IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 19 March 2024, accepted 8 April 2024, date of publication 12 April 2024, date of current version 26 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3387938

RESEARCH ARTICLE

CA2Det: Cascaded Adaptive Fusion Pyramid Network Based on Attention Mechanism for Small Object Detection

JITING ZHOU, QIAN XU, XINRUI ZHAO, ZHIHAO ZHOU, AND PU ZHANG
Shanghai Film Academy, Shanghai University, Shanghai 200072, China
Corresponding author: Jiting Zhou (zjting@shu.edu.cn)

ABSTRACT How to achieve fast and accurate small object detection holds crucial theoretical and practical significance. However, this task encounters substantial challenges due to scale differences among instances in the scene, along with the scarcity of inherent features and weak representation of small instances. To alleviate the above problems, we propose a novel attention-based cascaded adaptive feature pyramid fusion network, CA2Det, to effectively improve the small object detection performance. First, to prevent the information degradation of small-instance features during training, we introduce the efficient and lightweight Shuffle Attention mechanism to highlight the features of small instances. Second, to mitigate the information conflicts arising from the scale inconsistency among instances, we design a double-layer cascaded adaptive fusion pyramid module, CAFP, which can effectively suppress the information conflicts while enabling full information exchange across layers. Finally, we combine sparse convolution to achieve efficient high-resolution input, providing richer geometric information of the instances. Compared to the baseline network, on the COCO benchmark dataset and the popular UAV dataset VisDrone, both of which contain a large number of small instances, the proposed method improves the detection accuracy mAP values by 1.1% and 2.2%, respectively, while having a good real-time detection speed.

INDEX TERMS Small object detection, feature fusion, attention mechanism, multi-scale detection, deep learning.

I. INTRODUCTION

Object detection aims to accurately classify and regressively locate objects of interest in the scene. As a crucial foundation for tasks like object segmentation and tracking and localization, object detection has been widely used in computer vision. In recent years, thanks to the power of deep Convolutional Neural Networks (CNNs) in image feature extraction and the support of expansive datasets, remarkable achievements [1], [2], [3] have been made in general object detection. However, research on small object detection has progressed at a comparatively slower pace. The main reasons that hinder small object detection include instance information conflicts due to the inconsistency of object scales in the scene, fuzzy appearance of small objects, and feature scarcity. In practical applications such as intelligent traffic management, industrial inspection, and aviation detection, the identification and tracking of numerous small instances are frequently required. Therefore, implementing fast and accurate small object detection is both important and challenging.

Recently, in order to enhance small object detection performance, many works have appeared. These mainly include data augmentation [4], [5], increasing the input resolution [6], [7], [8], multi-scale perception [9], [10], [11], and enhancement of contextual information [12], [13], [14]. Among them, the most typical approaches include increasing the input resolution and improving multi-scale perception methods to construct a feature pyramid for multi-scale detection. In order to solve the problem of small instances

The associate editor coordinating the review of this manuscript and approving it for publication was Gustavo Olague.

being easily confused with the background due to their blurred geometric appearance in the scene, the most direct and effective approach is to increase the input resolution to provide rich geometric information. However, simply adding high-resolution input results in huge computational overhead, limiting practical applications. To address this issue, many works utilized sparse convolution networks [15], [16] and knowledge distillation [17] models to obtain a balance between model detection speed and accuracy.

For general object detection, pyramid multi-scale detection leverages CNNs to extract features at various scale layers, and features at each layer are responsible for detecting objects at the corresponding scales. Specifically, shallow layers, capturing finer geometric details, typically predict smaller objects, while deep layers, containing abundant semantic information, generally predict larger objects. This approach markedly alleviates the issue of missed detections in object detection and is now widely adopted in object detection models. It is worth noting that object detection is a task that combines both classification and regression localization. The accuracy of both object classification and localization is indispensable, so it is difficult to accurately predict targets with single-layer information. Traditional FPN [18] adopts a horizontal and top-down structure to fuse the information of each layer. However, the semantic information of deep-layer features inevitably degrades after being passed from level to level, and the shallow geometric information is not sufficiently communicated. Consequently, various FPN structures have been improved for specific tasks and have achieved excellent results.

Small object detection tasks demand attention to two primary issues. First, how to solve the information conflicts among instances, arising from the heuristic-guided allocation strategy of the FPN architecture. In other words, when instances of different sizes appear at the same spatial location simultaneously, if a deeper layer selects this location as a large instance prediction, other layers tend to prioritize that there are no instances of other scales at this location. This situation leads to feature conflicts among instances of different sizes, causing small instance features to be ignored and submerged. Second, despite the shallow layers containing rich geometric detail information, which is very helpful for small object prediction, the semantic information of the shallow layers is relatively lacking, which is still insufficient for accurate prediction. Previous works, such as FPN [18], BIFPN [19], and NAS-FPN [20], have demonstrated that effective feature fusion can enhance the accuracy of object detection. However, there remains considerable room for improvement in the balance between speed and accuracy.

For fast and accurate detection of small objects, in this paper, we propose CA2Det, an improved double-layer cascaded adaptive fusion pyramid network based on QueryDet [21]. The aim is to alleviate the feature degradation problem resulting from the susceptibility of small instances to noise in training and information conflicts problem among instances of different sizes in multi-scale detection, thereby improving the performance of small object detection in a targeted way. Firstly, we introduce efficient Shuffle Attention (SA) [22] to enhance the features of each layer in the ResNet50 backbone, focusing increased attention on small instances and preventing them from being contaminated by background and noise during training. Additionally, considering the information degradation caused by the down-sampling process of FPN layers at the neck, we also apply Shuffle Attention for specific P6 and P7 layers to enhance the instances' information. Secondly, to alleviate the problem of small instances being submerged due to instance scale inconsistency, we improve the Adaptively Spatial Feature Fusion (ASFF) [23] and design a novel double-layer cascaded adaptive fusion pyramid, CAFP, for multi-scale perception. This design dynamically learns the fusion weights of each layer of features during training, effectively filtering conflicts in instance information and facilitating efficient information exchange concurrently. Finally, we leverage the cascaded sparse query (CSQ) [21] algorithm to introduce a high-resolution feature input layer, acquiring richer geometric features for small instances while ensuring the real-time detection speed of the model.

In summary, our work achieves the following contributions:

- To achieve rapid and accurate detection of small objects, we proposed a novel attention-based cascaded adaptive feature pyramid fusion network, CA2Det.
- We designed a double-layer cascaded adaptive fusion pyramid module CAFP, capable of dynamically learning the fusion weights of each layer by each feature position, thereby enriching the instances features.
- We introduced SA attention to emphasize the focus on small instances to mitigate their feature degradation and utilized the CSQ algorithm to obtain richer instance features while ensuring efficient detection speed.

