeper layers of the network during the feature extraction process, thereby enhancing the expression of semantic information.

is the mismatch between low-resolution semantic information from deep networks and high-resolution spatial information from shallow networks. Feature pyramids (2017) address this by integrating deep and shallow features, enhancing small object localization and multi-scale feature expression while improving computational efficiency. However, as shown in Fig. 1(a), backbone networks still struggle with integrating and preserving shallow information, leading to feature mismatch issues.

To address the challenges associated with object detection in aerial images, we aim to achieve a more effective network design that meets the requirements for both accuracy and efficiency in real-time aerial image analysis. In this paper, we propose a novel network that includes two lightweight modules: the Feature Complementary Mapping Module (FCM) and the Multi-Kernel Perception Unit (MKP). Firstly, to alleviate information imbalance within the backbone network and promote better integration of semantic and spatial location information, we introduce the Feature Complementary Mapping Module (FCM). FCM implicitly encodes the target's spatial location information into high-dimensional vectors, guiding the complementary learning of spatial and
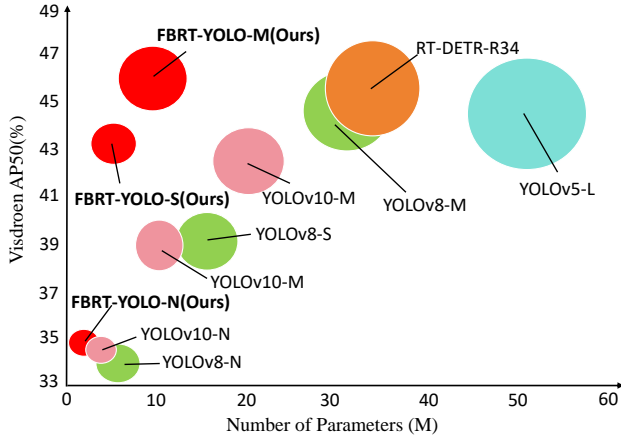
Figure 2: Our FBRT-YOLO is compared with other real-time detectors in terms of accuracy and efficiency on Vis-Drone dataset. The radius of the circle represents GFLOPs.

channel information across different stages of the backbone network. This facilitates the fusion of shallow spatial location information with deep semantic information, enhancing the consistency of spatial and semantic representations. This approach helps transfer shallow spatial location information to deeper layers of the network, improving feature alignment and enhancing the localization of small objects, as shown in Fig. 1(b).

Secondly, due to the minimal representation of small objects in aerial images, which often comprise just a few pixels, these objects are susceptible to feature disappearance during convolutional neural network (CNN) feature extraction. To fully utilize the limited feature information and enhance the network's perception of targets at different scales, we investigate the network's receptive field and propose a Multi-Kernel Perception Unit (MKP). MKP consists of convolutional kernels of different sizes and incorporates spatial point convolutions between these sizes to focus on details at various scales and highlight multi-scale feature representation. We replace the final downsampling layer of the network with MKP. This approach enables multi-scale perception of targets, improving the network's ability to capture features across different scales while further simplifying the network structure.

To meet the requirements of real-time detection in aerial images, we propose FBRT-YOLO, which boasts fewer training parameters and reduced computational load compared to the baseline YOLOv8 model (2023). Extensive experiments conducted on widely-used aerial image benchmarks such as VisDrone (2018), UAVDT (2018), and AI-TOD (2021) demonstrate that our FBRT-YOLO significantly outperforms previous state-of-the-art YOLO series models in terms of the trade-off between computation and accuracy across various model scales. The results are displayed in Fig. 2. Our contributions can be summarized as follows:

- We introduce a new family of real-time detectors for aerial image detection across different model scales, named FBRT-YOLO, achieving a highly balanced trade-off between accuracy and efficiency.

- We propose a Feature Complementary Mapping Module (FCM) that enhances feature matching for small targets in deep networks by integrating rich semantic information with precise spatial positional information.

- We introduce Multi-Kernel Perception Unit (MKP) to replace the final downsampling operation, enhancing multi-scale target perception, and simplifying the network for high efficiency.

Related Work

Real-time Object Detectors. Real-time object detectors are crucial for resource-constrained platforms, emphasizing model size, memory, and computational efficiency. Currently, YOLO (2016) and FCOS (2020) are mainstream frameworks for state-of-the-art real-time object detection. While existing real-time detectors have shown significant performance improvements on public benchmarks such as COCO (2014; 2024) for low-resolution natural images, their performance on high-resolution aerial images remains unsatisfactory. We introduce FBRT-YOLO, a specialized real-time object detector designed to excel in high-resolution aerial settings, demonstrating superior performance compared to existing models.

