sciendo

JAISCR, 2024, Vol. 14, No. 3, pp. 251 – 265
10.2478/jaiscr-2024-0014

SINEXTNET: A NEW SMALL OBJECT DETECTION MODEL FOR AERIAL IMAGES BASED ON PP-YOLOE

Wenkang Zhang, Zhiyong Hong*, Liping Xiong, Zhiqiang Zeng, Zhishun Cai, Kunyu Tan

School of Electronics and Information Engineering, Wuyi University, Jiangmen, Guangdong, China

**E-mail: hongmr@163.com*

Submitted: 23rd January 2024; Accepted: 2nd May 2024

Abstract

Although object detection has achieved great success in the field of computer vision in the past few years, the performance of detecting small objects has not yet achieved ideal results. For instance, UAV aerial photography object detection plays an important role in traffic monitoring and other fields, but it faces some great challenges. The objects in aerial images are mainly small objects, the resolution of whom is low and the feature expression ability of whom is very weak. Information will be lost in high-dimensional feature maps, and this information is very important for the classification and positioning of small objects. The most common way to improve small object detection accuracy is to use high-resolution images, but this incurs additional computational costs. To address the above-mentioned problems, this article proposes a new model SINextNet, which uses a new dilated convolution module SINext block. This module is based on depth-separable convolution, and can improve the receptive field of the model. While extracting small object features, it can combine small object features with background information, greatly improving the feature expression ability of small objects. The experimental results indicate that the method proposed in this paper can achieve advanced performance across multiple aerial datasets.

Keywords: object detection, UAV aerial photography, small object

1 Introduction

Object detection aims to classify and locate objects of interest in images, and is one of the most fundamental and challenging problems in the field of computer vision. Early object detection methods mainly detect and identify objects through the sliding window algorithm [1] and HOG algorithm [2]. With the emergence of deep convolutional networks [3, 4], deep learning has excellent performance and powerful feature extraction capabilities in large-scale data processing, and there are more and more object detection methods based on deep neural networks proposed [5, 6, 7, 8, 9, 10], and object detection has entered the era of deep learning. Object detection techniques based on deep learning have been widely used in many areas, such as autonomous driving, face recognition, robot navigation, etc.

Existing popular object detection methods primarily focus on detecting objects of standard sizes, but, aerial object detection refers to identify and detect the small objects on the ground from the air, typically small objects such as vehicles, pedestrians, and animals. Aerial object detection tech-

niques hold significant importance in various scenarios, for example, disaster monitoring, forestry, power, environmental protection, drone scene analysis [11], pedestrian detection [12], and traffic [13]. However, In existing object detection methods the accuracy of small object detection is significantly lower compared with the detection of normal-sized objects and large-sized objects. In the coco [14] evaluation index, the pixel area occupied by small objects is less than 32×32, the pixel area occupied by normal objects is greater than 32×32 but less than 96×96, and the pixel area occupied by large objects is greater than 96×96. Taking PP-YOLOE [15] as an example, it is one of the most advanced models for object detection. Its accuracy in detecting normal-sized and large-sized objects on the COCO2017 dataset test set is 53.9% AP and 66.2% AP respectively; while the accuracy of detecting small-sized objects is only 31.8% AP. Another small object detection model QueryDet [16] uses the cascade query method to detect small objects in images. In this model, the accuracy on the aerial small object dataset VisDrone2019 is much lower, only 28.35% AP. Therefore, compared with the detection accuracy of ordinary or large size objects, the accuracy of small object detection lags far behind.

The detection effect of small objects lags far behind the detection effect of normal and large-sized objects for the following reasons or problems: (1) small objects in images often exhibit lower resolution and weaker feature representation. During the downsampling process in neural networks, when mapping the features of these small objects to higher dimensions, the critical features of the small objects are susceptible to loss [17]; (2) small objects lack sufficient contextual information for accurate identification [18]; (3) detecting the precise location of small objects is more challenging compared with larger objects due to the increased sensitivity of the Intersection over Union (IoU) metric to even minor perturbations in the bounding box [16]; (4) the characteristics of small objects in images are easily affected by noise [19]. Because the image quality captured by the UAV from the air during flight is poor, the objects in the image are relatively small, and there are a large number of occluded objects. Therefore UAV aerial photography object detection also has the above four problems.

In recent years, a variety of solutions have been proposed to solve the existing problems of small object detection. On one hand, using high-resolution images as input images [20], this method can make use of the features of small objects occupying more pixels, making it easier to extract and retain the features of small objects. However, with the input of high-resolution images, the cost of the calculation of the network will increase significantly. On the other hand, some work [21, 22, 23, 24] have proposed methods of multi-scale feature map reuse to construct feature pyramids of different scales. By integrating features from multiple scales, these approaches enhance the accuracy of detecting small objects and compensate for their low resolution and weak feature representation. Despite this, the amount of calculation in the networks is still huge, and the characteristics of small objects cannot be well preserved.

In order to address the above problems, inspired by InceptionNeXt [25], this paper proposes SINext block, which is a small object feature extractor based on depth-separable convolution. Then based on SINext block and PP-YOLOE, a new network SINextNet is proposed to detect small objects in UAV aerial photography. SINextNet uses the PP-YOLOE architecture as the backbone architecture, combined it with the feature extraction module SINext block. This module mainly addresses two key issues in feature extraction of small objects: (1) small objects occupy very few pixels in the image, and when their features are mapped to high dimensions by the neural network, crucial information is at risk of being lost; (2) the weak feature representation of small objects often necessitates their combination with surrounding environmental information for more effective extraction by the model. In the SINext block, skip connections are added before and after feature extraction, and between different branches of depth-wise separable convolutions. This method can effectively retain the characteristics of small objects in high-dimensional feature maps. At the same time, dilated convolution is added to the SINext block, and can effectively increase the receptive field of the convolution kernel and better aggregate the multi-scale context information of the feature map, making the features of small objects easier identify and extract.

This paper evaluates SINextNet on the three UAV aerial photography datasets, they are Vsidrone2019 [26], UAVDT [27], and RAIVD [28]. These datasets are composed of UAV aerial images, where most of the objects are small objects. The main categories include cars, pedestrians, bicycles, etc. The model SINextNet proposed here reaches the advanced level on these datasets.

2 Related Work

In this section, the current mainstream object detection algorithms will be reviewed, including two-stage detectors [22, 29, 30, 31], one-stage detectors [7, 9, 23, 32, 33], small object detection algorithms [7, 16]. The one-stage detectors include anchor-based detectors and anchor-free detectors.

