Heliyon 9 (2023) e17730

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Small object intelligent detection method based on adaptive recursive feature pyramid

Jie Zhang [a], Hongyan Zhang [a], Bowen Liu [a], Guang Qu [c,*], Fengxian Wang [a], Huanlong Zhang [a], Xiaoping Shi [b]

[a] College of Electrical and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China
[b] School of Astronautics, Harbin Institute of Technology, Harbin 150080, China
[c] Department of navigation, Airforce communication NCO Academy, DaLian 116000, China

ARTICLE INFO

Keywords:
AR-PANet
Recursive structure
Small object detection
Adaptive method
CBAM

ABSTRACT

As we all know, YOLOv4 can achieve excellent detection performance in object detection and has been effectively applied in many fields. However, the inconsistency of scale features affects the prediction accuracy of the path aggregation network (PANet) in YOLOv4 for small objects, resulting in low detection accuracy. This paper presents YOLOv4, which uses an adaptive recursive path aggregation network (AR-PANet) to improve the detection accuracy of small objects. First, the output characteristics of the PANet are fed back into the backbone network by using a recursive structure to enrich the characteristic information of the object. Second, an adaptive approach is developed to eliminate conflicting information in multi-scale feature space, thereby enhancing scale invariance and promoting feature extraction accuracy for small objects. Finally, the CBAM is used to map the multi-scale features obtained from the AR-PANet to independent channels and spatial dimensions to achieve feature refinement, thus improving the detection accuracy of small objects. Experimental results show that our proposed method can effectively improve the accuracy of small object detection in multiple datasets, addressing this challenging problem with impressive results. Thus, our proposed approach has great potential and valuable applications in the fields of remote sensing and intelligent transportation.

1. Introduction

Object detection is a popular and extensively researched area in computer vision and digital image processing. Over the years, it has gained considerable attention and has emerged as a hot research topic. This field has a wide range of applications, including but not limited to robot navigation, industrial detection, and intelligent video surveillance [1].

To date, object detection techniques can be broadly divided into two categories: traditional detection and deep learning-based methods. Traditional detection approaches use manually designed features that do not require learning and training, and facilitate object detection through simple calculations and statistics. However, it is easily affected by human factors, the image generalization ability is poor, and the pixel requirements are high. The deep learning object detection method completes the detection task through the depth features extracted by the convolutional neural network. The depth feature is less impacted by lighting and orientation,

* Corresponding author.
E-mail address: zzuli_407@163.com (G. Qu).

https://doi.org/10.1016/j.heliyon.2023.e17730
Received 29 November 2022; Received in revised form 26 June 2023; Accepted 27 June 2023

Available online 3 July 2023

making it more effective in accurately reflecting the nature of an object. Effective mining of depth feature information is an important problem in object detection algorithms.

With the continuous development of technology, significant improvements have been made in computer processing power. As a result, the practicality of object detection based on deep neural networks has been enhanced [2–5]. Deep convolutional neural networks (CNNs) form the basis of modern object detection methods, which can be broadly classified into two categories: two-stage detectors and one-stage detectors. Two-stage detectors involve the generation of a “region of interest” proposal in the first stage, followed by object classification and position regression in the second stage. Examples of two-stage detectors include R-CNN [6], Fast R-CNN [7] and Faster R-CNN [8]. Unlike the two-stage detector, the one-stage detector performs object classification and regression directly on the image using a deep neural network, which includes SSD [9], YOLOv3 [10], YOLOV4 [11], CenterNet [12] and RetinaNet [13], etc.

Although the deep neural network greatly facilitates the progress of object detection. The accuracy of small object detection is still low. In practical applications, small objects such as vehicle and road sign detection in the field of autonomous driving, as well as disaster analysis and object search in the field of search and rescue, have a relatively high occupancy rate. Therefore, the design of algorithm framework for small object detection has become extremely important. Small objects occupy fewer pixels in the image, and there is a serious problem of semantic information loss after multiple convolutions. The specific performance is as follows:

(1) Few available features. It is difficult to extract distinguishing features from small objects with low resolution, due to less visual information. The detection model is easily disturbed by environmental factors, making it difficult to accurately locate and identify small objects.

(2) Maintaining high positional accuracy is essential for successful object detection in computer vision. Because small objects typically occupy only a small fraction of an image, their bounding box localization poses greater challenges than larger objects. These challenges are compounded by pixel shifts in the prediction boundary during the prediction process. In particular, the error effect of these shifts is significantly more pronounced for smaller objects than for their larger counterparts.

(3) Sample imbalance. When the manually set anchor frame is quite different from the real boundary frame of the small object, the positive training sample of the small object will be far smaller than the positive sample of the large-scale object, which will cause the training model to pay more attention to detecting the large-scale object and ignore the detection of the small object.

(4) In object detection, small object clustering refers to the phenomenon where small objects tend to aggregate or cluster together. When this happens, small objects adjacent to the aggregation area may be reduced to a single point on the deep feature map after multiple down-samplings, making them difficult for the detection model to distinguish from each other. In particular, when many similar small objects appear densely, the model’s post-processing non-maximum suppression operation may filter out many correctly predicted bounding boxes, leading to missed detections. In addition, the close proximity of bounding boxes in the aggregation region can hinder box regression and hinder model convergence. After an extensive review of the available literature, this formulation avoids redundancy while still conveying all critical information.

To solve these problems, the researchers optimize the small object detection method based on various optimization strategies, such as data enhancement [14–18], multi-scale learning [19–22], context learning [23–27], and generative confrontation learning [28–34], which are analyzed as follows:

(1) By augmenting the dataset with techniques such as object clustering, object extension and infrared methods, data augmentation increases the robustness and generalization capacity of the detection model by enriching the diversity within it and refining its feature details. This practice paves the way for a more comprehensive and expansive dataset, which in turn contributes to improved detection model performance. However, it also increases the computational cost. Improperly designed data augmentation strategies may introduce new noise and affect the performance of feature extraction, which poses challenges to the design of object detection algorithms.

(2) The multi-scale feature fusion technique integrates both shallow representational information and deep semantic information to improve the model’s perceptual range and enhance the object’s detailed information, leading to better extraction of small objects and ultimately improving the performance of small object detection. Despite its advantages, this technique can increase the complexity of the model, making the training process prone to overfitting. In addition, the feature function process can be affected by noise, making it a significant challenge to avoid its negative effects.

(3) The context learning approach uses information about the objects in the image to improve the model’s understanding and perception of the object’s environment. This in turn reduces noise and clutter in the input image, leading to better performance and robustness in detecting small objects. While this method is highly effective, it is limited in scenes where contextual information is lacking. In such scenarios, it is challenging to use easily recognizable results from the scene to support the detection of small objects.

(4) The object detection method based on generative adversarial models combines existing generative adversarial models with detection models to enhance the feature information of small objects by increasing the understanding of data distributions and improving detection performance. However, achieving a favorable balance between generator and discriminator in the training of generative adversarial networks is a challenging task, as the process involves many complex steps. Additionally, the diversity of samples generated by the generator during training is limited, and the improvement of performance after training to a certain extent is limited.

