Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.039410
Article

Faster RCNN Target Detection Algorithm Integrating CBAM and FPN

Wenshun Sheng*, Xiongfeng Yu, Jiayan Lin and Xin Chen

School of Computer and Communication Engineering, PuJiang Institute, Nanjing Tech University, Nanjing, 211200, China

*Corresponding Author: Wenshun Sheng. Email: sws@njpji.edu.cn

Received: 27 January 2023; Accepted: 10 March 2023; Published: 28 July 2023

Abstract: Small targets and occluded targets will inevitably appear in the image during the shooting process due to the influence of angle, distance, complex scene, illumination intensity, and other factors. These targets have few effective pixels, few features, and no apparent features, which makes extracting their efficient features difficult and easily leads to false detection, missed detection, and repeated detection, affecting the performance of target detection models. An improved faster region convolutional neural network (RCNN) algorithm (CF-RCNN) integrating convolutional block attention module (CBAM) and feature pyramid networks (FPN) is proposed to improve the detection and recognition accuracy of small-size objects, occluded or truncated objects in complex scenes. Firstly, the CBAM mechanism is integrated into the feature extraction network to improve the detection ability of occluded or truncated objects. Secondly, the FPN-featured pyramid structure is introduced to obtain high-resolution and vital semantic data to enhance the detection effect of small-size objects. The experimental results show that the mean average precision of target detection of the improved algorithm on PASCAL VOC2012 is improved to 76.1%, which is 13.8 percentage points higher than that of the commonly used Faster RCNN and other algorithms. Furthermore, it is better than the commonly used small sample target detection algorithm.

Keywords: Target detection; attention mechanism; CBAM; FPN; CF-RCNN

1 Introduction

As a prevailing direction of digital image processing and computer vision, target detection is widely used in autonomous driving, intelligent traffic surveillance, pedestrian counting, medical detection, face recognition, intelligent robot, and other fields [1,2]. With the rapid development of artificial intelligence, convolutional neural network [3] emerges as the times require. The deep learning model [4] has gradually replaced the traditional machine vision method and become the main flow algorithm in the field of target detection. Recently, many detection methods based on the convolutional neural network have been proposed to improve the accuracy and speed of target detection tasks. However, detecting small targets and truncated objects is still challenging for all detectors.

In recent years, there have been two primary solutions to small target and truncated target detection. One is to use a traditional image pyramid [5] and a multi-scale sliding window detection network [6]. The image is scaled in different proportions, and the classifier with fixed anchor size slides on the feature layers with different resolutions to detect the target. The small target is detected at the bottom of the pyramid. The other is a feature fusion network based on a feature pyramid [7]. According to the method, a single-scale image of any size is taken as input, a multi-scale feature image of proportional sizes is output in a full convolution mode, more comprehensive feature expression is carried out on the image through different dimensions, and a featured image with rich representation information is generated. These two models show good performance in practical applications. However, they are ineffective in detecting small-size, truncated or occluded objects because they do not fully consider the influence of global context information of local features on model detection performance.

At present, research on the detection of truncated objects is very rare. Therefore, it is a challenging task to detect small-size targets, especially those that are truncated or occluded. This paper proposes an improved faster region convolutional neural network (RCNN) integrating convolutional block attention module (CBAM) [8] and feature pyramid networks (FPN) [9–11] algorithm (CF-RCNN). CBAM is a lightweight module that combines channel attention and spatial attention. CBAM can focus on local high-utility information of feature images, reduce the interference of invalid objects, and improve the detection ability of occluded or truncated objects. Furthermore, a feature pyramid structure of FPN is incorporated to get high-resolution and strong semantic data, high-level feature data and low-level feature data are conveniently coupled, and the detection impact of tiny target objects is improved. CF-RCNN algorithm is an innovative achievement and a new contribution to this field because it solves the detection problem of small-size objects and truncated objects, ensures the correct recognition rate of the algorithm, and has good real-time performance and robustness. CF-RCNN algorithm is superior to the current conventional detection methods.

The rest of this present article is organized as follows: The second portion covers the current and relevant work in the areas of special target detection acknowledgment. Section 3 details the contributions of the proposed work. Section 4 depicts the results obtained from this work with suitable discussions. Finally, Section 5 derives the conclusions on this work and gives some possible future works.

2 Related Works

In recent years, many scholars have made active exploration in the field of small-scale target detection and truncated target detection and achieved good results.

Salau et al. [12] presented a modified GrabCut algorithm for localizing vehicle plate numbers. The approach extended the traditional GrabCut algorithm by adding a feature extraction method that uses geometric information for accurate foreground extraction.

Yang et al. [13] proposed a multi-scale feature attention fusion network named parallel feature fusion with CBAM (PFF-CB) for occlusion pedestrian detection. Feature information of different scales can be integrated effectively into the PFF-CB module. The PFF-CB module uses a convolutional block attention module (CBAM) to enhance the vital feature information in space and channels.

The authors in [14] proposed the DF-SSD algorithm based on a dense convolutional network (DenseNet) for detection and feature fusion. It uses the algorithm framework of Single Shot MultiBox Detector (SSD) [15] for reference and introduces DenseNet-S-32-1 to replace the original VGG16 network. In addition, DF-SSD also uses a multi-scale feature fusion mechanism. Their proposed residual prediction module is designed to enhance feature propagation along the feature extraction network, i.e., to use a 1 × 1 small convolution filter to predict the target class and the offset of the bounding box position.

The authors in [16] proposed an improved double-head RCNN small target detection algorithm. A transformer and a deformable convolution module are introduced into ResNet-50, and a feature pyramid network structure CARAFE-FPN based on content perception feature recombination is proposed. In that regional recommendation network, the anchor generation scale is reset according to the distribution characteristics of the small target scale so that the detection performance of the small target is further improved.

In [17], the authors suggested a BiFPN structure for the EfficientDet network, consisting of a weighted bidirectional feature pyramid network, adding cross-scale connections to improve feature representation and corresponding weight to each input to improve performance on small target detection tasks.

In [18], the authors performed deep optimization based on YOLOv5 architecture and proposed a multi-scale multi-attention target detection algorithm YOLO-StrVB based on STR (Swin Transformer) network structure. The method mainly aims at small target detection under a complex background, reconstructs a framework to build a multi-scale network, increases a target detection layer, and improves the target feature extraction capability under different scales. Then, a bi-directional feature pyramid network is added for multi-scale feature fusion, and a jump connection is introduced. On this basis, the baseline backbone network end integrates STR architecture.

Fu et al. [19] proposed a DSSD algorithm using ResNet with stronger learning ability as the backbone network. In addition, a deconvolution layer is introduced to reduce the missing rate of small targets further.

Singh et al. [20] started from the perspective of training and considered the level of data, and selectively back-propagated the gradient of target instances with different sizes according to different changes in image scale to realize scale normalization on image pyramid (SNIP). As a result, while capturing the object change as much as possible, the model was efficiently trained with objects with appropriate proportions, and the detection performance was significantly improved.

Although the above algorithms improve the detection performance of the model to a certain extent, they do not fully consider the effect of local features on truncated target detection and fail to optimize the lightweight of the feature extraction module. The CF-RCNN algorithm proposed in this paper is based on group convolution, which greatly reduces the weight of the feature extraction network. CBAM attention mechanism is introduced to consider local features fully. The FPN structure is integrated to improve the detection ability of small-size targets and truncated targets.

The following are the primary contributions of this work:

- The ResNet-50 network model is optimized, and a VS-ResNet network model with stronger expression ability is proposed, which improves the classification accuracy in the target recognition process.
- The improved model of non-maximum suppression is applied to solve the problem of eliminating prediction frame error in classification and regression.
- A CF-RCNN algorithm is proposed and implemented for small-size target and truncated target detection.

