




# APNet: Accurate Positioning Deformable Convolution for UAV Image Object Detection

Peiran Zhang , Guoxin Zhang , and Kuihe Yang 

**Abstract**—Unmanned aerial vehicle (UAV) image object detection, in recent years, has been receiving increasing attention for its wide application in military and civil fields. Current object detection methods perform well in generic scenarios, while vast small objects and extremely dense distribution in UAV images make it difficult to capture them, resulting in sub-optimal performance. In this paper, we propose a UAV image object detection framework APNet, which addresses the issue mentioned above by fine-grain deformable convolution (DC) and effective feature fusion. First, we design an *accurate positioning deformable convolution* (APDC), which changes the kernel shape dynamically to enforce refined features, especially in regions where objects gather densely. Specifically, a *positional information enhancement attention* (PEA) is designed to generate more accurate convolutional position offsets depending on the object position. Therefore, APDC alleviates inflexible deformation in vanilla DC and exhibits better adaptability to the shapes of different objects, which discriminates multi-objects in densely distributed areas in a fine-grain way. Second, we propose an *effective cross-layer feature fusion* (ECF) to integrate multi-scale features effectively and aggregate attentive features dynamically. Extensive experiments conducted on VisDrone and UAVDT demonstrate the universality and effectiveness of our APNet, achieving 29.8 and 48.7 in mAP and mAP50, respectively. Compared to the state-of-the-art (SOTA) method, our APNet achieves an improvement of 2.2 and 3.5 in mAP and mAP50, respectively.

Link to graphical and video abstracts, and to code: <https://latamt.ieeet9.org/index.php/transactions/article/view/8716>

**Index Terms**—Object detection, unmanned aerial vehicle (UAV) images, deformable convolution (DC), attention mechanism.

## I. INTRODUCTION

UAVs equipped with high-definition cameras have shown tremendous potential in detection tasks. Currently, they are widely applied in various practical scenarios, including agriculture, logistics transportation, environmental monitoring, and urban planning. In this paper, we are committed to improving the detection accuracy of UAV image object detection and exploring its potential, which is helpful for practical application in the UAV community.

Two main categories encompass present object detection works: two-stage detectors and one-stage detectors. Typical two-stage detectors include R-CNN series [1]–[3]. Typical one-stage detectors include YOLO series [4]–[6] and RetinaNet [7]. Two-stage detectors are well-known for their exceptional accuracy. However, they often experience slower

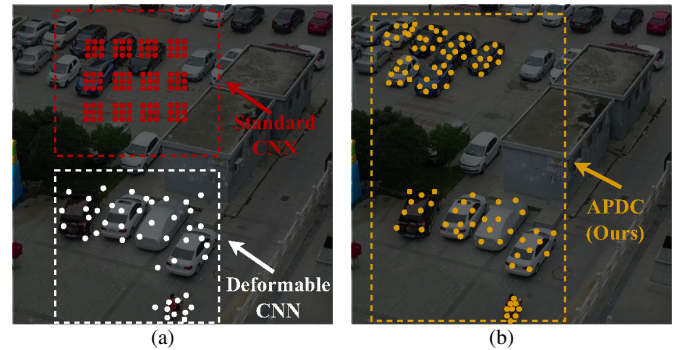


Fig. 1. Illustration of the sampling locations in  $3 \times 3$  standard CNN, deformable CNN, and APDC in UAV images. (a) regular sample locations of standard convolution (e.g. red) and deformed sampling locations of DC (e.g. white), in different regions. (b) accurate positioning sampling locations of APDC (e.g. yellow). APDC aggregates intensive refined features via accurate positioning sampling locations in regions where objects gather densely. For clear display, redundant sampling locations are ignored, and best viewed in zoom-in.

speeds. One-stage detectors possess the opposite characteristics, achieving real-time speeds but typically sacrificing accuracy, particularly for small objects. DETECTION TRANSFORMER (DETR) [8] pioneered the use of the Transformer [9] structure in object detection. [10] considers DETR as a one-stage detector. As research progresses, the DETR series [11]–[13] has surpassed convolutional neural network (CNN) [14]-based methods in general scenarios. RT-DETR [13] designs an efficient hybrid encoder module that decouples the intra-scale interaction and cross-scale fusion, achieving real-time speed. It surpasses the SOTA YOLO series detectors in both speed and accuracy.

However, when dealing with images obtained from UAVs, these detectors tend to show sub-optimal performance. This issue mainly stems from the substantial disparities between images in general scenes and UAV views, such as small objects, uneven distribution, complex backgrounds, and large variations. To tackle these problems, a multitude of algorithms have been proposed. To balance accuracy and efficiency, CEASC [15] leverages sparse convolution to optimize the head. Additionally, it incorporates a context-enhanced group normalization layer to mitigate the loss of context information. ClusDet [16] revolutionizes the detection process through the integration of clustering, which leads to a significant reduction in the overall number of chips. Furthermore, the inclusion of a scale estimation procedure further enhances the accuracy of detecting small objects. DSHNet [17] utilizes two

G. Zhang, P. Zhang and K. Yang are with Hebei University of Science and Technology, Shijiazhuang, China (e-mails: zhangpeiran@stu.hebust.edu.cn, zhangguoxin@gmail.com and ykh@hebust.edu.cn).