The YOLO series is a popular end-to-end object detection technique in computer vision that provides robust real-time detection capabilities. Among these methods, YOLOv4 stands out as a widely used approach for object detection due to its ability to maintain an optimal balance between accuracy and speed. In Yolov4, PANet is used for multi-scale learning, which solves the problem of mesoscale change in object detection. However, the parameters of the up-sampling and down-sampling methods on different scales of feature extractors are different at different levels of PANet, resulting in different spatial resolutions and semantic information

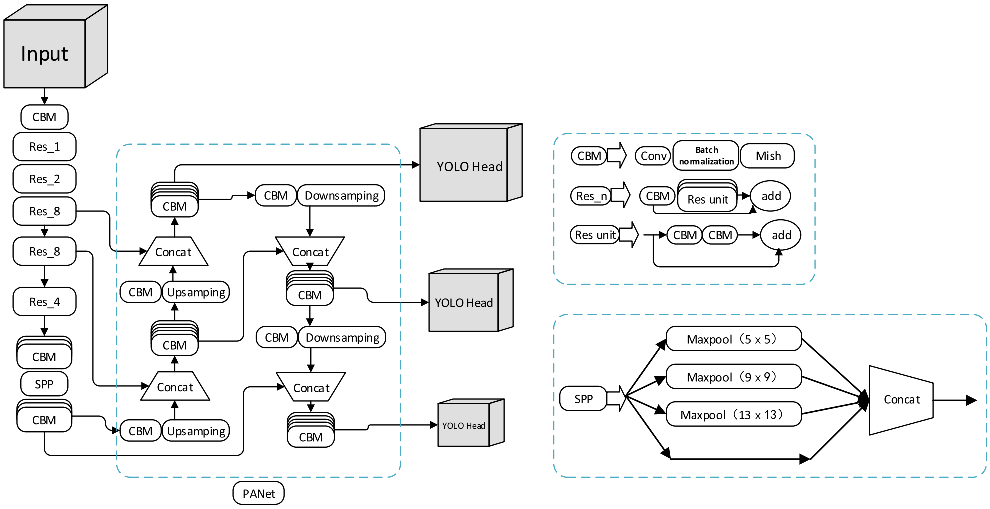


Fig. 1. YOLOv4 Structure Diagram.

of the extracted features at different levels. This leads to inconsistencies between different feature scales. Such inconsistencies may cause the model to allocate weights to certain features improperly, or fail to capture certain important features. This could result in bias or error in the model's prediction results, thereby affecting the accuracy and reliability of the model, as well as the detection accuracy of small objects. The proposed AR-PANet algorithm successfully improves the accuracy of small objects in the YOLOv4 algorithm by weighting and fusing feature maps of the same size before and after pyramid adaptive learning.

This paper makes the following notable contributions:

(1) To address the problem of insufficient feature information for small objects, a recursive architecture has been proposed. Due to the complementary relationship between the output features of PANet and the backbone network, missing information can be retrieved in the second feature extraction through the recursive structure. By examining the first pyramid and image feature repeatedly, the backbone network gains access to more valuable feature details, which grants it a more robust and potent feature representation.

(2) To address the problem of spatial conflicting information caused by feature scale mismatch between the output feature layers of the feature pyramid. By using adaptive learning methods to filter conflicting information in the feature space, different scale inconsistencies are suppressed by weighting the sampled feature layer pixels through up-sampling and down-sampling on different feature layers, thereby enhancing the fusion effect of the model on the features of small objects.

(3) The convolutional attention module provides a viable solution to the problem of inadequate modelling of global features and channel correlation. This module works by independently mapping the feature maps of each scale to the corresponding channels and spatial dimensions. Adaptive learning and adjustment of the weights of each channel and spatial location allows the representation of useful information to be enhanced while suppressing the influence of irrelevant information. Ultimately, this approach achieves feature refinement and improves the accuracy of small object detection.

This paper presents an innovative approach to small object detection using a YOLOv4 method based on AR-PANet. By exploiting the aforementioned innovations, the proposed method effectively addresses the challenges associated with small object detection. Quantitative experimental results provide conclusive evidence of the effectiveness of the method in accurately detecting small objects.

This paper is structured as follows. Section 2 provides a detailed review of related work. Section 3 describes the proposed method, a YOLOv4 method based on AR-PANet. Section 4 presents the results of comparative experiments conducted specifically for the detection of small objects. Finally, Section 5 provides concluding remarks.

2. Related work

As shown in Fig. 1, YOLOv4 uses a structure similar to YOLOv3 to develop a more accurate and faster object recognition model. YOLOv4 method mainly includes three sections: Cspdarknet-53 backbone network, multi-scale feature fusion and location prediction network. The above three sections are described in detail below.

2.1. CSPDarknet-53

YOLOv4 extracts features from input data using Cspdarknet-53 [35,36]. The backbone network of Cspdarknet-53 consists of 53 convolution layers and 23 residual layers. 3×3 and 1×1 convolution kernels are employed to extract features in the convolution layer. The 23 residual layers are composed of 1, 2, 8, 8, and 4 smaller residual units. By using this method, the gradient flow is segmented into different paths by the network, resulting in more diverse combinations of gradients when outputting residuals, thus avoiding problems such as gradient explosion and vanishing. Therefore, this method improves the computational efficiency of parameters

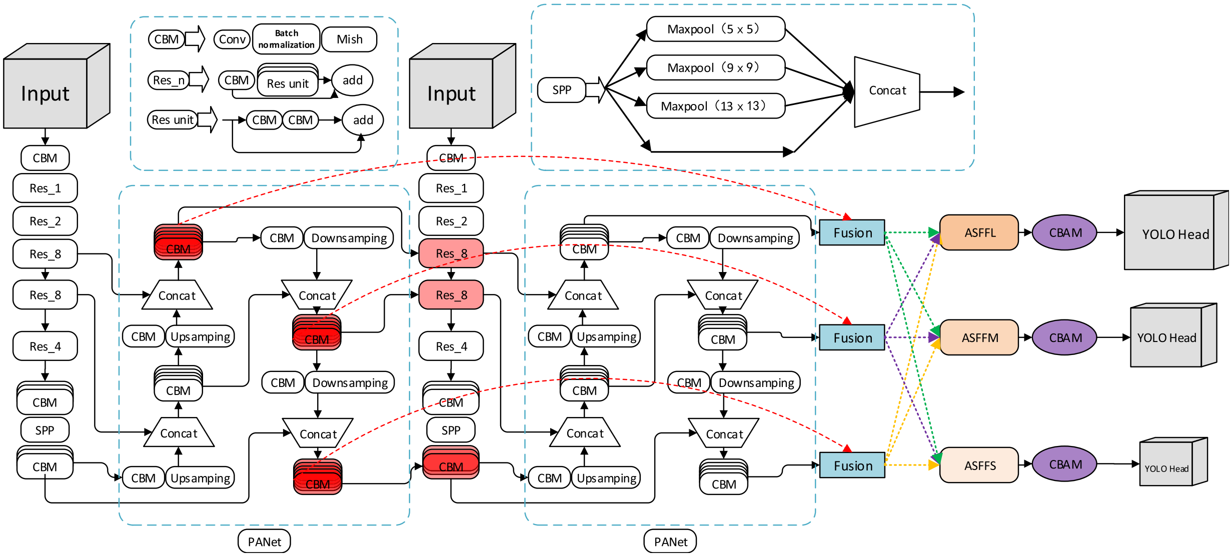


Fig. 2. Adaptive recursive pyramid network detection structure based on YOLOv4.

during model training and speeds up the model's inference time. But the network backbone performs only a single extraction of the image, making it difficult to identify the image's concealed details. Therefore, a recursive structure is proposed to ensure that the backbone network retrieves missing information during the second feature extraction and extracts a feature representation with strong generalization ability, thus improving the detection accuracy of small objects [37,38].

2.2. *Feature fusion module*

YOLOv4 performs feature fusion using Spatial Pyramid Pooling (SPP) and PANet [39]. SPP [40] uses four different sizes of maximum pooling: 1×1, 5×5, 9×9, and 13×13 to process the input feature mapping. The SPP structure removes the constraints on the input size for the convolutional neural network. By embedding the architecture within the feature extraction network, the proposed approach can extend the receptive field of the model and capture important contextual features while maintaining network speed. To enhance the top-down feature pyramid, PANet introduces a bottom-up path structure. Through multiple iterations of feature extraction and fusion at different scales, the model obtains three fused features with improved positioning and semantic properties, thereby enhancing the model's detection performance. However, the feature layers of different scales of PANet directly enter the detector, resulting in inconsistency between features, which the detection accuracy of small objects is reduced. The use of self-learning methods to filter spatial conflict information at different scales has improved scale invariance [41] and small objects detection accuracy.

2.3. *Prediction network*

Using three different scales of detection heads [10], YOLOv4 regresses and predicts the position, category and confidence of the objects, and selects the detection boxes using set thresholds. Set the score of the most predictive bounding box to 1 so that only one bounding box is assigned to each marked object. To locate the center of an object, the grid dimensions in the feature map are resized to 1 and the prediction offset is constrained between 0 and 1 using the sigmoid function [11]. In addition, the Bounding Box Regression Loss function uses CIOU (introduced in [42]) instead of mean square loss to improve the speed and accuracy of the bounding box regression process. Meanwhile, the cross-entropy loss function remains the primary method for calculating confidence and classification probability.