3 Model Design and Implementation

The CF-RCNN model proposed in this paper improves the Faster-RCNN [21,22] model by fusing the CBAM attention mechanism and the FPN feature pyramid structure. The design idea is as follows:

(1) To focus on the local high-utility information of the feature image, reduce the interference of invalid targets and improve the detection capability of the bloc or truncated objects, a CBAM mechanism is integrated into the feature extraction network.

(2) To enhance the detection effect of the small target, the FPN feature pyramid structure is introduced to connect the high-level and low-level feature data to obtain high-resolution and strong semantic data.

(3) To alleviate the gradient disappearance phenomenon, reduce the size of hyperparameters, and preserve the information of activation function in a high-dimensional environment, a new ResNet-50 [23] with inverse reactive structure (VS-ResNet) network model with more substantial expressive power is proposed to replace the commonly used VGG 16 network [24]. VS-ResNet network modifies part of the hierarchical structure based on the original ResNet-50, adds an auxiliary classifier, and uses the mode of reciprocal residual and group convolution to improve detection accuracy.

(4) To make up for the defect that the Non-Maximum Suppression (NMS) [25] algorithm erroneously eliminates the overlapping detection frames, the candidate frame score calculation method is reset.

The model was trained and tested on public datasets of CIFAR-10 [26] and PASCAL VOC2012 [27].

3.1 Faster RCNN Model Design

Faster RCNN is a classic work proposed by Ross Girshick in 2015 [21]. It consists of four modules, namely the feature extraction network module (Conv layers) [28], the Region Proposal Network (RPN) [29,30] module, the ROI Pooling [31] module, and the Classification and Regression [32] module. The frame diagram of Faster RCNN is shown in Fig. 1.

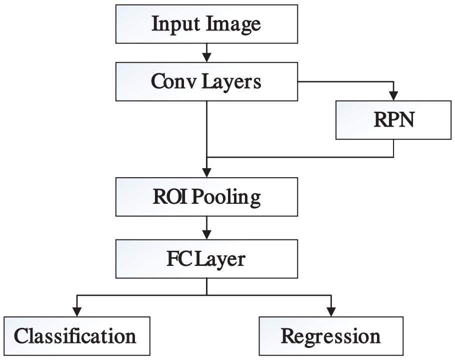


Figure 1: Schematic diagram of fast RCNN algorithm framework

After the original image is input, the shared convolution layer computes it. The results can be shared with the RPN. RPN makes region suggestions (about 300) on the feature map after CNN convolution, extracts feature maps according to the region suggestions generated by RPN, and makes ROI pooling on the extracted features. Then it classifies and regresses the processed data through the full connection layer.

The feature extraction network module [28] is an important part of the Fast RCNN algorithm. The flow of the feature extraction module is shown in Fig. 2.

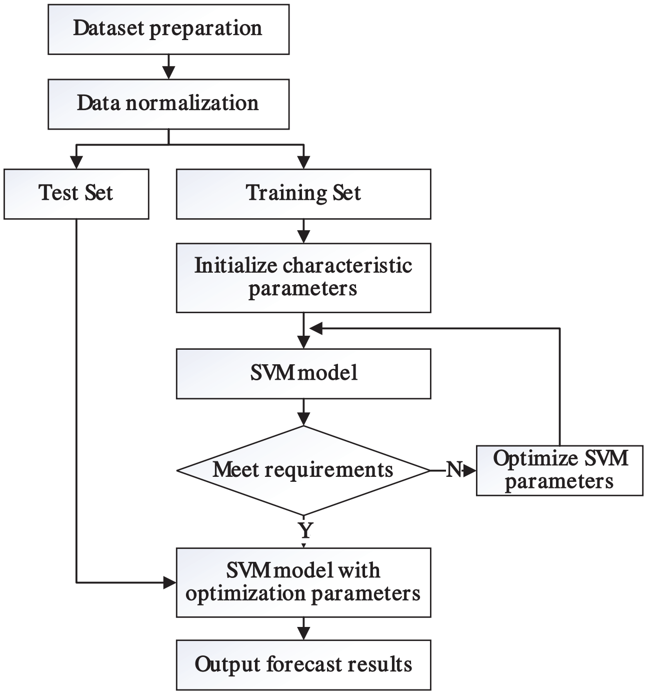


Figure 2: Feature extraction flow chart

After normalization, the prepared data set is divided into the training and test sets. When the training set is trained by the Support Vector Machine (SVM) [33] model, better parameters are obtained by iteration, and finally, the SVM model meets the requirements is obtained. Then, the test data set is input into the optimized SVM model, and the output result is finally obtained.

VGG16 is used as the feature extraction network in Faster RCNN. To break through the bottleneck of region selection in the predecessor target detection model, the new Fast RCNN algorithm model innovatively proposes the RPN algorithm, replacing the selective search (SS) [34] algorithm model used in Fast RCNN and RCNN, that is, Faster RCNN algorithm can be understood as RPN + Fast RCNN. SS algorithm is a simple image processing without the training function. RPN is a fully convolutional network. Its first several convolutional layers are the same as the first five layers of Faster R-CNN, so RPN can share the calculation results of convolutional layers to reduce the time consumption of region suggestions. The RPN module replaces the candidate region selection mode of the original SS algorithm, thus greatly reducing the time cost. RPN selects candidate regions according to image color, shape, and size and usually selects 2000 candidate boxes that may contain identification objects.

When the detected image is input to the feature extraction network module, the feature image is generated after the module convolution operation. Then, it is input to the RPN module to select candidate regions. This module uses the center point of the sliding window as the anchor. All the pixels in the feature picture correspond to k anchor points, generating 128, 256, and 512-pixel areas and 1:1, 1:2, and 2:1 scale windows, respectively. The combined result is a total of 9 windows, as shown in Fig. 3.

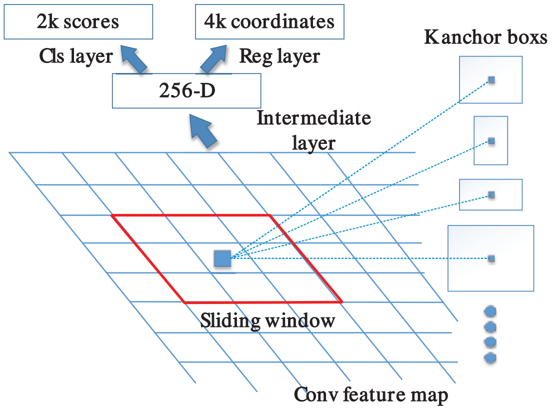


Figure 3: RPN structure diagram

The generated candidate area and convolution layer feature map are input to the ROI Pooling module. The ROI Pooling pools feature maps of different sizes in a uniform size, which facilitates their input to the next layer. Following the ROI Pooling module, the target classification and position regression module processes the data output from the ROI Pooling layer to obtain the object category of the candidate area and the modified image block diagram, and its processing formula can be expressed by Eq. (1).

$$\begin{aligned}\hat{G}x &= Pwdx\,(P) + Px\\ \hat{G}y &= Phdy\,(P) + Py\\ \hat{G}w &= Pw\exp\,(dw\,(P))\\ \hat{G}h &= Ph\exp\,(dh\,(P))\end{aligned} \tag{1}$$

wherein Px, Py, Pw, Ph are the horizontal and vertical coordinates and width and height values of the candidate box center pixel, respectively, and dx, dy, dw, dh are the regression parameters of $\mathrm{N} + 1$ categories of candidate boxes, with a total of $4 \times (\mathrm{N} + 1)$ node, exp is an exponential function with the natural number e as the base.