Small Object Detection. Detecting small objects has long been challenging. Recent solutions include augmenting small object datasets (2019) and using high-resolution images to retain detailed features. However, these methods often result in more complex models and slower detection speeds. ClusDet (2019) employs a cluster-based object scale estimation network to effectively detect small objects. DM-Net (2020a) utilizes a density map-based cropping method to leverage spatial and contextual information among objects for improved detection performance. Despite their effectiveness in small object detection, these methods suffer from long inference times and low detection efficiency. QueryDet (2022), while leveraging high-resolution features, incorporates a novel query mechanism to accelerate inference speed for object detectors based on feature pyramids. CEASC (2023) introduces a context-enhanced sparse convolution to capture global information and enhance focal features, striking a balance between detection accuracy and efficiency. These works propose lightweight decoupled heads that to some extent accelerate networks. However, achieving real-time detection remains challenging.

Multi-scale Information Extraction and Representation. Small objects are often represented in feature maps by only a few pixels, necessitating multi-scale information to enhance the feature representation of these small objects. Many works have also been carried out from this aspect (2018; 2024). Feature Pyramid Network (FPN) integrates the deep features with the richest semantics information and the shallow features with spatial location information, which alleviates the problem of feature imbalance to a certain extent. PANet (2018) adds a bottom-up path on the basis of FPN, which promotes the propagation of bottom-layer information and enhances information exchange. IPG-Net (2020) in-
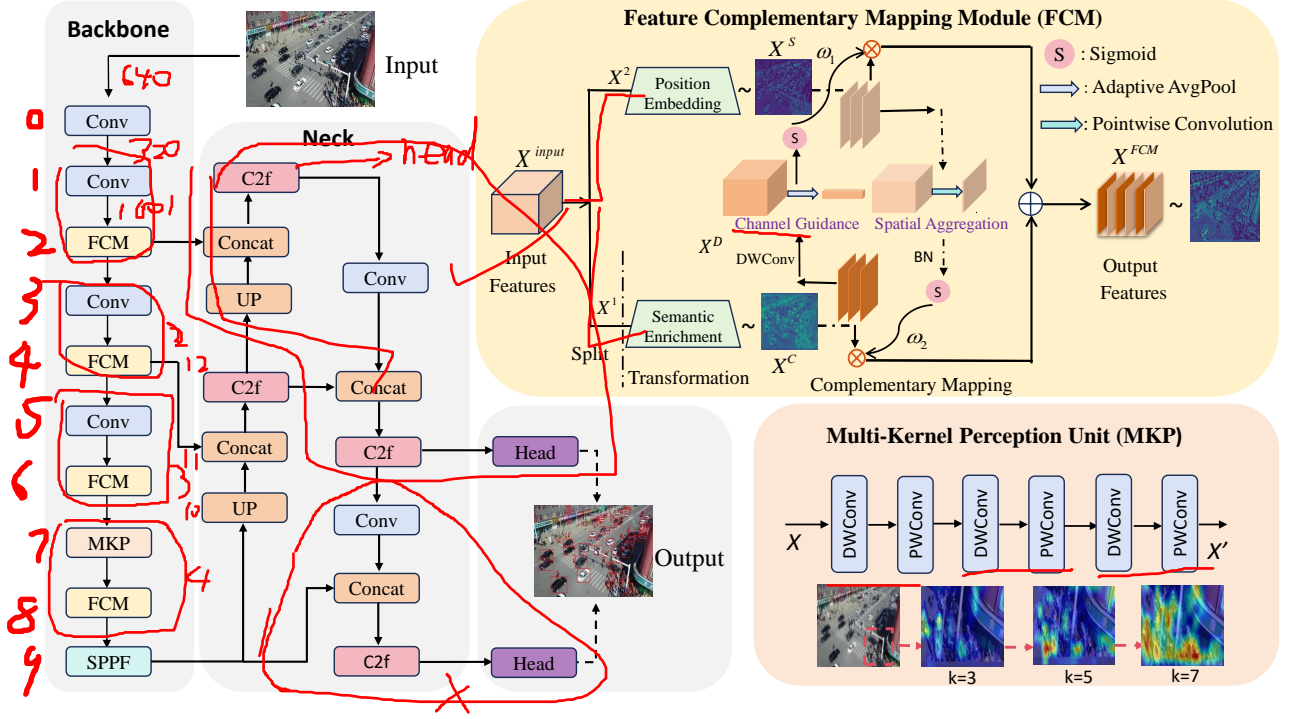
Figure 3: Framework of FBRT-YOLO. FCM module is embedded into each stage of the backbone network to integrate spatial positional information into deeper semantic information. In the final (fourth) stage of the backbone network, MKP units are introduced along with multi-scale convolutions to enhance perception of targets at various scales. It's worth noting that MKP replaces the final downsampling layer while also reducing the corresponding detection heads.

troduces image pyramid into the backbone network to solve the problem of information imbalance. The whole process consumes a lot of computing resources, which is not conducive to real-time detection. In our work, we focus on integrating deep semantic information with shallow spatial positional information in the backbone network. This integration alleviates the imbalance in information extraction during feature extraction, thereby enhancing the representation of small objects. We employ multi-scale convolutional kernels to strengthen the feature representation of targets across various scales.