3. Methodology

To overcome the accuracy limitations in small object detection resulting from the direct prediction of feature maps by the path aggregation network in YOLOv4, a novel method based on AR-PANet is proposed in this paper. Its architecture is illustrated in Fig. 2 and is specifically tailored to improve the performance of small object detection. Firstly, the proposed approach involves the recursive use of PANet [37] to provide feedback on the feature map, which is subsequently fed back into the backbone network for further feature map refinement and fusion. Second, the output features are refined using adaptive spatial feature fusion methods. The specific methods are described in Sections 3.1 and 3.2.

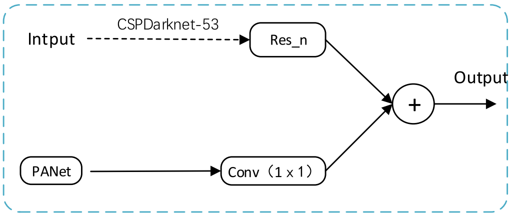


Fig. 3. Structure of recursive connection module.

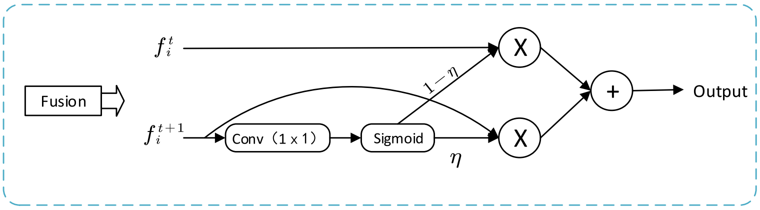


Fig. 4. Fusion module structure. η assigns weights to different features.

3.1. Recursive structure

Obtaining sufficient image detail from single feature learning in the backbone network can be challenging, but is essential for improving the accuracy of small object detection. To address this issue, this paper proposes a recursive structure that utilizes PANet output features as inputs to the backbone network, which is illustrated in Fig. 2. By incorporating the PANet features into three feature layers of the backbone network and exploiting iterative learning through backpropagation, crucial information lost in the initial learning can be recovered. The learned feature layers are then fed back into PANet for feature fusion to increase the generalization capacity of the network. In addition, the fusion module weights and fuses the output features of two PANets to obtain powerful feature representations, thereby improving detection accuracy. Next, the working principles of the recursive structure and the fusion module are introduced.

3.1.1. Recursive module

As shown in Fig. 2, the recursive module is used to allow the backbone network to simultaneously receive the first feedback features of PANet and the input features of the backbone network. It's different from DetectoRS [43], we directly adjust the features obtained from the first iteration of PANet to the same size as the backbone network through a 1×1 convolutional layer, saving computation in the ASPP part. By combining the original and adjusted features, we obtain a set of input features for the backbone network, which can be trained by backpropagation. An adaptive method is then applied to select the most informative feature layers, thereby improving the accuracy of small object detection. The recursive module is shown in Fig. 3, which incorporates the first PANet feature into the backbone network for feature extraction. Specifically, the feature extraction part of YOLOv4 is represented in the upper part of the recursive connection module. The lower part of the module processes the first feature of PANet by convolution. Finally, shortcut links are used to connect the output features, which are then fed into the backbone network for backpropagation. This could extract more refined and informative details, facilitating adaptive feature selection for small object detection.

3.1.2. Fusion module

The fusion module is shown in Fig. 4. f_i^t and f_i^{t+1} represent the first and second output features of PANet, respectively. f_i^{t+1} is adjusted to the same size as f_i^t using a 1×1 convolutional layer and a sigmoid operation is applied to obtain a weight η, ranging from 0 to 1. The weight of f_i^t is $1-\eta$. Then, a weighted sum of f_i^t and f_i^{t+1} is computed to generate a new feature, which then enters the subsequent adaptive feature fusion module to enhance the feature by eliminating spatially inconsistent information and refining it further. The formula of this method is shown in Formula (1).

$$Output = f_i^t \cdot (1-\eta) + f_i^{t+1} \cdot \eta \tag{1}$$

3.2. Adaptive spatial feature fusion

According to FPN [10], a feature pyramid is constructed through a series of upsampling and downsampling operations. Upsampling low-resolution feature maps and downsampling high-resolution feature maps produces feature maps with different scales, which is crucial for detecting objects of different sizes and shapes, thus improving the detection accuracy of the model. PANet initially proposed bottom-up secondary fusion and improved multi-scale feature fusion via lateral connections. Lateral connections enable the exchange and fusion of multi-scale feature maps, leading to improved detection performance, accuracy, adaptability, and

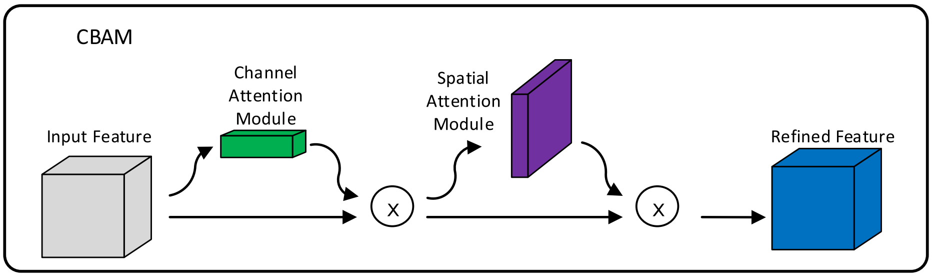


Fig. 5. Convolutional block attention module (CBAM) module structure.

robustness of the model. Small objects usually have smaller sizes and areas, thus requiring higher spatial resolution for fine detection and positioning. The high-resolution feature map can extract detailed information that is useful for small object detection, but inconsistent features at different scales hinder accuracy. To overcome this problem, an adaptive approach is used to learn the fusion of each feature level [41]. The feature maps of multiple scales are upsampled or downsampled to a uniform size, and an adaptive method is used to calculate the weight of each pixel. The feature maps of different branches are then merged by weighted averaging. The pixel weight is calculated by normalizing the response value of each pixel in the feature maps of different scales, and then determining the weight of each scale to obtain the feature vector for each layer. By adjusting the significance of the feature maps of different scales, it is possible to refine the positional information of the lower network and the semantic information of the upper network, resulting in better detection of small object features. The CBAM [44] is used for the obtained fusion features to distinguish them from the channel and space by the attention mechanism. By assigning weights, it can better extract useful information, remove noise information, achieve feature refinement, and thus be more conducive to detecting small objects. Next, the adaptive fusion and attention modules are described in detail.

The final three layers of PANet features differ in resolution and channel numbers, which makes them challenging to integrate directly. To mitigate this problem, an adaptive fusion module is incorporated into the network, allowing each feature to be up- or downsampled as required. The most important thing is to be able to map feature maps at other scales to the correct location, as shown in Formula (2).

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

Here, vector $x_{ij}^{n\to l}$ represents the operational relationship between pixels in different layers, where l ranges from 1 to 3, corresponding to features at three different scales. A 1×1 convolution is used to normalize the three-layer feature maps to a consistent resolution, and the weight parameters are derived by applying conventional backpropagation learning. The weights are defined as in Formula (3).

$$\alpha_{ij}^{l} = \frac{e^{\lambda_{\alpha_{ij}}^{l}}}{e^{\lambda_{\alpha_{ij}}^{l}} + e^{\lambda_{\beta_{ij}}^{l}} + e^{\lambda_{\gamma_{ij}}^{l}}} \tag{3}$$

Where the value range of the parameters α, β and γ is within [0,1], and it satisfies the constraint that $\alpha_{ij}^{l} + \beta_{ij}^{l} + \gamma_{ij}^{l} = 1$. The specific values are obtained by the softmax activation function. This technique enables the adaptive accumulation of features from each level and scale, resulting in the enhancement of features across varying scales. After fusing, the outputs y1, y2 and y3 are passed through spatial and channel selection and are fed into the detection head of the YOLOv4 for detection.

Through this method, the location information of the low-level network is fully fused with the high-level network's voice information enhancing the model's acceptance domain, enabling the model to exact more detailed information from the depth features and obtain more semantic information from the shallow features.