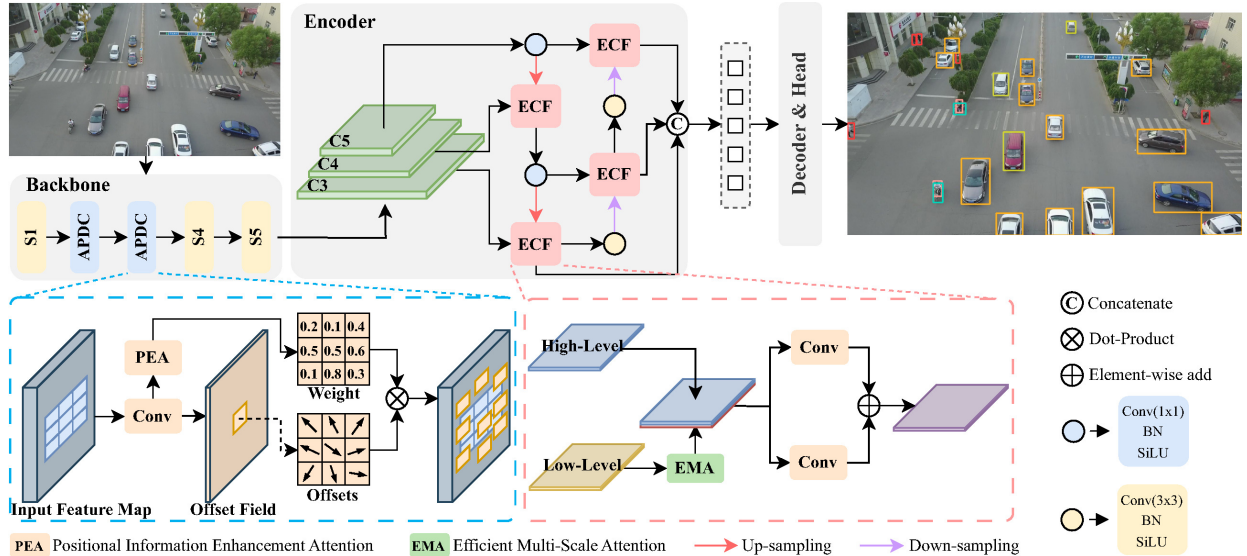


Fig. 2. The overall architecture of APNet. UAV images are input into the backbone network to obtain multiple feature maps. The backbone consists of stage 1, stage 4, and stage 5 from ResNet, as well as the proposed APDC module. We use the last three feature maps as the input to the encoder, and these feature maps are denoted as  $\{C3, C4, C5\}$ . The encoder first utilizes the ECF module to aggregate multi-scale features and then converts them into a sequence. The decoder and head precisely adjust parameters by minimizing the loss function, which ensures accurate predictions of the object category and boundary.

distinct samplers to sample proposals for tail-class and head-class objects individually. These proposals are then processed by two diverse heads, effectively tackling the problem of class imbalance through a dual-path approach. Considering that high-resolution images can effectively preserve contextual information even in deep networks, HRDNet [18] adopts a shallow network to handle high-resolution features, while a deep network is utilized to handle low-resolution features. This approach reduces computation costs while retaining more positional and semantic information. DMNet [19] utilizes a network to generate a density map for each image, which provides valuable information about the positions and density of objects. This enables more precise cropping of images, resulting in improved detection of tiny objects. QueryDet [20] introduces a query mechanism where coarse positions are predicted on low-resolution features. These positions are then used to guide in predicting accurate positions on high-resolution features. By adopting this approach, the model can effectively utilize the benefits of high-resolution features while avoiding useless computation costs.

Although these methods have improved the detection performance in UAV images, they generally suffer from the following issues: First, they overlook the limitations of standard convolutions in capturing features of densely populated and occluded objects. As shown in Fig. 1, standard convolutions struggle to accurately extract information from different objects in crowded regions. In the presence of occlusions, they mistakenly capture information from the occluding objects. Second, while employing multi-scale features, they fail to effectively preserve information from small objects, and the semantic gap between features at different scales is not bridged.

To solve the above problems, this paper proposes an efficient detection method for UAV images based on APDC. First, we have developed a novel convolutional approach

that differs from existing convolutions. This approach places a stronger emphasis on the positional information of the objects and guides feature extraction based on their locations. Unlike standard convolutions, it can dynamically adjust the shape of the kernel. Moreover, compared to DC [21] and modulated deformable convolution (MDC) [22], it closely approximates the true shape of the object. Second, we have designed the ECF module. This module not only effectively preserves information from small objects but also reduces the semantic gap between features at different scales, facilitating the comprehensive fusion of multi-scale features. Finally, we employ inner-IoU [23] to accelerate convergence and improve accuracy. Our contributions are given as follows:

1) We propose an accurate positioning deformable convolution (APDC) method that utilizes the positional information enhancement attention (PEA) module to obtain accurate positional information. This information is used to dynamically alter the shape of the kernel, enabling finer extraction of features and enhancing detection performance.

2) An effective cross-layer feature fusion (ECF) module is designed. This module is used for aggregating multi-scale features and addresses the issue of information loss in small objects.

