FGHDet: Delving Into Fine-grained Features With Head Selection for UAV Object Detection

Yanchao Bi, Yang Ning, Xiushan Nie
Shandong Jianzhu University
Jinan, P.R. China
2022110101@stu.sdjzu.edu.cn
ningyang20@sdjzu.edu.cn
niexsh@hotmail.com

Xiankai Lu
Shandong University
Jinan, P.R. China
carrierlxk@gmail.com

Ruiheng Zhang
Beijing Institute of Technology
Beijing, P.R. China
ruiheng.zhang@bit.edu.cn

Huanlong Zhang
Zhengzhou University of Light Industry
Zhengzhou, P.R. China
hlzhang@zzuli.edu.cn

Abstract

Detecting small objects in UAV imagery is a challenging and crucial task in computer vision. Most current methods struggle to tackle the challenges of the small object: fine-grained feature mining, multiple-layer feature fusion, and a mismatch in scale between anchors and feature maps. To alleviate the aforementioned issues, we present the FGHDet, focusing on delving into fine-grained features in low-level features with a head selection mechanism. Firstly, our approach introduces a Detail-preserving Semantic Information Enhancement Module to retain fine-grained information while excavating coarse-grained semantic details relevant to fine-grained information. Then, we devise a Coarse-to-fine Feature Guidance Module that leverages coarse-grained semantic information and fine-grained information to co-guide feature enhancement, further improving the model's classification ability. Finally, we introduce a multi-scale detection strategy based on anchor-head matching, ensuring scale-level matching between anchors and feature maps to prevent over-fitting due to overly fine anchor divisions. Extensive experiments on VisDrone, CARPK, and Drone-vs-Bird datasets demonstrate the effectiveness of FGHDet. It achieves notable mAP improvements (IoU range [0.5:0.95]) of 4.9, 4.1, and 2.2, respectively, showcasing strong competitiveness against state-of-the-art methods. The code is available at https://github.com/b-yanchao/UAVDetection.git.

***Keywords:** Drone-view image, Fine-grained information extraction, Learning fine-grained semantics, Anchor-head based scale-level matching.*

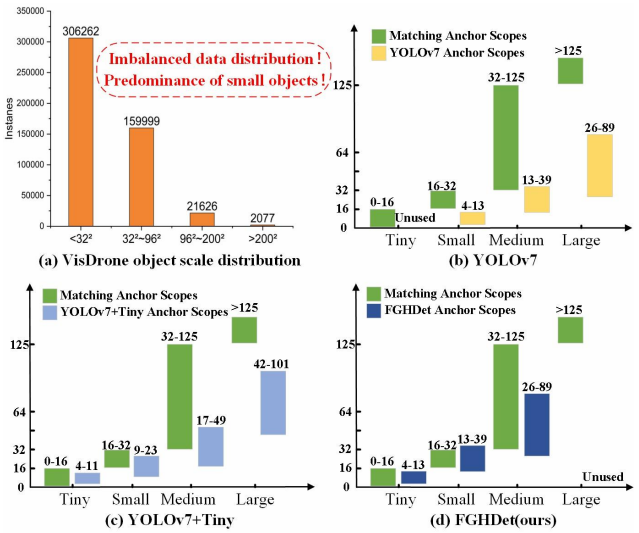

Figure 1. Our main motivation. (a) illustrates the imbalanced distribution (long-tailed) of UAV objects, predominantly consisting of small-size objects, emphasizing the importance of mining fine-grained features. (b) demonstrates the mismatch between anchors and detection heads in YOLOv7, making object regression challenging. (c) showcases the issue of over-fitting caused by excessively fine-grained anchor partitioning, due to adjacent detection heads learning highly similar features. (d) Our solutions.

1. Introduction

Unmanned aerial vehicles (UAVs) object detection plays an important role in a wide range of scenarios, such as infrared wildlife detection [9], autonomous driving, and intelligent surveillance [19]. The pervasive small size of UAV objects makes it difficult to acquire sufficiently effective

features, resulting in a large performance gap between small and normal-scale objects [41] (Fig. 1 (a, b)).

Recent advancements in boosting the performance of small object detection can be summarized as *context-based* [16], *generative-based* [23], and *multi-scale-based* [53] categories.

Firstly, the *context-based* methods exploit enhancing the exploration of interactions or context among small objects, effectively complementing the restricted feature information provided by these small objects [16]. The *generative-based* strategies involve the generation of high-resolution images from low-resolution counterparts with GAN [23]. While these methods enhance the efficacy of small object detection, they come with a considerable associated cost. Not all objects inherently possess easily learnable contextual information or recoverable fine-grained information. For *multi-scale-based* ones, an advanced method known as the Feature Pyramid Network (FPN) [24] has been proposed for constructing a feature pyramid by integrating multi-scale feature maps from diverse convolutional layers. This innovative approach adeptly exploits unique receptive field features across various layers, resulting in a notable improvement in small object detection performance. This study focuses on the third strategy.

As depicted in Fig. 1 (c), applying the multi-scale strategy on UAV images directly presents two challenges: Firstly, in the feature extraction stage, extensive downsampling operations are carried out to capture semantic information with large receptive fields [8]. Due to the nature of small and densely distributed objects in UAV imagery, this strategy frequently results in the loss of a significant amount of fine-grained details. Secondly, the performance of existing detection models heavily relies on anchor design. But this manually designed anchor with a fixed value can not adapt to both small and regular-sized object detection well [38]. To address this issue, a common approach involves introducing tiny detection heads. This method is subject to too fine a division of the anchor, which can lead to multiple detection heads learning redundant features, resulting in over-fitting [24]. Therefore, enhancing small object detection by utilizing fine-grained features and mitigating the effects of mismatches between anchors and feature maps remains a significant challenge.

In this study, we present a novel plug-and-play method, FGHDet, as a straightforward yet effective solution to address the previously mentioned challenges under UAV scenarios. We explore a novel paradigm to comprehensively boost UAV object detection from **learning semantic information related to fine-grained information** [47] and **anchor-head alignment** views. Specifically, we devise the Detail-preserving Semantic Information Enhancement Module (DSIEM) that utilizes modified dilated convolutions to obtain high-resolution feature maps with substantial semantic information and a large receptive field. Compared with the ASPP module [2], the DSIEM module mitigates the fusion conflict caused by the poor correlation between different receptive fields of semantic information and fine-grained information by learning semantic information related to fine-grained information on low-level feature map. Additionally, the DSIEM module effectively solves the problem of loss of fine-grained information that may occur in the process of small-object feature extraction. We have considered incorporating rich semantic information to enhance classification capabilities [27]. To this end, we introduce the Coarse-to-fine Feature Guidance Module (CFGM). This module leverages both enhanced fine-grained and acquired coarse-grained features to co-guide the mid-level features to be enhanced from coarse to fine. This approach aims to bolster the model's detection performance across various object scales, especially medium-scale objects.

In *multi-scale-based* detection methods, objects at different scales require distinct-scale prediction heads. Ensuring the proper alignment between anchors and their corresponding detection heads becomes essential. To facilitate the accurate learning of features for objects at various scales, it is necessary to appropriately match the scope of anchors with their corresponding detection heads. Building on this observation, we propose a multi-scale detection method grounded in anchor-head matching (Fig. 1 (d)). This approach involves selecting the optimal detection heads based on the anchor scopes determined through the $K\text{-}means$ clustering algorithm [41]. This ensures that the anchor operates effectively on the feature map at an appropriate scale, mitigating potential over-fitting issues due to excessively fine anchor partitioning (Fig. 1 (c)). In this way, our FGHDet achieves scale-level alignment of anchors with the feature maps, avoiding the over-fitting problem that occurs when the anchors are divided too finely and improving the accuracy of multi-scale detection.

We evaluated our FGHDet on the VisDrone [6], CARPK [12], and Drone-vs-Bird [3] datasets, achieving mAP (IoU range [0.5:0.95]) improvements of 4.9%, 4.1%, and 2.2%. In summary, we make three main contributions:

- We introduce a Detail-preserving Semantic Information Enhancement Module (DSIEM) designed to learn semantic information associated with fine-grained information. This module enhances the expressive ability of low-level feature map for small objects and consequently improves their recall.
- We have designed a Coarse-to-fine Feature Guidance Module (CFGM) that enhances the semantic information within the mid-level feature map through the co-guidance of both coarse-grained and fine-grained features. This module further improves the model's clas-

sification performance and enhance its robustness to objects at various scales.