3.3. Attention module

The attention mechanism uses a neural network to create a mask whose values correspond to the attentional weights assigned to different items. CBAM can focus on channels and feature spaces that better express object information during the neural network training process, thereby improving detection efficiency. The module structure of CBAM is shown in Fig. 5.

It can be seen from Fig. 5 that the processing steps of the feature map are channel first and then space, and the final result is obtained by multiplying the weights of the feature map. The mathematical expression is as in Formula (4):

$$\begin{aligned} U' &= T_m(U) \otimes U \\ U'' &= T_n(U') \otimes U' \end{aligned} \tag{4}$$

In the equation presented, $\otimes$ denotes element-wise multiplication, U represents the input feature map, $T_m(U)$ corresponds to the channel attention map generated by the channel attention module, and $T_n(U')$ represents the spatial attention map generated by the spatial attention module.

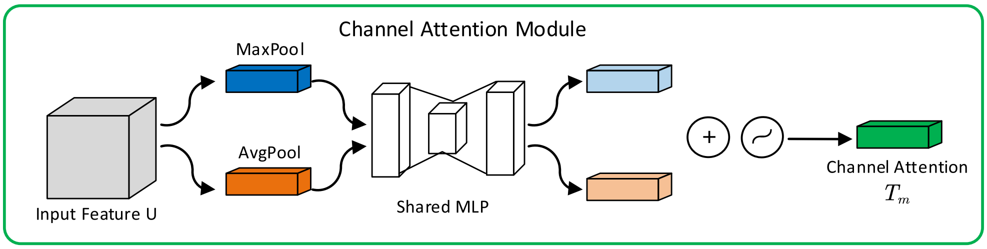


Fig. 6. Channel attention module structure.

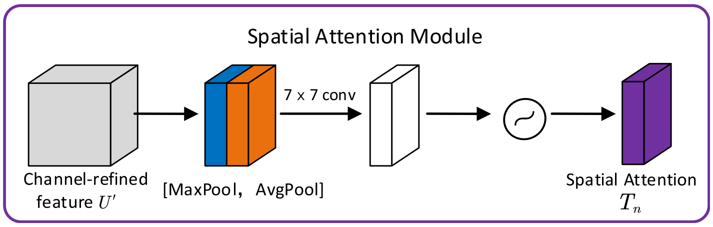


Fig. 7. Spatial attention module structure.

Fig. 6 shows the channel attention module. The techniques used in this approach are the application of global max-pooling and global average-pooling [45] to the input feature graph U. These operations are performed on the width and height of the graph to reduce redundancy. Next, a multilayer perceptron (MLP) is introduced, with shared weights. The MLP obtains the final channel feature map by adding the two output results, dividing the weights between different layers and then applying the sigmoid activation function. The implementation process of the mathematical expression is as in Formula (5):

$$\begin{aligned} T_m(U) &= \sigma(MLP(AP(U)) + MLP(MP(U))) \\ &= \sigma(D_1(D_0(U^m_{AP})) + (D_1(D_0(U^m_{MP}))) \end{aligned} \tag{5}$$

The equation presented includes the sigmoid activation function, denoted by σ, and the MLP weights, denoted by $D_0 \in R^{m/r\times m}$ and $D_1 \in R^{m\times m/r}$, where r is the dimensionality reduction factor.

Fig. 7 illustrates the spatial attention module. The module takes as input U', which is obtained by applying channel-based global maximum and average pooling to the feature map. U^n_{AP} and U^n_{MP} are merged to produce a two-channel feature map, which is later transformed into a single-channel feature map by a 7 × 7 convolutional layer. Finally, the spatial attention map $T_n(U)$ is generated by applying the sigmoid activation function. Its mathematical expression is in Formula (6):

$$\begin{aligned} T_n(U) &= \sigma(f^{7\times7}([AP(U); MP(U)])) \\ &= \sigma(f^{7\times7}([U^n_{AP}; U^n_{MP}])) \end{aligned} \tag{6}$$

In the given equation, 7 × 7 denotes the convolutional kernel's size, σ refers to the activation function, and $f^{7\times7}$ represents the convolution operation.

Through the above process, the fused features obtained by the adaptive spatial feature fusion module are distinguished in channels and space through the attention mechanism. It can enable the detector to extract important information with greater efficiency and accuracy, achieve feature refinement and, in particular, improve the accuracy of small objects.

4. Experiment

The training batch size is 8, and the initial learning rate is 0.0001. The IOU is set to 0.5 by default. The experimental environment is Windows 10, Pytorch 1.2, and CUDA 10.0. Computer hardware parameters are 256gb RAM, Intel NVIDIA TITAN RTX graphics card and 8163 processor.

4.1. Experimental dataset

To validate the effectiveness of the small object detection algorithm proposed in this article, the NWPU VHR-10 dataset, which contains 10 types of ground objects, and the RSOD dataset, which contains 4 types of ground objects, were selected as the primary datasets for experimentation. At the same time, we use the VOC dataset and the KITTI dataset to prove the universality of our algorithm.

Table 1
The outcomes obtained on the NWPU VHR-10.

Detector	AP (%)										mAP (%)
	HB	BR	VH	BC	GTF	BD	TC	PL	SP	ST	
YOLOv4	44.35	37.88	84.83	79.11	94.21	95.10	83.81	99.10	81.97	98.58	79.90
Ours	67.03	44.32	86.59	91.07	95.25	97.37	89.85	96.49	75.09	99.50	84.26