3) Extensive experiments on the VisDrone and UAVDT benchmarks demonstrate that our proposed model exhibits compelling performance.

## II. METHOD

We choose RT-DETR as the baseline and propose a more efficient algorithm called APNet. The overall structure of APNet is illustrated in Fig. 2. The backbone is a ResNet [24] architecture with the addition of APDC. In the encoder part, we propose the ECF module to fuse multi-scale features from the backbone. The decoder and auxiliary prediction

heads optimize parameters iteratively using the loss function, which ensures accurate predictions of the object category and boundary.

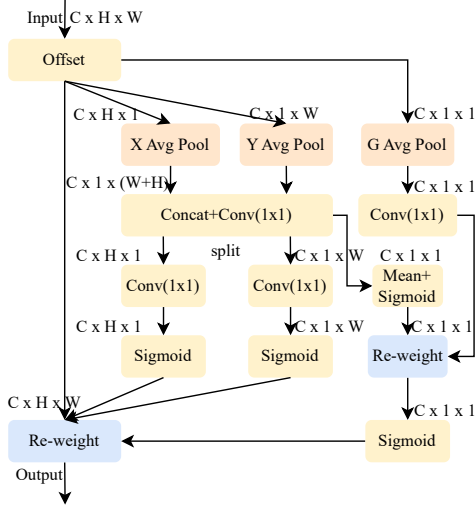


Fig. 3. The structure of PEA. Here, ‘X Avg Pool’ and ‘Y Avg Pool’ correspond to 1D horizontal global pooling and 1D vertical global pooling, respectively. ‘G Avg Pool’ refers to 2D global average pooling.

#### A. Accurate Positioning Deformable Convolution

In traditional CNNs, the convolution is performed using fixed-shaped kernels, typically  $3 \times 3$ ,  $5 \times 5$ , or  $7 \times 7$ . Consequently, the entire network has a fixed-shaped receptive field. However, this receptive field fails to accurately capture the intricate spatial variations. To overcome the limitations of conventional convolution, deformable convolutional networks (DCN) [21] introduced a new module called DC. The DC replaces the fixed sampling locations in standard convolution with learnable sampling locations. These learnable sampling locations are composed of fixed sampling locations added with learnable offsets, where the learnable offsets are obtained through convolution to the input features, enabling the shape of the kernel to dynamically adapt to various objects. MDC [22] introduced a modulation mechanism that allows the DC to adjust the learnable offsets and input features from different spatial locations.

However, the presence of numerous small objects and densely distributed objects in UAV images demands higher accuracy in sampling locations. Although DC and MDC have demonstrated improved performance in general scenes, it does not effectively enhance performance in UAV images. It may even result in performance degradation due to the utilization of extremely deviation sampling locations. The accurate generation of sampling locations relies on effectively learning the positional information encoded within the input feature, which is not easily achieved by a simple convolution. Building upon this insight, we propose the APDC module.

The key to the DC lies in the learnable offsets. To generate more accurate offsets, inspired by coordinate attention (CA) [25], we designed the PEA module, whose structure is shown

in Fig. 3. PEA consists of three branches: two 1D pooling operations along the  $x$ -axis and  $y$ -axis orientations, which aggregate information along the respective spatial directions, and one 2D pooling operation that globally aggregates information. We concatenate the features from the two 1D branches along the spatial direction and then apply a shared  $1 \times 1$  convolution to the concatenated features. This enables the attention module to capture precise positional information. The 2D branch is used to enhance the global modeling capability. These interconnected information are combined using re-weight. PEA excels at encoding global dependencies and precise positional information, leading to more accurate offset estimation.

Given the input feature map  $x$ , a standard  $3 \times 3$  convolution samples  $x$  using a regular grid  $R = \{(-1, -1), (-1, 0), \dots, (1, 0), (1, 1)\}$ . For each location  $p$  on the output feature map  $y$ , the output value  $y(p)$  is calculated as follows:

$$y(p) = \sum_{p_n \in R} w(p_n) \cdot x(p + p_n) \quad (1)$$

Here,  $p_n$  enumerates the locations in  $R$ ,  $w(p_n)$  denotes the weight at location  $p_n$  in the kernel, and  $x(p + p_n)$  represents the feature at location  $p + p_n$  in  $x$ . DC introduces learnable offsets, denoted as  $\{\Delta p_n\}_{n=1}^N$ , where  $N = |R|$ , resulting in the following modified formulation:

$$y(p) = \sum_{p_n \in R} w(p_n) \cdot x(p + p_n + \Delta p_n) \quad (2)$$

MDC utilizes a modulation scalar  $\{\Delta m_n\}_{n=1}^N$  to achieve the modulation function, where  $\Delta m_n$  lies in the range  $[0, 1]$ . The formula is as follows:

$$y(p) = \sum_{p_n \in R} w(p_n) \cdot x(p + p_n + \Delta p_n) \cdot \Delta m_n \quad (3)$$

Building upon the equation, the proposed APDC utilizes the PEA module to optimize the learnable offsets, denoted as  $PEA(\cdot)$ . The formulation is as follows:

$$y(p) = \sum_{p_n \in R} w(p_n) \cdot x(p + p_n + PEA(\Delta p_n)) \cdot \Delta m_n \quad (4)$$

Based on the analysis that accurate offsets require effective learning of positional information, we integrate APDC into the 2 and 3 stages of ResNet, replacing the original regular  $3 \times 3$  convolutions. APDC empowers the network’s receptive field to dynamically and precisely adapt to the shape of objects. This flexible receptive field, closely resembling the object’s shape, facilitates fine-grained extraction of object information, effectively enhancing the detection performance. Furthermore, accurately learning the true shape is essential for distinguishing between different objects, particularly in crowded regions. Therefore, APDC is well-suited for detection tasks in UAV images, and subsequent experiments have demonstrated the validity of our approach.