Method

We present the entire structure of FBRT-YOLO in Fig. 3. This includes two core lightweight modules: the Feature Complementary Mapping Module and the Multi-Kernel Perception Unit. FCM aims to integrate more spatial positional information into rich semantic features, enhancing the representation of small objects. MKP utilizes diverse convolutional kernels to capture target information across multiple scales. Additionally, for aerial image detection, we streamline the baseline network by removing non-critical or redundant computations, further refining the network.

Feature Complementary Mapping Module

Insufficient integration of spatial positional and semantic information can lead to mismatches and misalignments in target information. To address this limitation, we propose the Feature Complementary Mapping Module. This module implicitly encodes more low-level spatial information into high-dimensional vectors, transmitting it to deeper layers of the network. This enables the detector to capture stronger structural information, thereby enhancing the expression of semantic information. The detailed structure of FCM is shown in Fig. 3, which utilizes a split, transformation, complementary mapping strategy and feature aggregation. The following is a detailed introduction to this module.

Channel Split. We first split the channels of the input feature($X^{\text{input}} \in \mathbb{R}^{C \times H \times W}$) into two parts with αC channels and $(1-\alpha)C$ channels, where $0 \leq \alpha \leq 1$ is the split ratio. The value of α is quite important in the network. As the network deepens, the branch with lower-level spatial information becomes more prominent, with increasing amounts of low-level spatial information being implicitly encoded into high-dimensional vectors. Enhancing the acquisition of low-level information at appropriate times can improve performance. The split stage can be formulated as:

$$(X^1, X^2) = \text{Split}(X^{\text{input}}), \quad (1)$$

where $X^1 \in \mathbb{R}^{\alpha C \times H \times W}, X^2 \in \mathbb{R}^{(1-\alpha)C \times H \times W}$.

Orientation Transformation. To separately obtain spatial mappings of semantic and positional information, we send the obtained X^1 to the branch composed of standard 3×3 convolution, more rich feature information is extracted on each channel, it is represented as X^C in Fig. 3. X^2 is sent to the branch composed of point-wise convolution, the point-wise convolution extracts relatively weak information, preserving a large amount of shallow spatial position information, it is represented as X^S. This transformation process is represented by the formula:

$$(X^C, X^S) = \phi_1(X^1, X^2), \tag{2}$$

where ϕ_1 represents learning a mapping relationship between spatial and semantic information, $X^C \in \mathbb{R}^{C \times H \times W}$ contains rich channel information, $X^S \in \mathbb{R}^{C \times H \times W}$ retains more original spatial location information.

Complementary Mapping. Currently, the features we obtain, X^C and X^S, while effective, are discrete. This can lead to imprecise matching of target features. Therefore, we perform complementary mapping between them to compensate for their respective missing feature mappings, achieving efficient feature matching. We take X^C, which has richer channel information, into channel interaction. It can assign unique weights to the important information on each channel. This is then mapped to X^S, which has low-level spatial location information features, for complementary feature fusion. This allows the information after interaction to obtain higher-level features. Similarly, X^S with richer low-level spatial position information, through spatial interaction, assigns unique weights to the important information on each position, and maps it to X^C with rich channel information features, to achieve complementary integration and obtain higher-level features. This process achieves the guidance of stronger features to guide weaker ones, thereby alleviating the problem of information imbalance.

Channel Interaction: First, we use a the depthwise convolution to perform convolution operations on each channel, cutting off the information between the channels, which is calculated as:

$$X_i^D = \phi_2(k_i, X_i^C), \tag{3}$$

where ϕ_2 represents the mapping of each feature layer channel, k_i is the i-th convolution kernel, X_i^C is the i-th input channel, X_i^D is the corresponding single output channel. The output result after depthwise convolution is $X^D \in \mathbb{R}^{C \times H \times W}$.

then perform global average pooling to obtain the global information on each channel, and finally obtain the key information weights through the sigmoid layer. The unique weight $\omega_1 \in \mathbb{R}^{C \times 1 \times 1}$ generated on the channel can be represented as:

$$\omega_1 = \mathcal{R} \left(\frac{1}{H \times W} \sum_{i=0}^{H} \sum_{j=0}^{W} X^D(i, j) \right), \tag{4}$$

where $\mathcal{R}$ represents an activation function.