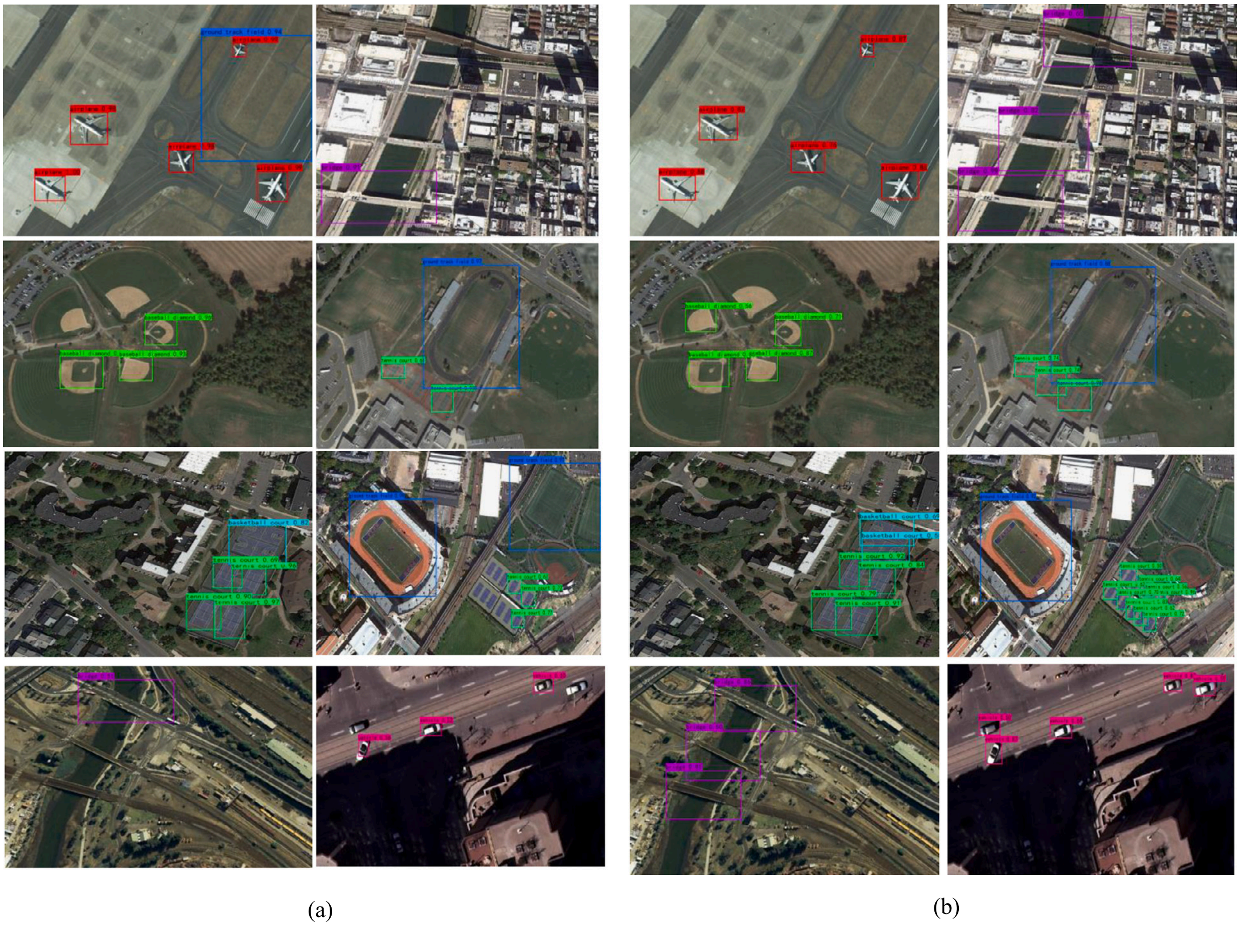


Fig. 8. The NWPU VHR-10 dataset's visualization outcomes are exhibited using color-coded rectangular boxes that distinguish different categories. The visual results of YOLOv4 are displayed in (a), while the results of AR-PANet are shown in (b).

4.2. NWPU VHR-10 dataset

This dataset is a remote sensing image dataset, which contains 3775 objects (757 airplanes (PL), 655 storage tanks (ST), 524 tennis courts (TC), 477 vehicles (VH), 390 baseball diamonds (BD), 302 ships (SP), 224 harbors (HB), 163 ground track fields (GTF), 159 basketball courts (BC), and 124 bridges (BR)). We used 650 of these images for training in the experiment. Using the same test conditions as previously described, we compared the test result from the AR-PANet based YOLOv4 network and the original YOLOv4 network and presented the results in Table 1. The evaluation shows a mAP of 84.26%, which is an improvement of 4.36% compared to the original YOLOv4 network.

Figs. 8 (a) and (b) show the test results on the NWPU VHR-10. By using rectangles to label objects, different colored rectangles display categories and confidence levels so that the results can be viewed more clearly. Compared with traditional YOLOv4, our method detects more small objects and identifies small objects more accurately, thus improving the problem of object detection errors. Our method shows superior accuracy and reliability in detecting small objects. This rephrased sentence conveys the same message more clearly and concisely, while avoiding redundancy.

Table 2
The outcomes obtained on the RSOD.

Method	AP (%)				mAP (%)
	aircraft	oiltank	overpass	playground	
YOLOv3	90.35	97.34	76.73	91.18	88.90
YOLOv4	90.70	98.87	81.92	93.93	91.36
CenterNet	73.64	97.36	85.77	92.86	87.41
RS-YOLOX [46]	-	-	-	-	93.07
RA-BiFPN [38]	68.00	95.56	85.04	99.76	87.09
Our method	91.79	99.46	91.37	94.98	94.40

- means no data was given in the original paper.

Table 3
The PASCAL VOC 2007+2012 test result.

Algorithm	Backbone	Input	mAP (%)	GPU
SSD	VGG	512*512	77.50	GTX 1080Ti
YOLOv4	CSPDarknet-53	416*416	81.69	Titan RTX
R-FCN	ResNet-101	-	73.20	Titan X
Faster-yolo	-	-	77.90	Titan X
MDFN500	ResNet-101	500*500	78.30	RTX 2080Ti
CenterNet	ResNet-101	512*512	78.70	Titan X
RA-BiFPN	VGG	512*512	81.72	Titan RTX
YOLOv5-s	Modified CSP v5	640*640	77.8	-
YOLOv7	New ELANCSP	640*640	80.7	-
YOLOX-m	Modified CSP v5	640*640	81.54	-
Our	CSPDarknet-53	416*416	83.77	Titan RTX

- means no data was given in the original paper.
The mAP performance of an object detection model is significantly influenced by the backbone network and the input size. A deeper backbone network is typically better at extracting complex features, which can greatly improve the model's detection accuracy and performance. A larger input size can also improve accuracy, but at the cost of increased computational and memory requirements. In addition, the GPU plays a critical role in the model's training and inference speed, indirectly affecting the model's mAP performance.

4.3. RSOD dataset

The remote sensing image data set RSOD has 976 images. There are 446 images of playgrounds, 189 of oil tanks, 176 of overpasses, and 165 of airplanes. Some of the objects in it perfectly meet the requirements of small objects with only dozens of pixels. We randomly selected 80% as the training set and 20% as the test set.

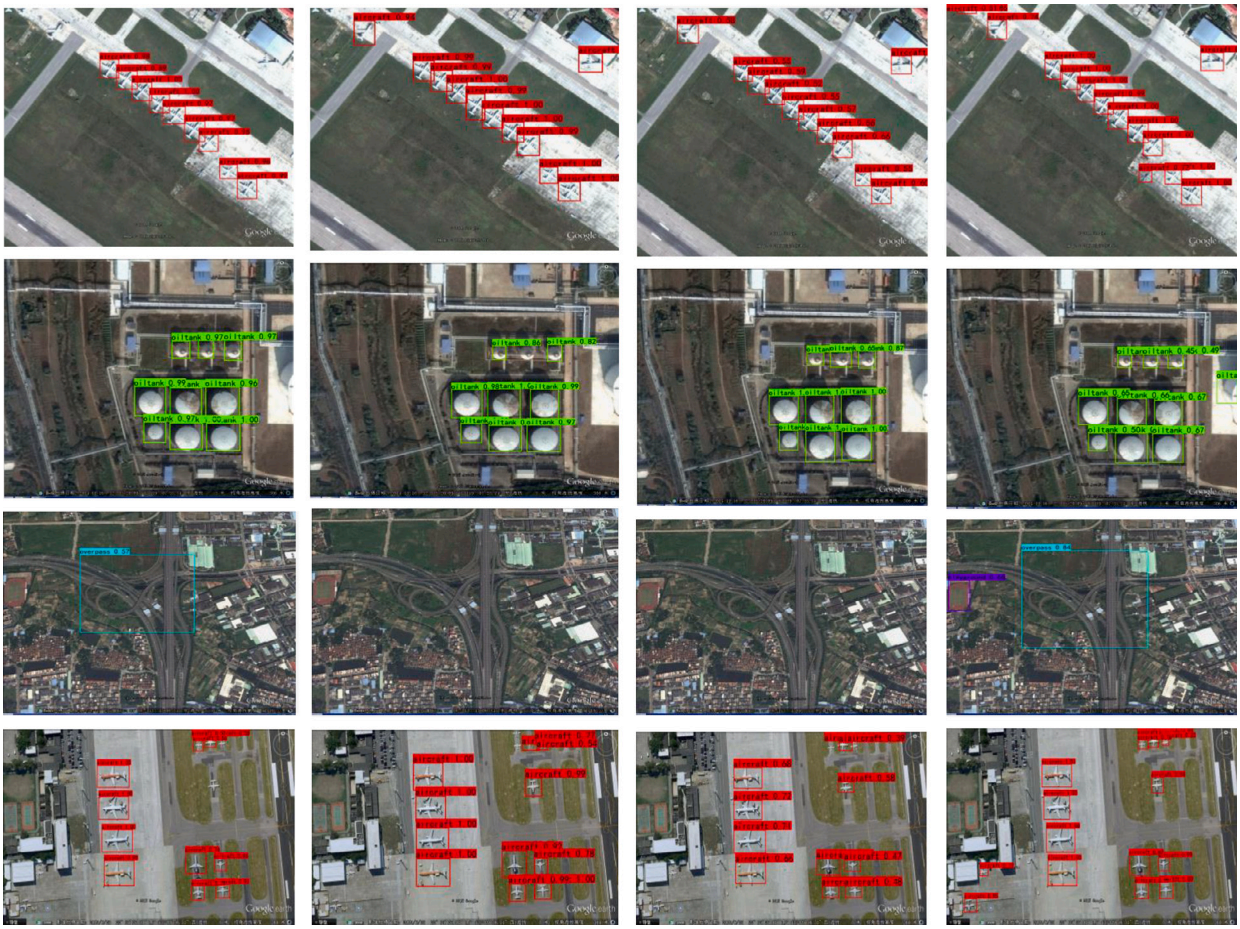
Our approach was compared with YOLOv4, YOLOv3, CenterNet, and RA-BiFPN on the RSOD dataset under identical circumstances. Table 2 shows the mAP results obtained using our proposed method, which yielded a score of 94.4%, surpassing the performance of the other methods evaluated. The accuracy of each type of object is improved by this method. Fig. 9 shows the detection results obtained using YOLOv4, YOLOv3, CenterNet, and the proposed method on the RSOD dataset, as presented in (a), (b), (c), and (d), respectively. In particular, our method outperforms YOLOv4, YOLOv3 and CenterNet in the detection of small objects.