II. RELATED WORK

A. GENERAL OBJECT DETECTION

Object detection methods in deep learning can be mainly classified into two categories: two-stage and one-stage detectors, which are distinguished by whether candidate regions are generated first. Two-stage detectors typically use separate networks for foreground and background classification to generate regions of interest (ROI) that may contain objects. Subsequently, feature extraction and final detection are performed on these regions. Common algorithms include Faster R-CNN [2], Mask R-CNN [24], Cascade R-CNN [25], and DetectoRS [26]. On the contrary, one-stage detectors perform feature extraction and generate anchor boxes directly on the image for direct classification and localization. Its representative algorithms include the YOLO-series [27], [28], [3], [29], SSD [1],RetinaNet [30].

Compared with the two-stage detectors, single-stage detectors have a simpler model structure, so they consume less computing resources and have faster detection speed, but the accuracy tends to be slightly lower. With the

advantage of detection speed, single-stage detectors have received extensive attention and research. For example, the focal loss proposed by RetinaNet [30] has greatly alleviated the problem of imbalanced distribution of positive and negative samples in training. Yolov4 [28] introduced the data augmentation strategy, Yolov5 [31] devised an automatic anchor box strategy, Yolov7 [3] devised an efficient aggregation network and introduced a dynamic label assignment strategy, and Yolov8 [32] designed a novel anchor-free detection head and optimized the backbone network, all of which effectively to the enhancement of model performance. Consequently, the accuracy gap between one-stage detectors and two-stage detectors has now been markedly diminished. Recently, to reduce the resource consumption caused by the prediction of anchor boxes, some anchor-free detectors such as CenterNet [33], FCOS [34], and FSAF [35] have been proposed. The approach of these detectors is to regress the centroid and width-height of each object, but they exhibit lower recall and precision. Thus, there is large room for improvement in their structural design.

B. SMALL OBJECT DETECTION

In real-world scenarios, due to small instances with limited geometric details and blurred contours, make their features susceptible to noise interference during training. This susceptibility results in the degradation of instance feature information, posing huge challenges in the field of small object detection. To address these issues, numerous targeted methods have emerged, which can be summarized as follows:

1) Using data augmentation [4], [5] and formulating corresponding sample allocation strategies [36] to enrich the instance samples and enhance the robustness of the detection model.

2) Increasing the input resolution [6], [7] to provide rich geometric information. In particular, some models further propose to incorporate knowledge distillation [37], [38] and sparse convolution [21], [39] to save computational resources and ensure a reasonable detection speed.

3) Leveraging contextual information [13], [14], [40] to acquire more information related to small instances in the scene to provide more valuable clues for model predictions.

4) Utilizing multi-scale feature-aware fusion [9], [10], [41] to enable effective communication and integration of semantic and geometric information of instances at each layer, thereby obtaining richer representations of instances.

5) Introducing the attention mechanism [42], [43] to enhance the features of small instances and prevent instance feature decay during training.

C. MULTI-SCALE FEATURE AWARE

The multi-scale detection network has become a fundamental framework in recent years in the field of object detection to address the inconsistency of instance scales in scenes. SSD [1], a pioneer in this research, utilized feature maps from different levels of the CNN network to predict instances of different scales, significantly enhancing the recall of the model. However, the insufficient information of instances at each level resulted in low detection accuracy. To address this problem, the Feature Pyramid Network (FPN) [18] proposed a top-down fusion path connected horizontally and vertically to enhance the feature information at each layer, achieving commendable outcomes. Inspired by this, BIFPN [19] adopted a composite scaling method to achieve bi-directional feature fusion of the model. NAS-FPN [20] extended FPN by integrating the neural architectural search network and reinforcement learning to achieve cross-scope fusion. ASFF [23] proposed an adaptive cross-layer fusion method, effectively filtering information contradictions among different instances and reinforcing valuable features. SSPNet [11] designed CAM and SEM strategies to enhance instance features at specific scales and fully leverage the close relationship of adjacent layers for efficient feature fusion across layers. QueryDet [21] designed an efficient sparse query head and incorporated high-resolution input layers to capture richer geometric details. CEASC [40] formulated an adaptive masking strategy for automatically learning the mask ratio of each layer to obtain more positive training samples. Therefore, the effective construction of the feature pyramid network and the improvement of the feature-aware model is an important research direction for the object detection task.

D. ATTENTION MECHANISM

Similar to the neural system of the human brain, the attention mechanism in deep neural networks highlights the features of the instances by focusing more on the regions of interest. This approach can effectively mitigate the feature degradation resulting from instances being interfered by noise during training. In recent years, a large amount of lightweight and highly effective attention mechanisms, such as SE [44], CA [45], SA [22], ECA [46], EMA [47], and EA [48], have been proposed in the field of object detection and segmentation. Numerous works have shown that designing an appropriate attention mechanism at the corresponding position in the network can effectively improve the performance of the model. For example, Zhang et al. [42] devised a global-to-local fusion module through self-attention, enhancing the model's ability to distinguish between foreground and background. The Dynamic head [43] adopted an independent self-attention mechanism, sequentially applying scale-aware, spatially-aware, and task-aware on the feature maps, synergizing with the detection head to enhance the overall model performance. Zhang et al. [49] utilized an attention pyramid network to enhance instance edge features from local to global. MANet [50] designed a multidimensional attention network to efficiently aggregate instance information. The attention mechanism, valued for its flexibility and effectiveness, has found widespread application in the field of computer vision.