#### B. Effective Cross-Layer Feature Fusion

Due to the flying altitude, images captured by UAVs typically contain more small objects compared to general

scenes, and these small objects often contain limited information. Most existing detection algorithms are built using CNN, where during the process of continuous convolution, the information of small objects is easily polluted by other objects or background information. In current detection algorithms, it is common to utilize a backbone for extracting multi-scale features and subsequently fuse them to address the aforementioned issue. Features at different scales play distinct roles. Low-level features have higher resolution and more accurate positional information, making them suited for precisely localizing objects, especially small ones. However, they often lack sufficient semantic information and may not meet the requirements of classification tasks. On the contrary, high-level features contain rich semantic information, enabling accurate classification. However, they lack precise positional information, making them less suitable for localization tasks.

Feature pyramid network (FPN) [26] and its variants are dedicated methods for fusing multi-scale features and have been widely applied in numerous detection algorithms. They employ aggregation paths, either top-down or bottom-up, or a combination of both, to ensure that features contain rich semantic information and accurate positional information. While these methods effectively improve the capability to preserve information from objects, they overlook the issue of lost information on small objects in low-level features. Furthermore, these methods often employ simplistic techniques such as direct addition or concatenating features from different layers to achieve feature fusion. However, a notable semantic gap exists between detailed high-level features and coarse low-level features, and this straightforward fusion approach fails to bridge this gap effectively. As a result, a potential issue may arise where the coarse low-level features can overshadow the fine-grained high-level features.

To address these issues observed in current multi-scale feature fusion methods, we introduce an innovative ECF module. ECF first enhances the low-level features and then concatenates the high-level features with them along the spatial dimension. The concatenated feature is processed by parallel  $1 \times 1$  convolution layers, and the output features from different branches are element-wise added. This approach not only preserves the information of small objects but also facilitates more comprehensive interaction between features at different scales. As a result, ECF can generate features that retain both precise positional information and rich semantic information. We incorporate RepBlock [27] into one of the parallel branches. RepBlock consists of parallel  $1 \times 1$  and  $3 \times 3$  convolutions, allowing for diverse receptive fields. This design is advantageous for detecting objects of various sizes. We set the number of RepBlock in ECF to 3.

In addition, we incorporate the efficient multi-scale attention module (EMA) proposed by Ouyang *et al.* [28], whose structure is shown in Fig. 4. The group plays a vital role in EMA. Its main objective is to enhance the representation of diverse semantic information. To achieve this, EMA initially partitions the input features into  $G$  groups along the channel dimension, generating several sub-features. These sub-features are subsequently fed into sub-networks, each equipped with either small or large receptive fields. The purpose is to capture

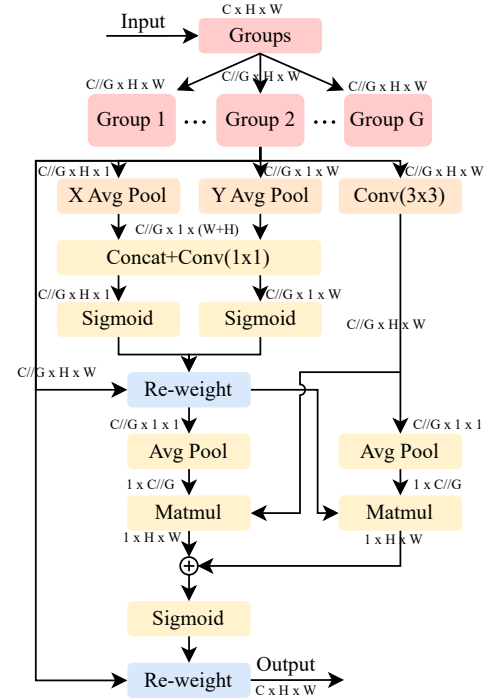


Fig. 4. The Structure of EMA. Here,  $G$  denotes the number of divided groups. ‘Avg Pool’ denotes 2D global average pooling.

spatial information at different scales. EMA achieves the enhancement of object information and suppression of irrelevant information such as background and noise by reweighting the extracted features. We utilize the EMA mechanism to emphasize the information on small objects and refine the low-level features.