In the two-stage detectors [34], such as Faster R-CNN [5], R-FCN [35], candidate frames that may contain the objects are first extracted on the feature map, and then the objects are classified. Their detection results are often more accurate, but their real-time performance is poor and they require more computing resources. In order to solve these problems, one-stage anchor-based detectors are proposed. Compared with the two-stage detectors, the one-stage anchor-based detectors directly perform the object detection task in a single neural network, without dividing it into two stages. There are many one-stage anchor-based detectors [32, 36, 37], among them, YOLO [32] is the most representative. YOLO is an one-stage anchor-based detector that applies a convolutional neural network to the entire image. The network divides the feature map into multiple grids, and in each grid there is a set of labels generated by a clustering algorithm. Anchors of multiple sizes are responsible for predicting the location information and category information of the objects in each grid. Compared with the two-stage detectors, the detection speed of the one-stage anchor-based detectors has been greatly improved, but the detection accuracy has decreased, especially for those small objects.

With the development of one-stage detectors, one-stage anchor-free detectors [38, 39, 40, 41] have attracted people's attention. One stage anchor-free detectors require less computing resources to achieve the accuracy of anchor-based detectors. Compared with traditional one-stage detectors, one-stage anchor-free detectors do not have predefined anchors. Each feature map grid only outputs a set of position information and category information. The algorithm structure of one-stage anchor-free detectors is relatively simple and does not require additional design parameters, reducing the burden of manual tuning. Since it does not require a predefined anchor, it is more flexible and can adapt to various object sizes and shapes, especially for the detection of small and irregular objects. Excellent anchor-free detectors in recent years include YOLOX [42] and PP-YOLOE [15]. For instance, PP-YOLOE is the current mainstream one-stage anchor-free object detection model, its backbone network is CSPRepResNet. In PP-YOLOE, ESE attention mechanism module [43] is added to the backbone network to reduce the loss of feature map channel information. The Feature Pyramid Network (FPN) [6] built by RepResBlock and CSPRepResStage is used in Neck to fuse low-dimensional features and high-dimensional features. PP-YOLOE uses the ET-Head detection head, and different network branches in the detection head output category information and location information respectively. At the same time, in PP-YOLOE, the TAL [44] label allocation strategy is used to solve the problem of uneven distribution of positive and negative samples when training the network.

By employing one-stage object detection algorithms or two-stage detection methods, satisfactory results can be achieved when detecting large or medium-sized objects. However, when it comes to detecting small objects, the performance tends to be very poor due to the low feature resolution of small objects in the image. In response to the problem of low resolution of small objects in images, people have proposed a variety of methods to solve this problem: (1) improve the resolution of training network images [6, 20]; (2) use strong data enhancement strategies [23, 45]; (3) combine small objects in the image with background information [46, 47]. [3] proposes RetinaNet, whose loss function is a simple but very practical Focal Loss function, which solves the problem of extreme imbalance between positive and negative samples in object detection. Balancing the distribution of positive and negative samples helps ensure that the model pays full attention to small objects during training, improves the detection accuracy of small ob-

jects, and reduces the risk of missed detections and false detections. QueryDet [16] is one of the current mainstream small object detection networks. In QueryDet, a novel cascade query feature pyramid structure is proposed: it first predicts the coarse locations of small objects on low-resolution features and then computes the accurate detection results using high-resolution features sparsely guided by those coarse positions. This approach enables the extraction of critical features from small objects while avoiding unnecessary computations on background regions. In QueryDet, cascade queries require careful tuning of thresholds and parameters at each level to ensure superior performance across the entire cascade, requiring more training time and experience because tuning parameters at one level will affect the performance of other levels.

3 Method

In this section, the main model SINextNet to detect small objects in UAV aerial images will be introduced in detail. This model is mainly built on the architecture of PP-YOLOE, replacing PP-YOLOE's feature extraction module by the new SINext block. Compared with PP-YOLOE, the SINext block feature extraction module pays more attention to small object features in the image. In the following, we introduce the structure of SINextNet, explain SINext block, and then present the loss function of this model.

3.1 SINextNet model

The SINextNet network architecture is illustrated in Figure 1. It is constructed based on the excellent model architecture of PP-YOLOE and the proposed SINext block in this paper. By integrating the SINext block feature module into the PP-YOLOE framework, SINextNet can improve capabilities in extracting features of small objects within images.

SINextNet's backbone network SINextResNet is a ResNet structure consisting of the Stem Layer and four SINextStages. The Stem Layer, responsible for initially extracting features in images, is composed of three 3×3 convolution operations. However, the design of SINextStages introduces a more intricate composition. Each SINextStage is primarily composed of 1×1 convolution, the SINext block feature extraction module, and the ESE attention mechanism [47]. Among them, SINext block is the main feature extraction module in SINextResNet, responsible for extracting features in images; the ESE module uses a global average pooling layer and a fully connected layer to solve the problem of channel information loss caused by the reduction of channel dimensions. SINextNet retains the Neck structure of PP-YOLOE. Information of different scales are aggregating by using SINext block and SINextStage in the Neck structure. Moreover, SINextNet uses PP-YOLOE Head as the detection head, which is mainly composed of ESE module and 3×3 convolution. Different network branches output category information and position information respectively.

3.2 SINext block

The SINext block proposed in this paper is a feature extraction module based on the InceptionNeXt block [25], shown in Figure 2.

Firstly we recall the InceptionNeXt block.The InceptionNeXt block is the primary feature extraction module of the classification network InceptionNeXt [25]. It utilizes depth-wise separable convolutions with large kernel sizes to extract features from the image. As shown in Figure 2, the InceptionNeXt block first groups the input feature map along the channel dimension, and each group of features is processed through depth-wise separable convolutions with different kernel sizes, resulting in respective output feature maps. After the convolution is completed, the output features from each group are concatenated along the channel dimension, yielding an output feature map with the same shape as the input feature map. Subsequently, this output feature map undergoes a Norm layer, where Batch normalization is applied to normalize the features in batches, thereby enhancing the stability and convergence speed of the model. After normalization, the neurons are activated through the MLP layer, and the final output is obtained after residual connection between the activated output and the original input features.