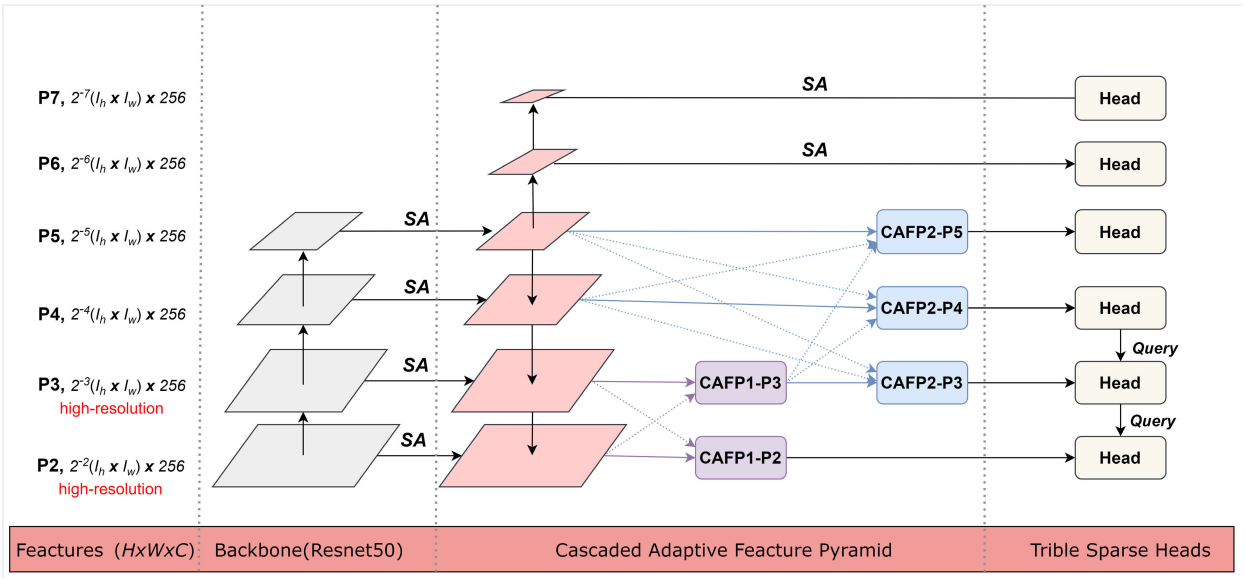


FIGURE 1. The overall pipeline of the proposed CA2Det. The image is first passed through the backbone and cascaded fusion networks for feature extraction to generate a feature pyramid. The width and height of the resolution of each layer of P_l is adjusted to 2^{-l} of the input features, and the number of channels is 256. The feature layers are then fed in parallel to the triple detection head (classification, regression, and query) for prediction.

III. METHOD

To achieve accurate and rapid small object detection, we propose the CA2Det algorithm based on QueryDet [21] (a detector with superior detection performance). The overall pipeline of CA2Det is shown in Fig. 1. The backbone part incorporates ResNet50 and SA attention, utilizing enhanced CNN features at each layer to construct the foundation of the multi-scale feature map. The neck part comprises the FPN structure, double-layer cascaded adaptive fusion pyramid module, and SA attention together, forming a novel pyramidal multi-scale feature fusion network to efficiently process the associated information across layers. The detection head consists of three parallel subnets: classification subnet, regression subnet, and query subnet, to make the final prediction of the objects. Next, we elaborate on the implementation details of our CA2Det algorithm from the three main parts: backbone, neck, and detection head.

A. ENHANCING MULTI-LEVEL FEATURES OF BACKBONE

In this work, we utilize the four feature layers (Res2, Res3, Res4, Res5) of ResNet50 [51], an outstanding convolutional neural network, as the backbone of the multi-scale detection model. Typically, small instances exhibit smaller sizes and ambiguous geometric appearances, making them susceptible to confusion with the background during training, consequently resulting in missed detections or misjudgments. In order to make the network focus more on small instances and prevent feature degradation of instances caused by background noise interference during training, we introduce efficient SA attention into the Res2-Res5 layers to enhance the features of small instances. The SA attention is characterized by its high efficiency and few parameters, which does not impose too much burden to the network.

Drawing inspiration from ShuffleNet [52], the SA attention [22] processes spatial and channel attention in parallel. The characteristic of SA attention lies in its implementation of the "Channel Shuffle" algorithm, facilitating the exchange of information across feature groups. The pipeline of the Shuffle Attention (SA) is shown in Fig. 2.

For each input feature map $F \in \mathbb{R}^{C\times H\times W}$, where C denotes the number of channels, and H and W denote the height and width of the feature map, respectively. Firstly, the SA module partitions F into N groups along the channel direction, which obtains $F = [F_1, \cdots, F_N]$, where $F_i \in \mathbb{R}^{C/N\times H\times W}$, and subsequently learns each group of features concurrently. Specifically, each unit F_1 is further divided into two branches, channel and spatial, $F_{i_1}, F_{i_2} \in \mathbb{R}^{C/2N\times H\times W}$. At this stage, F_{i_1} and F_{i_2} respectively use channel and spatial attention strategies to enhance the semantic and geometric information of each layer instance.

In the channel attention branch, global information is first extracted through global average pooling (GAP) according to Eq. (1).

$$s = f_{gap}(F_{i_1}) = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W} F_{i_1}(i,j) \tag{1}$$

Following this, the corresponding semantic information of the original features is augmented according to Eq.(2), to obtain the channel attention output.

$$F'_{i_1} = \sigma(f_c(s)) \cdot F_{i_1} = \sigma(W_1 \cdot s + b_1) \cdot F_{i_1} \tag{2}$$

where σ represents the sigmoid activation function.

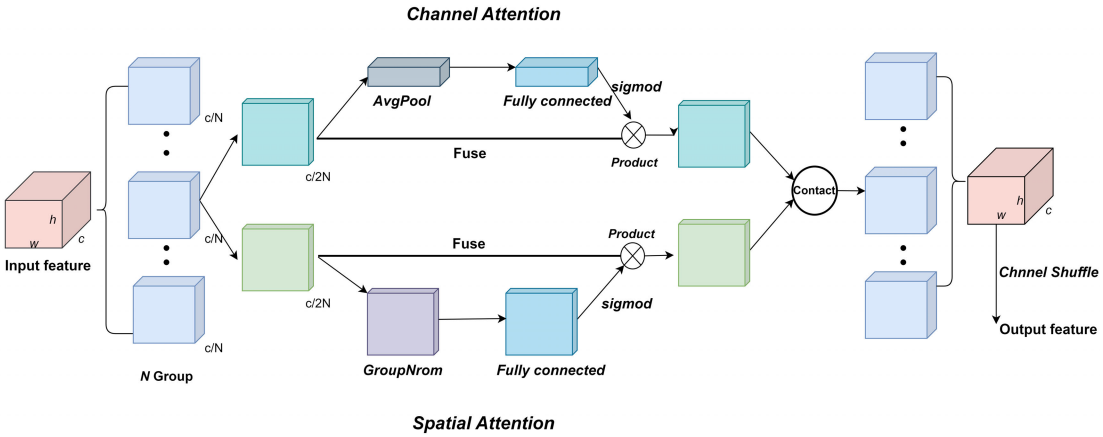


FIGURE 2. The pipeline of the Shuffle Attention module, enhancing instances features from both spatial and semantic branches.

In the spatial attention branch, the spatial information is first acquired through group normalization (GN). Subsequently, the corresponding geometric information of the original features is enhanced according to Eq. (3), to obtain the spatial attention output.

$$F'_{i_2} = \sigma(W_2 \cdot GN(F_{i_2}) + b_2) \cdot F_{i_2} \tag{3}$$

Subsequently, each subunit utilizes the ''shuffle unit'' operation to effectively integrate spatial and channel information, enabling the model to focus more on the spatial location (''where'') and semantic content (''what'') of each instance.

Finally, the ''channel shuffle'' algorithm is employed to share the information of each unit and then efficiently integrated as the output features. The introduction of the SA module significantly enhances the features of small instances, which is adequately prepared for the cascaded adaptive pyramidal fusion network at the neck.