Spatial interaction: To further aggregate spatial information, we adopt a simple design, as shown in Fig. 3, which

consists of a 1×1 spatial convolution layer, BN (2015) and sigmoid. Finally, we generate a spatial attention map, which is similar to channel interaction, and map it to the branch that go through the 3×3 standard convolution, making it more focused on spatial information. The spatial information weight ω_2 generated can be calculated as:

$$\omega_2 = \mathcal{R}(\mathcal{F}(X^S)), \tag{5}$$

where $\mathcal{F}$ represents convolution mapping with spatial aggregation, $\omega_2 \in \mathbb{R}^{1 \times H \times W}$.

Feature Aggregation. After obtaining the channel information weight ω_1 and spatial information weight ω_2, they are respectively mapped to the features containing X^S and X^C. Then the two branches are connected together to obtain the feature X^{FCM}, which contains features with dual mappings of spatial and semantic relationships. X^{FCM} is calculated as:

$$X^{FCM} = (X^C \otimes \omega_2) \oplus (X^S \otimes \omega_1), \tag{6}$$

where $\otimes$ is element-wise multiplication.

Overall, the FCM module adopts an information complementary fusion method with relatively low computational resources. It propagates shallow-level spatial positional information into deeper layers of the network, alleviating the loss of object spatial location information in the backbone network downsampling process.

Multi-Kernel Perception Unit

Small targets in aerial images are often obscured by background noise, resulting in limited effective information. To fully leverage the available feature information, we employ a multi-kernel perception unit to detect targets at different scales and establish spatial relationships across these scales, thereby enhancing the feature representation of contextual and small target information. As illustrated in Fig. 3, the Multi-Kernel Perception Unit (MKP) concatenates convolutional kernels of various sizes sequentially, incorporating point-wise convolutions between kernels of different scales. The entire process can be mathematically represented as follows:

$$X' = \mathcal{T}_{2k+1}(\mathcal{A}(\cdots \mathcal{A}(\mathcal{T}_k(X))\cdots)), \tag{7}$$

where X represents local features of the input, while X' represents globally mapped features across multiple scales. $\mathcal{T}_k$ represents depthwise convolution with kernel size k. In our experiment, we set $k = 3$. $\mathcal{A}$ represents point-wise convolution transformation.

Targeted Reduction of Redundancy-Driven Network Design

Currently, real-time detection models are primarily designed for traditional low-resolution image detection, but this does not apply well to high-resolution aerial image detection, resulting in significant structural redundancy. For spatial downsampling in feature extraction, channel expansion precedes depthwise convolution sampling (2024). Post depthwise convolution, there is interference between channels, leading to a loss of spatial information, which is disadvantageous for detecting aerial images in complex environments.

Model	AP	AP$_{50}$	FLOPs	Params	FPS
YOLOv5-L (2020)	27.3	44.5	109.1 G	46.5 M	65
YOLOv8-N (2023)	19.6	33.3	8.7 G	3.2 M	174
YOLOv8-S (2023)	23.6	39.6	28.6 G	11.2 M	134
YOLOv8-M (2023)	27.1	44.4	78.9 G	25.9 M	83
YOLOv8-L (2023)	28.4	45.9	165.2 G	43.7 M	61
YOLOv8-X (2023)	28.9	46.8	257.8 G	68.2 M	47
YOLOv9-M (2024)	27.2	44.6	76.3 G	20.0 M	77
YOLOv10-S (2024)	23.8	39.3	21.6 G	7.2 M	135
YOLOv10-L (2024)	27.6	44.6	120.3 G	24.4 M	64
YOLOv10-X (2024)	28.7	46.1	160.4 G	29.5 M	49
RT-DETR-R34 (2024)	27.2	46.0	92.0 G	31.0 M	88
RT-DETR-R50 (2024)	28.8	48.3	136.0 G	42.0 M	65
FBRT-YOLO-N (Ours)	20.2	34.4	6.7 G	0.9 M	192
FBRT-YOLO-S (Ours)	25.9	42.4	22.9 G	2.9 M	143
FBRT-YOLO-M (Ours)	28.4	45.9	58.7 G	7.2 M	94
FBRT-YOLO-L (Ours)	29.7	47.7	119.2 G	14.6 M	70
FBRT-YOLO-X (Ours)	30.1	48.4	185.8 G	22.8 M	52

Table 1: Comparison of AP(%) and Params/FPS on VisDrone by using our methods with different real-time object detectors.