- We propose an anchor-head matching mechanism that selectively adds or removes corresponding heads based on anchors obtained through clustering. To the best of our knowledge, this marks the first instance in which such a mechanism has been proposed. This mechanism enables scale-level matching between anchors and feature maps, mitigating the over-fitting problem.

2. Related Work

2.1. Unmanned Aerial Vehicle Object Detection

Detecting objects in unmanned aerial vehicle (UAV) imagery poses a highly challenging and crucial task in the application of computer vision [51]. Existing models encounter challenges in extracting adequate effective features due to the limited and densely distributed features, leading to sub-optimal performance [35]. In recent years, substantial advancements have been achieved in employing deep learning approaches for small object detection, addressing the associated challenges. Current strategies utilized for enhancing small object detection in UAV scenarios encompass data augmentation techniques [4, 41, 34], which involve operations such as direct duplication, pasting, or scaling to augment the number of small object samples. Besides, multi-scale feature fusion methods [10] effectively integrate deep and shallow features to improve detection capabilities. Contextual information learning strategies [5] efficiently leverage relationships among the environment, objects, or among objects to facilitate object and scene recognition. Super-resolution techniques [23] directly generate high-resolution images with detailed information. Furthermore, alternative methods [30, 43] encompass refinements in loss functions and attention mechanisms. These methods employ various strategies to effectively utilize the fine-grained information and semantic information of small objects, enhancing UAV object detection performance.

3. Multi-scale Object Detection Strategy

As an important solution to handle object scale variation, a multi-scale object detection strategy considering high-level features generally encompasses more semantic information and larger receptive fields, well-suited for detecting large objects. In contrast, low-level features contain finer details and are more conducive to detecting small objects.

Traditional computer vision approaches often analyze and process images by extracting features at a single scale. However, this approach tends to exhibit limitations in detecting objects of different sizes or scenes with diverse proportions. To address this, researchers have devised feature pyramids that encompass various scales [24, 7]. By integrating feature maps from different levels, these pyramids leverage the semantic information inherent in each scale, overcoming the constraints associated with using a single-scale feature. This enhancement significantly improves the detection performance, particularly for small objects. However, there is a lack of fusion between high-level and low-level features in this procedure. To address this, PAFPN [28] extends FPN by incorporating a bottom-up pathway, enabling the retrieval of fine details in low-level features by high-level features. Considering that different input features typically contribute unequally to output features at different resolutions, BiFPN [37] introduces a weighted bidirectional feature pyramid network, achieving a simple and efficient multi-scale feature fusion. RaFPN [53] calculates the similarity between pixels situated on cross-scale features to establish the relationships among objects, preventing the dilution of relational features contained in non-adjacent layers.

Despite enhancing low-level features through feature fusion, the limitations stemming from inconsistent gradient computations across different layers hinder the full exploitation of shallow layers in FPN for detecting tiny objects. To address this, SSPNet [11] leverages relationships between adjacent layers to facilitate appropriate feature sharing between deep and shallow layers. This approach mitigates the inconsistency in gradient calculations between different layers and has demonstrated significant improvements in small object detection. In contrast to the aforementioned approach, our method processes low-level features, learning semantic features relevant to fine-grained characteristics. By enhancing the correlation between fine-grained information and semantic information with large receptive fields, our approach strengthens the expressive capability of low-level features for small object representation.

4. Method

Figure 2 shows the overview of the proposed method. The image is first fed into the backbone to generate a series of feature maps of different resolutions. Then, we utilize the DSIEM to learn semantic information relevant to fine-grained details, thereby enhancing the expressive capability of low-level feature map for small objects. After that, the CFGM enhances mid-level features progressively from coarse to fine under the co-guidance of coarse-grained features and fine-grained features, aiming to robustly detect objects at different scales. Finally, by retaining matched detection heads to achieve scale-matching between anchors and feature maps, we can avoid over-fitting and improve the detection performance of UAV objects.

The details of the aforementioned components are elaborated in sections §4.1, §4.2, and §4.3.

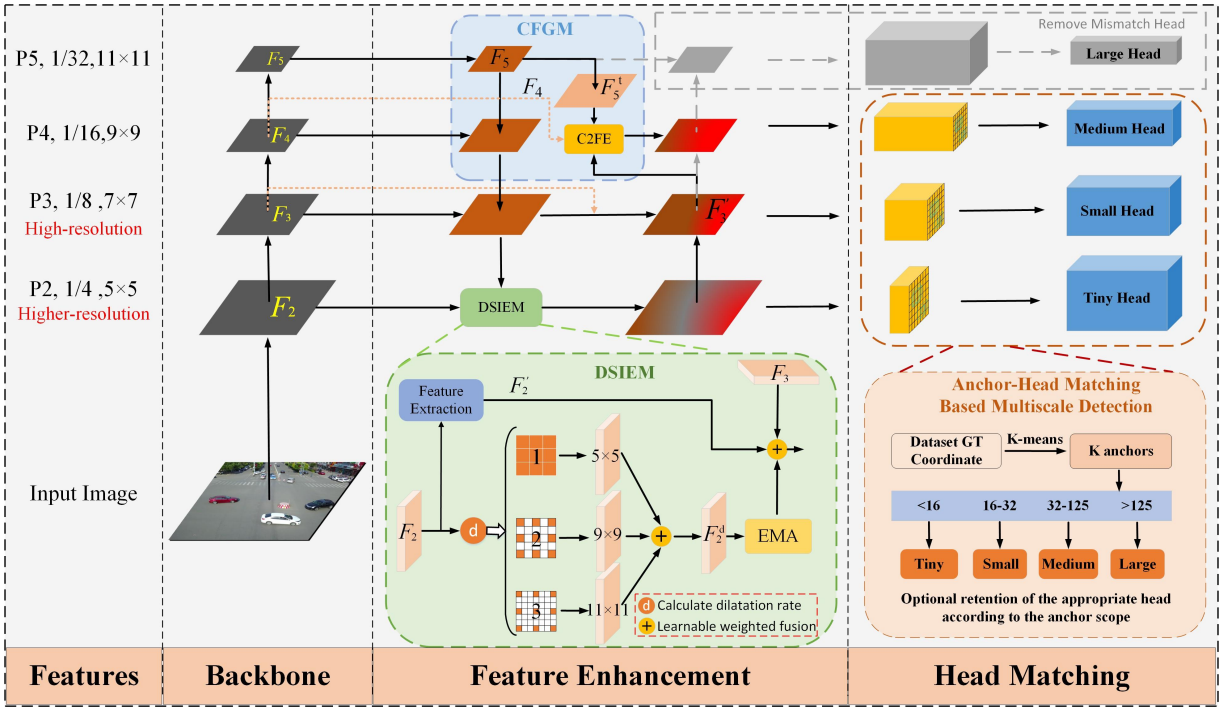

Figure 2. The whole architecture of the proposed FGHDet. Deliver the image to the backbone to produce a series of feature maps of different resolutions. In the DSIEM module, learning semantic information relevant to fine-grained information from F_2 enhances the expression capability of low-level features for small objects. After that, the CFGM module uses P_5 and P_3 to co-guide the mid-level feature map, enhancing semantic features from coarse to fine to robustly detect objects at different scales. Finally, retaining matched detection heads to achieve scale-matching between anchors and feature maps improves the accuracy of UAV object detection.

4.1. Detail-preserving Semantic Information Enhancement Module

Existing multi-scale methods employ extensive down-sampling operations during feature extraction to acquire a larger receptive field of semantic information [24]. However, this operation tends to lose a substantial amount of fine-grained information and even reduce the object to a singular pixel which harms small object detection. To alleviate this issue, we propose a Detail-preserving Semantic Information Enhancement Module (DSIEM). The whole architecture of the DSIEM module is illustrated in Figure 2.

Specifically, the DSIEM initially calculates dilation rates for dilated convolutions by considering the receptive fields of high-level features. The input image is subjected to down-sampling through successive convolutions with a 3 $\times$ 3 kernel and a stride of 2, progressively halving the feature map sizes. This process yields feature maps at levels designated as F_1, F_2, F_3, F_4, and F_5. Each of these levels corresponds to receptive field sizes of 3 $\times$ 3, 5 $\times$ 5, 7 $\times$ 7, 9 $\times$ 9, and 11 $\times$ 11, respectively. Typically, the F_1 feature map is not employed (see Fig. 2).