B. CASCADED ADAPTIVE FUSION PYRAMID

In the model's neck, the enhanced Res2-Res5 feature layers are first horizontally and top-down fused, following the FPN structure, then generating the P2-P7 multi-scale feature maps. After the FPN stage, to facilitate comprehensive and efficient information exchange and sharing among the FPN layers, we improve Adaptive Spatial Feature Fusion (ASFF) [23] to design a double-layer cascaded adaptive fusion pyramid module CAFP. This module can dynamically integrate the contextual information of each layer while filtering the information conflicts among different instances at the same spatial location, thereby enhancing the representation of instances.

As illustrated in Fig. 3, in order to avoid the semantic and geometric gaps between large cross-layers, we adopt a progressive adaptive fusion approach. First, the feature layers, P2 and P3, are sent to the CAFP1 module for adaptive fusion, followed by utilizing the fused P3 (CAFP1-P3) to further fuse with P4 and P5. This cascaded fusion structure greatly aids in the effective integration of semantic and geometric features of instances across each layer.

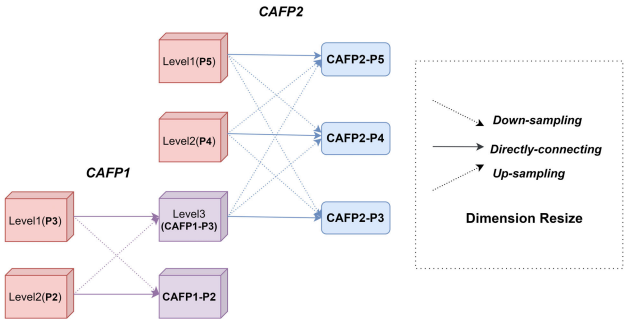


FIGURE 3. The pipeline of the double-layer cascaded adaptive fusion pyramid.

Prior to fusion, dimensional scaling is initially applied to each Level l, (in CAFP1, $l \in \{1, 2\}$; in CAFP2, $l \in \{1, 2, 3\}$). Specifically, the resolution is increased using up-sampling and bilinear interpolation operations, while the resolution is decreased using down-sampling and max-pooling operations, ensuring dimensional consistency before the fusion process at each level. As shown in Fig. 3, we take the second-level adaptive fusion module CAFP2 to describe the fusion details.

For each level of features, according to Eq. (4), the features of level1 (P5), level2 (P4), and level3 (CAFP1-P3) are multiplied and then summed with the learned corresponding weights α_{ij}^l, β_{ij}^l, and γ_{ij}^l, respectively. Thus, the three fused output layers are obtained.

$$y_{ij}^l = \alpha_{ij}^l \cdot x_{ij}^{1\to l} + \beta_{ij}^l \cdot x_{ij}^{2\to l} + \gamma_{ij}^l \cdot x_{ij}^{3\to l} \tag{4}$$

where α_{ij}^l, β_{ij}^l, γ_{ij}^l respectively represent the weights of each of the other layers relative to the fusion of this layer, which are calculated according to Eqs. (5)∼(7). And $\alpha_{ij}^l + \beta_{ij}^l + \gamma_{ij}^l = 1$. In training, α_{ij}^l, β_{ij}^l, γ_{ij}^l are optimized by gradient

back-propagation for automatic learning.

$$\alpha_{ij}^{l} = \frac{e^{\lambda_{\alpha_{ij}}^{l}}}{e^{\lambda_{\alpha_{ij}}^{l}} + e^{\lambda_{\beta_{ij}}^{l}} + e^{\lambda_{\gamma_{ij}}^{l}}} \tag{5}$$

$$\beta_{ij}^{l} = \frac{e^{\lambda_{\beta_{ij}}^{l}}}{e^{\lambda_{\alpha_{ij}}^{l}} + e^{\lambda_{\beta_{ij}}^{l}} + e^{\lambda_{\gamma_{ij}}^{l}}} \tag{6}$$

$$\gamma_{ij}^{l} = \frac{e^{\lambda_{\gamma_{ij}}^{l}}}{e^{\lambda_{\alpha_{ij}}^{l}} + e^{\lambda_{\beta_{ij}}^{l}} + e^{\lambda_{\gamma_{ij}}^{l}}} \tag{7}$$

As previously discussed, the P2-P5 feature layers acquire more accurate and comprehensive instance information through the double-layer cascaded adaptive perception fusion module. For P6 and P7, they are two feature layers obtained by down-sampling the P5 layer. To prevent the loss of instance information during the down-sampling process, we utilize efficient SA attention once again to enhance the instance features. At this point, after the feature layers are fused and processed by the cascaded adaptive pyramid network, they are transmitted in parallel to the head for prediction.

C. TRIPLE DETECTION HEAD

In the model's head, we adopt a structure consistent with QueryDet [21], as shown in Fig. 4. The prediction head at each layer consists of three parallel subnets: classification subnet, regression localization subnet, and query subnet, respectively. For pyramidal networks, small instances are often predicted at shallow layers because shallow layers contain richer geometric information. However, the distribution of small instances in real-world scenarios is typically sparse. In order to avoid computational redundancy, we expedite the inference by using the sparse convolution algorithm. Specifically, the classification subnet, regression localization subnet, and query subnet utilize four 3×3 convolutional networks, each coupled with a prediction network for object classification and localization regression, respectively. For efficient model training, all feature layers share classification and regression subnets parameters.

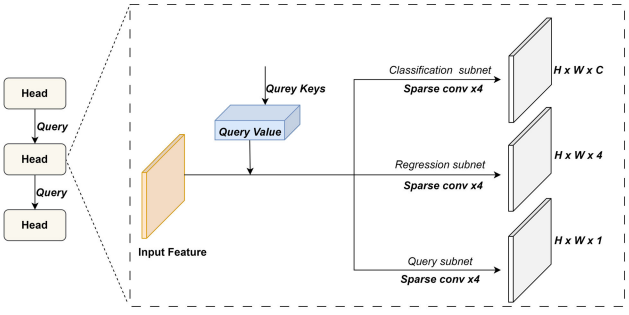


FIGURE 4. Structure of the triple detection head: classification subnet, regression subnet, query subnet.