The VS-ResNet network is used to replace the VGG16 network in order to address the issue of the weak expression ability of the VGG16 network caused by its few layers. The VS-ResNet network is based on the ResNet-50 network improvement. In the original Faster RCNN, the CBAM attention mechanism and FPN feature pyramid structure are integrated to enhance the detection capability for small truncated or occluded objects. The specific improvement process is as follows.

3.2 Design of Group Convolution and Inverse Residual Structure

The structure of the packet convolution network refers to the multi-dimensional convolution [35] combination principle of inception [36], which changes the single path convolution kernel convolution operation of the characteristic graph into a multi-channel convolution kernel convolution stack, reducing the parameters in the network and effectively reducing the complexity of the algorithm model. However, its accuracy will not be greatly affected. For example, VS-ResNet refers to the ResNeXt network [37] and uses 32 groups of the 8-dimensional convolution kernel.

When a deep convolution network is used for convolution, the convolution parameter of the partial convolution kernel is 0, which results in partial parameter redundancy. The experiment proves that when the dimension is too low, the image feature information of Rectified Linear Unit (ReLU) [38] activation function is lost too much. The ReLU function is shown in Eq. (2).

$$\mathrm{Re}LU\,(x) = \max\,(x, 0) = \begin{cases} x, x \geq 0 \\ 0, x < 0 \end{cases} \tag{2}$$

To solve this problem, the VS-ResNet network uses a reciprocal residual structure, with the size of 1 × 1, 3 × 3, 1 × 1 size convolution kernel, as shown in Fig. 4.

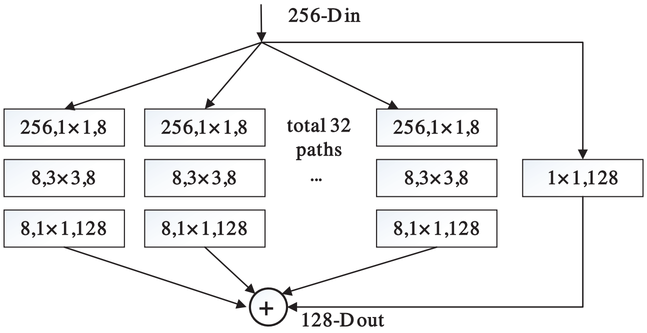


Figure 4: Convolution block structure diagram

The common residual structure is first based on the output basis of the previous layer, using 1 × 1 dimension reduction by convolution kernel of size; pass 3 × 3 size convolution kernel again to extract image features; last used 1 × 1 Dimension elevation of size convolution. Unlike the ordinary residual structure, the inverted residual structure has the opposite order of dimension increase and reduction. That is, 1 × 1 size convolution is first used to increase dimension. And then, get a 3 × 3 convolution; finally, a 1 × 1 convolution kernel is used to reduce the dimension to the original feature map size. When the ReLU function is activated linearly in the region of $x > 0$, it may cause the function value to be too large after activation, thus affecting the stability of the model. To eliminate too much value, VS-ResNet uses the ReLU6 function to replace the ReLU function, as shown in Eq. (3).

$$\mathrm{Re}LU6(x) = \min[\max(x, 0), 6] \tag{3}$$

3.3 Reference to Auxiliary Classifier

With the deepening of the network, the convergence of the feature extraction network becomes more difficult. Therefore, retaining a certain degree of front-end network propagation gradient is necessary to alleviate the gradient disappearance. VS-ResNet adds an auxiliary classifier [39] after convolution layer two and convolution layer four of the ResNet-50 network, respectively, to retain the low dimensional output information of convolution layer two and convolution layer four. In the final classification task, combined with the actual application, set a fixed utility value, and use shallow feature reuse for auxiliary classification. For example, set the utility value of the auxiliary classifier in the VS-ResNet network to 0.1. The auxiliary classifier consists of an average pooling layer, a convolution layer, and two full connection layers, as shown in Fig. 5. After the image is extracted by convolution layer two in ResNet-50, a feature map is generated and inputted to the first auxiliary classifier. Firstly, the network parameters are reduced by an average pooling downsampling layer, which uses a pool size of 14 × 14 with a step distance of 6. Firstly, the network parameters are reduced by an average pooling downsampling layer, which uses a pool size of 14 × 14 with a step distance of 6. Then enter the convolution layer, which uses 128 convolution kernels of 1 × 1 with a sliding step of 1. The resulting results are then flattened and fed into the fully connected layer that follows this layer.

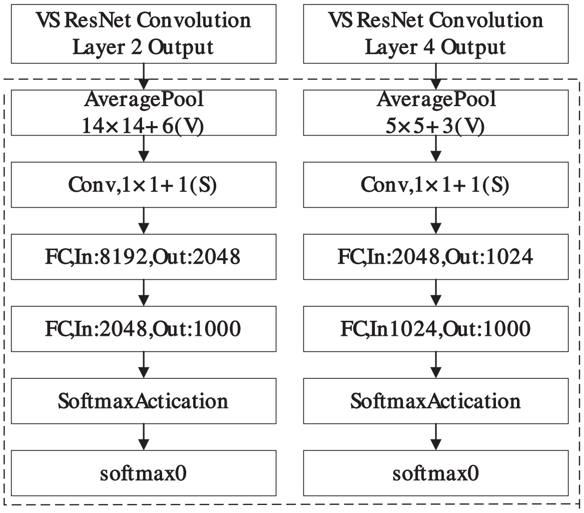


Figure 5: Auxiliary classifier diagram

Dropout [40] at 50% was performed in two fully connected layers to prevent overfitting. Some parameters of the second auxiliary classifier differ from those of the first auxiliary classifier. For example, the pool size is 5 × 5 instead of 14 × 14, and the step distance is changed from 6 to 3 based on the position of the auxiliary classifier in the convolutional neural network. The input of the two fully connected layers is 2048 and 1024 neurons, respectively. The output results of the two auxiliary classifiers were multiplied by the utility ratio set in VS-ResNet and then added to the final classification result. Adding the auxiliary classifiers increased the gradient of network backpropagation and alleviated the phenomenon of gradient disappearance.

3.4 Detailed Structure of VS-ResNet Network

For the feature extraction network module, VS-ResNet changes the residual block structure from the funnel model to the bottleneck model so that the activation function information can be better preserved, as shown in Table 1.

Table 1: Improved ResNet-50 network architecture

Network layer	Output	52 layer
Convolution layer 1	112 × 112	7 × 7, 64, Step-length 2
Convolution layer 2	56 × 56	3 × 3, Max pooling, Step-length 2
		$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256,\ \ C = 32 \\ 1 \times 1, 128 \end{bmatrix} \times 3$

(Continued)

Table 1 (continued)

Network layer	Output	52 layer
	The first auxiliary classifier	
Convolution layer 3	28 × 28	$\begin{bmatrix} 1\times 1, 512 \\ 3\times 3, 512,\ \mathrm{C}=32 \\ 1\times 1, 256 \end{bmatrix}\times 3$
Convolution layer 4	14 × 14	$\begin{bmatrix} 1\times 1, 1024 \\ 3\times 3, 1024,\ \mathrm{C}=32 \\ 1\times 1, 512 \end{bmatrix}\times 9$
	The second auxiliary classifier	
Convolution layer 5	7 × 7	$\begin{bmatrix} 1\times 1, 2048 \\ 3\times 3, 2048,\ \mathrm{C}=32 \\ 1\times 1, 1024 \end{bmatrix}\times 3$
1 × 1		Convolutional pooling layer, 1000 neurons, softmax classifier

Referring to the structural parameters of ResNeXt-50, the number of convolution channels of the first residual structure is changed from the original [64, 64, 256] to [256, 256], and the last three residual structures are also modified successively from [128, 128, 512] to [512, 512, 256]. [256, 256, 1024] is [1024, 1024, 512], and [512, 512, 2048] is [2048, 2048, 1024]. The group convolution method is referenced in the inverted residual structure, and the number of groups is set to 32. By referring to the model of the Swin Transformer algorithm [41], the original hierarchical structure [3, 4, 6, 3] is modified to [3, 3, 9, 3], and auxiliary classifiers are added after the second and fourth layers to accelerate the convergence speed of the network to a certain extent and alleviate the phenomenon of gradient disappearance.