To learn the same receptive field semantic information on the F_2 feature map as the F_4 and F_5 feature maps, employs the dilation convolution of 3×3 kernel size where the required dilation rates d are 2 and 3, respectively. Note that the kernel size of 3 $\times$ 3 has been experimentally verified to offer the optimal accuracy and the fastest speed. The computation of dilation rates d is as follows:

$$d(RF_{i+1}) = \frac{(RF_{i+1} - RF_i)}{S_i(k-1)}, \tag{1}$$

where RF_i, RF_{i+1}, S_i, and k denote the current receptive field, target receptive field, convolution stride, and convolution kernel size, respectively.

Subsequently, we utilize dilated convolution to acquire high-resolution feature maps with distinct receptive fields. Thus, semantic information with a different receptive field relevant to fine-grained information is computed by:

$$F_{2d} = \begin{cases} DConv(F_2, d(RF) = 1), & RF = 5 \times 5, \\ DConv(F_2, d(RF) = 2), & RF = 9 \times 9, \\ DConv(F_2, d(RF) = 3), & RF = 11 \times 11, \end{cases} \tag{2}$$

where $DConv(\cdot)$ refers dilated convolution operation with different dilation rates d, and $d(\cdot)$ refers to Eq. 1. For the high-resolution feature map F_2, we employ dilated convolutions with dilation rates of $d = 2$ and $d = 3$ to obtain high-resolution feature maps F_{22} and F_{23}, which possess individually the same receptive fields as F_4 and F_5 feature

maps. To mitigate potential issues arising from the use of multiple dilated convolutions, we apply standard convolution $(d=1)$ to produce the feature map F_{21} that preserves local information. We directly employ dilated convolution and the high-resolution feature map to learn the semantic information relevant to the fine-grained information, addressing the issue of poor correlation between the acquired semantic information and the fine-grained information.

Learned the fine-grained relevant semantic information for different receptive fields, we utilize a learnable weight mechanism to fuse these features for each dilation rate d to obtain semantic features relevant to low-level features, denoted as F_2^d:

$$F_2^d = EMA(LWFusion(F_{21}, F_{22}, F_{23})), \quad (3)$$

where $LWFusion(\cdot)$ denotes the learnable weighted fusion method and $EMA(\cdot)$ denotes the Efficient Multi-scale Attention [33], which can be used to highlight the foreground objects and reduce the influence of background information on the small objects.

Finally, as shown in Fig.2, the feature map $F_2^{'}$ is obtained by applying a feature extraction operation to the low-level feature map F_2. By combining the learned semantic information related to fine-grained information F_2^d with the 5×5 receptive field feature map F_3 into $F_2^{'}$ through weighted fusion method, we achieve semantic feature enhancement of the low-level feature map, denoted as $F_2^{'}$:

$$F_2^{'} := LWFusion(F_2^{'}, F_2^d, F_3). \quad (4)$$

This module alleviates the issue of poor correlation between low-level and high-level features during feature fusion, bolsters the expressive ability of the low-level feature map for small objects and improves their recall rate.

4.2. Coarse-to-fine Feature Guidance Module

In section 4.1, we implement feature enhancement for low-level features to solve the problem of poor correlation between low-level features and high-level features that occurs with feature fusion in the original feature pyramid structure, bolstering the expressive capacity of fine-grained features for small objects. However, achieving robust detection of UAV objects with large-scale variations becomes challenging when relying solely on the augmentation of low-level features.

The high-level features have rich semantic information and large receptive fields, which help to achieve robust detection and object classification of objects at different scales. Given that objects in UAV imagery are predominantly small to medium-sized, we have introduced the Coarse-to-Fine Feature Guidance Module (CFGM) to specifically bolster the detection of medium-sized objects.

As shown in Fig.2, the coarse-grained guidance obtained from F_5 and the fine-grained feature $F_3^{'}$, enhanced by the DSIEM, co-guide the feature enhancement from coarse to fine in the mid-level feature map F_4:

$$F_4 := C2FE(F_4, F_3^{'}, TransposeConv(F_5)), \quad (5)$$

where $TransposeConv(\cdot)$ refers to the transpose convolution operation and $C2FE(\cdot)$ refers to the learnable weighted fusion approach from coarse to fine. Up-sampling the high-level feature map F_5 using transposed convolution yields coarse-grained guidance information that contains rich semantic information. Enhanced by the DSEIM module, the low-level feature map $F_3^{'}$ contains more fine-grained information and semantic details related to fine-grained information.

Using this coarse-grained guidance information allows for the precise addition of fine-grained information to medium-scale objects, from a global level to a local level, which substantially improves the recall rate for detecting medium-scale objects. This learnable weighted fusion method retains semantic information from large receptive fields, which helps to mitigate the negative effects of removing the large detection head on large object detection. It also enhances the classification performance for both large and medium objects.

4.3. Anchor-Head Matching Based Multi-scale Detection

In addition to the feature-level enhancement implemented by the aforementioned modules, this study further investigates the relationship between anchors, detection heads, and feature maps to improve UAV detection.

Existing multi-scale detection strategies rely on performing object detection at different scales with different anchor sizes. However, manually designed anchors are difficult to adapt to detect both small and normal-scale objects simultaneously, thereby resulting in scale-level mismatching issues. Therefore, we urgently need a strategy that matches anchors with detection heads, addressing the scale-level mismatch between anchors and feature maps.

One intuitive approach for improving the YOLO series often involves introducing the extra tiny detection head [54] to leverage low-level fine-grained features for small object detection. However, the extra detection heads make anchor segmentation excessively fine and mislead multiple detection heads learning highly similar features, causing overfitting problems. To address these issues, we introduce a multi-scale detection strategy that employs anchor-based head matching. This strategy retains scale-level matching detection heads by utilizing anchor ranges determined by an adaptive anchor method. The model achieves scale-level matching between anchors and feature maps by adjusting its detection heads interactively before training.

The adaptive anchors introduced in the YOLO series are first used to calculate the corresponding anchors for the UAV dataset. This involves preprocessing the widths and

heights of all ground truth (GT) boxes within the dataset to obtain candidate anchors, denoted as "*bbox*":

$$bbox = \{b \in Scale(GT, img\_size) \mid w_b > 2 \vee h_b > 2\}, \tag{6}$$

where the $Scale(\cdot)$ function transforms the dimensions of Ground Truth (GT) boxes in the dataset from relative to absolute sizes, with "*img_size*" representing the input size required by the model. Subsequently, filter the scaled widths (w_b) and heights (h_b) to eliminate boxes with dimensions smaller than two pixels, ensuring a more significant overlap with the Ground Truth (GT) boxes.

Candidate anchors are usually clustered into a set of anchors using the $K\text{-}means$ algorithm, which achieves clustering by optimizing the following objective function:

$$J = \sum_{i=1}^{K} \sum_{b \in bbox} ||b - \mu_i||^2, \tag{7}$$

where *J* denotes the Within-Cluster Sum of Squares (WCSS), which is used to measure the clustering effect and is optimized to a minimum to achieve the best clustering effect. "*K*" denotes the number of clusters, in this case, corresponding to the number of detection heads. "*b*" denotes each candidate anchor. "μ_i" denotes the clustering center point of the i-th category, using the formula $\mu_i = \frac{1}{|C_i|} \sum_{x \in C_i}$ to continuously update the clustering center until the stopping condition is satisfied.

To ensure that the clustering algorithm focuses more on shape than size, the widths and heights of the candidate anchors are whitened to eliminate scale size differences within the dataset. Then, the $K\text{-}means$ algorithm clusters the obtained candidate anchors (*bbox*) to actively generate a set of anchors for each detection head, denoted as "anchors":

$$\text{anchors} = K\text{-}means\left(\left\{\left(\frac{w_i - \mu_w}{\sigma_w}, \frac{h_i - \mu_h}{\sigma_h}\right) \mid w_i, h_i \in bbox\right\}\right), \tag{8}$$

where w_i and h_i denote the widths and heights of the candidate anchors, respectively. μ_w and μ_h denote the mathematical expectations of the widths and heights of the candidate boxes, respectively. σ_w and σ_h denote the standard deviations of the widths and heights of the candidate boxes, respectively. $K\text{-}means(\cdot)$ is the clustering algorithm described in Equation 7 that clusters a set of anchors for each detection head based on candidate anchors (*bbox*).