From Fig. 8-9, our proposed method has achieved remarkable results in addressing the challenge of small object detection in remote sensing images. The detection of small objects in remote sensing images is accurately achieved, and even in low confidence detection scenarios, our model accurately positions and identifies the objects. This demonstrates the superiority of the recursive method in extracting more feature information, which not only increases object detection efficiency, but also improves small object detection. Overall, the present study provides a valuable solution for small object detection in remote sensing images.

4.4. PASCAL VOC 2007+2012 dataset

For testing, we utilized the PASCAL VOC 2007+2012 dataset and a Pytorch 1.2 experimental environment to ensure fair comparisons among all methods. Table 3 displays the performance results of our approach and other commonly used detectors, such as SSD [9], R-FCN [47], Faster-yolo [48], MDFN500 [49], CenterNet, RA-BiFPN [38] YOLOv5-s [50], YOLOv7 [50] and YOLOX-m [50]. We refer to their homepages for the outcome stats. When the resolution is 416×416, the contrast between ours and the original YOLOv4 is improved by 2.08%. The proposed method demonstrates superior object detection performance for smaller input sizes, even when compared to other state-of-the-art detection methods.

Fig. 10 shows the comparative results of different methods on the PASCAL VOC 2007+2012 dataset. In particular, our proposed method shows superior detection performance compared to YOLOv4. This is mainly because AR-PANet can obtain richer object feature information through the second time learning features, which also proves that our algorithm has sufficient adaptability and robustness.



(a) (b) (c) (d)

Fig. 9. Visualization of YOLOv4, YOLOv3, CenterNet and AR-PANet results on the RSOD.

Table 4
The KITTI test result.

Algorithm	AP (%)			mAP (%)
	Person	Car	Cyclist	
YOLOv4	51.91	87.58	61.49	66.99
Our method	50.92	88.34	70.29	69.85

4.5. *KITTI dataset*

KITTI dataset is a more popular computer vision evaluation set for evaluating algorithm performance. Table 4 shows the comparative results of YOLOv4 and our proposed method. The results show that our method outperforms YOLOv4 with an improvement of 2.86%. These experimental results demonstrate the ability of the proposed method to achieve accurate detection of small objects in the traffic domain, which has significant value for intelligent transport applications.

Fig. 11 shows a comparison between the detection outcomes of YOLOv4 (a) and our approach (b) on the KITTI dataset. Our method has dramatically improved the detection of vehicles and people, and the probability of detection error is lower.

4.6. *Ablation study*

To better compare the improvement of small object detection by each element, we make a comparison with YOLOv4 under the same test conditions. The experiment was performed on the NWPU VHR-10 dataset with an input size of 512×512. The results of the ablation study are presented in Table 5.

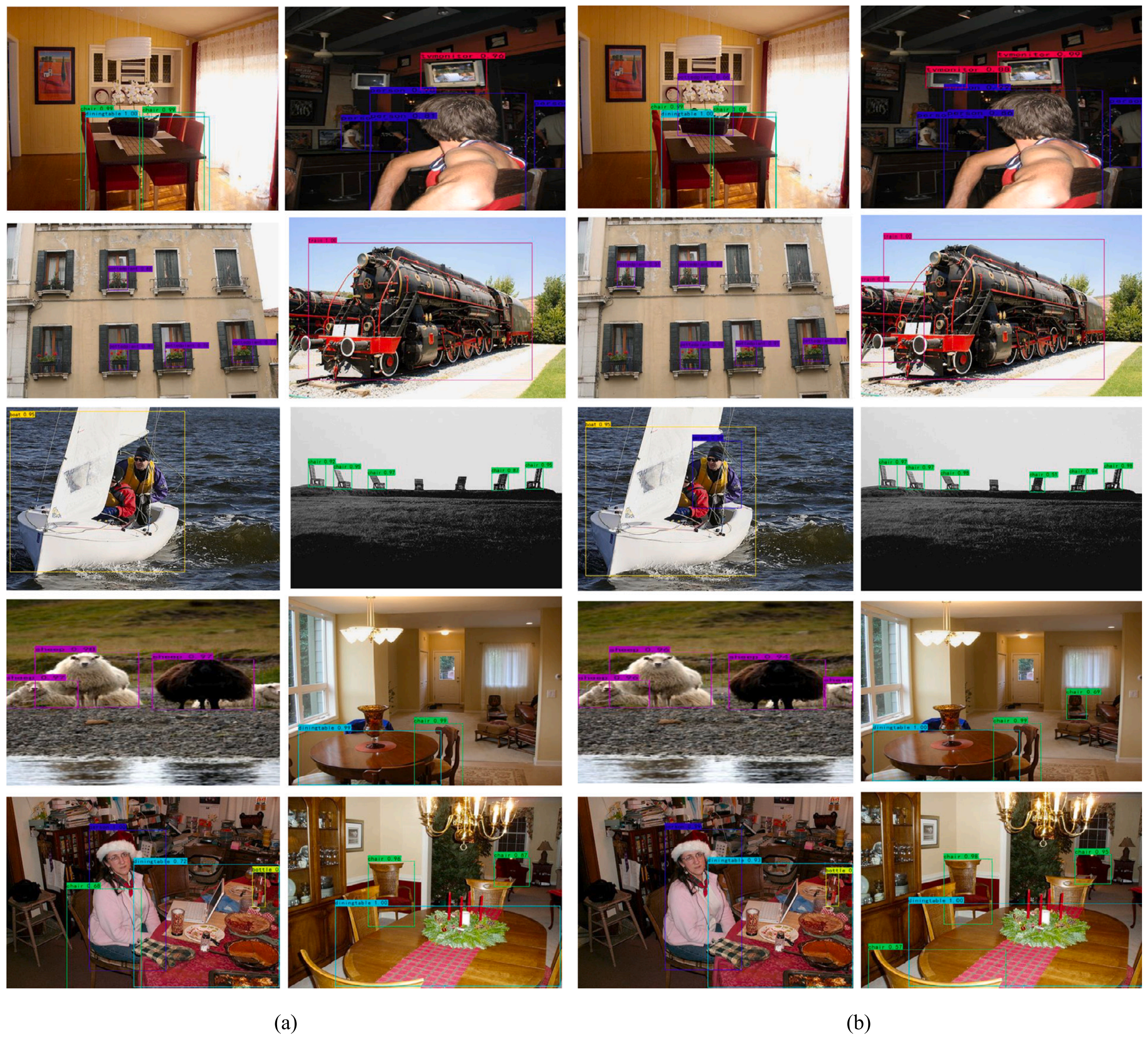


Fig. 10. Visualization results of the PASCAL VOC 2007+2012 dataset.