3.5 Incorporate the CBAM Attention Mechanism

The attention mechanism can selectively ignore some inefficient information in the image, focus on the efficient information, reduce the resource consumption of the inefficient part, improve the network utilization, and enhance the target detection ability. Therefore, the CBAM attention mechanism is integrated into the feature extraction network. The channel and spatial attention mechanisms are connected [42] to form a simple but effective attention module, whose structure is shown in Fig. 6.

In the channel attention module, the global average pooling and maximum pooling of the same input feature space are performed respectively to obtain the spatial information of the feature map. Then the obtained feature space information is input into the multi-layer awareness mechanism module of the next layer for dimension reduction and dimension increase processing. First, the weight of the two shared convolution layers in the multi-layer awareness network is shared. Then add the characteristics of the perceptual network output and process it with the sigmoid activation function [43] to obtain channel attention. The calculation formula is shown in Eq. (4).

$$Mc(F) = \varepsilon[MLP(F^{c}_{avg}) + MLP(F^{c}_{\max})] \tag{4}$$

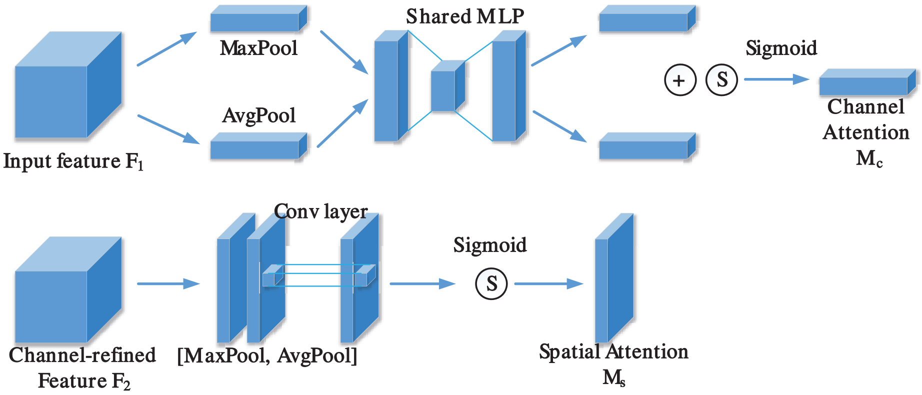


Figure 6: Schematic diagram of CBAM attention mechanism

Spatial attention features are complementary to channel attention and reflect the importance of input values in spatial dimensions. The calculation formula is shown in Eq. (5).

$$Ms(F) = \varepsilon\{conv_{7\times 7}[unit(F^{s}_{avg}, F^{s}_{\max})]\} \quad (5)$$

First, global average pooling and maximum global pooling of one channel dimension are performed on the feature map. Next, the two features are spliced, and finally, the dimension is reduced to one by 7 × 7 convolution post-channel processing using the sigmoid function to generate spatial attention feature maps [44]. Among them, Mc is the channel attention module calculation factor, Ms is the space attention module calculation factor, ε is the sigmoid activation function, MLP is the multi-layer perceptron, F represents the feature vector, *unit* is the channel combination, and *conv* is the convolution operation.

The CBAM is not embedded in all convolutional residual blocks but only acts after different convolutional layers to facilitate the use of pre-trained models during the experiment.

3.6 Introduce FPN Feature Pyramid Structure

To alleviate the unsatisfactory detection ability of the Faster RCNN algorithm for small-sized targets, the FPN feature pyramid network model was introduced into the Faster RCNN target detection algorithm. The FPN network model is divided into two network routes. One of the network routes produces multi-scale features from bottom to top, connecting the high-level features with high semantics and low resolution and the low-level features with high resolution and low semantics; another network route is from top to bottom, after some layer changes, the rich semantic information contained in the upper layer is transferred layer by layer to the low layer features for fusion. The FPN uses multi-level features and multi-scale anchor frames compared to the SSD [15] technique. In contrast, the SSD only predicts low-level data, making it difficult to ensure robust semantic features and having a poor detection effect for small objects.

Fig. 7 is a schematic diagram of the FPN feature pyramid structure. In the figure, the bottom-up feature map with computed order on the left is {C2, C3, C4, C5}, and the top-down feature pyramid structure on the right is {P2, P3, P4, P5}. The CF-RCNN algorithm uses the above VS-ResNet as the backbone extraction network of the FPN feature pyramid structure. The image part on the left side of Fig. 7 is a down-sampling model. The step size value is set as {4, 8, 16, 32} during this model's feature extraction operation. In the up-sampling model of the right image, the upper feature map is convolved with a convolution kernel of 1 × 1 size, the step size is set to 1, and the number of channels is 256 to adjust the dimension to be consistent so that it can be fused with the lower feature. Then, after 3 × 3 size convolution, the aliasing situation in the 2-fold up-sampling process is eliminated to obtain the feature map. {P2, P3, P4, P5} share the weight of RPN and Fast RCNN and use different anchor sizes of $\{32^2, 64^2, 128^2, 256^2\}$ and anchor ratio of {1:2, 1:1, 2:1} on {P2, P3, P4, P5} feature map to select candidate boxes.

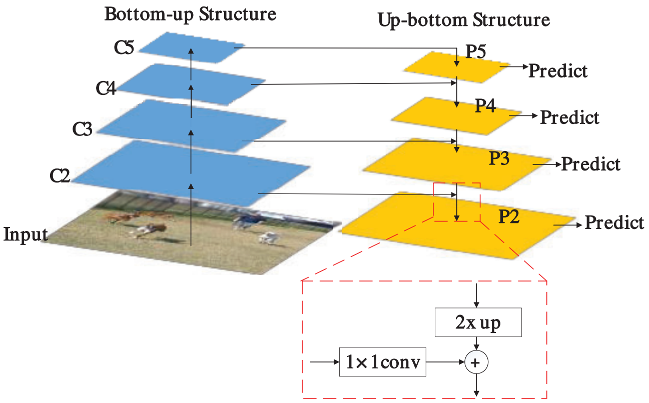


Figure 7: FPN feature pyramid structure

3.7 Improvement of Non-Maximum Suppression NMS Algorithm

Non-maximum suppression (NMS) [45] is an important part of the target detection algorithm, and its primary function is to eliminate redundant candidate boxes generated in the RPN network. The elimination criterion is shown in Eq. (6).

$$Si = \begin{cases} Si, IOU(M, bi) < Nt \\ 0, IOU(M, bi) \geq Nt \end{cases} \tag{6}$$

Si is the score of the *i*th candidate box; *M* indicates the candidate box with the highest score; *bi* is the candidate box to be scored; *IOU* is the combined ratio of *bi* and *M*; N_t is the threshold value of *IOU* set in NMS. When the ratio between the candidate box to be scored and the candidate box with the largest score exceeds the threshold, the NMS deletes the candidate box to be scored. As a result, NMS mistakenly deletes overlapping candidate boxes in partially crowded and truncated complex scenarios. To solve this problem, CF-RCNN uses the Soft-NMS algorithm [45].

$$Si = \begin{cases} Si, IOU(M, bi) & < Nt \\ Si(1 - IOU(M, bi)), IOU(M, bi) & \geq Nt \end{cases} \tag{7}$$

The Soft-NMS score function is shown in Eq. (7). If the combined ratio is less than the threshold, the score remains unchanged. If it is greater than or equal to the threshold, first calculate the difference between 1 and the combined ratio, then use the product of the difference and *Si* as the value of new *Si*. Compared with NMS, Soft-NMS is not directly overlapping candidate boxes but resets *Si* to a smaller score.