### C. Loss Function

The loss function plays a pivotal role in the training process. Its purpose is to quantify the difference between the predicted and the real labels. By minimizing the loss function, the parameters undergo continuous optimization, resulting in improved accuracy in both classification and localization. In the baseline, the bounding box regression loss function consists of two parts: the L1 loss and the generalized intersection over union (GIoU) loss [29]. The GIoU loss takes into account the degree of overlap between different bounding boxes and introduces a new loss term. It effectively mitigates the problem of gradient vanishing that occurs when there is no overlap between the anchor box and the ground truth (GT) box. This makes the GIoU loss more accurate in localizing objects. The formula for the GIoU loss function is as follows:

$$L_{GIoU} = 1 - IoU + \frac{|C - B \cap B^{gt}|}{|C|} \quad (5)$$

$B$  and  $B^{gt}$  represent the predicted and the GT box, respectively.  $C$  is the smallest box covering  $B$  and  $B^{gt}$ .

However, GIoU loss overlooks the limited generalization capability of the intersection over union (IoU) loss. Zhang *et al.* [23] proposed a novel loss function called Inner-IoU, which can be easily incorporated into existing loss functions that involve IoU loss. We apply the Inner-IoU loss to the GIoU

TABLE I  
COMPARISON RESULTS OF DIFFERENT MODELS ON VISDRONE

Model	Backbone	Resolution	mAP	mAP50	FLOPs	Parameters	FPS
Faster-RCNN [2]	ResNet50	1000x600	19.1	31.8	118.3G	41.2M	37.6
Cascade-RCNN [3]	ResNet50	1000x600	21.8	37.6	120.3G	69.2M	30.2
RetinaNet [7]	ResNet50	640x640	11.1	19.5	83.4G	36.3M	57.7
Yolov5-m [5]	CSPDarknet53	640x640	20.7	36.9	49.0G	21.2M	69.4
Yolov8-m [6]	Darknet53	640x640	25.9	42.5	78.9G	25.9M	93.5
VAMYIOX-m [32]	CSPDarknet53	640x640	27.2	45.1	151.4G	27.1M	54.9
DSHNet [17]	ResNet50	1000x600	24.6	44.4	-	-	-
DMNet [19]	ResNet50	1000x600	28.2	47.6	-	-	-
QueryDet [20]	ResNet50	2400x2400	28.3	48.1	-	-	-
RT-DETR [13]	ResNet18	640x640	27.6	45.2	60.0G	20.0M	96.2
APNet	ResNet18+APDC	640x640	29.8	48.7	61.9G	21.3M	65.3

loss, resulting in the Inner-GIoU [23]. The formula for the Inner-GIoU loss function is as follows:

$$L_{Inner-GIoU} = L_{GIoU} + IoU - IoU^{Inner} \quad (6)$$

The Inner-GIoU loss function utilizes auxiliary bounding boxes to calculate the loss, effectively speeding up the bounding box regression process and achieving better regression results. At the same time, the retained loss term in Giou accurately describes the degree of overlap between the bounding boxes. A series of experiments demonstrated the effectiveness of the Inner-GIoU loss function in APNet.

### III. EXPERIMENTS

#### A. Datasets

1) VisDrone: The VisDrone [30] comprises 10,209 images, with a train set of 6,471 images, a validation set of 548 images, and a test set of 3,190 images. These images are captured by UAVs across various areas, including urban and rural, and at different altitudes. These images have a resolution of approximately  $2,000 \times 1,500$ . The annotations classify the objects into ten categories: pedestrian, person, bicycle, car, van, truck, tricycle, awning-tricycle, bus, and motor. Due to the unavailability of the test set, we follow prior work [20] to report evaluation scores on the validation set.

2) UAVDT: The UAVDT [31] comprises 40,735 images, with a train set of 24,143 images and a test set of 16,592 images. These images are captured using UAVs primarily in urban areas, with a resolution of approximately  $1,024 \times 540$  pixels. The dataset encompasses objects classified into three categories: car, bus, and truck.

#### B. Implementation Details

We implemented the proposed method on the following hardware configuration: an Intel i5-12600K CPU, 16GB RAM, and NVIDIA 3090 GPU. The corresponding software configuration included Windows 10 system, CUDA 11.1, and Python 3.7. During the training of APNet, we employ the AdamW optimizer with a fixed momentum of 0.9 and weight decay of 0.0001. We also set the initial learning rate to 0.0001 and utilize the batch size of 4. All training was conducted without pre-trained weights. The size of the input image is the same for the training and the validation. Other strategies and hyperparameters follow the baseline.

TABLE II  
COMPARISON RESULTS OF DIFFERENT MODELS ON UAVDT

Model	mAP	mAP50
ClusDet [16]	13.7	26.5
DMNet [19]	14.7	24.6
GLSAN [33]	17.0	28.1
RT-DETR [13]	16.3	27.0
APNet	17.9	29.4

#### C. Evaluation Metrics

To assess the effectiveness of the proposed method, we utilized multiple evaluation criteria, including mean average precision (mAP), floating point operations (FLOPs), parameter size, and inference time as frames per second (FPS). The mAP was calculated by considering ten IoU thresholds ranging from 0.5 to 0.95 in increments of 0.05. An IoU threshold of 0.5 was used to measure the mAP50. The computation procedure for mAP is outlined as follows:

$$Precision = \frac{TP}{TP + FP} \quad (7)$$

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

$$mAP = \frac{\sum_{n=1}^N \int_0^1 p(r) dr}{N} \quad (9)$$

Where true positive ( $TP$ ) denotes the quantity of correctly identified objects. False positive ( $FP$ ) denotes the quantity of objects that are misidentified as positive. False negative ( $FN$ ) denotes the quantity of objects that are misidentified as negative.  $N$  represents the number of categories, and  $p(r)$  stands for *Precision-Recall* curve.