Model	AP	AP$_{50}$	AP$_{75}$
YOLOv3-SPP3 (2019)	26.4	-	-
YOLOv3-SPP3 (2019)	26.4	-	-
DMNet (2020a)	29.4	49.3	30.6
QueryDet (2022)	28.3	48.1	28.8
CEASC (2023)	28.7	50.7	28.4
YOLC (2024)	28.9	51.4	28.3
FBRT-YOLO (Ours)	**30.1**	48.4	**31.7**

Table 2: Comparison of AP(%) with state-of-the-art detectors on VisDrone.

However, we decouple this process by first applying group convolutions for spatial downsampling and then using point convolutions for channel expansion. The parameter calculations for both approaches are as follows:

$$P^{'} = 3 \times 3 \times C_1 \times C_2, \tag{8}$$

$$P = 3 \times 3 \times C_1 \times \frac{C_1}{g} + 1 \times 1 \times C_1 \times C_2, \tag{9}$$

where $P^{'}$ represents the parameter count for standard convolution, and P represents the parameter count for our method. C_1 and C_2 denote the input and output channel numbers, respectively. During network downsampling, the channel expansion typically results in $C_2 = 2C_1$. g represents the number of groups.

Experiments

Implementation Details

We conduct extensive experiments on three object detection benchmarks based on aerial images, $i.e.$ Visdrone, UAVDT, and AI-TOD. All experiments are conducted on an NVIDIA GeForce RTX 4090 GPU, except that the inference speed is test on a single RTX 3080 GPU. Our network is trained for 300 epochs using the stochastic gradient descent (SGD) optimizer with a momentum of 0.937, a weight decay of 0.0005, a batch size of 4, and an initial learning rate of 0.01.

Results on Visdrone Dataset

State-of-the-art Comparison. As shown in Table 1, we compare FBRT-YOLO with existing real-time detectors. Our FBRT-YOLO achieves superior performance and faster detection efficiency across various model scales. For resource-limited aerial operation equipment, we demonstrate results of FBRT-YOLO models at various scales compared to other real-time state-of-the-art object detectors. For small models, FBRT-YOLO-N/S reduces parameter count by 72% and 74% respectively compared to YOLOv8-N/S, while achieving an improved detection accuracy of 0.6% and 2.3% in average precision (AP). For medium models, FBRT-YOLO-M reduce GFLOPs by 26% and 23% compared to YOLOv8-M and YOLOv9-M, respectively, while achieving improvements in AP of 1.3% and 1.2%, respectively. For large models, compared to YOLOv8-X and YOLOv10-X, our FBRT-YOLO-X shows 66% and 23% fewer parameters, respectively, and achieves a significant improvement in AP of 1.2% and 1.4%. Moreover, compared to RT-DETR-R34/R50, FBRT-YOLO-M/L achieves fewer parameters, lower GFLOPs, higher detection speed, and better detection performance. These experimental results demonstrate the superiority of our FBRT-YOLO as a real-time aerial image detector.

As shown in Table 2, it shows the comparison results of our method with other state-of-the-art methods on VisDrone. which indicates that our FBRT-YOLO can effectively detects aerial images.

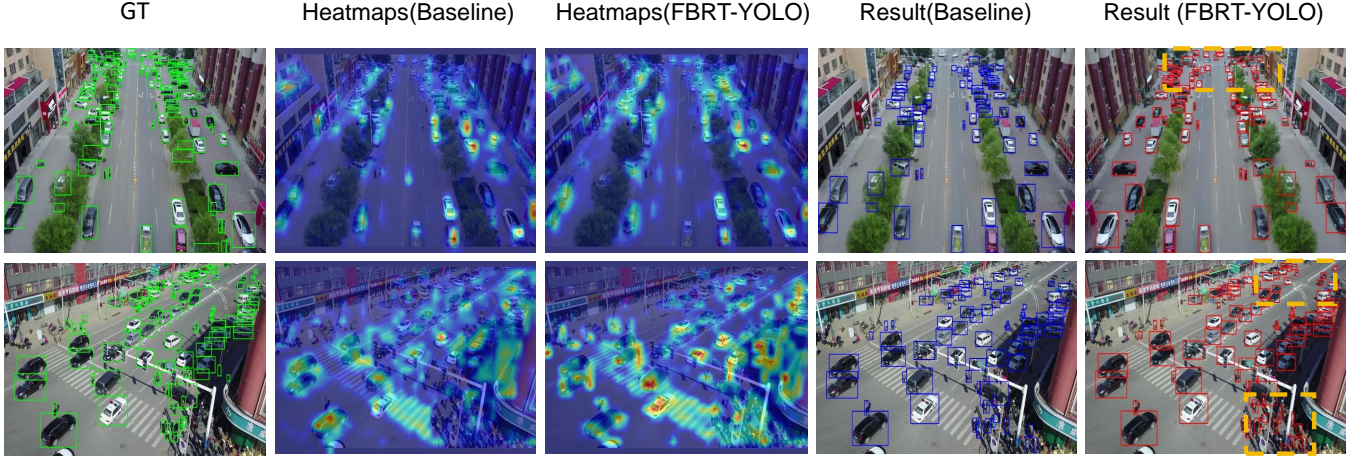| GT | Heatmaps(Baseline) | Heatmaps(FBRT-YOLO) | Result(Baseline) | Result (FBRT-YOLO) |