3.8 Overall Structure of CF-RCNN

The detailed structure of the CF-RCNN algorithm is shown in Fig. 8.

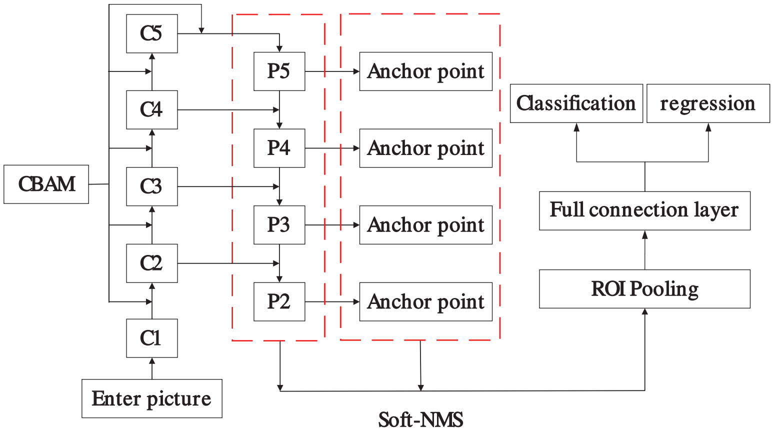


Figure 8: Overall structure of CF-RCNN

Firstly, CF-RCNN takes the image stream as input, extracts image features based on the inverted residual ResNet50 model, and obtains intermediate feature vector graphics; secondly, the CBAM attention mechanism is used to calculate the attention weight in the two dimensions of space and channel, and adjust the parameters of the intermediate feature map; thirdly, based on the FPN feature pyramid structure, the feature map is multi-scaled; fourthly, on each feature map, the RPN network uses a single scale to select regions that may contain objects, and balances the positive and negative sample ratios through Soft-NMS; finally, the classification and position regression operations are performed on the objects in the selected area.

4 Experimental Results and Analysis

The experiment is based on the deep learning network framework PyTorch 1.81. The programming language is Python 3.8, the CPU configuration is a 10-core Intel(R) Xeon(R) Gold 5218R CPU @ 2.10 GHz, and the memory is 64 GB. The GPU is configured as RTX 3090 + CUDA 1.1, with 24 GB of video memory. The operating system uses Ubuntu 18.04 version.

4.1 Public Dataset Preparation

To verify the effectiveness of the improvement measures in CF-RCNN for improving target detection performance, object recognition and target detection experiments are performed on the CIFAR-10 dataset [26] and the PASCAL VOC2012 [27] dataset, respectively. The CIFAR-10 dataset has a total of 60,000 images, divided into 10 types of objects to be recognized. Each type of object to be recognized has a capacity of 6,000 images, and the ratio of the training set and test set is 5:1. The PASCAL VOC2012 dataset contains 20 detection categories and a total of 17,125 images, of which 5,718 images are used in the training set and 5,824 images in the validation set. In addition, the expanded dataset of the PASCAL VOC2012 provides additional annotation enhancements. There are 13,487 images in the training, verification, and test sets, including 10,582 images in the training set. Among them, 1,456 images for testing and 1,449 for verification were selected for the experiment.

4.2 Object Recognition Experiment Results Analysis

The experiment compares the change in the error rate of VGG16, ResNet-50 and VS-ResNet network on the CIFAR-10 dataset with the increase in the number of iterations. The number of iterations is set to 35, the training batch is 16. When the learning rate is too high, it is easy to cause the LOSS function value to miss the lowest point. On the other hand, when the learning rate is too small, the convergence rate is high. As shown in Fig. 9, comparing the LOSS function values obtained after 35 iterations of different learning rates, the learning rates of the subordinate experiments are set to 0.0001.

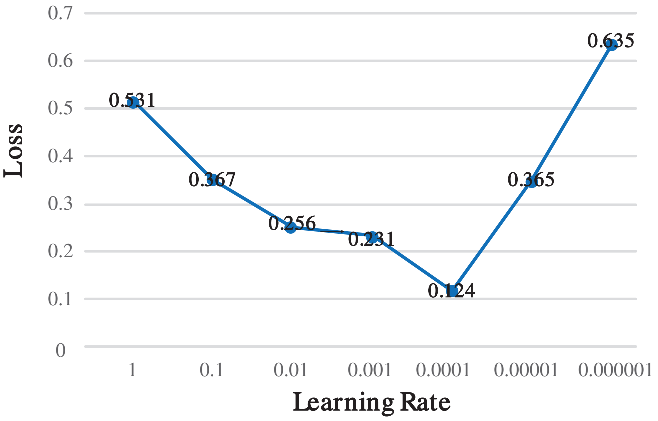


Figure 9: Comparison of different learning rates

The experimental results are shown in Table 2 and Fig. 10. The data in Table 2 shows that the ResNet-50 network performs better in the object classification task than the traditional VGG16 network, with an increase of 1.65 percentage points in the accuracy rate. Compared with the initial ResNet-50, the VS-ResNet network has increased by 1.32 percentage points and by 2.97 percentage points compared with the VGG16 model. Moreover, it can handle the most layers and the lowest error rate, which reflects the best processing effect.

Table 2: Comparison of the lowest error rates of different convolutional networks

Network model	Number of plies	Minimum error rate
VGG16	16	0.1106
ResNet-50	50	0.0941
VS-ResNet-50	52	0.0809

In Fig. 10, the abscissa is the number of iterations, and the ordinate is the test error rate. It can be seen from Fig. 10 that the VGG16 model has a specific advantage in the convergence speed, which is evident in the first 10 iterations. After 15 iterations, the descending gradient of the model becomes smaller and gradually tends to be saturated. The VS-ResNet model alleviates the problem of the slow convergence speed of ResNet-50 to a certain extent. The convergence improvement effect of the first 5 iterations is noticeable, and the classification error rate after 15 iterations has apparent advantages over ResNet-50 and VGG16. To sum up, the VS-ResNet network accelerates the convergence speed of ResNet50 and reduces the classification error rate.

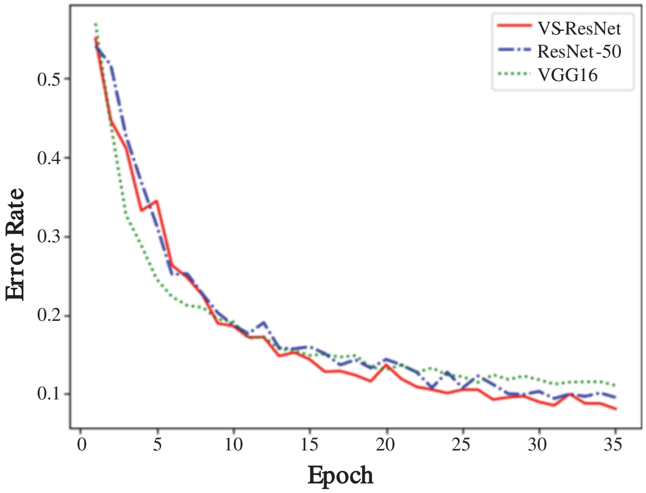


Figure 10: Comparison of error rates of different convolutional networks

4.3 Analysis of Target Detection Experiment Results

The target detection experiment is based on the traditional Faster RCNN algorithm, combined with the above optimization measures, to verify the impact of different optimization strategies on the algorithm's performance.