#### D. Results and Analysis

To evaluate the effectiveness of the proposed model in UAV images, we compared it with other SOTA methods on VisDrone and UAVDT. The comparison results on VisDrone are presented in Table I. Despite a slight increase in computation cost, with FLOPs and parameters increasing by 1.9G and 1.3M respectively, APNet effectively improved performance.

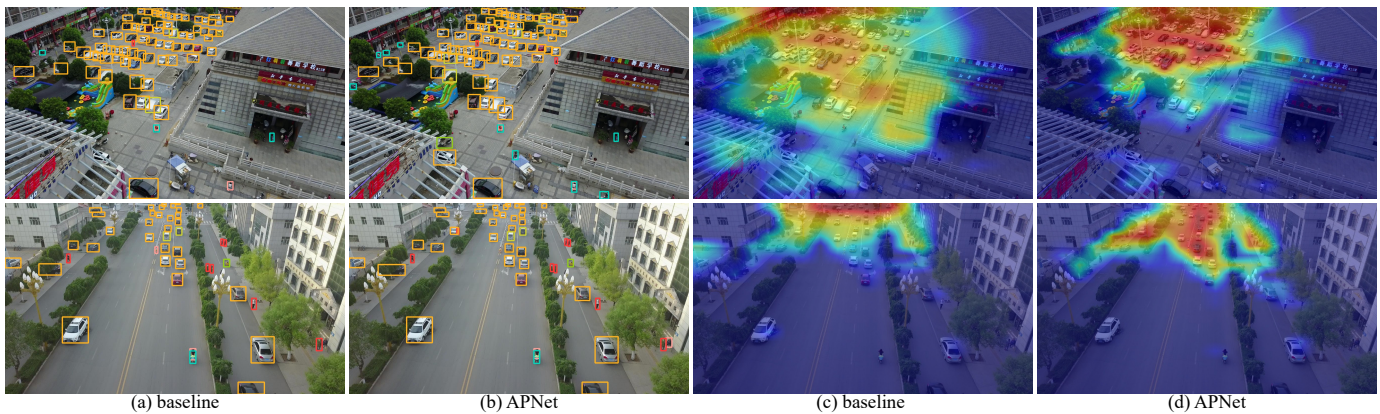


Fig. 5. Qualitative results for visualization of result and feature heatmap. (a)(b) are visualization results of baseline and our method. (c)(d) are visualization results of feature heatmap of baseline and our method.

TABLE III  
MAP50 OF EACH CATEGORY IN THE VALIDATION SET OF VISDRONE

Model	Resolution	mAP50	Pedestrian	Person	Bicycle	Car	Van	Truck	Tricycle	Awning-tricycle	Bus	Motor
Baseline	416x416	36.0	37.9	35.3	11.0	78.5	40.0	26.1	23.3	14.6	48.3	44.9
	640x640	45.2	53.2	45.7	19.6	85.0	48.7	36.1	31.1	16.2	59.0	57.1
	800x800	49.6	59.9	50.5	24.9	86.6	54.9	40.1	35.3	17.9	63.8	62.1
	1000x1000	53.1	63.7	54.9	30.8	88.2	55.6	42.2	40.1	21.7	67.6	65.8
APNet	416x416	38.1	41.0	39.0	12.4	79.7	41.8	30.0	26.1	14.9	48.0	48.1
	640x640	48.7	55.8	50.2	23.1	85.8	50.9	41.6	35.7	17.6	64.9	61.4
	800x800	51.3	60.8	53.2	27.5	87.5	54.7	41.9	38.7	19.8	65.5	63.7
	1000x1000	54.9	65.7	55.4	34.9	88.4	56.5	46.8	40.3	22.8	71.6	66.8

The mAP and mAP50 increased by 2.2 and 3.5 compared to the baseline, reaching 29.8 and 48.7 respectively, surpassing other advanced methods. According to the results presented in Table II, our model demonstrates compelling performance on UAVDT. Compared to the baseline, our model shows an improvement of 1.6 in mAP and 2.4 in mAP50. Notably, our method achieves impressive mAP and mAP50 of 17.9 and 29.4, respectively, surpassing other SOTA methods.

We visualize the detection results and feature heatmap of the baseline and APNet in Fig. 5. Based on the comparative results, it is evident that APNet demonstrates significant effectiveness in detecting small objects and distinguishing objects in crowded regions. Additionally, APNet showcases its capability to accurately recognize occluded objects that remain undetected by the baseline.

To investigate the robustness of the proposed method, we use input images with resolutions of  $416 \times 416$ ,  $640 \times 640$ ,  $800 \times 800$ , and  $1,000 \times 1,000$ . Table III presents the mAP50 scores of the baseline and APNet under different resolutions. It can be observed that mAP50 is improved at each of the four resolutions, with respective improvements of 2.1, 3.5, 1.7, and 1.8. The detection accuracy for almost every category is enhanced, demonstrating that our proposed method can extract features in a more granular manner.