In the query subnet, we adopt a cascaded approach to query information about small objects layer by layer. The query value of each layer corresponds to the query key of the previous layer, denoted as the query feature map. Specifically, during the training process, the distance between the centroid of each prediction box $b' = (x', y')$ and all the centroids of the instances' ground truth $b_i = \{(x_i, y_i)\}$ in each layer is calculated according to Eq. (8). The minimum anchor box scale on each feature layer P_l is s_l, and the query feature map of each layer is defined as Eq. (9).

$$D_l[x_i][y_i] = \min_i \{\sqrt{(x' - x_i)^2 + (y' - y_i)^2}\} \tag{8}$$

$$Q_l^*[x_i][y_i] = \begin{cases} 1 & \text{if } D_l[x_i][y_i] < s_l \\ 0 & \text{if } D_l[x_i][y_i] \geq s_l \end{cases} \tag{9}$$

Throughout the model, we use Focal Loss [30] for training the classification and query subnets, and use L1 Smooth Loss [53] for training the regression subnet. Consequently, the loss function of each layer P_l is:

$$\mathcal{L}_l(C_l, R_l, Q_l) = \mathcal{L}_{FL}(C_l, C_l^*) + \mathcal{L}_1(R_l, R_l^*) + \mathcal{L}_{FL}(Q_l, Q_l^*) \tag{10}$$

where C_l, R_l, and Q_l refer to the outputs of classification, regression, and query, respectively, while C_l^*, R_l^*, and Q_l^* correspond to the ground truth of the classification, regression, and query maps. $\mathcal{L}_{FL}$ represents Focal Loss, and $\mathcal{L}_1$ represents L1 Smooth Loss. For the whole network, the total loss function is as follows:

$$\mathcal{L}_{all} = \sum_l \mu_l * \mathcal{L}_l, \tag{11}$$

where μ_l is a hyperparameter representing the weights assigned to each feature layer by the model. In this paper, layers of P2-P7 contain parallel classification and regression subnets. In particular, we start to use the sparse query subnet in layers of P4-P2 to gradually query the locations of shallow small instances from coarse to fine. This approach makes efficient use of high-resolution feature maps, thereby balancing the accuracy and speed of small object detection.

IV. EXPERIMENTS

A. DATASET

In this paper, we evaluated our model using the benchmark dataset MS COCO [54] and the UAV dataset VisDrone [55], both of which contain a large number of small instances.

COCO [54] categorizes the instances into small, medium, and large classes based on varying scale sizes. The dataset comprises 117k images in the training set and 5k images in the validation set, covering 80 categories of general scene instances. The majority of COCO images have a resolution ranging from 500 to 800 pixels. Generally, instances smaller than 32×32 pixels are defined as small targets in the COCO dataset, constituting 30% of instances statistically. COCO has become one of the most widely used datasets in the field of object detection.

VisDrone [55] is a high-resolution dataset with multi-angle UAV aerial photography, containing a substantial number

of small dense instances. The dataset contains common instances: pedestrians, people, bicycles, cars, vans, trucks, tricycles, sunshade tricycles, buses, and motorbikes, totaling 10 categories. There are 6471 images in the training set and 548 images in the validation set, with resolutions ranging from 960 to 1360 pixels. Therefore, VisDrone is more challenging and is now widely used in remote sensing image detection and small object detection tasks.

B. EVALUATION METRICS

In this paper, we use Average Precision (AP), Mean Average Precision (mAP), Average Recall (AR), and Frames Per Second (FPS) as model evaluation metrics.

For a certain instance, all ground truth instances are categorized as either positive or negative cases before prediction. After prediction, there are two prediction situations: true or false, as shown in Table 1 of the confusion matrix.

TABLE 1. Confusion matrix.

Ground Truth	Prediction is Positive	Prediction is Negative
Positive	TP(True Positive)	FN(False Negative)
Negative	FP(False Positive)	TN(True Negative)

Then precision and recall can be expressed as:

$$Precision = \frac{TP}{TP + FN}, \tag{12}$$

$$Recall = \frac{TP}{TP + FP}. \tag{13}$$

The Average Precision (AP) is the area formed by the Precision-Recall (P-R) curve and the coordinate axis, defined by Eq. (14). A larger AP value indicates higher detection accuracy.

$$AP = \int_0^1 P(R)dR \tag{14}$$

The mAP represents the mean of the Average Precision (AP) across all categories. It is a measure of the overall performance and robustness of the model. It is calculated as Eq. (15).

$$mAP = \frac{1}{C}\sum_{i=1}^{C} AP_i \tag{15}$$

where C refers to the number of categories.

FPS is the number of picture frames processed per second. A higher FPS corresponds to a faster detection speed.

C. IMPLEMENTATION DETAILS

In this paper, all experiments were performed on an NVIDIA RTX 3060 12G GPU with Ubuntu 22.04 as its operating system. We conducted experiments based on the PyTorch framework and Detectron2(a popular object detection toolkit) [56]. For the whole model, we trained it using the stochastic gradient descent (SGD) optimizer with a batch size of 2 and an initial learning rate of 0.001. The COCO dataset was trained for 7200k iterations and the VisDrone dataset was trained for 200k iterations.

D. EXPERIMENTAL RESULTS AND ANALYSIS

On the COCO dataset, the experimental results are shown in Table 2, the proposed method exhibits significant improvements compared with the original network RetinaNet [30] and the baseline network QueryDet [21]. Specifically, the proposed method improves the mAP values by 2.0% and 1.1%, respectively, and the AP_s values show corresponding increments of 3.4% and 1.3%, respectively. Moreover, there are parallel improvements in AP_m and AP_l. When compared to RetinaNet, our FPS improves, but remains slightly lower than QueryDet. This is because we improve the model structure based on QueryDet to obtain higher accuracy, which introduces some computational overhead. Nevertheless, the proposed method achieves an optimal balance of model accuracy and detection speed with a mAP value of 39.5%, AP_s of 26.1%, and FPS of 6.93.

TABLE 2. Experimental results on COCO dataset.

Method	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	FPS
RetinaNet [30]	37.5	56.9	40.0	22.7	41.5	48.1	8.84
QueryDet [21]	38.4	59.2	41.1	24.8	42.0	49.2	**9.67**
CA2Det(ours)	**39.5**	**60.0**	**42.3**	**26.1**	**42.9**	**49.5**	6.93

TABLE 3. Experimental results on VisDrone dataset.

Method	mAP	AP_{50}	AP_{75}	AR_1	AR_{10}	AR_{100}	AR_{500}	FPS
RetinaNet [30]	26.3	45.9	27.1	**0.52**	5.46	34.7	37.4	1.97
QueryDet [21]	28.4	48.3	28.9	0.51	**5.87**	36.6	39.5	**2.06**
CA2Det(ours)	**30.6**	**53.8**	**30.3**	0.51	5.75	**39.0**	**42.9**	1.68

On the VisDrone dataset, the proposed method performs comparably to COCO. As shown in Table 3, the proposed method achieves the best detection accuracy mAP of 30.6% and the highest recall AR_{100} of 39.0% with the FPS of 1.68, which not only significantly enhances the model's detection accuracy, but also greatly improves the recall, thus mitigating the issue of small target miss-detection. This further illustrates the effectiveness of the proposed method. Relative to RetinaNet [30] and QueryDet [21], the proposed method improves the mAP by 4.3% and 2.2%, respectively, with corresponding increases in AR_{100} values of 4.3% and 2.4%. Moreover, in Table 4, compared with various mainstream detectors, including the anchor-free(FSAF [35]), FCOS [34]), the two-stage(Faster RCNN [2], Cascade R-CNN [25], DetectoRS [26]), and the one-stage(GFL V1 [57], CEASC [40]). It can be clearly observed that the proposed method CA2Det achieves significant improvements in both detection precision(mAP) and recall(AR). Notably, CA2Det shows a greater improvement compared to the anchor-free

TABLE 4. Comparison of experimental results with different detectors on VisDrone dataset.