4.3.1 AP_{50} Detection Accuracy Comparison

The experiment's initial learning rate is 0.005, the momentum parameter is 0.9, and the weight attenuation coefficient is 0.0005. AP_{50} refers to the average precision of target detection when the Intersection over the Union (IoU) [45–47] threshold is 0.5. The batch size is 16, and the iterations are 20 k times. The experimental results are shown in Table 3. Compared with the data in Table 3, the detection accuracy AP_{50} of CF-RCNN is improved by 13.8 percentage points, reaching 76.1%, and the average processing time is slightly lower than that of Faster RCNN. However, it can still meet real-time requirements.

Table 3: Impact of different optimization strategies

VGG16	VS-ResNet	CBAM	FPN	Soft-NMS	Detection accuracy AP_{50} (%)	Average processing time (ms)
√					62.3	262
	√				65.8	269
	√	√			68.2	273
	√	√	√		73.9	286
	√	√	√	√	76.1	289
√		√	√	√	71.8	283

Note: √ indicates that the optimization strategy uses this module.

4.3.2 AP Detection Accuracy Comparison

The targets smaller than 32 × 32 pixels are selected as the experimental data set to compare the detection accuracy to verify the improvement measures' impact on small-sized targets' detection ability. The AP value is the average AP accuracy achieved for ten different IoU thresholds of 0.50:0.05:0.95. The results are shown in Table 4.

Table 4: Effect of FPN structure on fast RCNN performance

Algorithm	AP (%)
Faster RCNN (VGG16)	6.9
Faster RCNN (ResNet-101)	3.2
Faster RCNN + FPN	18.8
Faster RCNN + FPN + Soft-NMS	20.8

It can be seen from Table 4 that after FPN and Soft-NMS are introduced into the Faster RCNN model. ResNet-101 [48] has many layers, and higher-level data extraction is not conducive to detecting small-size objects. As the data shows, after ResNet-101 is integrated into Faster RCNN, the target detection accuracy of Faster RCNN decreases. The AP value for small-sized target detection is increased by 13.9 percentage points, alleviating the weak detection performance of the target detection algorithm based on small-sized objects.

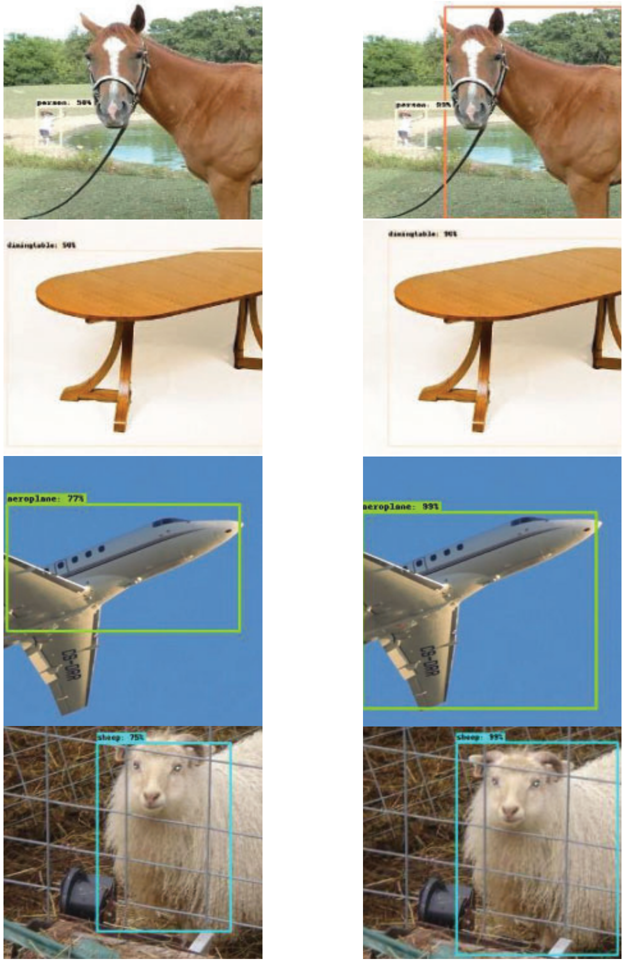
4.3.3 Truncated Objects Detection Accuracy Comparison

To verify whether the improved algorithm optimizes the detection ability of the original algorithm for truncated targets. First, manually edit the picture, intercept some detected objects, and use CF-RCNN and Faster RCNN to detect them. The test results are shown in Fig. 11. The experimental results show that the accuracy of CF-RCNN is higher than that of Fast RCNN, and the accuracy of the image frame and the possibility of classification are also higher.

In the first row, CF-RCNN identified a horse with only half body, but Faster RCNN did not do this. The recognition rate of the pictures in other corresponding picture groups is also improved. The detection results of the third row of aircraft show that CF-RNN can effectively identify small truncated targets and large-size targets, and the recognition of aircraft body is more complete.

4.3.4 Recall Rate Comparison

The data in Table 5 are the recall rate changes on the PASCAL VOC2012 dataset before and after algorithm optimization. AR is the average recall rate [49,50] which refers to taking the maximum recall rate for different IoUs and then calculating the average. The experimental data shows that the recall rate of the improved algorithm on PASCAL VOC 2012 has changed from the original 57.2% to 66.5%, which increases by 9.3 percentage points.



(a) Faster RCNN algorithm (b) CF-RCNN algorithm

Figure 11: Comparison of truncated target detection results

Table 5: Comparison of recall rate before and after algorithm optimization

Algorithm	AR (%)
Faster RCNN	57.2
CF-RCNN	66.5

4.3.5 Algorithm Efficiency Comparison

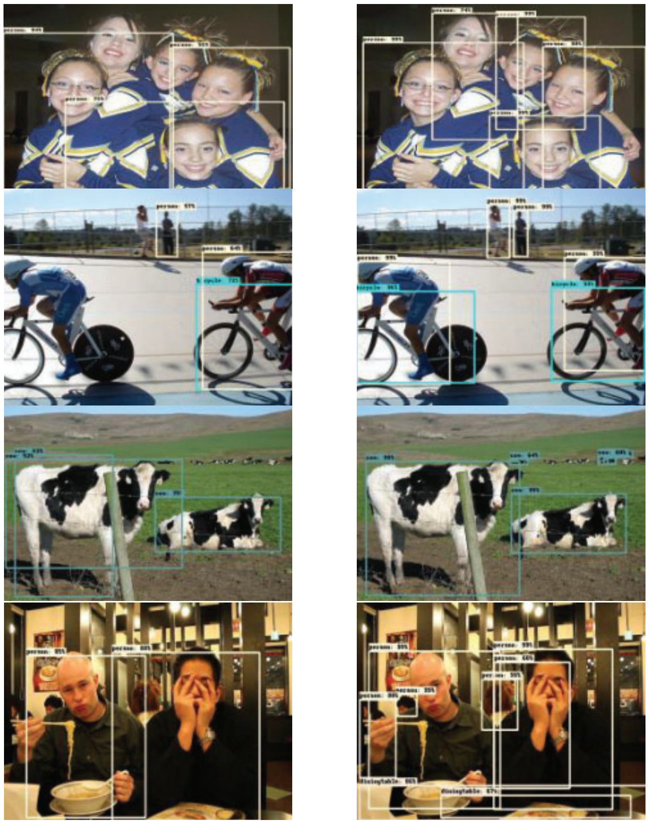
Different classical and CF-RCNN algorithms are used for comparative experiments, and several algorithms' AP_{50} values and average processing time changes are compared. The experiments are based on the same data set to ensure that the experimental environment of several algorithms is consistent. As shown in Table 6, the classical algorithms of the experiment include YOLOv5 [51] and SSD algorithms. The experimental results show that the algorithm proposed in this paper still has better advantages than the commonly used art methods such as YOLOv5 and SSD.