Figure 4: Visualization of the detection results and heatmaps on VisDrone. The highlighted areas represent the regions that the network is focusing on.

Model	AP	AP_{50}	AP_{75}
ClusDet (2019)	13.7	26.5	12.5
GLSAN (2020)	17.0	28.1	18.8
DREN (2019)	15.1	-	-
GFL (2020b)	16.9	29.5	17.9
CEASC (2023)	17.1	30.9	17.8
FBRT-YOLO (Ours)	**18.4**	**31.1**	**18.9**

Table 3: Comparison of AP(%) with state-of-the-art detectors on UAVDT.

Model	AP	AP_{50}	Params	FLOPs	FPS
YOLOv8-S	19.1	43.6	11.2 M	28.6 G	131
FBRT-YOLO-S	**20.2**	**45.8**	**2.9 M**	**22.9 G**	**142**

Table 4: Comparison of AP(%) and Params/FPS on AI-TOD by using our methods with baseline.

Qaulitative Resutls. To better demonstrate the superior performance of FBRT-YOLO in detecting aerial images, we visualize the heatmaps of both the baseline model and our method in Fig. 4. From the results, we observe that FBRT-YOLO enhances focus on small and densely packed targets, showcasing the method's superiority in enhancing spatial and multiscale information within the network.

Results on UAVDT Dataset

Quantitative Result. Table 3 reports our comparison results on the UAVDT dataset. Our proposed method surpasses existing methods, such as GLSAN (2020) and CEASC (2023). The results clearly show that our proposed FBRT-YOLO achieves superior performance with an AP of 18.4%, outperforming other state-of-the-art methods in aerial image detection. This demonstrates the effectiveness of our detection framework.

RR	FCM	MKP	AP	AP_{50}	Params	FLOPs
✗	✗	✗	23.6	39.6	11.2 M	28.6 G
✓	✗	✗	23.4	39.2	9.1 M	25.5 G
✓	✓	✗	24.3	40.6	7.2 M	23.2 G
✓	✓	✓	**25.9**	**42.4**	**2.9 M**	**22.9 G**

Table 5: Ablation on FCM, MKP and RR Module on VisDrone. 'RR' represents operations aimed at reducing inherent redundancy in the network. Replace the final downsampling layer with MKP and remove the corresponding detection head.

Qaulitative Resutls. A complex background can significantly limit the effective information about the target. Our method focuses on effectively propagating the spatial information of the target through network layers to enhance feature representation. Visualization of detection results, as shown in Fig. 5, proves that our method significantly improves detection performance in complex backgrounds.

Results on AI-TOD Dataset

The AI-TOD dataset contains a significant proportion of small objects. To better validate the superiority of our method in small object detection, we also evaluate FBRT-YOLO on AI-TOD. As reported in Table 4, our method reduces the parameter count by 74%, the GFLOPs by 20%, while achieving a 2.2% increase in AP_{50} and a 1.1% increase in AP compared to the baseline.

Ablation Study

To validate the effectiveness of the core module design in FBRT-YOLO, we design a series of ablation experiments on the VisDrone dataset. We use YOLOv8-S as the baseline model in all the ablation experiments.

Effect of Key Components. Experimental results in Table 5 exhibit the effectiveness of all contributions in this
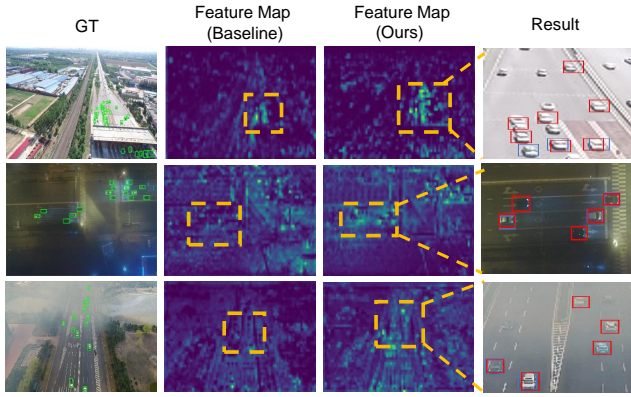
Figure 5: Visualizations of the detection results of baseline and our proposed method under low light and similar background conditions on UAVDT. The blue boxes represent the prediction results using the baseline model, while the red boxes represent the prediction results using our method.

