

Received April 10, 2020, accepted April 23, 2020, date of publication April 27, 2020, date of current version May 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2990477

# SSD Object Detection Model Based on Multi-Frequency Feature Theory

JINLING LI<sup>1</sup>, QINGSHAN HOU<sup>1</sup>, JINSHENG XING<sup>2</sup>, AND JIANGUO JU<sup>1</sup><sup>3</sup>

<sup>1</sup>School of Economics and Management, Shanxi Normal University, Linfen 041004, China

<sup>2</sup>School of Mathematics and Computer Science, Shanxi Normal University, Linfen 041004, China

<sup>3</sup>School of Information Science and Technology, Northwestern University, Xi'an 710127, China

Corresponding authors: Qingshan Hou (18435202842@163.com), Jinsheng Xing (xjs19640408@163.com), and Jianguo Ju (jig\_1227@163.com)

This work was supported in part by the Soft Science Foundation of Shanxi Province under Grant 2011041037-02, and in part by the Scholarship Council of Shanxi Province under Grant 2011-8.

**ABSTRACT** In order to further improve the accuracy and real-time performance of the traditional Single Shot Multibox Detector (SSD) object detection model, an improved SSD multi-object detection model is proposed. Firstly, aiming at the defect of weak correlation between prediction object score and positioning accuracy in the traditional SSD model, the improved model enhanced the correlation between the two by adding Intersection Over Union (IoU) prediction loss branch. Secondly, in order to reduce the spatial redundancy of traditional SSD model, a multi-frequency feature component convolution module is designed, which greatly reduces the calculation overhead and hardware overhead of the traditional model. Finally, in order to accelerate the convergence speed of the improved model, the Adaptive and Momental Bound (AdaMod) optimizer is introduced to modify the adaptive learning rate of the improved model which is too large in the training process. Experimental results show that the improved model has stronger detection capabilities, better overall detection results, and improved detection accuracy and real-time detection.


**INDEX TERMS** Object detection, IoU predicting loss, SSD, multi-frequency feature component convolution, AdaMod optimizer.

## I. INTRODUCTION

With the continuous improvement and development of the object detection technology, there are more ideal use experience and a wide range of applications. In the aspect of traffic [1], object detection can be efficiently completed by detecting pedestrian, vehicle, road signs, traffic lights and other objects on the road to assist traffic management. In the medical field [2], object detection is often applied to the pathological detection and recognition of images, which makes a significant contribution to the prevention and cure of diseases. The application of these aspects shows that the work done by the object detection instead of the human is very efficient and convenient. In the military field [3], object detection is usually used to track the missile hitting the target, which plays an important role in the intelligent development of the military field. In the field of security [4], object detection is used to track suspicious vehicles and

detect illegal and criminal behaviors in real time, providing a favorable guarantee for social security protection. At present, there are two kinds of common object detection algorithms, which are based on two-stage and single-stage processes. In the two-stage object detection algorithm, the candidate boxes area are first generated, and then the candidate boxes are classified and adjusted. Its main representative algorithms are R-CNN series algorithms. Such algorithms have high accuracy, but there are problems such as slow detection speed and low real-time detection of algorithms. However, the object detection algorithm based on single-stage process generates object bounding box and probability classification directly, without the need to generate the candidate box areas, which improves the speed of object detection, but the accuracy rate has decreased. Representative object detection algorithms based on single-stage and two-stage processes include SSD [5] and Faster R-CNN [6].