Now we introduce SINext block in detail. The structure of SINext block as depicted in Figure 3, serves as the primary feature extraction module in SINextNet, effectively capturing features of small

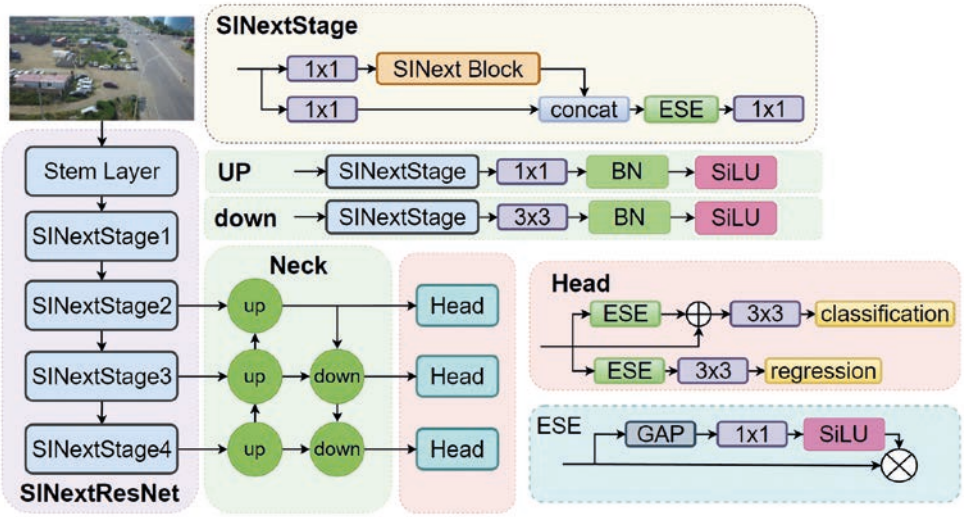


Figure 1. SINextNet network architecture. The backbone network of SINextNet is SINextResNet composed of four SINextStages. The detection head of this article uses the detection head of PP-YOLOE (ET-head).

objects in images. In the InceptionNeXt block, large-kernel depthwise separable convolutions are employed, effectively increasing the model's receptive field. Inspired by this, SINext block is proposed, and its overall structure is illustrated in Figure 3. To enable the model to capture more features of small objects in low-resolution images, in the SINext block, dilated convolution [46] is introduced into the InceptionNeXt block's $\text{DWConv}_{3\times3}$. Additionally, shortcut connections [48] are added between parallel convolutions and both before and after convolution operations. Dilated convolution increases the size of the convolutional kernel by inserting gaps between the convolution kernel, expanding the range of feature extraction to make the model pay more attention to the background information of the object in the image. Adding shortcut connections between parallel convolutions allows the information from feature maps in different groups to interact, increasing the receptive field between feature map channels. These approaches do not introduce additional parameters and enable better integration of features related to small objects and background information, thus improving the accuracy of small object recognition.

Illustrated in Figure 3. The SINext block first groups the input feature map along the channel dimension, and each group of feature maps undergoes different convolution operations to obtain output feature maps. After completing the convolution operations, their output feature maps are concatenated along the channel dimension. Next, a shortcut connection is established with the input, and after passing through the Norm layer and the MLP layer, a final shortcut connection is made with the original input to obtain the ultimate output. The input X is divided into four groups of features, as shown in Equation 1, resulting in $X_{c1}, X_{c2}, X_{c3}, X_{c4}$.

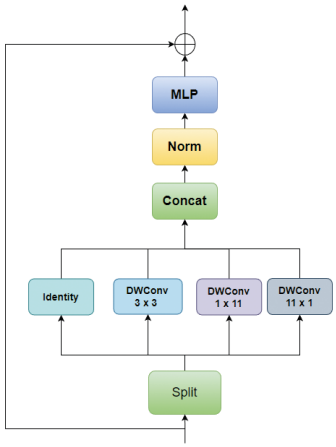


Figure 2. Illustration of the InceptionNext block [25].

$$X_{c1}, X_{c2}, X_{c3}, X_{c4} = Split(X) \tag{1}$$

Then features X_{c1}, X_{c2}, X_{c3} undergo $DWConv$ operations separately. Here, in Equation 2-4, $DWConv_{(n\times n, d=q)}$ denotes a depthwise separable

convolution with a convolution kernel size of $n \times n$ and an expansion ratio of q. After X_{c1} undergoes $DWConv_{(3\times3,d=3)}$, it yields $X_{c1}^{'}$, when added to X_{c2} and subjected to $DWConv_{1\times11}$, produces $X_{c2}^{'}$. Similarly, $X_{c2}^{'}$ added to X_{c3} and subjected to $DWConv_{11\times1}$ results in $X_{c3}^{'}$. After obtaining $X_{c1}^{'}, X_{c2}^{'}, X_{c3}^{'}$, the Concat function in Equation 5 concatenates these features along the channel dimension with X_{c4} to obtain X'. These processes can be expressed as:

$$X_{c1}^{'} = DWConv_{(3\times3,d=3)}(X_{c1}) \tag{2}$$

$$X_{c2}^{'} = DWConv_{(1\times11)}(X_{c2} + X_{c1}^{'}) \tag{3}$$

$$X_{c3}^{'} = DWConv_{(11\times1)}(X_{c3} + X_{c2}^{'}) \tag{4}$$

$$X^{'} = Concat(X_{c1}^{'}, X_{c2}^{'}, X_{c3}^{'}, X_{c4}) \tag{5}$$

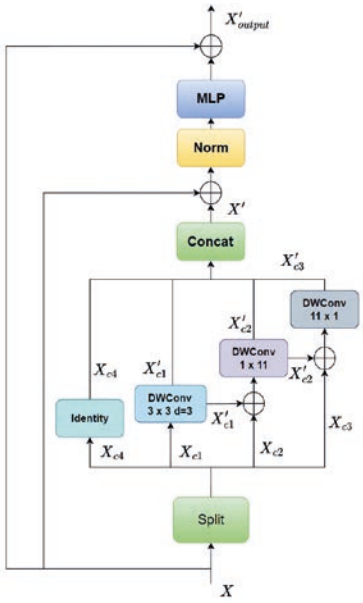


Figure 3. Illustration of the SINext block.

After establishing a shortcut connection between X' and X, the combined features undergo normalization through the Norm layer and pass through the MLP layer. The output of the MLP layer, along with a shortcut connection to X, results in the final output feature denoted as X'_{output}.

$$X'_{output} = MLP(Norm(X' + X)) + X \tag{6}$$

As shown in Figure 2 and Figure 3, compared with the InceptionNeXt block, the SINext block incorporates dilated convolutions in feature extraction and includes shortcut connections between parallel convolutions. These two approaches enable the model to focus more on the features of small objects in the image, thereby enhancing the accuracy of the model in detecting small objects.

3.3 Loss functions of the SINextNet

SINextNet shares the same loss function as PP-YOLOE, comprising of three parts, they are classification loss varifocal loss (VFL) [49], the bounding box prediction loss GIoU [50], and the bounding box localization loss distribution focal loss (DFL) [51]. VFL uses the object category score to weight the positive sample loss, making the model pay more attention to high-quality samples. DFL uses a generalized distribution method to detect high-quality bounding boxes. In all SINextNet loss function is illustrated in Equation 7,

$$Loss = \frac{\alpha \cdot loss_{VFL} + \beta \cdot loss_{GIoU} + \gamma \cdot loss_{DFL}}{\sum_{i}^{N_{pos}} \hat{t}} \tag{7}$$

Here, N_{pos} represents the number of detected objects in the grid cells of the feature pyramid, and $\hat{t}$ is the normalized value of the combined standardization of classification scores and IoU. The weights α, β, γ are coefficients for three different loss terms.