Method	Backbone	mAP	AP_{50}	AP_{75}	AR_1	AR_{10}	AR_{100}	AR_{500}
FCOS [34]	ResNet50	24.5	42.7	22.6	0.43	5.25	32.1	36.9
FSAF [35]	ResNet50	24.7	44.3	23.5	0.45	5.37	33.8	37.3
Faster RCNN [2]	ResNet50	23.9	43.6	25.0	0.42	5.38	33.2	35.9
Cascade R-CNN [25]	ResNet50	24.4	43.1	25.3	0.48	5.47	34.3	35.7
DetectoRS [26]	ResNet50	27.8	45.4	28.3	**0.54**	5.62	35.2	38.1
RetinaNet [30]	ResNet50	26.3	45.9	27.1	0.52	5.46	34.7	37.4
GFL V1 [57]	ResNet50	27.6	49.2	26.9	0.51	5.56	35.6	41.5
QueryDet [21]	ResNet50	28.4	48.3	28.9	0.51	**5.87**	36.6	39.5
CEASC [40]	ResNet50	27.9	50.2	28.1	0.52	5.67	35.8	41.7
CA2Det(ours)	ResNet50	**30.6**	**53.8**	**30.3**	0.51	5.75	**39.0**	**42.9**

TABLE 5. Ablation studies on VisDrone.

SA	CAFP	CSQ	mAP	AP_{50}	AP_{75}	AR_1	AR_{10}	AR_{100}	AR_{500}	FPS
			28.5	48.2	28.8	**0.51**	**5.87**	36.5	39.5	0.92
✓			29.6	52.6	29.2	0.49	5.49	37.8	42.4	0.88
	✓		29.9	53.0	29.5	0.49	5.49	38.0	42.4	0.83
		✓	28.4	48.3	28.9	**0.51**	5.87	36.6	39.5	**2.06**
✓		✓	29.6	52.5	29.2	0.49	5.49	37.8	42.3	1.93
	✓	✓	29.8	52.9	29.5	0.49	5.49	37.8	42.3	1.89
✓	✓		**30.7**	**53.9**	**30.4**	**0.51**	5.75	**39.0**	**43.0**	0.75
✓	✓	✓	30.6	53.8	30.3	**0.51**	5.75	**39.0**	42.9	1.68

method and a smaller improvement compared to the two-stage method. This is because the anchor-free approach sacrifices model precision to reduce model complexity, while the two-stage method sacrifices model complexity for higher precision. The above results fully demonstrate that the proposed method CA2Det significantly improves the performance of small object detection while ensuring the real-time detection speed, and highlights the strong robustness.

E. ABLATION STUDIES

To validate the efficacy of the proposed model, we conducted ablation experiments on the VisDrone dataset, and Table 5 shows the results of the experiments that sequentially add each module to the baseline QueryDet detector. Next, we specifically analyze the effect of each module.

1) DETAILED ANALYSIS OF SA

To mitigate the feature degradation due to the interference of small instances by noise during training, we introduced SA attention in the backbone network ResNet50 and FPN-specific layers P6 and P7. In Table 5, in comparison to the baseline QueryDet, when SA is introduced to the network, the mAP improves by 1.1%, and the AR_{100} improves by 1.3%. This effectively proves that the addition of SA targetedly enhances the small-instance features.

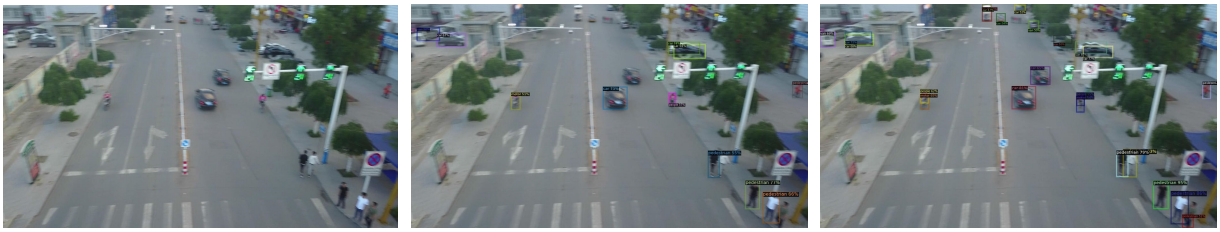
In addition, Table 6 compares the model performance of adding several other attention algorithms to the baseline QueryDet network respectively, and it can be observed that the introduction of SA attention is the most effective. Consequently, SA attention brings fewer parameters while exhibiting superior performance compared to other attention mechanisms.

TABLE 6. Comparison of different attention mechanisms under baseline.

Method	mAP	AP_{50}	AP_{75}	AR_{10}	AR_{100}	AR_{500}	FPS
-	28.4	48.3	28.9	**5.87**	36.7	39.5	**2.06**
CA [45]	28.5	49.5	26.8	5.54	37.6	39.7	1.85
SE-Net [44]	28.8	51.6	28.2	5.58	37.3	40.8	1.95
ECA [46]	29.4	52.2	29.1	5.36	37.4	42.0	1.87
SA [22]	**29.6**	**52.5**	**29.2**	5.49	**37.8**	**42.3**	1.93

2) DETAILED ANALYSIS OF CAFP

To mitigate the issue of small instances being submerged due to inconsistency in instance scale, we design a double-layer cascade adaptive perceptual fusion module. This module aims to filter information conflicts among feature layers and facilitate effective information exchange across feature layers. In Table 5, the network's performance exhibits a noticeable improvement with the incorporation of the CAFP module compared to the baseline QueryDet. Specifically, the