#### E. Ablation Experiments

To objectively evaluate the effectiveness of each module, we performed ablation studies on the VisDrone, employing identical experimental conditions, and the results are shown in Table IV. Compared to the baseline, the inclusion of

ECF improves mAP50 by 0.7, with minimal increases in FLOPs and parameters. With the addition of APDC, there is a slight increase in computation cost, but mAP50 improves by 2.3. Lastly, by introducing the superior Inner-GIoU without altering the computation cost, there is a 0.5 increase in mAP50.

Furthermore, to demonstrate the superiority of each module and analyze the impact of different hyperparameters, we conducted additional experiments.

1) *Effect of APDC*: Table V illustrates the optimization capability of different attention mechanisms on MDC. These experiments were conducted in the second stage of the backbone. It can be observed that regular MDC cannot adapt well to the characteristics of objects in UAV images, resulting in a modest improvement of only 0.2 in mAP50. Moreover, incorporating attention mechanisms to optimize the offsets significantly enhances performance. When using CA and EMA, mAP50 improves by 0.9 and 1.1, respectively. Our proposed PEA demonstrates more effective extraction of positional information and learning more accurate object shapes, achieving a mAP50 of 47.5 with a 1.6 improvement, surpassing previous methods.

Table VI demonstrates the effects of inserting APDC into different stages of the backbone. Inserting APDC in stage 1 does not improve performance because the early extracted feature maps have high resolution but poor representation capability. Stage 2 and stage 3 show effective improvements in detection performance, with mAP50 reaching 47.5 and 48.2, respectively. Stage 4 and stage 5 have limited positional information, thus being unable to effectively enhance the performance.

TABLE IV  
ABLATION EXPERIMENT ON VISDRONE

Baseline	ECF	APDC	Inner-GIoU	mAP50	Precision	Recall	FLOPs	Parameters
✓				45.2	59.6	43.7	60.0G	20.0M
✓	✓			45.9	60.4	44.3	60.0G	20.0M
✓	✓	✓		48.2	61.3	46.7	61.9G	21.3M
✓	✓	✓	✓	48.7	61.5	47.0	61.9G	21.3M

TABLE V  
PERFORMANCE OF DIFFERENT ATTENTION IN THE MDC

Method	mAP50	FLOPs	Parameters
baseline+ECF	45.9	60.0G	20.0M
+MDC	46.1	60.6G	20.3M
+MDC-CA	46.8	60.6G	20.3M
+MDC-EMA	47.0	60.7G	20.3M
+APDC(ours)	47.5	60.6G	20.3M

TABLE VI  
IMPACT OF DIFFERENT STAGES OF APDC

Stage	mAP50	Precision	Recall
baseline+ECF	45.9	60.4	44.3
1	45.6	60.1	43.6
2	47.5	59.6	46.0
2,3	48.2	61.3	46.7
2,3,4	47.7	61.9	45.9
2,3,4,5	47.8	61.0	46.3

TABLE VII  
IMPACT OF RATIO ON INNER-GIOU

Ratio	mAP50	Precision	Recall
1.2	48.0	61.5	46.1
1.23	48.1	62.2	46.3
1.25	48.7	61.5	47.0
1.28	47.8	61.6	45.8
1.3	47.6	62.4	45.6

2) *Effect of Inner-GIoU*: We conducted experiments with different ratios to investigate the impact of Inner-GIoU on performance, and the results are presented in Table VII. The ratio controls the size of the auxiliary bounding boxes used in calculating the loss. For different datasets, it is important to select an appropriate ratio to achieve the best performance. In the case of the VisDrone, the highest mAP50 value of 48.7 is achieved when the ratio is set to 1.25.

#### IV. CONCLUSION

In this paper, we propose a novel one-stage method called APNet for object detection in UAV images. Firstly, we propose the APDC module, which endows the model to dynamically and accurately extract object information. By incorporating the APDC, the model can precisely learn the shapes of objects, thereby solving the problem of distinguishing objects in crowded regions and significantly improving detection accuracy. Secondly, we introduce the ECF module to address the

issue of information loss for small objects. The ECF module enhances valuable information while suppressing irrelevant details, effectively preserving the information of small objects. Additionally, it promotes comprehensive multi-scale feature fusion through parallel processing. Finally, we utilize the Inner-GIoU loss function to accelerate the convergence and enhance detection accuracy. Extensive experimental results demonstrate that APNet achieves SOTA.