In the following, three loss functions in Equation 7 and $\hat{t}$ are explained in sequence. Firstly, the $loss_{VFL}$ is the classification loss for SINextNet, as shown in Equation 8.

$$loss_{VFL}(p,q) = \begin{cases} -q(q\log(p) \\ +(1-q)\log(1-p)) & q > 0 \\ \\ -\alpha_1 p^{\gamma_1}\log(1-p) & q = 0 \end{cases} \tag{8}$$

Here, p represents the predicted localization score, and q is the object score. Detection of a object occurs when q is greater than 0, and the loss is calculated based on the localization score and object score. When q equals 0, indicating the detection of background, the coefficients α_1 and γ_1 are used to scale the loss, reducing the contribution of the negative samples (q=0) to the loss without affecting the loss of positive samples. This approach addresses the issue of class imbalance between positive and negative samples.

Secondly, the $loss_{GIoU}$ is the bounding box prediction loss for SINextNet, as illustrated in Equation 9.

$$loss_{GIoU} = 1 - IoU + \frac{|C - (A \bigcup B)|}{|C|} \tag{9}$$

Here, A represents the predicted box, B is the label, $A \bigcup B$ is the area of the union of the predicted box and the label, and C is the area of the minimum rectangle that encloses both A and B. IoU is the ratio of the intersection area of the predicted box and the true box to the union area of the intersection and union. This paper utilizes the $loss_{GIoU}$ to compute the loss for predicting bounding boxes. The GIoU loss function takes into account the contributions of both overlapping and non-overlapping regions between the object boxes, making it more adaptable to variations in the shape and scale of the objects.

Thirdly, the $loss_{DFL}$ is the bounding box localization loss for SINextNet, as expressed in Equation 10.

$$\begin{aligned} loss_{DFL}(S_i, S_{i+1}) = -((y_{i+1} - y) log(S_i) \\ +(y - y_i) log(S_{i+1})) \end{aligned} \tag{10}$$

Here, y denotes the position coordinates of the label, y_{i+1} is the coordinate prediction value closest to y that is greater than y, and y_i is the coordinate prediction value closest to y that is smaller than y. S_i denotes the softmax operation applied to y_i. By introducing y_i and y_{i+1}, the network can more quickly focus on values near y, maximizing the probability of utilizing them effectively.

Lastly, the normalized value $\hat{t}$ corresponds to the standardized value of t, where t is a higher-order combination of classification scores and IoU, as shown in Equation 11.

$$t = s^{\alpha_2} \times u^{\beta_2} \tag{11}$$

Here, s and u represent the classification score and IoU value, respectively, while α_2 and β_2 are weighting coefficients. This higher-order combination guides the network to dynamically focus on high-quality predicted bounding boxes.

4 Experiment

The experiments in this paper have conducted on three popular aerial datasets: Vsidrone2019 dataset [26], UAVDT dataset [27], and RAIVD dataset [28]. In this Section, we will introduce the datasets used in the experiments, experimental details and ablation experiments in detail, and verifies the generalization ability of the model.

4.1 Dataset introduction

Now we introduce three prevalent aerial datasets: the Vsidrone2019 dataset, UAVDT dataset, and RAIVD dataset. The basic information for these three datasets is presented in Table 1. In this table, "size" refers to the dimensions of the images in the dataset; while "train", "val" and "test" represent the number of images in the training, validation, and test sets, respectively; "class" indicates the number of categories in the dataset. In the following, each dataset is introduced in detail.

Table 1. Basic information about three datasets

Dataset	size	train	val	test	class
Vsidrone2019	640×640	6471	548	1610	10
UAVDT	1080×540	20686	3458	16591	3
RAIVD	1920×1080	9286	1547	4641	4

Firstly, the VisDrone2019 dataset, collected by the AISKYEYE team at Tianjin University's Machine Learning and Data Mining Laboratory, consists of aerial images. It includes 10 prediction categories, they are pedestrians, people, cars, trucks, buses, vans, motorcycles, bicycles, awning tricycles, and tricycles. VisDrone2019 is specifically designed for detecting objects in images captured by drones, with a focus on small objects dominating the scale distribution.

Secondly, the UAVDT dataset, curated by teams from the University of Chinese Academy of Sciences and Harbin Institute of Technology, comprises of 100 aerial video sequences. The captured scenes in these video sequences include daytime, nighttime, and foggy weather conditions, spanning locations such as squares, main roads, toll stations, highways, and intersections. This dataset encompasses three prediction categories: car, truck, and bus, with a total of 840,000 annotated bounding boxes in PASCAL VOC format.

Lastly, the RAIVD dataset, collected by the team at Universidad Europea de Madrid, consists of aerial images captured by drones over Spanish

roundabouts. The dataset includes images taken above roundabouts with varying traffic flows. The primary categories are car, truck, cycle, and bus, and the annotations follow the PASCAL VOC format too.

4.2 Implementation details and evaluation metrics

This paper implements its models based on the PyTorch and MMYOLO frameworks. All models are trained on an NVIDIA 3090 GPU. The AdamW optimizer is employed with a batch size of 16, an initial learning rate of 0.001, and a weight decay of 0.05. Multi-scale images are used during training, with input image sizes randomly selected from 320×320 to 768×768, with a stride of 32. The values of the hyperparameters α, β and γ in the loss function (Equation 7) are 1, 2, and 0.25 respectively. Experiments are conducted on the VisDrone2019 dataset, UAVDT dataset, and RAIVD dataset using the aforementioned configurations. The training time of PP-YOLOE-l and SINextNet for 300 epochs on the VisDrone2019 dataset is approximately 30 hours.

In these experiments, the following evaluation metrics are considered, they are AP, AP_{50}, AP^{val}, AP^{test}, AP_S, AP_M, AP_L. Among them, AP is the average precision for each category; AP_{50} is the average precision computed for bounding boxes with an overlap exceeding 50% with the ground truth bounding boxes; AP^{val} represents the average precision on the validation set; AP^{test} corresponds to the average precision on the test set; and AP_S, AP_M, AP_L denote the precision for detecting small, medium, and large objects, respectively. Here small objects occupy a pixel area smaller than 32×32, medium objects have a pixel area larger than 32×32 but smaller than 96×96, and large objects occupy a pixel area larger than 96×96.