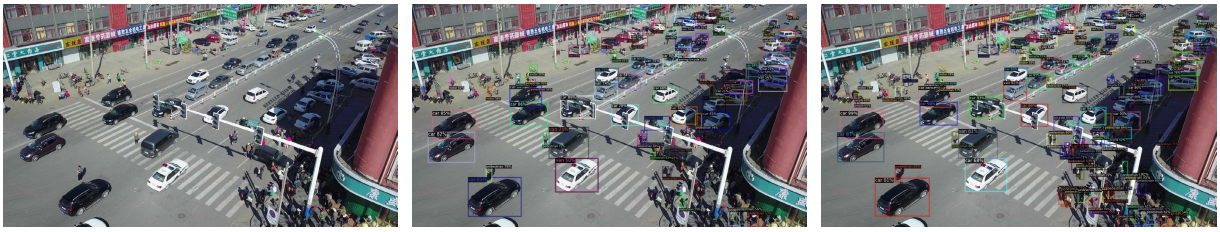


(a) Blurred Appearance

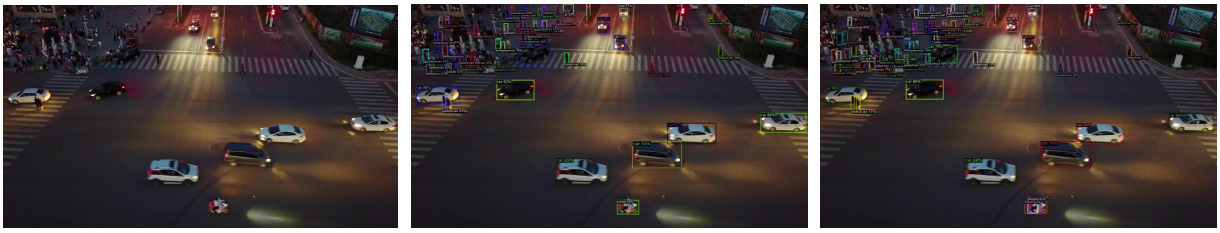
(b) Near-Far Instances

(c) Complex Backgrounds

(d) Dense Instances



(e) Multiple Classes and Different Sizes



(f) Insufficient Light

FIGURE 5. Visualization of detection results. The first column shows the original image, the second column shows the QueryDet detection, and the third column shows the proposed method detection.

accuracy mAP value and recall AR_{100} of the network are significantly improved by 1.4% and 1.5%, respectively, while the FPS value is slightly decreased, which strongly validates that CAFP is a lightweight and efficient feature-aware fusion module. The module effectively integrates instance information from each layer by dynamically learning the fusion weights of each feature layer, thereby obtaining a more comprehensive instance representation to support model predictions.

3) DETAILED ANALYSIS OF CSQ

In Table 5, following the incorporation of the CSQ algorithm, it can be observed that the model's detection speed FPS improves by 2.2×, from 0.75 to 1.68, while the mAP decreases by 0.1%, but this decrease is negligible. The reason is that the use of sparse convolution networks can efficiently leverage feature maps to avoid computational redundancy, which makes the model have good real-time detection performance.

Smaller objects are typically predicted at shallower layers. Table 7 compares the results of querying small instances, starting from different layers. It is noteworthy that the optimal balance of the overall model performance can be achieved when starting queries from the P4 layer. The mAP value decreases more when querying from the P5 layer. This is because the P5 layer mainly contains information about medium-scale instances instead of small-scale instances, which does not match the information about small instances contained in the shallow layer. Besides, querying from the P5 layer adds unnecessary computational overhead, which also leads to a decrease in FPS. Furthermore, the mAP value when querying from layer P3 is slightly lower than that from layer P4. The reason is that the P4 layer contains semantic information of small instances, which can support shallow layers' further predictions. Querying information of small instances layer by layer from P4 can fully utilize the contextual information, providing more valuable predictive cues for the model.

TABLE 7. Comparison of starting query Layer on VisDrone.

Satrt Layer	mAP	AP_{50}	AP_{75}	AR_{10}	AR_{100}	AR_{500}	FPS
-	**30.7**	**53.9**	**30.4**	5.75	**39.0**	**43.0**	0.75
P3	28.3	49.3	28.2	5.81	37.1	39.3	1.59
P4	30.6	53.8	30.3	5.75	**39.0**	42.9	**1.68**
P5	27.5	46.9	28.0	**5.83**	36.2	38.6	1.62

F. VISUALIZATION

We visualized the detection results on the VisDrone dataset. In the following visualization, the first column shows the original image, the second column shows the results detected by the baseline QueryDet, and the third column shows the results detected by the proposed method. Fig. 5(a) visualizes the detection results in an outdoor scene, which includes some small instances with blurred appearance. In Fig. 5(b), CA2Det improves the detection of both near and far instances in the realistic traffic scene. In Fig. 5(c) and Fig. 5(d), CA2Det effectively identifies numerous dense small instances in complex backgrounds, such as active crowds, bustling streets, and closely spaced vehicles. In Fig. 5(e), CA2Det exhibits better detection performance for objects of multiple classes and different sizes in the same scene. In Fig. 5(f), CA2Det accurately detects more objects in insufficient light scenes. These images encompass various shooting angles and diverse complex backgrounds. Figs. 5(a) to 5(c) compare the detection performance under different instance densities, while Figs. 5(c) to 5(e) compare the effectiveness for single-multiple-class object detection. It can be seen that the proposed method has good detection results for these small instances. However, CA2Det's performance for some mutual occlusion, blurred outlines, long distances, and extremely small objects in the scene still requires enhancement. It also needs to further optimize the lightweight of the model to broaden its applicable scenes. Overall, the proposed CA2Det method excels in achieving both fast and accurate detection of small objects.

V. CONCLUSION

In this paper, we propose CA2Det, a double-layer cascaded adaptive fusion pyramid approach aimed at improving the performance of small object detection. To address the feature degradation during training, resulting from the scarcity of inherent characteristics of small instances, we first introduce efficient shuffle attention to enhance the features of small instances. Secondly, we design a novel cascaded adaptive perception fusion method to effectively facilitate information exchange across layers, thereby alleviating the issue of instance-scale inconsistency in the scene. Finally, we leverage the cascade sparse query algorithm to efficiently utilize high-resolution feature maps. Experimental results on the COCO dataset and the VisDrone dataset show that our method significantly improves the performance of small object detection while ensuring an efficient model detection speed.

In the future, we will investigate how to better utilize contextual cues to improve the detection of occluded and inconspicuous objects, enhance the robustness of the models, and develop lightweight models to facilitate practical applications. We also plan to extend the model for 3D object detection and video target tracking localization tasks.

REFERENCES

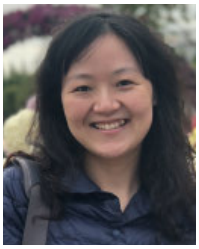
[1] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Computer Vision—ECCV*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer, 2016, pp. 21–37.