Finally, the obtained anchors are optimized using a genetic algorithm. The decision to retain the mutation results is based on calculating the average of the maximum matches between the mutated anchors and all candidate anchors is greater than the fitness ("*th*") before mutation. The formula is as follows:

$$anchors = \begin{cases} GenA(anchors), & fit(GenA(anchors)) > th, \\ anchors, & fit(GenA(anchors)) \leq th, \end{cases}$$
$$\text{s.t. } fit(m\_anchors) = \frac{1}{|bbox|} \sum_{b \in bbox} \max_{a \in m\_anchors} r(a, b), \tag{9}$$

where $GenA(\cdot)$ operates stands for a genetic algorithm that actively mutates and selects anchors by simulating the process of natural selection, to identify a set of anchors that offer superior performance. $fit(\cdot)$ refers to a fitness calculation algorithm that employs $r(\cdot)$ to assess the direct fitness of the mutation anchors (*m_anchors)* derived from the genetic algorithm to the candidate box. Repeat the above operation for n rounds to compute a set of anchor sizes for each detection head that best matches this dataset.

It is well known that each detection head corresponds to a type of scale feature map, and each scale feature map is associated with an optimal detection object scale scope. However, from the above calculation process, it is evident that the adaptive anchor method primarily considers obtaining the most closely matching anchors within the dataset and assigning them to each detection head. It does not, however, consider whether the anchors assigned to each detection head are scale-level matched with the corresponding feature map, as can be referred to in Table 1. For example, for the small detection head in YOLOv7, the best matching object size is between 16 and 32 pixels. However, when performing clustering on the VisDrone set to determine the anchors, the resulting anchor scope of [4, 6, 8, 14, 15, 11] indicates a scale-level mismatch issue between the anchors and the feature maps.

Table 1. Comparison of the generated anchors with the optimal anchor scopes of the current head. **Anchor Scope** stands for the optimal anchor scopes of the current head, and **Anchor** stands for the anchors generated by YOLOv7.

Head	Anchor Scope	Anchor
Tiny Head	<16	Unused
Small Head	16∼32	[4, 6, 8, 14, 15, 11]
Medium Head	32∼125	[12, 16, 19, 36, 40, 28]
Large Head	>125	[36, 75, 76, 55, 72, 146]

To address this problem, we investigate the optimal correspondence between the detection head and anchor scopes for achieving scale-level matching between anchors and detection heads. Specifically, we first use the anchors generated by the adaptive anchor method to obtain the scale-level matching detection head. Then, we adjust the model's detection head and recompute the anchors for the new detection head using the adaptive anchor method, achieving a scale-level match between the anchor points

and the feature map. Taking VisDrone as an example to demonstrate the algorithm, the original YOLOv7 employs three detection heads: *small*, *medium*, and *large*. We firstly utilize the $K\text{-}means$ algorithm and current dataset with the labels to calculate the most matching anchors on each head, respectively resulting in $[4, 6, 8, 14, 15, 11]$, $[12, 16, 19, 36, 40, 28]$, and $[36, 75, 76, 55, 72, 146]$. Afterward, to understand the distribution of object sizes in the current dataset, we need to find the minimum and maximum values in each group of anchors, *i.e.* (4,15), (12,28), and (36,146).

Next, we match the optimal detection heads in Table 1 according to the obtained minimum and maximum values, and the results are roughly *tiny*, *small*, and *medium*. Subsequently, we adjust FGHDet based on these matching results, which means adding the tiny head and removing the large head. Finally, the anchors clustering results of tiny, small, and medium detection heads on FGHDet are $[4, 6, 8, 14, 15, 11]$, $[12, 16, 19, 36, 40, 28]$, and $[36, 75, 76, 55, 72, 146]$, which realize the scale-level matching of anchors with the feature maps.

In this way, we attain scale-level matching between anchors and feature maps, mitigating over-fitting issues and enhancing the performance of small object detection.

5. Experiments

Due to the latest YOLO series method, YOLOv10 [39] is the anchor-free method, it cannot all be directly combined with our method. Thus, we integrate FGHDet with the latest anchor-based YOLO model, YOLOv7 [41], and conduct comprehensive ablation experiments to evaluate the effectiveness against state-of-the-art methods.

5.1. Datasets and Metrics

We evaluate the proposed method on three primary benchmarks for UAV aerial imagery: VisDrone [6], CARPK [12], and Drone-vs-Bird [3]. VisDrone comprises 7,019 high-resolution (2000×1500) aerial images across 10 categories, featuring small and densely distributed objects. It employs 6,471 images for training, 548 for validation, and 1,610 for testing. CARPK contains 989 training images and 459 test images captured by drones, presenting challenges such as small and densely distributed objects. Additionally, the dataset exhibits significant variations in lighting, darkness, and background conditions. Drone-vs-Bird consists of 1,387 training images and 434 test images, incorporating extensive drone and environmental data. Furthermore, it includes birds that have a resemblance to drones, intensifying the challenges associated with classification. We utilize Average Precision (AP), Average Recall (AR), and mean Average Precision as an evaluation metric for accuracy, where mAP_{50} indicates IoU of 0.5 and mAP represents the average of all ten IoU thresholds ranging from 0.5 to 0.95.

5.2. Implementation Details

We implement our network based on PyTorch. All models are trained for 300 epochs with the original YOLOv7 parameter configuration. Our method employs the loss function consistent with YOLOv7, including object classification loss and bounding box regression loss, where the classification loss utilizes $BCELoss$[41] and $FocalLoss$[22], and the regression loss uses $CIoULoss$ [41]. Employing the Anchor-Head Matching based Multi-scale Detection strategy, we consider the correspondence between anchors and detection heads, resulting in customized anchor configurations for the three datasets. On VisDrone, in order to balance small objects and conventionally sized objects in the dataset, we adjust the anchor size corresponding to the detection head and obtain the configurations for the tiny, small, and medium detection heads as $[5, 6, 8, 14, 15, 11]$, $[12, 16, 19, 36, 40, 28]$, and $[36, 75, 76, 55, 72, 146]$, respectively. On CARPK, object sizes are concentrated between 15 and 60, which is not suitable for small detection heads. Therefore, we remove the tiny detection head and obtain the configurations for the small and medium detection heads as $[19, 35, 39, 18, 22, 45]$ and $[38, 26, 56, 25, 30, 55]$. Given that the data distribution in Drone-vs-Bird is similar to VisDrone, we adopt the same anchor configuration. The input image sizes are set to 640×640, 1280×1280, and 1536×1536 on VisDrone, and 640×640 on CARPK and Drone-vs-Bird. All experiments are conduct on a NVIDIA RTX A5000 GPU. And employ the Adam optimization algorithm with an initial learning rate of 0.01 and a decay rate of 1e-5. The batch sizes of 4, 4, and 2 for input resolutions of 640, 1280, and 1536, respectively.

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

To demonstrate the effectiveness of our method, we conduct a comprehensive comparison with state-of-the-art methods on the VisDrone validation set. We select state-of-the-art two-stage methods such as CZ Det [31], UFPMP-Det [14], and HRDNet [29], along with one-stage methods like CZ FCOS Det [31], SDPDet [52], and TPH-YOLOv5-l [54]. Due to variations in the input resolution among these models, we conduct experiments with input resolutions of 640×640, 1280×1280, and 1536×1536, respectively, to ensure fair comparisons. As illustrated in Table 2, our method significantly outperforms existing one-stage methods at all three input resolutions and approaches the accuracy of two-stage methods. At a resolution of 1536×1536, FGHDet achieves a 4.1% performance promotion to the baseline YOLOv7 in terms of mAP_{50}.

We also evaluate FGHDet on CARPK and Drone-vs-Bird. As reported in Tables 3 and 4, our method improves mAP by 4.1% and 2.2% compared to the baseline, respectively. Notably, the present method performs better on datasets with higher object densities, highlighting the

Table 2. Comparison of mAP and mAP_{50} with the state-of-the-art methods for different resolutions input on VisDrone. The symbol '†' indicates the baseline of FGHDet, and '-' indicates that the result is not reported.