4.3 Comparsion with other mainstream detectors

In order to evaluate the small object detection performance of SINextNet, we compare it with PP-YOLOE-1 conducted on the VisDrone2019 dataset. The VisDrone2019 dataset is an aerial dataset, where the objects in the images are primarily small objects. The detection performance of models is evaluated by comparing their accuracy on this dataset. Currently, PP-YOLOE-1 is an important detection model for detecting small object model, which represents a model variant of PP-YOLOE, here, -l denotes a large-sized model with additional feature extraction modules and more output feature map channels. The training set is consisted of 6471 images from the VisDrone2019 dataset, the validation set is comprised of 548 images, and the testing set includes 1610 images. The models are trained for a total of 300 epochs. The experimental results, as shown in Table 2, reveal that SINextNet outperforms PP-YOLOE-l on the VisDrone2019 dataset. Specifically, SINextNet exhibits a 0.7% improvement in AP^{val}, a 0.8% improvement in AP^{val}_{50}, a 1.2% improvement in AP^{test}, and a 1.6% improvement in AP^{test}_{50}. This indicates that SINextNet demonstrates superior generalization on the VisDrone2019 dataset while reducing the parameter count by 1.8 million compared with PP-YOLOE-l.

Table 3 presents the detection results of SINextNet and PP-YOLOE-l on small, medium, and large objects in the VisDrone2019 test set. Among these, recall that small objects refer to those occupying a pixel area smaller than 32×32 in the image, medium objects occupy a pixel area greater than 32×32 but smaller than 96×96, and large objects occupy a pixel area greater than 96×96. As shown in Table 3, SINextNet outperforms PP-YOLOE-l with a 0.9% improvement in AP^{test}_S, a 0.9% improvement in AP^{test}_M, and a 1.8% improvement in AP^{test}_L. Therefore SINextNet demonstrates superior detection performance for objects of varying scales compared with PP-YOLOE-l.

Table 3. Detection results for different-sized objects on the VisDrone2019 test set for both PP-YOLOE-l and SINextNet.

Model	AP^{test}_S (%)	AP^{test}_M(%)	AP^{test}_L(%)
PP-YOLOE-l	13.7	34.6	40.2
SINextNet	**14.6**	**35.5**	**42.0**

In order to better assess the model's detection capability, this paper visualizes the output results of SINextNet and PP-YOLOE-l, as depicted in Figure 4. The first and second rows represent the results of visualizing different images. The first column shows the original images, the second col-

Table 2. Comparison between SINextNet and PP-YOLOE-l on the VisDrone2019 validation set and test set. FPS indicates the model's inference speed on one NVIDIA 3090 GPU.

Model	size	$AP^{val}(\%)$	$AP^{val}_{50}(\%)$	$AP^{test}(\%)$	$AP^{test}_{50}(\%)$	Parameters(M)	FPS(ms)	GFLOPs
PP-YOLOE-l	640	29.2	47.3	23.5	39.1	51.3	**42.2**	53.6
SINextNet	640	**29.9**	**48.1**	**24.7**	**40.7**	**49.5**	39.4	**51.8**

umn shows the visualization results of PP-YOLOE-l, and the third column shows the visualization results of SINextNet. As shown by the arrow in the first row, second column image, PP-YOLOE-l incorrectly identified the obstacle on the roadside as a truck, while SINextNet did not exhibit this issue. As shown by the arrow in the second row, second column image, PP-YOLOE-l failed to detect the occluded objects in the center of the image, whereas SINextNet successfully detected the occluded objects in the middle of the image. The visualization results indicate that SINextNet excels in recognizing partially obscured small objects.

Further, in Table 4, this paper compares SINextNet with two mainstream small object detection networks on the evaluation indicators AP and AP_{50}, they are QueryDet [16] and RetinaNet [7]. AP is the average accuracy of each category; AP_{50} is the average accuracy where the predicted bounding box overlaps with the real bounding box by more than 50%; these two evaluation indicators are more effective Demonstrates the model's ability to detect small objects. Trained on images of size 640×640, SINextNet achieves a 3.69% higher AP^{val} than RetinaNet and a 1.55% higher AP than QueryDet.

Table 4. Comparison between SINextNet and mainstream small object detection models on the VisDrone2019 validation set.

Model	size	$AP^{val}(\%)$	$AP^{val}_{50}(\%)$
RetinaNet	640	26.21	44.90
QueryDet	640	28.35	**48.21**
SINextNet	640	**29.90**	48.10

Additionally, Table 5 compares SINextNet with the popular YOLO series models (YOLOv5-l, YOLOv6-l and YOLOX-l) in the evaluation indicators AP_S and AP_L, the purpose of which is to evaluate SINextNet's ability to detect normal-sized objects. All models are trained using the training set from the VisDrone2019 dataset and the results are compared using the validation set. SINextNet outperforms YOLOv6-l by 3.6%, YOLOv5-l by 8.1%, and YOLOX-l by 8.6% in terms of AP^{val}. Although SINextNet's AP^{val}_S improves compared with the other three models, AP^{val}_L experiences a slight decline. This is attributed to YOLOv5 and YOLOv6 being anchor-based detectors, employing clustering algorithms to define default anchor sizes for objects in images. These anchors tend to perform better when detecting large-sized objects.

Table 5. Comparison between SINextNet and other mainstream YOLO models.

Model	size	$AP^{val}_{50}(\%)$	$AP^{val}_S(\%)$	$AP^{val}_L(\%)$	Parameters(M)
YOLOX-l	640	39.5	14.1	40.5	54.1
YOLOv5-l	640	40.0	14.1	**51.2**	**46.6**
YOLOv6-l	640	44.5	17.3	50.2	58.5
SINextNet	640	**48.1**	**20.6**	47.2	49.5

4.4 Ablation experiments

This paper conducted ablation experiments on the feature extraction module of SINextNet, the SINext block, to analyze the impact of each component on detection results, as shown in Table 6. When the shortcut connection between parallel *DWConv* in SINext block is removed, AP^{val} decreases by 0.4, AP^{val}_{50} decreases by 0.3%, AP^{val}_S decreases by 0.3%, and AP^{val}_L decreases by 1.0%. Removing the dilated convolution in $DWConv_{3\times3}$ of SINext block results in a decrease of 0.2 in AP^{val}, 0.1% in AP^{val}_{50}, 0.4% in AP^{val}_S, and 1.3% in AP^{val}_L. Simultaneously removing both shortcut connection and dilated convolution leads to a decrease of 0.4 in AP^{val}, 0.2% in AP^{val}_{50}, 0.4% in AP^{val}_S, and 1.0% in AP^{val}_L.