Table 6: Performance comparison between classical target detection algorithm and CF-RCNN algorithm

Algorithm	AP_{50} (%)	Average processing time (ms)
YOLOv5	74.2	76
SSD	61.3	83
CF-RCNN	76.1	99

4.3.6 Adaptability Comparison

Compared with the other two classical target detection algorithms, CF-RCNN has a longer processing time but meets the basic real-time requirements. The detection accuracy is higher than the other two algorithms. To test the adaptability of the CF-RCNN algorithm to the truncation and occlusion of objects, the images with object occlusion or truncation were screened out on the PASCAL VOC 2012 dataset, and Faster RCNN and CF-RCNN were used for comparative experiments. Fig. 12 shows part of the experimental results.



(a) Faster RCNN algorithm (b) CF-RCNN algorithm

Figure 12: Comparison of detection results before and after algorithm optimization

Fig. 12 shows that the traditional Faster RCNN algorithm can meet the detection requirements of truncated or occluded objects in a simple environment. Furthermore, the object classification results are not significantly different from those of the CF-RCNN algorithm but are inferior to the CF-RCNN algorithm in object location selection. Traditional Faster RCNN exhibits the highest rates of false detection and missed detection phenomena in the case of truncation or occlusion of objects in complex images.

5 Conclusion and Recommendation

In recent years, more and more scholars have entered the field of computer vision and artificial intelligence to explore and study, which has led to considerable development of technology in this field. The original Faster RCNN algorithm will gradually solve the problem of ignoring the detection results of small-size objects and occlusion or truncation objects. The improved CF-RCNN algorithm proposed in this paper is an exploratory attempt in this field, and it solves this problem well and makes up for the defects. Compared with the VGG16 network, the VS-ResNet network not only retains the character of the VGG16 network with fast iterative convergence speed but also improves the recognition accuracy of the network. Introducing FPN and CBAM modules enhances the detection ability of the algorithm for size or truncation objects. The experimental results show that the CF-RCNN algorithm can achieve good detection results in simple scenes and good stability in complex environments. However, it is easy to cause the problem of multiple prediction frames for a single object. The implementation process of the algorithm is detailed, but it also increases the algorithm's time complexity. The next step will explore and study the method of cascading multiple detectors to solve this problem.

Funding Statement: This work was sponsored by the Natural Science Research Program of Higher Education Jiangsu Province (19KJD520005), Qing Lan Project of Jiangsu Province (Su Teacher's Letter [2021] No. 11) and the Young Teacher Development Fund of Pujiang Institute Nanjing Tech University ([2021] No. 73).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] L. Rukhsar, W. H. Bangyal, K. Nisar and S. Nisar, "Prediction of insurance fraud detection using machine learning algorithms," *Mehran University Research Journal of Engineering & Technology*, vol. 41, no. 1, pp. 33–40, 2022.

[2] S. Pervaiz, W. H. Bangyal, A. Ashraf, K. Nisar, M. R. Haque *et al.,* "Comparative research directions of population initialization techniques using PSO algorithm," *Intelligent Automation & Soft Computing*, vol. 32, no. 3, pp. 1427–1444, 2022.

[3] K. Zhang, X. H. Feng, Y. R. Guo, Y. K. Su, K. Zhao *et al.,* "Overview of deep convolutional neural networks for image classification," *Journal of Image and Graphics*, vol. 26, no. 10, pp. 2305–2325, 2021.

[4] X. Tang, "Research on identification method for rice pests with small sample size problem based on deep learning," M.S. Dissertation, Anhui University, 2021.

[5] S. You and Z. H. Zhang, "Channel attention and spatial pyramid-based improved YOLOv3 and its application," *Intelligent Computer and Applications*, vol. 13, no. 2, pp. 179–186, 2023.

[6] B. Yang, X. S. Xu, J. Li and H. Zhang, "Landmark generation in visual place recognition using multi-scale sliding window for robotics," *Applied Sciences*, vol. 9, no. 15, pp. 3146, 2019.

[7] Y. X. Gu, X. L. Qin, Y. C. Peng and L. Li, "Content-augmented feature pyramid network with light linear spatial transformers for object detection," *IET Image Processing*, vol. 16, no. 13, pp. 3567–3578, 2022.

[8] H. X. Fu, G. Q. Song and Y. C. Wang, "Improved YOLOv4 marine target detection combined with CBAM," *Symmetry*, vol. 13, no. 4, pp. 623, 2021.

[9] Y. C. Li, S. L. Zhou and H. Chen, "Attention-based fusion factor in FPN for object detection," *Applied Intelligence*, vol. 52, no. 13, pp. 15547–15556, 2022.

[10] T. Feng, J. G. Liu, X. Fang, J. Wang and L. B. Zhou, "A double-branch surface detection system for armatures in vibration motors with miniature volume based on ResNet-101 and FPN," *Sensors*, vol. 20, no. 8, pp. 2360, 2020.

[11] Z. Y. Liu, L. Yuan, M. C. Zhu, S. S. Ma and L. Z. T. Chen, "YOLOv3 traffic sign detection based on SPP and improved FPN," *Computer Engineering and Applications*, vol. 57, no. 7, pp. 164–170, 2021.

[12] A. O. Salau, T. K. Yesufu and B. S. Ogundare, "Vehicle plate number localization using a modified GrabCut algorithm," *Journal of King Saud University–Computer and Information Sciences*, vol. 33, no. 4, pp. 399–407, 2019.

[13] G. Y. Yang, Z. Y. Wang, S. N. Zhuang and H. Wang, "PFF-CB: Multiscale occlusion pedestrian detection method based on PFF and CBAM," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 3798060, 2022.

[14] S. P. Zhai, D. R. Shang, S. H. Wang and S. S. Dong, "DF-SSD: An improved SSD object detection algorithm based on DenseNet and feature fusion," *IEEE Access*, vol. 8, pp. 24344–24357, 2020.

[15] K. M. T. Jawad, B. Maisha and S. Nazmus, "Targeted face recognition and alarm generation for security surveillance using single shot multibox detector (SSD)," *International Journal of Computer Applications*, vol. 177, no. 22, pp. 8–13, 2019.

[16] D. W. Wang, L. C. Hu, J. Fang and Z. J. Xu, "Small object detection algorithm based on improved double-head RCNN for UAV aerial images," *Journal of Beijing University of Aeronautics and Astronautics*, 2023. https://doi.org/10.13700/j.bh.1001-5965.2022.0591

[17] M. Tan, R. Pang and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. 2020 IEEE Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, USA, pp. 10778–10787, 2020.

[18] C. Y. Zhang, S. Zhang, H. T. Wang and X. K. Ran, "Multi-head attention detection of small targets in remote sensing at multiple scales," *Computer Engineering and Applications*, 2022. https://doi.org/10.3778/j.issn.1002-8331.2210-0366

[19] C. Y. Fu, W. Liu, A. Ranga, A. Tyagi and A. C. Berg, "DSSD: Deconvolutional single shot detector," in *Proc. 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2017)*, Honolulu, HI, USA, pp. 2881–2890, 2017.