Method	Publication	Size	Backbone	#Params.(M)	mAP	mAP_{50}
Two-stage						
DetectoRS+RFLA [46]	ECCV22	1333×800	ResNet50	-	27.40	45.30
CZ Det [31]	CVPR23	1333×800	ResNet50+FPN	-	33.22	58.30
OGMN [20]	ISPRS23	-	ResNeXt101	-	35.00	59.70
HRDNet [29]	ICME21	3800×2800	ResNeXt50+101	152.2	35.10	62.00
UFPMP-Det [14]	AAAI22	1333×800	ResNeXt-101	-	39.20	65.30
Anchor-free						
FCOS+RFLA [46]	ECCV22	1333×800	ResNet50	32.00	15.10	27.30
CZ FCOS Det [31]	CVPR23	1333×800	ResNet50	-	33.91	56.20
YOLOv9 [42]	arXiv24	640×640	CSPDarknet53	25.30	29.70	47.90
YOLOv10 [39]	arXiv24	640×640	CSPDarknet53	31.60	27.90	44.90
One-stage						
RetinaNet+CEASC [5]	CVPR23	1333×800	ResNet50	-	20.80	35.00
RetinaNet+QueryDet [49]	CVPR22	1333×800	ResNet50	-	19.60	35.70
YOLOv7† [41]	CVPR23	640×640	CSPDarknet53	37.20	27.20	48.60
YOLOv5x+EMA [33]	ICASSP23	640×640	CSPDarknet53	91.18	30.40	49.70
GFL V1+CEASC [5]	CVPR23	1333×800	ResNet18	-	28.70	50.70
FCOS+FGE+SAW [13]	PR24	1333×800	ResNet50	-	-	51.50
HRDNet [29]	ICME21	1333×800	ResNeXt18+101	152.2	31.40	53.30
HTH-YOLOv5 [26]	CVM24	1504×1504	CSPDarknet53	-	34.70	57.10
SDPDet [52]	TMM24	1333×800	ResNeXt101	-	34.20	57.80
OGMN [21]	ISPRS23	1360×765	ResNeXt101	141.8	35.00	59.70
STF-YOLO [15]	MEASUREMENT24	1280×1280	CSPDarknet53	46.74	36.73	60.14
YOLOv7† [41]	CVPR23	1536×1536	CSPDarknet53	37.20	37.30	61.10
TPH-YOLOv5-l [54]	ICCVW21	1920×1920	CSPDarknet53	60.40	40.70	62.70
FGFDet(ours)	-	640×640	CSPDarknet53	43.90	**32.10**	**53.80**
FGFDet(ours)	-	1280×1280	CSPDarknet53	43.90	**39.40**	**63.90**
FGFDet(ours)	-	1536×1536	CSPDarknet53	43.90	**40.80**	**65.20**

Table 3. Comparison of AP, AR, and mAP with the state-of-the-art methods for 640×640 input on CARPK.The symbols have the same meaning as in Table 2.

Method	AP	AR	mAP	mAP_{50}
CZ Det [31]	-	-	-	92.18
QueryDet [49]	-	-	-	93.96
YOLOv5l [32]	-	-	62.3	95.30
Car-Det [32]	-	-	63.1	95.80
YOLOv7† [41]	97.6	94.4	65.8	97.50
MHA-YOLOv5 [36]	-	-	64.2	97.75
FGHDet(ours)	98.2	95.2	**69.9**	**98.00**

Table 4. Comparison of AP, AR, and mAP with the state-of-the-art methods for 640×640 input on Drone-vs-Bird. The symbols have the same meaning as in Table 2.

Method	AP	AR	mAP	mAP_{50}
DETR [1]	-	-	25.1	66.7
YOLOv5 [3]	-	-	-	74.6
DETR+MNMS [18]	-	-	41.9	82.2
YOLOv7† [41]	88.9	89.4	49.2	93.0
FGHDet(ours)	93.5	87.4	**51.4**	**94.0**

model's excellent robust performance.

5.4. Ablation Studies

We validate the principal components of FGHDet, where we also employ YOLOv7 as the baseline in all the ablation studies. On the VisDrone validation set, we analyze the effectiveness of each component, using input resolution of 640×640 and continuing to adopt mAP and mAP_{50} as evaluating indicators (See Table 5).

Effect of Extra Prediction Head. Adding the tiny detection head proves effective in utilizing fine-grained information for small object detection, given that low-level features are better suited for detecting small objects. Although leading to a slight parameter increase (37.2M → 37.7M), the performance improvement in mAP is substantial (mAP : 27.2% → 29.00%, mAP_{50} : 48.60% → 49.50%).

Table 5. Ablation of each component on VisDrone validation set. **With** P_2 stands for addition of a tiny detection head. **DSIEM** stands for Detail-preserving Semantic Information Enhancement Module. **Removing Large Head** stands for anchor-head matching results. **CFGM** stands for Coarse-to-fine Feature Guidance Module.

Baseline	With P_2	DSIEM	Removing Large Head	CFGM	#Param.(M)	AP	AR	mAP	mAP_{50}
✓					37.2	58.7	48.8	27.20	48.60
✓	✓				37.7	58.2	48.9	29.00(↑ 1.8)	49.50(↑ 0.9)
✓	✓	✓			38.8	58.3	51.7	30.40(↑ 1.4)	51.50(↑ 2.0)
✓	✓		✓		26.7	61.5	48.7	30.00(↑ 1.0)	50.50(↑ 1.0)
✓	✓	✓	✓		27.7	60.4	52.7	31.40(↑ 1.0)	52.80(↑ 1.3)
✓	✓	✓	✓	✓	43.9	63.6	52.1	**32.10**(↑ 0.7)	**53.80**(↑ 1.0)

Effect of Detail-preserving Semantic Information Enhancement. As shown in Table 3, this lightweight module significantly improves the mAP_{50} performance (+2%) by adding only a few parameters (37.7M → 38.8M). This module enhances the expression ability of small objects within low-level feature map, elevates the model's average recall for small objects (AR: 48.9% → 51.7%). This trade-off between a slight increase in computational cost and a significant boost in performance highlights the importance of mining fine-grained features.

Effect of Removing Large Head. We further investigate the effect of adding P_2 and removing the large detection head from YOLOv7. This operation reduces the model parameters from 37.7M to 26.7M, increasing the mAP_{50} by 1.0%. Implementing the anchor-head matching strategy to achieve the scale level of the anchor and feature map, also prevents the over-fitting problem due to too fine a division of the anchors, improving the average accuracy (AP: 58.2% → 61.5%) of the model for small object detection. Subsequently, incorporating the DSIEM module further boosts the mAP_{50} by 1.3%. The combination of these two operations achieves a more compact meanwhile excellent detection model.

Effect of Coarse-to-fine Feature Guidance Module. Finally, we analyze the effect of the coarse-grained feature guidance module. When employing the above module, there is an increase in model parameters (27.7M → 43.9M). However, it enhances the model's ability to adapt to objects at different scales and further improves overall model performance. Combining all components, the model parameters are reduced, and the accuracy is further improved (mAP: 31.40% → 32.10%, mAP_{50}: 52.80% → 53.80%), improving model robustness to objects at different scales (AP: 60.4% → 63.6%).

5.5. Visualization

Figure 3 presents detection results within typical scenarios captured by drones (VisDrone), dense distribution scenes (CARPK), and instances involving small objects (Drone-vs-Bird). It is observed that FGHDet demonstrates exceptional performance in scenarios with dense distributions and tiny objects, effectively identifying small objects.

Table 6. Ablation of each component with a 640 × 640 input resolution on VisDrone test set. **Tiny** stands for the addition of a tiny detection head and introduces a P_2 feature map. **DSIEM** stands for Detail-preserving Semantic Information Enhancement Module. **RLH** stands for removing the large head. **CFGM** stands for Coarse-to-fine Feature Guidance Module.