Experimental results indicate that incorporating shortcut connection and dilated convolution into the module improves the detection performance of the model. Shortcut connection enhances information exchange between different convolutional groups, improving the overall receptive field of the model. Dilated convolution allows the module to extract more contextual information during feature extrac-



Figure 4. Visualization of the output results of PP-YOLOE-l and SINextNet. In the figures, (a) represents the input image, (b) shows the visualization of the output results from PP-YOLOE-l, and (c) displays the visualization of the output results from SINextNet.

tion, enabling better integration of features related to small objects and background information, there by improving the model's accuracy in recognizing small objects. The addition of shortcut connection and dilated convolution does not alter the model's parameter count. In the case of shortcut connection, the feature maps undergo addition, which does not introduce additional parameters. Similarly, dilated convolution inserts fixed values of 0 in the convolution kernel to expand the pixel extraction range, also without increasing extra parameters.

Table 6. Ablation experiments on the SINext block. Here, "s" denotes the shortcut connection between parallel *DWConv* in the SINext block, and "d" represents the dilated convolution in $DWConv_{3\times3}$ of the SINext block.

Model	$AP^{val}(\%)$	$AP^{val}_{50}(\%)$	$AP^{val}_{S}(\%)$	$AP^{val}_{L}(\%)$	Parameters(M)
SINextNet	**29.9**	**48.1**	**20.6**	**47.2**	49.5
w/o s	29.5	47.8	20.3	46.2	49.5
w/o d	29.7	48.0	20.2	45.9	49.5
w/o s and d	29.5	47.9	20.2	46.2	49.5

Ablation experiments are conducted on the dilation factor of dilated convolution, and the experimental results are shown in Table 7. The model achieves the highest accuracy when the dilation is set to 3. When the dilation factor is too small, the model's receptive field becomes smaller, and it fails to effectively aggregate contextual information. Conversely, when the dilation factor is too large, the insertion of too many zero values in the convolution kernel results in fewer extracted pixels from the image, leading to a decrease in accuracy.

Table 7. Perform ablation experiments on the expansion coefficient of $DWConv_{3\times3}$ convolution in SINext block, and use dilation=1, 2, 3, 4, and 5 to perform ablation experiments.

SINext block	$AP^{val}(\%)$	$AP^{val}_{50}(\%)$	$AP^{val}_{S}(\%)$	$AP^{val}_{L}(\%)$	Parameters(M)
dilation=1	29.7	48.2	20.2	45.9	49.5
dilation=2	29.8	48.2	20.5	46.6	49.5
dilation=3	**29.9**	48.1	20.6	**47.2**	49.5
dilation=4	29.8	**48.3**	**20.7**	43.0	49.5
dilation=5	29.7	48.0	20.2	44.6	49.5

In order to validate whether the shortcut connection in the SINext block affects the model's receptive field, a visual analysis is conducted on the output feature maps of SINextNet. This analysis includes the output feature maps of SINextNet with SINext blocks lacking shortcut connections, as well as the overall output feature maps of SINextNet, as shown in Figure 5. Figure 5(a) represents the feature maps outputted by the model with shortcut connection removed in SINext block, while Figure 5(b) displays the feature maps outputted by SINextNet. It can be observed from the figures that SINextNet has a wider receptive field, indicating a stronger focus on each object compared with the model without shortcut connection.

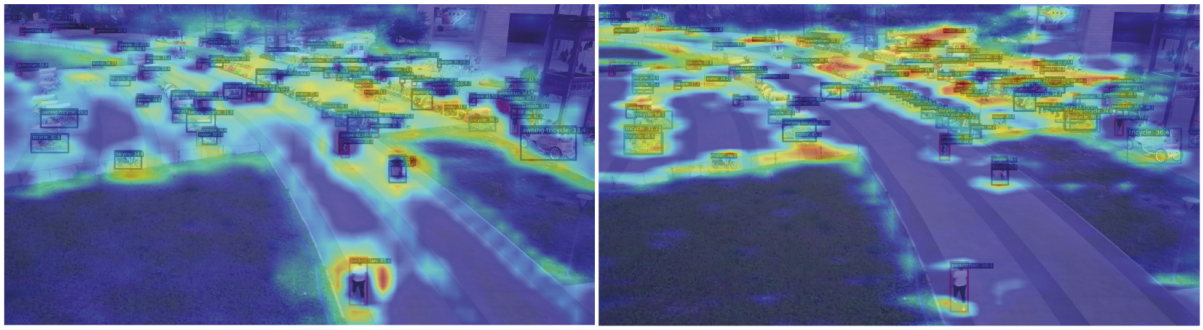


Figure 5. (a) is the feature map output by the SINext block model with the shortcut connection deleted, and (b) is the feature map output by SINextNet.

4.5 Verify the generalization ability of the model

In order to verify the generalization ability of the model, comparative experiments have been conducted on the UAVDT dataset, RAIVD dataset, and the VisDrone2019 dataset with fewer images for both the PP-YOLOE model and SINextNet.

The UAVDT test set is employed to evaluate the generalization capabilities of the PP-YOLOE-l and SINextNet models trained on the VisDrone2019 dataset. Test is conducted using images of size 640×640, and the results are presented in Table 8. On the UAVDT test set, SINextNet shows a 2.1% improvement in AP^{test}, a 3.1% improvement in AP^{test}_{50}, a 0.2% improvement in AP^{test}_{S}, a 3.9% improvement in AP^{test}_{M}, and a 0.6% decrease in AP^{test}_{L} compared with PP-YOLOE-l. SINextNet tends to focus more on features of objects with regular sizes on this dataset, leading to a decrease in accuracy for detecting large-sized objects.

Table 8. Experimental results of PP-YOLOE-l and SINextNet on the UAVDT test set.

Model	$AP^{test}(\%)$	$AP^{test}_{50}(\%)$	$AP^{test}_{S}(\%)$	$AP^{test}_{M}(\%)$	$AP^{test}_{L}(\%)$
PP-YOLOE-l	17.3	32.2	8.2	31.4	**27.6**
SINextNet	**19.4**	**35.3**	**8.4**	**35.3**	27.0

To analyze the training performance of PP-YOLOE-l and SINextNet on another dataset, this paper retrained and tested the models on the aerial dataset RAIVD. PP-YOLOE-l and SINextNet were trained for 120 epochs on this dataset, and the experimental results are presented in Table 9. On the RAIVD test set, SINextNet exhibits a 0.7% improvement in AP^{test}, a 1.0% improvement in AP^{test}_{50}, a 1.9% improvement in AP^{test}_{S}, a 5.4% improvement in AP^{test}_{M}, and a 0.3% improvement in AP^{test}_{L} compared with PP-YOLOE-l. It is evident that SINextNet demonstrates better performance and higher accuracy on the RAIVD dataset.