Channel Mapping	Spatial Mapping	AP	AP$_{50}$
✗	✗	25.0	40.4
✓	✗	25.6	41.7
✗	✓	25.3	41.3
✓	✓	**25.9**	**42.4**

Table 6: The experiment validates the optimal configuration of the mapping relationship.

work. We reduce inherent redundancy in the baseline model, optimize it, and achieve a 18% reduction in parameters and a 11% decrease in computational load, albeit with a slight decrease in accuracy. Introducing the FCM module into various stages of the backbone network incorporates spatial positional information in deeper layers, resulting in a 1.4% increase in AP$_{50}$ and further reducing network computational resources. We replace the downsampling operation of the backbone network's final layer with MKP units to detect targets at multiple scales, thereby increasing AP by 1.6%. It's worth noting that our network converges faster during the training process compared to the baseline network.

Effect of Mapping Relationship. Table 6 shows the results of the proposed channel and spatial complementary mapping. In order to obtain the optimal configuration of two mapping relationships, we design a series of variant experiments. According to the experimental results, we find that models using channel or spatial mapping are superior to models without mapping relationships. Combining the two can achieve better results. Compared with the model without mapping relationships, this optimal configuration has improved AP$_{50}$ by 2.0%.

Effect of Split Ratio. Table 7 shows the impact of different parameters α on the experimental results, where α represents the split ratio of spatial feature information and channel feature information. From the experimental results, we can see that as the downsampling process progresses, the

Split Ratio (α)	AP	AP$_{50}$	Params
(0.5,0.5,0.5,0.5)	25.7	42.2	3.2 M
(0.25,0.25,0.25,0.25)	25.5	42.1	**2.6 M**
(0.75,0.75,0.75,0.75)	25.8	42.3	4.0 M
(0.75,0.75,0.25,0.25)	**25.9**	**42.4**	2.9 M

Table 7: Experimental verification of the impact of the spatial and channel feature partition ratio of α in the FCM module at each stage of the backbone network.

Kernel Size	AP	AP$_{50}$	Params	FLOPs
(3,3,3)	25.4	41.7	**2.82 M**	**22.7 G**
(5,5,5)	25.6	42.0	2.88 M	22.8 G
(7,7,7)	25.8	42.2	2.94 M	23.1 G
(3,5,7)	**25.9**	**42.4**	2.90 M	22.9 G

Table 8: Experiments with different kernel size in MKP.

proportion of the spatial feature part (undergoing point-wise convolution) increases, and the experimental effect will be better. We speculate that the reason for this phenomenon is that when α takes the values 0.75, 0.75, 0.25, 0.25, it retains more spatial location information in deeper networks, which is beneficial for the localization and matching of target features. Retaining more spatial position information in a deeper network is also consistent with the original intention of the FCM module design.

Effect of Kernel Size. Table 8 shows the experimental results of different kernel sizes in MKP. From the experimental results, it can be observed that smaller kernels provide limited receptive fields for the network, failing to establish strong contextual associations, while larger kernels introduce significant background noise, which is detrimental to detection. By using convolutional kernels of varying sizes, we capture multi-scale features of targets spanning different sizes. Additionally, we introduce point-wise convolutions between different kernel sizes to integrate spatial information across scales, achieving optimal performance.

Conclusion

In this paper, we propose a new family of real-time detectors for aerial image detection, named FBRT-YOLO. Specifically, it introduces two lightweight modules: the Feature Complementary Mapping Module (FCM), which aims to improve the fusion of rich semantic information with precise spatial location details, and the Multi-Kernel Perception Unit (MKP), which enhances multi-scale target perception and improves the network's ability to capture features across varying scales. For aerial image detection, we also reduce the inherent redundancies in conventional detectors, further accelerating the network. Extensive experimental results on the VisDrone, UAVDT, and AI-DOT datasets demonstrate that FBRT-YOLO achieves a highly balanced trade-off between accuracy and efficiency in aerial image detection.

References

Deng, S.; Li, S.; Xie, K.; Song, W.; Liao, X.; Hao, A.; and Qin, H. 2020. A global-local self-adaptive network for drone-view object detection. *IEEE Transactions on Image Processing*, 30: 1556–1569.

Du, B.; Huang, Y.; Chen, J.; and Huang, D. 2023. Adaptive sparse convolutional networks with global context enhancement for faster object detection on drone images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13435–13444.