Method	#P.(M)	GFLOPs	mAP	mAP_{50}
Baseline	37.2	105.3	21.7	40.5
+Tiny	37.7	116.9	23.7	41.4
+Tiny+DSIEM	38.8	175.0	24.4	42.8
+Tiny+RLH	26.7	108.6	24.2	42.3
+Tiny+DSIEM+RLH	27.7	166.0	24.9	43.7
FGHDet(ours)	43.9	238.9	25.6	44.5

5.6. Performance of the FGHDet on the VisDrone test set

To further demonstrate the effectiveness of the FGHDet, we showcase the parameters and GFLOPs (floating-point operations per second) increased, and the performance improvement of each module on the VisDrone test set.

From Tables 5 and 6, it can be found that employing the DSIEM module can substantially improve the model performance, especially the model recall, with a little increase in the GFLOPs and the parameters. The reason is that the DSIEM module enhances the expressiveness of the low-level feature map for small objects by learning semantic information related to fine-grained information. This reduction in the difficulty of distinguishing small objects from the background improves the recall rate for these objects. Employing the anchor-head matching strategy achieves scale-level matching between anchors and feature maps, thereby eliminating mismatched large detection heads. It not only reduces parameters and GFLOPs but also prevents the over-fitting problem that can arise when multiple detector heads learn the same features due to the fine division of the anchor. This approach further enhances the model's classification accuracy. The use of the CFGM module does increase the number of parameters and GFLOPs to some extent, but it plays a crucial role in enhancing the model's classification accuracy. The anchor-head matching strategy, which

Figure 3. The visualization results of FGHDet on the VisDrone (1^{st} and 2^{nd} row), CARPK (3^{rd} and 4^{th} row), and Drone-vs-Bird (5^{th} and 6^{th} row) test sets, using different colors to differentiate between the different classes, are still effective in scenarios with dense distributions and small objects.

removes mismatched large detection heads, can result in the loss of deep semantic information. This is not advantageous for object classification, particularly for medium and large objects. Analysis of Tables 5 and 6 reveals that the CFGM module can significantly improve the model's classification accuracy.

Overall, the FGHDet method can achieve large improvements on both test and validation sets with a little increase in parameters and GFLOPs, especially for densely distributed and small objects. We present the mAP_{50} results for each category on the VisDrone test set in Table 7. When we compare to YOLOv7 [41], YOLOv9 [42], and YOLOv10 [39] it is evident that our method achieves competitive results across almost all categories. In Fig. 4, we compare the detection effectiveness of YOLOv7+Tiny and our method on the VisDrone test set. The results demonstrate that our method effectively detects a greater number of small objects, reducing instances of missed and wrong detections. Additionally, most detected objects exhibit high confidence.

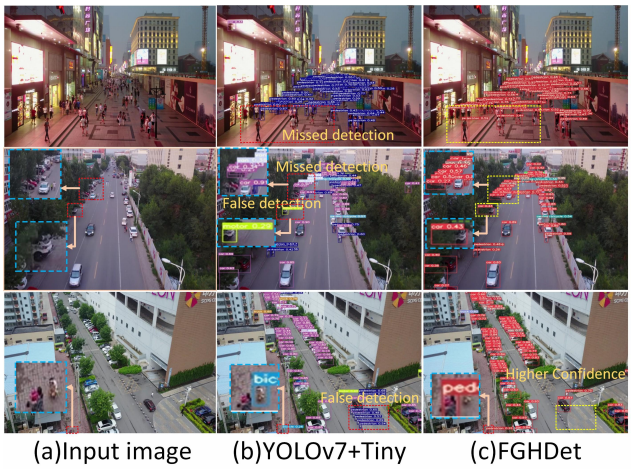

Figure 4. Comparison of the performance of YOLOv7+Tiny and FGHDet on the VisDrone test set.

5.7. Feature map visualization and analysis

To demonstrate the effectiveness of our method, we visually present the output feature maps corresponding to the detection heads used in YOLOv7, YOLOv7+Tiny, and the FGHDet method in Figure 5. Utilizing two images at varying scales, we present activation maps across four different scales. This is because only objects of corresponding size activated on scale-level matched feature maps can be combined with anchors to achieve effective object detection. Given that the detection heads correspond one-to-one with the feature maps, it highlights the necessity of achieving anchor-head matching.

From Fig. 5a, it can be seen that both images of different scales activate the object on mismatched feature maps, resulting in the object not being able to be regressed using the matching anchor, making it difficult to detect accurately. Due to the aforementioned reasons, the direct application of YOLOv7 to UAV detection results in poor detection per-

Table 7. Comparison of FGHDet models' performances on VisDrone test set for each category.

Method	#P.(M)	pedestrian	people	bicycle	car	van	trunk	tricycle	awning-tricycle	bus	motor	mAP_{50}
YOlOv10 [39]	31.7	29.4	17.6	13.1	75.1	43.2	48.1	23.4	24.7	61.0	33.1	36.9
YOLOv9 [42]	50.7	33.4	18.4	17.3	77.2	45.9	51.8	26.6	25.6	**65.9**	37.9	40.0
YOLOv7 [41]	37.2	37.4	26.3	16.3	78.9	44.1	47.8	26.7	23.2	62.4	41.5	40.5
FGHDet(ours)	43.9	**43.3**	**29.7**	**20.4**	**82.2**	48.3	**54.9**	26.6	**27.8**	65.0	**45.8**	**44.5**

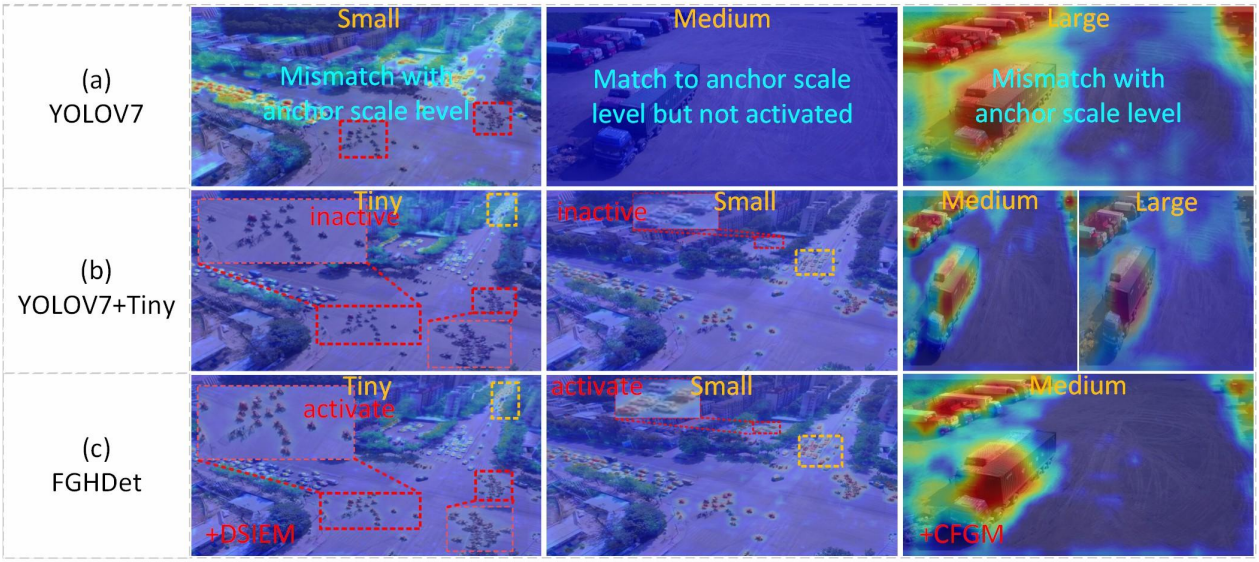

Figure 5. Comparison of activation feature maps for different scales of their detection heads for the YOLOv7, YOLOv7+Tiny, and FGHDet.

formance. Moreover, given that UAV objects are generally small, using the large head is not only unhelpful for detecting objects but also leads to mismatched anchors with feature map scales and results in a significant amount of redundant computations. Removing the large detection head is definitely the best option.

As shown in Fig. 5b, common existing improvements often involve adding a tiny detection head. Although this separates tiny objects from medium-sized ones, there is still a significant amount of information related to small objects that are challenging to activate, as indicated by the red-boxed region. Moreover, this approach employs multiple detection heads, leading to a fine subdivision of anchors. This results in each detection head learning numerous similar features, leading to over-fitting issues and making it difficult to truly address the problem of small object detection.