Table 9. The accuracy of PP-YOLOE-l and SINextNet on the RAIVD test set respectively.

Model	$AP^{test}(\%)$	$AP^{test}_{50}(\%)$	$AP^{test}_{S}(\%)$	$AP^{test}_{M}(\%)$	$AP^{test}_{L}(\%)$
PP-YOLOE-l	73.8	94.4	28.2	63.0	84.6
SINextNet	**74.5**	**95.4**	**30.1**	**68.4**	**84.9**

The scarcity of small object datasets is one of the reasons for the lower accuracy in small object detection compared with regular object detection. To investigate the generalization ability of the models when the dataset is limited, this paper trains models with a smaller number of images to compare their accuracy under conditions of dataset scarcity. Using only 70% of the original images from the VisDrone2019 train set, PP-YOLOE-l and SINextNet were separately trained for 300 epochs. The accuracy on the VisDrone2019 test set is presented in Table 10. SINextNet shows improvements in AP^{test} by 0.6%, AP^{test}_{50} by 1.2%, AP^{test}_{S} by 0.2%, AP^{test}_{M} by 1.1%, and AP^{test}_{L} by 1.7%, compared with SINextNet trained with the original training set, its AP^{test} decreased by 1.5%, AP^{test}_{50} decreased by 2.3%, AP^{test}_{S} decreased by 1.5%, AP^{test}_{M} decreased by 1.9%, and AP^{test}_{L} decreased by 2.8%. Through the analysis of experimental results, it is evident that SINextNet demonstrates better generalization ability when trained on a smaller dataset.

Table 10. The accuracy of PP-YOLOE-l and SINextNet on the RAIVD test set respectively.

Model	$AP^{test}(\%)$	$AP^{test}_{50}(\%)$	$AP^{test}_{S}(\%)$	$AP^{test}_{M}(\%)$	$AP^{test}_{L}(\%)$
PP-YOLOE-l	22.6	37.2	12.9	32.5	37.5
SINextNet	**23.2**	**38.4**	**13.1**	**33.6**	**39.2**

In this Section, PP-YOLOE-l and SINextNet are compared on the UAVDT dataset, RAIVD dataset, and a subset of the VisDrone2019 dataset with fewer images. Through experimentation, it is observed that SINextNet exhibits superior generalization ability on these datasets.

Conclusion

In this paper, inspired by the InceptionNext block feature extraction module, a new feature extraction module SINext block is proposed. The SINext block extends the InceptionNext block by incorporating dilated convolutions and shortcut connections. By integrating the architecture of the PP-YOLOE model and the SINext block, we propose a new network SINextNet to more accurately detect small objects in aerial images. Experimental results indicate that this network significantly improves the model's ability to extract features from small objects, enabling the model to focus more on the characteristics of small objects in images and thereby enhance the accuracy of small object detection. We have found in experiments that although compared with those state of the art methods, e.g. PP-YOLOE, the model has achieved good performance in detecting small objects, there is a slight decrease in accuracy when detecting large objects. Our next work will explore how to solve this problem. Finally, we hope that our work can provide inspiration to researchers engaged in area of small object detection.

References

[1] Viola, P., Jones, M., Rapid object detection using a boosted cascade of simple features., Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition, 2001, 511-518.

[2] Dalal, N., Triggs, B., Histograms of oriented gradients for human detection., IEEE computer society conference on computer vision and pattern recognition, 2005, 886-893.

[3] He, K., Zhang, X., Ren, S., Sun, J., Deep residual learning for image recognition., Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, 770-778.

[4] Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K., Aggregated residual transformations for deep neural networks., Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, 1492-1500.

[5] Ren, S., He, K., Girshick, R., Sun, J., Faster r-cnn: Towards real-time object detection with region proposal networks., Advances in neural information processing systems, 2015.

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

[7] Lin, T. Y., Goyal, P., Girshick, R., He, K., Dollár, P., Focal loss for dense object detection., Proceedings of the IEEE international conference on computer vision, 2017, 2980-2988.

[8] Tian, Z., Shen, C., Chen, H., He, T., FCOS: A simple and strong anchor-free object detector., IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 1922-1933.

[9] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., End-to-end object detection with transformers., European conference on computer vision, 2020, 213-229.

[10] Tao, M., Li, X., Ota, K., Dong, M., Single-Cell Multiuser Computation Offloading in Dynamic Pricing-Aided Mobile Edge Computing., IEEE Transactions on Computational Social Systems, 2023.

[11] Cheng, G., Han, J., A survey on object detection in optical remote sensing images., ISPRS journal of photogrammetry and remote sensing, 2016, 11-28.

[12] Geronimo, D., Lopez, A. M., Sappa, A. D., Graf, T., Survey of pedestrian detection for advanced driver assistance systems., IEEE transactions on pattern analysis and machine intelligence, 2009, 1239-1258.

[13] Jensen, M. B., Philipsen, M. P., Møgelmose, A., Moeslund, T. B., Trivedi, M., Vision for looking at traffic lights: Issues, survey, and perspectives., IEEE transactions on intelligent transportation systems, 2016, 1800-1815.

[14] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., Microsoft coco: Common objects in context., Proceedings of the European conference on computer vision (ECCV), 2014, 740-755.

[15] Xu, S., Wang, X., Lv, W., Chang, Q., Cui, C., Deng, K., Lai, B., PP-YOLOE: An evolved version of YOLO., arXiv preprint arXiv:2203.16250.

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

[17] He, K., Zhang, X., Ren, S., Sun, J., Deep residual learning for image recognition., Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, 770-778.

[18] Li, Y., Chen, Y., Wang, N., Zhang, Z., Scale-aware trident networks for object detection., Proceedings of the IEEE/CVF international conference on computer vision, 2019, 6054-6063.

[19] Chen, L., Zheng, H., Yan, Z., Li, Y., Discriminative region mining for object detection., IEEE Transactions on Multimedia, 2020, 4297-4310.

[20] Li, J., Liang, X., Wei, Y., Xu, T., Feng, J., Yan, S., Perceptual generative adversarial networks for small object detection., Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, 1222-1230.

[21] Cai, Z., Fan, Q., Feris, R. S., Vasconcelos, N., A unified multi-scale deep convolutional neural network for fast object detection., Proceedings of the European conference on computer vision (ECCV), 2016, 354-370.

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

[23] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., Berg, A. C., Ssd: Single shot multibox detector., Proceedings of the European conference on computer vision (ECCV), 2016, 21-37.

[24] Lou, H., Duan, X., Guo, J., Liu, H., Gu, J., Bi, L., Chen, H., DC-YOLOv8: small-size object detection algorithm based on camera sensor., Electronics, 12(10), 2323.