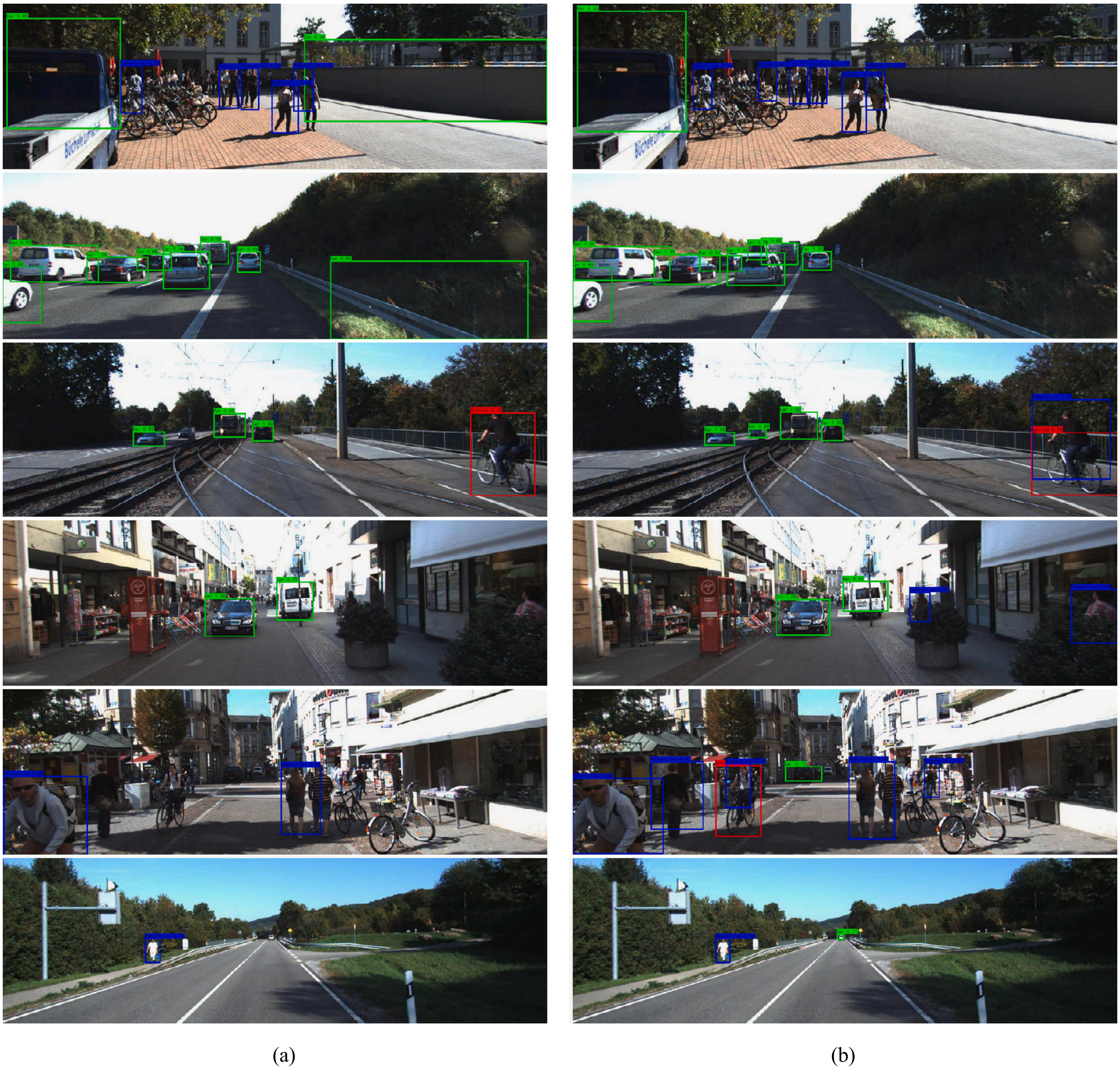
Table 5
The results of ablation experiments.

YOLOv4	R-PANet	AR-PANet	mAP (%)
√	-	-	79.90
√	√	-	82.67
√	-	√	84.26

From the experimental data in Table 5, it can be seen that the detection efficiency of AR-PANet based on YOLOv4 has been improved by 2.77%, and this mAP increases by 1.59% when we introduce adaptive structures into R-PANet, which indicates that the fusion of adaptive and recursive structures can effectively improve the detection effect.

5. Conclusion

This paper presents a novel technique, AR-PANet, specifically designed to improve the accuracy of small object detection performance in YOLOv4. The method uses a recursive structure to feed PANet's output features back into the backbone network, which effectively improves the feature representation capabilities of PANet. In addition, the obtained multi-scale feature information is adaptively fused to eliminate the inconsistency between different scales caused by multi-scale feature detection. The CBAM attention


(a)

(b)

Fig. 11. Visualization results on the KITTI.

mechanism refines the features of the obtained feature layer from two separate dimensions of channel and space, thus improving the object representation ability. The experimental results provide strong empirical evidence supporting the effectiveness of our proposed method in detecting small objects, which can potentially improve the prevailing small object detection challenges in various domains such as intelligent transportation and remote sensing. However, our method has introduced more parameters while improving accuracy, which has somewhat affected the real-time detection performance of the model.

Subsequent research efforts will prioritize the development of a highly effective adaptive recursive pyramid structure to minimize network parameters and increase detection efficiency while maintaining optimal detection accuracy. This revised statement uses concise technical terminology to convey the same message without redundancy. Additionally, we will explore the combination of recursive structures and attention to more precisely locate valuable feature information during the channel combination process.

CRediT authorship contribution statement

Jie Zhang; Bowen Liu: Conceived and designed the experiments.
Hongyan Zhang: Contributed reagents, materials, analysis tools or data; Wrote the paper.
Guang Qu; Xiaoping Shi: Analyzed and interpreted the data.
Fengxian Wang: Performed the experiments.

Huanlong Zhang: Contributed reagents, materials, analysis tools or data.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This project is funded by the National Science Foundation of China (Grant No. 62102373, 61873246, 62006213) and the Science and Technology Research Project of Henan Province (Grant No. 222102320321).