As illustrated in Fig. 5c, our approach demonstrates a stronger representational capacity for small objects on the tiny and small-scale feature maps. From the red boxes in the tiny-scale feature map, we can observe that our method activates more small objects. This also confirms that the DSIEM module can enhance the semantic information of small objects, reduce the difficulty of distinguishing them from the background, and improve their recall. Utilizing the FPN also enables the detection of more objects at corresponding scales within each subsequent level of the feature map. As observed from the yellow box in the activation graph, our method not only significantly improves the recall rate of the objects but also enhances the activation strength of objects at the matching scale. From the medium-scale feature map, it can be observed that the FGHDet method exhibits a higher activation strength than YOLOv7+Tiny. This demonstrates that the addition of the CFGM module improves the mid-level feature map, enhancing the classification effectiveness for medium and large objects.

The anchor-head matching strategy achieves scale-level matching between the anchors and the feature maps, thus preventing the over-fitting problem of the detection head. Combining the low-level feature map that has strong expressive ability for small objects, the mid-level feature map which is effective for classifying medium and large objects, and anchors that are scale-level matched to the feature maps. This integrated approach significantly enhances the performance of UAV object detection, particularly for scenarios with dense and very small object distributions.

5.8. Comparison with the ASPP module

The ASPP module [2] also employs dilation convolution to capture semantic information. It is designed to obtain richer semantic information by utilizing a larger dilation

Table 8. Comparison of mAP and mAP_{50} with the ASPP module and DSIEM module for 640×640 input on Drone-vs-Bird.

dilation rate	ASPP		DSIEM
	6,12,18	1,2,3	1,2,3
AP	91.6	88.5	**93.5**
AR	86.5	**91.3**	87.4
mAP	50.2	49.8	**51.4**
mAP_{50}	93.9	93.0	**94.0**

rate on high-level feature maps. This approach allows for the addition of more comprehensive contextual information, which is beneficial for object detection tasks. However, the DSIEM module is engineered to learn semantic information related to fine-grained information present on low-level feature map. It achieves this by calculating and applying a fixed and smaller dilation rate, which helps to enhance the representation of low-level feature map for small objects and improves their recall.

To verify the effectiveness of the DSIEM module, we also test the ASPP module as a substitute for the DSIEM module in learning fine-grained semantic information. We use default dilation rates of 6, 12, and 18 for the ASPP module and the calculated dilation rates of 1, 2, and 3 for the DSIEM module. As shown in Table 8, the ASPP module does not perform as well as the DSIEM module across both sets of dilation rates when compared to our approach. Analysis of the Average Recall (AR) indicates that using smaller dilation rates on low-level feature map is more effective for learning semantic information related to fine-grained details. This approach facilitates the separation of small objects from the background. Additionally, the Average Precision (AP) reveals that the DSIEM module is more adept at enhancing the expression of small objects, resulting in higher detection accuracy. The significance of semantic information related to fine-grained details for small object detection is further confirmed by these results, particularly in scenarios featuring dense distributions and small objects.

5.9. More experiments on the effectiveness of small objects on the COCO2017

Our method achieves competitive results in UAV object detection. To adequately verify the generalization ability of FGHDet, we also test its effectiveness in detecting small objects in the COCO2017 [25]. According to Table 9, our method is competitive with the state-of-the-art methods. It can detect more small objects in the scene and identify them accurately with fewer parameters (AP_s: 35.2% → 36.6%, AR_s: 53.7% → 56.5%). For medium-scale objects, our method also has significant enhancement (AP_m: 55.9% → 56.6%, AR_m: 73.5% → 74.3%). This demonstrates that our method is not only applicable to UAV-photographed scenes but also enhances the detection performance of medium and small objects in natural scenes.

Table 9. Comparison of AP_s, AR_s,and mAP_{50} with the state-of-the-art methods for 640×640 input on COCO2017 validation set. The symbol '†' stands for the baseline of FGHDet, and '-' stands for the result that is not reported.

Method	#P.(M)	AP_s	AR_s	mAP_{50}
ATSS+RaFPN [53]	39.6	24.7	-	60.8
ResNeXt+LFPN [45]	57.5	24.5	-	61.3
Faster RCNN+AFPN [50]	52.2	24.7	-	61.3
RTMDet-L [44]	52.3	-	-	68.8
PPYOLOE-L [48]	52.2	31.4	-	68.9
Gold YOLO-L [40]	75.1	34.1	-	68.9
YOLOv7† [41]	37.2	35.2	53.7	69.7
YOLOv8 [17]	43.7	35.4	52.2	70.0
YOLOv9 [42]	50.9	36.3	53.5	70.5
FGHDet(ours)	43.9	**36.6**	**56.5**	**70.6**

6. Conclusion

In this study, we introduce a novel plug-and-play method tailored for fine-grained feature enhancement and anchor-head matching in multi-scale detection, specifically crafted for UAV object detection in images. Firstly, we utilize the DSIEM to preserve original details and acquire coarse-grained semantic information relevant to fine-grained details, enhancing the ability of low-level features to express small objects. Secondly, we employ the CFGM to augment mid-level features through the co-guidance of coarse-grained and fine-grained features. This approach enhances the model's detection performance for medium-scale objects and bolsters its robustness to objects of various scales. Finally, by aligning the anchor scopes with the detection heads, we ensure a scale-level match between anchors and feature maps, averting over-fitting problems. Extensive experiments demonstrate the effectiveness of our approach in UAV object detection, attaining competitive accuracy.

However, due to the limited features of small objects and the unavoidable loss of these features during the acquisition of semantic information, there remains a significant challenge in detecting very small and distant dense objects. In our future work, we will continue to explore other methods to enhance small object features.

Acknowledgement

This work is supported in part by the Shandong Provincial Natural Science Foundation for Distinguished Young Scholars (ZR2021JQ26), National Natural Science Foundation of China (62176141), Major science and technology innovation project of Shandong Province (2021CXGC11204), Taishan Scholar Project of Shandong Province (tsqn202103088), Natural Science Foundation of Shandong Province (ZR202103010201).

References

[1] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*, pages 213–229. Springer, 2020. 8

[2] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 2, 11

[3] A. Coluccia, A. Fascista, A. Schumann, L. Sommer, A. Dimou, D. Zarpalas, F. C. Akyon, O. Eryuksel, K. A. Ozfuttu, S. O. Altinuc, et al. Drone-vs-bird detection challenge at ieee avss2021. In *2021 17th IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 1–8. IEEE, 2021. 2, 7, 8

[4] Y. Ding, K. Zhu, P. Wei, Y. Lin, and R. Wang. Deformable cnn with position encoding for arbitrary-scale super-resolution. In *International Conference on Computational Visual Media*, pages 93–108. Springer, 2024. 3

[5] B. Du, Y. Huang, J. Chen, and D. Huang. Adaptive sparse convolutional networks with global context enhancement for faster object detection on drone images supplementary material. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13435–13444, 2023. 3, 8

[6] D. Du, P. Zhu, L. Wen, X. Bian, H. Lin, Q. Hu, T. Peng, J. Zheng, X. Wang, Y. Zhang, et al. Visdrone-det2019: The vision meets drone object detection in image challenge results. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019. 2, 7

[7] H. Fang, P. Wu, Y. Li, X. Zhang, and X. Lu. Unified embedding alignment for open-vocabulary video instance segmentation. In *Proceedings of the European conference on computer vision*, pages 225–241. Springer, 2024. 3

[8] T. Hao, Y. Tao, M. Li, X. Ma, P. Dong, L. Cui, P. Lv, and M. Xu. Foreground and background separate adaptive equilibrium gradients loss for long-tail object detection. In *International Conference on Computational Visual Media*, pages 200–218. Springer, 2024. 2

[9] A. He, X. Li, X. Wu, C. Su, J. Chen, S. Xu, and X. Guo. Alss-yolo: An adaptive lightweight channel split and shuffling network for tir wildlife detection in uav imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2024. 1

[10] J.-Y. He, Z.-Q. Cheng, C. Li, W. Xiang, B. Chen, B. Luo, Y. Geng, and X. Xie. Damo-streamnet: Optimizing streaming perception in autonomous driving. *International Joint Conference on Artificial Intelligence*, pages 810–818, 2023. 3

[11] M. Hong, S. Li, Y. Yang, F. Zhu, Q. Zhao, and L. Lu. Sspnet: Scale selection pyramid network for tiny person detection from uav images. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2021. 3