[25] Yu, W., Zhou, P., Yan, S., Wang, X., Inceptionnext: When inception meets convnext., arXiv preprint arXiv:2303.16900.

[26] Zhu, P., Wen, L., Du, D., Bian, X., Ling, H., Hu, Q., Song, Z., Visdrone-det2018: The vision meets drone object detection in image challenge results., Proceedings of the European Conference on Computer Vision (ECCV) Workshops, 2018.

[27] Du, D., Qi, Y., Yu, H., Yang, Y., Duan, K., Li, G., Tian, Q., The unmanned aerial vehicle benchmark: Object detection and tracking., Proceedings of the European conference on computer vision (ECCV), 2018, 370-386.

[28] Puertas, E., De-Las-Heras, G., Fernández-Andrés, J., Sánchez-Soriano, J., Dataset: Roundabout Aerial Images for Vehicle Detection., Data, 2022, 47.

[29] Girshick, R., Donahue, J., Darrell, T., Malik, J., Rich feature hierarchies for accurate object detection and semantic segmentation., Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, 580-587.

[30] He, K., Zhang, X., Ren, S., Sun, J., Spatial pyramid pooling in deep convolutional networks for visual recognition., IEEE transactions on pattern analysis and machine intelligence, 2015, 1904-1916.

[31] GIRSHICK, Ross., Fast r-cnn., Proceedings of the IEEE international conference on computer vision, 2015, 1440-1448.

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

[33] Zhang, Z., Drone-YOLO: an efficient neural network method for target detection in drone images., Drones, 7(8), 526.

[34] He, K., Gkioxari, G., Dollár, P., Girshick, R., Mask r-cnn., Proceedings of the IEEE international conference on computer vision, 2017, 2961-2969.

[35] Dai, J., Li, Y., He, K., Sun, J., R-fcn: Object detection via region-based fully convolutional networks., Advances in neural information processing systems, 2016, 29.

[36] Redmon, J., Farhadi, A., Yolov3: An incremental improvement., arXiv preprint arXiv:1804.02767.

[37] Bochkovskiy, A., Wang, C. Y., Liao, H. Y. M., Yolov4: Optimal speed and accuracy of object detection., arXiv preprint arXiv:2004.10934.

[38] Tian, Z., Shen, C., Chen, H., He, T., Fcos: Fully convolutional one-stage object detection., Proceedings of the IEEE/CVF international conference on computer vision, 2019, 9627-9636.

[39] Uzkent, B., Yeh, C., Ermon, S., Efficient object detection in large images using deep reinforcement learning., Proceedings of the IEEE/CVF winter conference on applications of computer vision, 2020, 1824-1833.

[40] Kong, T., Sun, F., Liu, H., Jiang, Y., Li, L., Shi, J., Foveabox: Beyound anchor-based object detection., IEEE Transactions on Image Processing, 2020, 7389-7398.

[41] Law, H., Deng, J., Cornernet: Detecting objects as paired keypoints., Proceedings of the European conference on computer vision (ECCV), 2018, 734-750.

[42] Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J., Yolox: Exceeding yolo series in 2021., arXiv preprint arXiv:2107.08430.

[43] Lee, Y., Park, J., Centermask: Real-time anchor-free instance segmentation., Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, 13906-13915.

[44] Feng, C., Zhong, Y., Gao, Y., Scott, M. R., Huang, W., Tood: Task-aligned one-stage object detection., In 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021, 3490-3499.

[45] Zoph, B., Cubuk, E. D., Ghiasi, G., Lin, T. Y., Shlens, J., Le, Q. V., Learning data augmentation strategies for object detection., Proceedings of the European conference on computer vision (ECCV), 2020, 566-583.

[46] Yu, F., Koltun, V., Multi-scale context aggregation by dilated convolutions., arXiv preprint arXiv:1511.07122.

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

[48] He, K., Zhang, X., Ren, S., Sun, J., Identity mappings in deep residual networks., Proceedings of the European conference on computer vision (ECCV), 2016, 630-645.

[49] Yu, G., Chang, Q., Lv, W., Xu, C., Cui, C., Ji, W., Ma, Y., PP-PicoDet: A better real-time object detector on mobile devices., arXiv preprint arXiv:2111.00902.

[50] Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S., 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, 2019, 658-666.

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



Wenkang Zhang was born in Maoming, Guangdong, China, in 2000. He is a second year graduate student in College of Electronic and Information Engineering, Wuyi University. His research interests include Deep learning, Object detection, and image processing.
https://orcid.org/0009-0002-5484-7947



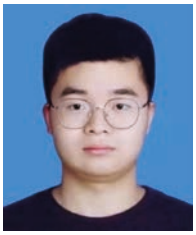
Zhiyong Hong is a professor in College of Electronic and Information Engineering, Wuyi University. He received his Ph.D. degree in computer science and technology from Southwest Jiaotong University, Chengdu, China, in 2014, and received the B.S. and M.S. degrees in computer science and technology from the Shenyang Institute of Technology, Shenyang, China, in 2001 and 2004, respectively. His research interests include intelligent information processing, AI, block chain, and rough set theory.
https://orcid.org/0000-0002-5409-2962



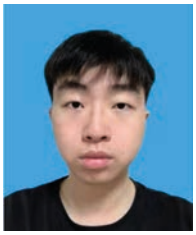
Liping Xiong was born in Yichun City, Jiangxi Province, China, in 1985. He received the Ph.D. degree from Sun Yat-sen University, Guangzhou, China, in 2016. Now, he has been a lecturer in the College of Electronic and Information Engineering, Wuyi University, China. His research interests include knowledge representation and reasoning, formal methods, and machine learning.
https://orcid.org/0000-0001-7924-8655



Zhiqiang Zeng was born in Shaoguan, Guangdong, China, in 1989. He received the Ph.D. degree from the South China University of Technology, Guangzhou, China, in 2018. Since then, he has been a professor with the College of Electronic and Information Engineering, Wuyi University, China. He is the author of a number of high-level articles. His research interests include machine learning, data mining, industrial big data, and production scheduling.
https://orcid.org/0000-0002-9544-5605



Zhishun Cai was born in Yancheng, Jiangsu, China, in 1998. He is a third year graduate student in College of Electronic and Information Engineering, Wuyi University. His research interests include Object detection, Deel learning and aerial image.
https://orcid.org/0009-0000-1813-3897



Kunyu Tan was born in Jiangmen, Guangdong, China, in 2000. He is a first year graduate student in College of Electronic and Information Engineering, Wuyi University. His research interests include Deep learning, Object detection, and Semantic Segmentation.
https://orcid.org/0009-0003-1961-0466