[20] B. Singh and L. S. Davis, "An analysis of scale invariance in object detection snip," in *Proc. 2018 IEEE Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, pp. 3578–3587, 2018.

[21] A. A. J. Pazhani and C. Vasanthanayaki, "Object detection in satellite images by faster R-CNN incorporated with enhanced ROI pooling (FrRNet-ERoI) framework," *Earth Science Informatics*, vol. 15, no. 1, pp. 553–561, 2022.

[22] X. S. Chen, T. Su and W. M. Qi, "Printed circuit board defect detection algorithm based on improved faster RCNN," *Journal of Jianghan University (Natural Science Edition)*, vol. 50, no. 1, pp. 87–96, 2022.

[23] I. S. Walia, D. Kumar, K. Sharma, J. D. Hemanth and D. E. Popescu, "An integrated approach for monitoring social distancing and face mask detection using stacked ResNet-50 and YOLOv5," *Electronics*, vol. 10, no. 23, pp. 2996, 2021.

[24] A. V. Bobkov and K. Aung, "Real-time person identification by video image based on YOLOv2 and VGG 16 networks," *Automation and Remote Control*, vol. 83, no. 10, pp. 1567–1575, 2022.

[25] Y. Zhang, "A novel non-maximum suppression strategy via frame bounding box smooth for video aerobics target location," *Sensing and Imaging: An International Journal*, vol. 23, no. 1, pp. 29–44, 2022.

[26] X. Y. Lv, "CIFAR-10 image classification based on convolutional neural network," *Frontiers in Signal Processing*, vol. 4, no. 4, pp. 100–106, 2020.

[27] Z. X., Li, J. Zhang, J. L. Wu and H. F. Ma, "Semi-supervised adversarial learning based semantic image segmentation," *Journal of Image and Graphics*, vol. 27, no. 7, pp. 2157–2170, 2022.

[28] A. O. Salau and S. Jain, "Feature extraction: A survey of the types, techniques, applications," in *Proc. 5th IEEE Int. Conf. on Signal Processing and Communication (ICSC)*, Noida, India, pp. 158–164, 2019.

[29] J. H. Seong, S. H. Lee, W. Y. Kim and D. H. Seo, "High-precision RTT-based indoor positioning system using RCDN and RPN," *Sensors*, vol. 21, no. 11, pp. 3701, 2021.

[30] M. Catelani, L. Ciani, D. Galar and G. Patrizi, "Risk assessment of a wind turbine: A new FMECA-based tool with RPN threshold estimation," *IEEE Access*, vol. 8, pp. 20181–20190, 2020.

[31] H. Akiyoshi, K. Shinya and N. Ryohei, "Computerized classification method for histological classification of masses on breast ultrasonographic images using convolutional neural networks with ROI pooling," *Electronics and Communications in Japan*, vol. 105, no. 3, pp. 586–592, 2022.

[32] D. Szostak, A. Włodarczyk and K. Walkowiak, "Machine learning classification and regression approaches for optical network traffic prediction," *Electronics*, vol. 10, no. 13, pp. 1578, 2021.

[33] B. Anissa, K. Jamal and Z. Arsalane, "Face recognition using SVM based on LDA," *International Journal of Computer Science Issues (IJCSI)*, vol. 10, no. 4, pp. 171–179, 2013.

[34] J. R. R. Uijlings, K. E. A. Sande, T. Gevers and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[35] Y. B. Meng, D. W. Shi, G. H. Liu, S. J. Xu and D. Jin, "Dense irregular text detection based on multi-dimensional convolution fusion," *Optical Precision Engineering*, vol. 29, no. 9, pp. 2210–2221, 2021.

[36] Z. J. Deng and Q. H. Tian, "Improved inception-v3 network for gesture image recognition," *Computer Systems & Applications*, vol. 31, no. 11, pp. 157–166, 2022.

[37] D. P. Yadav, A. S. Jalal and V. Prakash, "Human burn depth and grafting prognosis using ResNeXt topology based deep learning network," *Multimedia Tools and Applications*, vol. 81, no. 13, pp. 18897–18914, 2022.

[38] G. Plonka, Y. Riebe and Y. Kolomoitsev, "Spline representation and redundancies of one-dimensional ReLU neural network models," *Analysis and Applications*, vol. 21, no. 1, pp. 127–163, 2023.

[39] J. Oh and M. Kim, "PeaceGAN: A GAN-based multi-task learning method for SAR target image generation with a pose estimator and an auxiliary classifier," *Remote Sensing*, vol. 13, no. 19, pp. 3939, 2021.

[40] Q. Chen, W. Y. Zhang, K. Zhu, D. Zhou, H. Dai *et al.,* "A novel trilinear deep residual network with self-adaptive dropout method for short-term load forecasting," *Expert Systems with Applications*, vol. 182, pp. 115272, 2021.

[41] Z. H. Liao, N. Fan and K. Xu, "Swin transformer assisted prior attention network for medical image segmentation," *Applied Sciences*, vol. 12, no. 9, pp. 4735, 2022.

[42] L. Li, B. H. Fang and J. Zhu, "Performance analysis of the YOLOv4 algorithm for pavement damage image detection with different embedding positions of CBAM modules," *Applied Sciences*, vol. 12, no. 19, pp. 10180, 2022.

[43] S. R. Swamy, "Coefficient bounds for al-oboudi type bi-univalent functions based on a modified sigmoid activation function and horadam polynomials," *Earthline Journal of Mathematical Sciences*, vol. 7, no. 2, pp. 251–270, 2021.

[44] Q. Xu, X. M. Xi, X. J. Meng, Z. Y. Qin, X. S. Nie *et al.,* "Difficulty-aware bi-network with spatial attention constrained graph for axillary lymph node segmentation," *Science China (Information Sciences)*, vol. 65, no. 9, pp. 74–85, 2022.

[45] F. S. Wang, Q. S. Wang, J. G. Chen and F. R. Liu, "A redundacy-reduced candidate box accelerator based on soft-non-maximum suppression," *Laser & Optoelectronics Progress*, vol. 58, no. 24, pp. 405–416, 2021.

[46] S. K. Wu, J. R. Yang, X. G. Wang and X. P. Li, “IoU-balanced loss functions for single-stage object detection,” *Pattern Recognition Letters*, vol. 156, pp. 96–103, 2022.

[47] A. S. Al-Fahoum, E. B. Jaber and M. A. Al-Jarrah, “Automated detection of lung cancer using statistical and morphological image processing techniques,” *Journal of Biomedical Graphics and Computing*, vol. 4, no. 2, pp. 33–42, 2014.

[48] A. Singh and D. Kumar, “Detection of stress, anxiety and depression (SAD) in video surveillance using ResNet-101,” *Microprocessors and Microsystems*, vol. 95, no. 104681, pp. 1–15, 2022.

[49] X. L. Yang, “Survey for performance measure index of classification learning algorithm,” *Computer Science*, vol. 48, no. 8, pp. 209–219, 2021.

[50] M. X. Wang, B. Y. Fu, J. B. Fan, Y. Wang, L. K. Zhang *et al.,* “Sweet potato leaf detection in a natural scene based on faster R-CNN with a visual attention mechanism and DIoU-NMS,” *Ecological Informatics*, vol. 73, pp. 101931, 2023.

[51] H. Y. Huang, Z. J. You, H. Y. Cai, J. F. Xu and D. X. Lin, “Fast detection method for prostate cancer cells based on an integrated resnet50 and YOLOv5 framework,” *Computer Methods and Programs in Biomedicine*, vol. 226, pp. 107184, 2022.