#### REFERENCES

- [1] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440–1448, doi:10.1109/ICCV.2015.169.
- [2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 39, no. 6, pp. 1137–1149, 2017, doi:10.1109/TPAMI.2016.2577031.
- [3] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6154–6162, doi:10.1109/cvpr.2018.00644.
- [4] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018, doi:10.48550/arXiv.1804.02767.
- [5] G. Jocher, "YOLOv5 by ultralytics," May 2020. [Online]. Available: <https://github.com/ultralytics/yolov5>
- [6] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLO," Jan. 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [7] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2999–3007, doi:10.1109/iccv.2017.324.
- [8] 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 (ECCV)*, 2020, pp. 213–229, doi:10.1007/978-3-030-58452-8\_13.
- [9] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008, doi:10.48550/arXiv.1706.03762.
- [10] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proceedings of the IEEE*, vol. 111, no. 3, pp. 257–276, 2023, doi:10.1109/jproc.2023.3238524.
- [11] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020, doi:10.48550/arXiv.2010.04159.
- [12] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, "DINO: Detr with improved denoising anchor boxes for end-to-end object detection," *arXiv preprint arXiv:2203.03605*, 2022, doi:10.48550/arXiv.2203.03605.
- [13] W. Lv, S. Xu, Y. Zhao, G. Wang, J. Wei, C. Cui, Y. Du, Q. Dang, and Y. Liu, "Detsr beat yolos on real-time object detection," *arXiv preprint arXiv:2304.08069*, 2023, doi:10.48550/arXiv.2304.08069.
- [14] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi:10.1038/nature14539.
- [15] B. Du, Y. Huang, J. Chen, and D. Huang, "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 (CVPR)*, 2023, pp. 13 435–13 444, doi:10.1109/cvpr52729.2023.01291.

- [16] F. Yang, H. Fan, P. Chu, E. Blasch, and H. Ling, "Clustered object detection in aerial images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 8310–8319, doi:10.1109/iccv.2019.00840.
- [17] W. Yu, T. Yang, and C. Chen, "Towards resolving the challenge of long-tail distribution in uav images for object detection," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 3257–3266, doi:10.1109/wacv48630.2021.00330.
- [18] Z. Liu, G. Gao, L. Sun, and Z. Fang, "HRDNet: High-resolution detection network for small objects," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2021, pp. 1–6, doi:10.1109/icme51207.2021.9428241.
- [19] C. Li, T. Yang, S. Zhu, C. Chen, and S. Guan, "Density map guided object detection in aerial images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 737–746, doi:10.1109/cvprw50498.2020.00103.
- [20] 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 (CVPR)*, 2022, pp. 13 658–13 667, doi:10.1109/cvpr52688.2022.01330.
- [21] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 764–773, doi:10.1109/iccv.2017.89.
- [22] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable convnets v2: More deformable, better results," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 9300–9308, doi:10.1109/cvpr.2019.00953.
- [23] H. Zhang, C. Xu, and S. Zhang, "Inner-IoU: More effective intersection over union loss with auxiliary bounding box," *arXiv preprint arXiv:2311.02877*, 2023, doi:10.48550/arXiv.2311.02877.
- [24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778, doi:10.1109/cvpr.2016.90.
- [25] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 13 708–13 717, doi:10.1109/cvpr46437.2021.01350.
- [26] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 936–944, doi:10.1109/cvpr.2017.106.
- [27] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, and J. Sun, "RepVGG: Making vgg-style convnets great again," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 13 728–13 737, doi:10.1109/cvpr46437.2021.01352.
- [28] 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 *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5, doi:10.1109/icassp49357.2023.10096516.
- [29] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 658–666, doi:10.1109/cvpr.2019.00075.
- [30] Y. Cao, Z. He, L. Wang, W. Wang, Y. Yuan, D. Zhang, J. Zhang, P. Zhu, L. Van Gool, J. Han *et al.*, "VisDrone-DET2021: The vision meets drone object detection challenge results," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2021, pp. 2847–2854, doi:10.1109/iccvw54120.2021.00319.
- [31] D. Du, Y. Qi, H. Yu, Y. Yang, K. Duan, G. Li, W. Zhang, Q. Huang, and Q. Tian, "The unmanned aerial vehicle benchmark: Object detection and tracking," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 375–391, doi:10.1007/978-3-030-01249-6\_23.
- [32] Y. Yang, X. Gao, Y. Wang, and S. Song, "VAMYOLOX: An accurate and efficient object detection algorithm based on visual attention mechanism for uav optical sensors," *IEEE Sensors Journal*, vol. 23, no. 11, pp. 11 139–11 155, 2023, doi:10.1109/jsen.2022.3219199.
- [33] S. Deng, S. Li, K. Xie, W. Song, X. Liao, A. Hao, and H. Qin, "A global-local self-adaptive network for drone-view object detection," *IEEE Transactions on Image Processing*, vol. 30, pp. 1556–1569, 2021, doi:10.1109/tip.2020.3045636.



**Peiran Zhang** was born in Zhengzhou, Henan Province, China, in 1999. In 2022, he received his Bachelor's degree. He is now a master's student majoring in Computer Science and Technology at the Hebei University of Science and Technology (China). His research interest is computer vision.



**Guoxin Zhang** was born in 1998 in Xingtai, Hebei Province, China. In 2021, he received his Bachelor's degree. He is now studying for his master's degree at the Hebei University of Science and Technology (China). His research interest is computer vision.



**Kuihe Yang** was born in 1966, in Handan, Hebei Province, China. He received the B.S. degree from Tianjin University (China) in 1988, the M.S. degree from University of Science and Technology Beijing (China) in 1997, and the Ph.D degree in computer application technology from Xidian University (China) in 2004. From 2005 to 2007, he was a Postdoctoral Fellow in Army Engineering University of PLA (China). He went to Manchester University (UK) for short-term training in 2011. Currently, He is professor and master tutor with Hebei University of Science and Technology (China). His research interests include database application technology, artificial intelligence and machine learning.