[12] M.-R. Hsieh, Y.-L. Lin, and W. H. Hsu. Drone-based object counting by spatially regularized regional proposal network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 4145–4153, 2017. 2, 7

[13] S. Huang, S. Ren, W. Wu, and Q. Liu. Discriminative features enhancement for low-altitude uav object detection. *Pattern Recognition*, 147:110041, 2024. 8

[14] Y. Huang, J. Chen, and D. Huang. Ufpmp-det: Toward accurate and efficient object detection on drone imagery. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1026–1033, 2022. 7, 8

[15] Y. Hui, J. Wang, and B. Li. Stf-yolo: A small target detection algorithm for uav remote sensing images based on improved swintransformer and class weighted classification decoupling head. *Measurement*, 224:113936, 2024. 8

[16] J. Jiao, Y.-M. Tang, K.-Y. Lin, Y. Gao, A. J. Ma, Y. Wang, and W.-S. Zheng. Dilateformer: Multi-scale dilated transformer for visual recognition. *IEEE Transactions on Multimedia*, 25:8906–8919, 2023. 2

[17] G. Jocher, A. Chaurasia, and J. Qiu. Ultralytics YOLO, Jan. 2023. 12

[18] M. Kassab, R. A. Zitar, F. Barbaresco, and A. E. F. Seghrouchni. Drone detection with improved precision in traditional machine learning and less complexity in single shot detectors. *IEEE Transactions on Aerospace and Electronic Systems*, 2024. 8

[19] C. Lang, G. Cheng, B. Tu, C. Li, and J. Han. Base and meta: A new perspective on few-shot segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[20] X. Li, W. Diao, Y. Mao, P. Gao, X. Mao, X. Li, and X. Sun. Ogmn: Occlusion-guided multi-task network for object detection in uav images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 199:242–257, 2023. 8

[21] X. Li, W. Diao, Y. Mao, P. Gao, X. Mao, X. Li, and X. Sun. Ogmn: Occlusion-guided multi-task network for object detection in uav images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 199:242–257, 2023. 8

[22] X. Li, W. Wang, L. Wu, S. Chen, X. Hu, J. Li, J. Tang, and J. Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *Advances in Neural Information Processing Systems*, 33:21002–21012, 2020. 7

[23] J. Liang, H. Zeng, and L. Zhang. Details or artifacts: A locally discriminative learning approach to realistic image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5657–5666, 2022. 2, 3

[24] T.-Y. Lin, P. Dollar, and R. Girshick. Feature pyramid networks for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2117–2125, 2017. 2, 3, 4

[25] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European conference on computer vision*, pages 740–755. Springer, 2014. 12

[26] Y. Lin and Y. Liu. Improved yolov5 algorithm for small object detection in drone images. In *International Conference*

on Computational Visual Media, pages 354–373. Springer, 2024. 8

[27] A. Liu, S. Li, Y. Chang, W. Zhang, and Y. Hou. Coarse-to-fine cross-view interaction based accurate stereo image super-resolution network. *IEEE Transactions on Multimedia*, 2024. 2

[28] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8759–8768, 2018. 3

[29] Z. Liu, G. Gao, L. Sun, and Z. Fang. Hrdnet: High-resolution detection network for small objects. In *2021 IEEE International Conference on Multimedia and Expo*, pages 1–6. IEEE, 2021. 7, 8

[30] X. Lu, W. Wang, C. Ma, J. Shen, L. Shao, and F. Porikli. See more, know more: Unsupervised video object segmentation with co-attention siamese networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3623–3632, 2019. 3

[31] A. Meethal, E. Granger, and M. Pedersoli. Cascaded zoom-in detector for high resolution aerial images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2045–2054, 2023. 7, 8

[32] D.-L. Nguyen, X.-T. Vo, A. Priadana, and K.-H. Jo. Car detector based on yolov5 for parking management. In *The 12th Conference on Information Technology and Its Applications*, pages 102–113. Springer, 2023. 8

[33] D. Ouyang, S. He, G. Zhang, M. Luo, H. Guo, J. Zhan, and Z. Huang. Efficient multi-scale attention module with cross-spatial learning. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1–5. IEEE, 2023. 5, 8

[34] Z. Qin, C. Han, Q. Wang, X. Nie, Y. Yin, and L. Xiankai. Unified 3d segmenter as prototypical classifiers. In *Advances in Neural Information Processing Systems*, volume 36, pages 46419–46432. Curran Associates, Inc., 2023. 3

[35] D. Ren, Y. Zhang, L. Wang, H. Sun, S. Ren, and J. Gu. Fclgyolo: Feature constraint and local guided global feature for fire detection in unmanned aerial vehicle imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2024. 3

[36] G. Song, H. Du, X. Zhang, F. Bao, and Y. Zhang. Small object detection in unmanned aerial vehicle images using multi-scale hybrid attention. *Engineering Applications of Artificial Intelligence*, 128:107455, 2024. 8

[37] M. Tan, R. Pang, and Q. V. Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10781–10790, 2020. 3

[38] Z. Tian, C. Shen, H. Chen, and T. He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9627–9636, 2019. 2

[39] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding. Yolov10: Real-time end-to-end object detection. *arXiv preprint arXiv:2405.14458*, 2024. 7, 8, 10, 11

[40] C. Wang, W. He, Y. Nie, J. Guo, C. Liu, Y. Wang, and K. Han. Gold-yolo: Efficient object detector via gather-and-distribute mechanism. *Advances in Neural Information Processing Systems*, 36, 2024. 12

[41] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7464–7475, 2023. 2, 3, 7, 8, 10, 11, 12

[42] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao. Yolov9: Learning what you want to learn using programmable gradient information. *arXiv:2402.13616*, 2024. 8, 10, 11, 12

[43] P. Wu, X. Lu, J. Shen, and Y. Yin. Clip fusion with bi-level optimization for human mesh reconstruction from monocular videos. In *Proceedings of the 31st ACM International Conference on Multimedia*, page 105–115. Association for Computing Machinery, 2023. 3

[44] H. Xiang, N. Jing, J. Jiang, H. Guo, W. Sheng, Z. Mao, and Q. Wang. Rtmdet-r2: An improved real-time rotated object detector. In *Chinese Conference on Pattern Recognition and Computer Vision*, pages 352–364. Springer, 2023. 12

[45] J. Xie, Y. Pang, J. Nie, J. Cao, and J. Han. Latent feature pyramid network for object detection. *IEEE Transactions on Multimedia*, 2022. 12

[46] C. Xu, J. Wang, W. Yang, H. Yu, L. Yu, and G.-S. Xia. Rfla: Gaussian receptive field based label assignment for tiny object detection. In *Proceedings of the European Conference on Computer Vision*, pages 526–543. Springer, 2022. 8

[47] Q. Xu, J. Wang, B. Jiang, and B. Luo. Fine-grained visual classification via internal ensemble learning transformer. *IEEE Transactions on Multimedia*, 2023. 2

[48] S. Xu, X. Wang, W. Lv, Q. Chang, C. Cui, K. Deng, G. Wang, Q. Dang, S. Wei, Y. Du, et al. Pp-yoloe: An evolved version of yolo. *arXiv preprint arXiv:2203.16250*, 2022. 12

[49] C. Yang, Z. Huang, and N. Wang. Querydet: Cascaded sparse query for accelerating high-resolution small object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13668–13677, 2022. 8

[50] G. Yang, J. Lei, H. Tian, Z. Feng, and R. Liang. Asymptotic feature pyramid network for labeling pixels and regions. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024. 12

[51] J. Ye and Z. Yu. Fusing global and local information network for tassel detection in uav imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2024. 3

[52] N. Yin, C. Liu, R. Tian, and X. Qian. Sdpdet: Learning scale-separated dynamic proposals for end-to-end drone-view detection. *IEEE Transactions on Multimedia*, 2024. 7, 8

[53] Z. Zhou and Y. Zhu. Rafpn: Relation-aware feature pyramid network for dense image prediction. *IEEE Transactions on Multimedia*, 2024. 2, 3, 12

[54] X. Zhu, S. Lyu, X. Wang, and Q. Zhao. Tph-yolov5: Improved yolov5 based on transformer prediction head for object detection on drone-captured scenarios. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 2778–2788, 2021. 5, 7, 8