Du, D.; Qi, Y.; Yu, H.; Yang, Y.; Duan, K.; Li, G.; Zhang, W.; Huang, Q.; and Tian, Q. 2018. The unmanned aerial vehicle benchmark: Object detection and tracking. In *Proceedings of the European conference on computer vision (ECCV)*, 370–386.

Huang, B.; Li, J.; Chen, J.; Wang, G.; Zhao, J.; and Xu, T. 2024. Anti-UAV410: A Thermal Infrared Benchmark and Customized Scheme for Tracking Drones in the Wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5): 2852–2865.

Ioffe, S.; and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, 448–456. pmlr.

Jocher, G. 2020. YOLOv5 by Ultralytics.

Jocher, G.; Chaurasia, A.; and Qiu, J. 2023. Ultralytics YOLO.

Kisantal, M.; Wojna, Z.; Murawski, J.; Naruniec, J.; and Cho, K. 2019. Augmentation for small object detection. *arXiv preprint arXiv:1902.07296*.

Li, C.; Yang, T.; Zhu, S.; Chen, C.; and Guan, S. 2020a. Density map guided object detection in aerial images. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 190–191.

Li, J.; Liang, X.; Shen, S.; Xu, T.; Feng, J.; and Yan, S. 2018. Scale-Aware Fast R-CNN for Pedestrian Detection. *IEEE Transactions on Multimedia*, 20(4): 985–996.

Li, J.; Liang, X.; Wei, Y.; Xu, T.; Feng, J.; and Yan, S. 2017. Perceptual Generative Adversarial Networks for Small Object Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Li, X.; Wang, W.; Wu, L.; Chen, S.; Hu, X.; Li, J.; Tang, J.; and Yang, J. 2020b. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *Advances in Neural Information Processing Systems*, 33: 21002–21012.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 740–755. Springer.

Liu, C.; Gao, G.; Huang, Z.; Hu, Z.; Liu, Q.; and Wang, Y. 2024. YOLC: You Only Look Clusters for Tiny Object Detection in Aerial Images. *IEEE Transactions on Intelligent Transportation Systems*.

Liu, S.; Qi, L.; Qin, H.; Shi, J.; and Jia, J. 2018. Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8759–8768.

Liu, Z.; Gao, G.; Sun, L.; and Fang, L. 2020. IPG-net: Image pyramid guidance network for small object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 1026–1027.

Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2020. FCOS: A simple and strong anchor-free object detector. *IEEE transactions on pattern analysis and machine intelligence*, 44(4): 1922–1933.

Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; and Ding, G. 2024. Yolov10: Real-time end-to-end object detection. *arXiv preprint arXiv:2405.14458*.

Wang, C.-Y.; Bochkovskiy, A.; and Liao, H.-Y. M. 2023. 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*, 7464–7475.

Wang, C.-Y.; Yeh, I.-H.; and Liao, H.-Y. M. 2024. Yolov9: Learning what you want to learn using programmable gradient information. *arXiv preprint arXiv:2402.13616*.

Wang, J.; Sun, K.; Cheng, T.; Jiang, B.; Deng, C.; Zhao, Y.; Liu, D.; Mu, Y.; Tan, M.; Wang, X.; et al. 2020. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10): 3349–3364.

Wang, J.; Yang, W.; Guo, H.; Zhang, R.; and Xia, G.-S. 2021. Tiny object detection in aerial images. In *2020 25th international conference on pattern recognition (ICPR)*, 3791–3798. IEEE.

Wang, Y.; Zhang, X.; Yang, T.; and Sun, J. 2022. Anchor detr: Query design for transformer-based detector. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, 2567–2575.

Yang, C.; Huang, Z.; and Wang, N. 2022. QueryDet: Cascaded sparse query for accelerating high-resolution small object detection. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 13668–13677.

Yang, F.; Fan, H.; Chu, P.; Blasch, E.; and Ling, H. 2019. Clustered object detection in aerial images. In *Proceedings of the IEEE/CVF international conference on computer vision*, 8311–8320.

Zhang, J.; Huang, J.; Chen, X.; and Zhang, D. 2019. How to Fully Exploit The Abilities of Aerial Image Detectors. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 1–8.

Zhang, P.; Zhong, Y.; and Li, X. 2019. SlimYOLOv3: Narrower, Faster and Better for Real-Time UAV Applications. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 37–45.

Zhao, Y.; Lv, W.; Xu, S.; Wei, J.; Wang, G.; Dang, Q.; Liu, Y.; and Chen, J. 2024. Detrs beat yolos on real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16965–16974.

Zhu, P.; Wen, L.; Bian, X.; Ling, H.; and Hu, Q. 2018. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*.