[2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

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

[4] M. Kisantal, Z. Wojna, J. Murawski, J. Naruniec, and K. Cho, "Augmentation for small object detection," 2019, *arXiv:1902.07296*.
[5] X. Zhang, E. Izquierdo, and K. Chandramouli, "Dense and small object detection in UAV vision based on cascade network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 118–126.
[6] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021.
[7] Z. Liu, G. Gao, L. Sun, and Z. Fang, "HRDNet: High-resolution detection network for small objects," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2021, pp. 1–6.
[8] C. Deng, M. Wang, L. Liu, Y. Liu, and Y. Jiang, "Extended feature pyramid network for small object detection," *IEEE Trans. Multimedia*, vol. 24, pp. 1968–1979, 2022.
[9] T. Gao, Q. Niu, J. Zhang, T. Chen, S. Mei, and A. Jubair, "Global to local: A scale-aware network for remote sensing object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–14, 2023.
[10] Y. Gong, X. Yu, Y. Ding, X. Peng, J. Zhao, and Z. Han, "Effective fusion factor in FPN for tiny object detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 1159–1167.
[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 Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.
[12] J. Pang, C. Li, J. Shi, Z. Xu, and H. Feng, "R^2-CNN: Fast tiny object detection in large-scale remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5512–5524, Aug. 2019.
[13] G. Zhang, S. Lu, and W. Zhang, "CAD-net: A context-aware detection network for objects in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 10015–10024, Dec. 2019.
[14] L. Cui, P. Lv, X. Jiang, Z. Gao, B. Zhou, L. Zhang, L. Shao, and M. Xu, "Context-aware block net for small object detection," *IEEE Trans. Cybern.*, vol. 52, no. 4, pp. 2300–2313, Apr. 2022.
[15] M. Figurnov, M. D. Collins, Y. Zhu, L. Zhang, J. Huang, D. Vetrov, and R. Salakhutdinov, "Spatially adaptive computation time for residual networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1790–1799.
[16] Y. Yan, Y. Mao, and B. Li, "SECOND: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, p. 3337, Oct. 2018.
[17] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.
[18] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
[19] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10778–10787.
[20] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "NAS-FPN: Learning scalable feature pyramid architecture for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7029–7038.
[21] C. Yang, Z. Huang, and N. Wang, "QueryDet: Cascaded sparse query for accelerating high-resolution small object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 13658–13667.
[22] Q.-L. Zhang and Y.-B. Yang, "SA-net: Shuffle attention for deep convolutional neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 2235–2239.
[23] S. Liu, D. Huang, and Y. Wang, "Learning spatial fusion for single-shot object detection," 2019, *arXiv:1911.09516*.
[24] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
[25] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6154–6162.
[26] S. Qiao, L.-C. Chen, and A. Yuille, "DetectoRS: Detecting objects with recursive feature pyramid and switchable atrous convolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10208–10219.
[27] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.
[28] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.
[29] C.-Y. Wang and H.-Y. M. Liao, "YOLOv9: Learning what you want to learn using programmable gradient information," 2024, *arXiv:2402.13616*.
[30] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2999–3007.
[31] G. Jocher, A. Chaurasia, A. Stoken, J. Borovec, NanoCode012, Y. Kwon, and K. Michael, "ultralytics/yolov5: v7.0—YOLOv5 SOTA realtime instance segmentation," Nov. 2022, doi: 10.5281/zenodo.7347926.
[32] G. Jocher, A. Chaurasia, and J. Qiu. (Jan. 2023). *Ultralytics YOLO*. [Online]. Available: https://github.com/ultralytics/ultralytics
[33] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6568–6577.
[34] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9626–9635.
[35] C. Zhu, Y. He, and M. Savvides, "Feature selective anchor-free module for single-shot object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 840–849.
[36] 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 *Computer Vision—ECCV 2022*. Cham, Switzerland: Springer, 2022, pp. 526–543.
[37] Y. Zhu, Q. Zhou, N. Liu, Z. Xu, Z. Ou, X. Mou, and J. Tang, "ScaleKD: Distilling scale-aware knowledge in small object detector," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2023, pp. 19723–19733.
[38] Y. Yang, X. Sun, W. Diao, H. Li, Y. Wu, X. Li, and K. Fu, "Adaptive knowledge distillation for lightweight remote sensing object detectors op timizing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, 2022.
[39] L. Song, Y. Li, Z. Jiang, Z. Li, H. Sun, J. Sun, and N. Zheng, "Fine-grained dynamic head for object detection," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.* Red Hook, NY, USA: Curran Associates, 2020, pp. 1–11.
[40] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 13435–13444.
[41] Q. Zhao, T. Sheng, Y. Wang, Z. Tang, Y. Chen, L. Cai, and H. Ling, "M2Det: A single-shot object detector based on multi-level feature pyramid network," in *Proc. AAAI Conf. Art. Intel.*, vol. 33, Jan. 2019, pp. 9259–9266.
[42] Y. Zhang, C. Wu, T. Zhang, Y. Liu, and Y. Zheng, "Self-attention guidance and multiscale feature fusion-based UAV image object detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, 2023.
[43] X. Dai, Y. Chen, B. Xiao, D. Chen, M. Liu, L. Yuan, and L. Zhang, "Dynamic head: Unifying object detection heads with attentions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7369–7378.
[44] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020.
[45] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13708–13717.
[46] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11531–11539.
[47] 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 *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2023, pp. 1–5.
[48] M.-H. Guo, Z.-N. Liu, T.-J. Mu, and S.-M. Hu, "Beyond self-attention: External attention using two linear layers for visual tasks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 5436–5447, May 2023.
[49] J. Zhang, A. Ding, G. Li, L. Zhang, and D. Zeng, "A pyramid attention network with edge information injection for remote-sensing object detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, 2023.
[50] K. Jiang, J. Liu, W. Zhang, F. Liu, and L. Xiao, "MANet: An efficient multidimensional attention-aggregated network for remote sensing image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–18, 2023.
[51] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, vol. 16, 2016, pp. 770–778.

[52] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Computer Vision—ECCV*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham, Switzerland: Springer, 2018, pp. 122–138.
[53] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
[54] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Computer Vision—ECCV*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham, Switzerland: Springer, 2014, pp. 740–755.
[55] D. Du, P. Zhu, L. Wen, X. Bian, H. Lin, Q. Hu, T. Peng, J. Zheng, X. Wang, and Y. Zhang, "VisDrone-DET2019: The vision meets drone object detection in image challenge results," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 213–226.
[56] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick. (2019). *Detectron2*. [Online]. Available: https://github.com/facebookresearch/detectron2
[57] 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," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, Dec. 2020, pp. 21002–21012.



JITING ZHOU received the Ph.D. degree from Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences. She is currently a Lecturer with Shanghai University. Her research interests include computer vision, object detection, and information security.

QIAN XU is currently pursuing the M.S. degree in digital media creative engineering with Shanghai Film Academy, Shanghai University, China. Her research interests include deep learning, computer vision, and object detection.

XINRUI ZHAO is currently pursuing the master's degree in electronic information with Shanghai Film Academy, Shanghai University, China. Her research interests include computer vision, image forensics, and the applications related to copyright protection.

ZHIHAO ZHOU is currently pursuing the M.Sc. degree with Shanghai Film Academy, Shanghai University, China. His research interests include computer vision, open world object detection, and open vocabulary object detection.

PU ZHANG is currently pursuing the M.Sc. degree with Shanghai Film Academy, Shanghai University, China. His research interests include computer vision and adversarial attack.