References

[1] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Commun. ACM 60 (6) (2017) 84–90, https://doi.org/10.1145/3065386.
[2] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, M. Pietikäinen, Deep learning for generic object detection: a survey, Int. J. Comput. Vis. 128 (2) (2020) 261–318, https://doi.org/10.1007/s11263-019-01247-4.
[3] S.S.A. Zaidi, M.S. Ansari, A. Aslam, N. Kanwal, M. Asghar, B. Lee, A survey of modern deep learning based object detection models, Digit. Signal Process. 126 (2022) 103514, https://doi.org/10.1016/j.dsp.2022.103514.
[4] M. Wang, W. Deng, Deep face recognition: a survey, Neurocomputing 429 (2021) 215–244, https://doi.org/10.1016/j.neucom.2020.10.081.
[5] G. Guo, N. Zhang, A survey on deep learning based face recognition, Comput. Vis. Image Underst. 189 (2019) 102805, https://doi.org/10.1016/j.cviu.2019.102805.
[6] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 580–587.
[7] R. Girshick, Fast r-cnn, in: 2015 IEEE International Conference on Computer Vision (ICCV), IEEE Computer Society, Los Alamitos, CA, USA, 2015, pp. 1440–1448.
[8] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: towards real-time object detection with region proposal networks, IEEE Trans. Pattern Anal. Mach. Intell. 39 (6) (2017) 1137–1149, https://doi.org/10.1109/tpami.2016.2577031.
[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, A.C. Berg, Ssd: Single shot multibox detector, 2016, pp. 21–37, https://doi.org/10.1007/978-3-319-46448-0_2.
[10] J. Redmon, A. Farhadi, Yolov3: an incremental improvement, arXiv preprint, arXiv:1804.02767, https://doi.org/10.48550/arXiv.1804.02767.
[11] A. Bochkovskiy, C.-Y. Wang, H.-Y.M. Liao, Yolov4: optimal speed and accuracy of object detection, arXiv preprint, arXiv:2004.10934, https://doi.org/10.48550/arXiv.2004.10934.
[12] X. Zhou, D. Wang, P. Krähenbühl, Objects as points, arXiv preprint, arXiv:1904.07850, https://doi.org/10.48550/arXiv.1904.07850.
[13] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollar, Focal loss for dense object detection, IEEE Trans. Pattern Anal. Mach. Intell. 42 (2) (2020) 318–327, https://doi.org/10.1109/TPAMI.2018.2858826.
[14] G. Yuan, T. Ye, H. Fu, L. Wang, Z. Wang, Clustering based detection of small target pedestrians for smart cities, Sustain. Energy Technol. Assess. 52 (2022) 102300, https://doi.org/10.1016/j.seta.2022.102300.
[15] N. Su, J. He, Y. Yan, C. Zhao, X. Xing, Sii-net: spatial information integration network for small target detection in sar images, Remote Sens. 14 (3) (2022) 442, https://doi.org/10.3390/rs14030442.
[16] S. Liu, P. Chen, M. Woźniak, Image enhancement-based detection with small infrared targets, Remote Sens. 14 (13) (2022) 3232, https://doi.org/10.3390/rs14133232.
[17] W. Lin, Z. Zhang, L. Zhang, Infrared moving small target detection and tracking algorithm based on feature point matching, Eur. Phys. J. D 76 (10) (2022) 185, https://doi.org/10.1140/epjd/s10053-022-00505-4.
[18] K. Ren, Y. Gao, M. Wan, G. Gu, Q. Chen, Infrared small target detection via region super resolution generative adversarial network, Appl. Intell. 52 (10) (2022) 11725–11737, https://doi.org/10.1007/s10489-021-02955-6.
[19] C. Chen, S. Wang, S. Huang, An improved faster rcnn-based weld ultrasonic atlas defect detection method, Meas. Control 56 (3–4) (2023) 832–843, https://doi.org/10.1177/0020294022109203.
[20] L. Huang, C. Chen, J. Yun, Y. Sun, J. Tian, Z. Hao, H. Yu, H. Ma, Multi-scale feature fusion convolutional neural network for indoor small target detection, Front. Neurorobot. 16 (2022), https://doi.org/10.3389/fnbot.2022.881021.
[21] L. Zhou, C. Zheng, H. Yan, X. Zuo, Y. Liu, B. Qiao, Y. Yang, Repdarknet: a multi-branched detector for small-target detection in remote sensing images, ISPRS Int.l J. Geo-Inf. 11 (3) (2022) 158, https://doi.org/10.3390/ijgi11030158.
[22] Z. Zhou, Z. Cui, Z. Zang, X. Meng, Z. Cao, J. Yang, Ultrahi-prnet: an ultra-high precision deep learning network for dense multi-scale target detection in sar images, Remote Sens. 14 (21) (2022) 5596, https://doi.org/10.3390/rs14215596.
[23] S. Du, K. Wang, Z. Cao, From characteristic response to target edge diffusion: an approach to small infrared target detection, Infrared Phys. Technol. 124 (2022) 104214, https://doi.org/10.1016/j.infrared.2022.104214.
[24] Y. Xi, W. Jia, Q. Miao, X. Liu, X. Fan, H. Li, Fifonet: fine-grained target focusing network for object detection in uav images, Remote Sens. 14 (16) (2022) 3919, https://doi.org/10.3390/rs14163919.
[25] X. He, Application of deep learning in video target tracking of soccer players, Soft Comput. 26 (20) (2022) 10971–10979, https://doi.org/10.1007/s00500-022-07295-2.
[26] H. Xiao, Y. Li, Y. Xiu, Q. Xia, Development of outdoor swimmers detection system with small object detection method based on deep learning, Multimed. Syst. 29 (1) (2023) 323–332, https://doi.org/10.1007/s00530-022-00995-7.
[27] H. Lv, H. Yan, K. Liu, Z. Zhou, J. Jing, Yolov5-ac: attention mechanism-based lightweight yolov5 for track pedestrian detection, Sensors 22 (15) (2022) 5903, https://doi.org/10.3390/s22155903.
[28] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Commun. ACM 63 (11) (2020) 139–144, https://doi.org/10.1145/3422622.

[29] L. Wang, Q. Li, J. Qin, Rotating machinery fault diagnosis method based on improved semisupervised generative confrontation network, Sci. Program. 2021 (2021) 1–14, https://doi.org/10.1155/2021/1761446.

[30] L. Sixt, B. Wild, T. Landgraf, Rendergan: Generating Realistic Labeled Data, vol. 5, Frontiers Media SA, 2018, p. 66.

[31] W. Ma, Y. Zhang, J. Guo, K. Li, Abnormal traffic detection based on generative adversarial network and feature optimization selection, Int. J. Comput. Intell. Syst. 14 (1) (2021) 1170–1188, https://doi.org/10.2991/ijcis.d.210301.003.

[32] J. Ling, H. Wang, M. Xu, H. Chen, H. Li, J. Peng, Mathematical study of neural feedback roles in small target motion detection, Front. Neurorobot. 16 (2022), https://doi.org/10.3389/fnbot.2022.984430.

[33] Y. Bai, Y. Zhang, M. Ding, B. Ghanem, Sod-mtgan: small object detection via multi-task generative adversarial network, https://doi.org/10.1007/978-3-030-01261-8_13.

[34] J. Noh, W. Bae, W. Lee, J. Seo, G. Kim, Better to follow, follow to be better: towards precise supervision of feature super-resolution for small object detection, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE Computer Society, 2019, pp. 9724–9733.

[35] C.-Y. Wang, H.-Y.M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, I.-H. Yeh, Cspnet: a new backbone that can enhance learning capability of cnn, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE Computer Society, 2020, pp. 1571–1580.

[36] Z. Hong, T. Yang, X. Tong, Y. Zhang, S. Jiang, R. Zhou, Y. Han, J. Wang, S. Yang, S. Liu, Multi-scale ship detection from sar and optical imagery via a more accurate yolov3, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 14 (2021) 6083–6101, https://doi.org/10.1109/JSTARS.2021.3087555.

[37] C. Cao, X. Liu, Y. Yang, Y. Yu, J. Wang, Z. Wang, Y. Huang, L. Wang, C. Huang, W. Xu, et al., Look and think twice: capturing top-down visual attention with feedback convolutional neural networks, in: 2015 IEEE International Conference on Computer Vision (ICCV), IEEE Computer Society, 2015, pp. 2956–2964.

[38] H. Zhang, Q. Du, Q. Qi, J. Zhang, F. Wang, M. Gao, A recursive attention-enhanced bidirectional feature pyramid network for small object detection, Multimed. Tools Appl. (2022) 1–20, https://doi.org/10.1007/s11042-022-13951-4.

[39] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation, arXiv preprint, arXiv:1803.01534, https://doi.org/10.48550/arXiv.1803.01534.

[40] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (9) (2015) 1904–1916, https://doi.org/10.1109/TPAMI.2015.2389824.

[41] S. Liu, D. Huang, Y. Wang, Learning spatial fusion for single-shot object detection, arXiv preprint, arXiv:1911.09516, https://doi.org/10.48550/arXiv.1911.09516.

[42] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, D. Ren, Distance-iou loss: faster and better learning for bounding box regression, arXiv preprint, arXiv:1911.08287, https://doi.org/10.48550/arXiv.1911.08287.

[43] S. Qiao, L.-C. Chen, A. Yuille, Detectors: detecting objects with recursive feature pyramid and switchable atrous convolution, arxiv 2020, arXiv preprint, arXiv:2006.02334, https://doi.org/10.48550/arXiv.2006.02334.

[44] S. Woo, J. Park, J.-Y. Lee, I.S. Kweon, Cbam: Convolutional block attention module, 2018, pp. 3–19, https://doi.org/10.1007/978-3-030-01234-2_1.

[45] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, https://doi.org/10.48550/arXiv.1512.04150, 2015.

[46] L. Yang, G. Yuan, H. Zhou, H. Liu, J. Chen, H. Wu, Rs-yolox: a high-precision detector for object detection in satellite remote sensing images, Appl. Sci. 12 (17) (2022) 8707, https://doi.org/10.3390/app12178707.

[47] J. Dai, Y. Li, K. He, J. Sun, R-fcn: object detection via region-based fully convolutional networks, arXiv preprint, arXiv:1605.06409, https://doi.org/10.48550/arXiv.1605.06409.

[48] W. Ma, Y. Wu, F. Cen, G. Wang, Mdfn: multi-scale deep feature learning network for object detection, Pattern Recognit. 100 (2020) 107149, https://doi.org/10.1016/j.patcog.2019.107149.

[49] Y. Yin, H. Li, W. Fu, Faster-yolo: an accurate and faster object detection method, Digit. Signal Process. 102 (2020) 102756, https://doi.org/10.1016/j.dsp.2020.102756.

[50] Y. Dai, W. Liu, H. Wang, W. Xie, K. Long, Yolo-former: marrying yolo and transformer for foreign object detection, IEEE Trans. Instrum. Meas. 71 (2022) 1–14, https://doi.org/10.1109/TIM.2022.3219468.