In order to solve the problems in object detection, many scholars at home and abroad have done a lot of research work. In 2014, Ross Girshick *et al.* proposed R-CNN

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

(Region-based Convolutional Neural Networks) [7] algorithm, which introduced the deep learning [8] model CNN (Convolutional Neural Networks) into the object detection field for the first time, and the detection effect was significantly improved compared with the traditional object detection algorithms. However, the R-CNN algorithm needs to acquire the features of 2000 RoI (Region of Interest) from select search [9] no matter in the training or prediction stage of the model, which results in the slow detection speed of the network model. In addition, the process of feature extraction can't be updated because the relevant feature extractor of CNN is separated from the SVM used for prediction. Therefore, after R-CNN algorithm, Ross Girshick proposed Fast R-CNN [10] algorithm in 2015, which was optimized for the defects of R-CNN and improved the training and prediction speed of R-CNN detection algorithm to some extent. After that, Faster R-CNN was proposed by Shaoqing Ren *et al.* Based on Fast R-CNN, this algorithm constructed a region proposal network (RPN). The prediction network directly generated Region Proposals instead of the ROI obtained by selecting the search method. With the help of RPN, the detection speed of Faster R-CNN was further improved. Because R-CNN needs to obtain a large number of proposals, and the large amount of overlapping proposals causes a lot of unnecessary repetitive work. You only look once (YOLO) [11] modified the prediction idea based on proposal in the R-CNN series of algorithms, dividing the input image into several small cells, and making prediction in each small cell. YOLO algorithm realized the end-to-end detection effect and improved the detection rate, but the detection accuracy was deficient due to its coarse granularity. The SSD algorithm draws on the ideas of the YOLO cell and the anchor mechanism of Faster R-CNN, which is a multi-object detection algorithm that directly predicts the object categories and boundary boxes. The SSD algorithm uses the methods of generating default box and convolution prediction to achieve the purpose of multi-object detection by comprehensive utilization of the output feature maps of different convolution layers. The SSD algorithm generates multiple default boxes for each predicted position in the output feature maps and sets different aspect ratios and sizes. During the prediction, SSD algorithm generates categories scores for the object in each default box and processes the corresponding default boxes. However, there are some problems in SSD detection algorithm, high network space redundancy and the high redundancy between each feature map, which are not conducive to the accurate positioning of the objects in the input image. Therefore, Fu *et al.* improved the original algorithm by combining stronger feature extraction network and adding more context information through the deconvolutional module, and proposed the Deconvolutional Single Shot Detector (DSSD) [12] model. However, with the replacement of feature extraction network and the addition of deconvolution module, the real-time detection performance of model is greatly reduced. In order to improve the detection accuracy of SSD object detection algorithm, Jeong *et al.* [13]

improved the method of feature fusion, so as to make full use of the features of each output layer. Li *et al.* proposed the feature fusion single shot multibox detector (FSSD) [14] model, which obtained more details of the output feature layers through feature fusion and down-sampling, so as to improve the detection accuracy of the model.

In order to improve the detection accuracy and real-time performance of traditional SSD object detection algorithm, the model of this paper makes the following related work: Firstly, to enhance the correlation between object score and positioning accuracy, the IoU prediction loss branch was added to the improved model. Secondly, in order to reduce the spatial redundancy of the SSD model, the multi-frequency feature component convolution module is designed for the original model. Finally, in order to accelerate the convergence of the improved model, the abnormal adaptive learning rate during the model training process was modified by the AdaMod optimizer [15].

## II. RELATED WORK

### A. MULTI-FREQUENCY FEATURE THEORY

With the continuous improvement and development of related technologies in the field of computer vision, convolutional neural network has been applied in many fields such as object detection, image recognition [16], semantic segmentation [17] *et al.*, and has achieved great success. Although in recent years, convolutional neural network has made some achievements in reducing the redundancy of relevant model parameters [18]–[20] and channel dimension of network feature maps [21]–[24], the output feature maps generated by convolutional neural networks still have a lot of redundant information in the spatial dimension. In the output feature maps generated by the network model, each location separately stores the relevant feature information of its own location. However, the feature information stored in adjacent locations is often similar, and these public information can be stored and processed together. Due to the existence of a large amount of redundant information, the execution efficiency of the network model is reduced.

Perform a related Fourier transform on the natural image. The transformed image usually contains two parts: low frequency and high frequency. The region with slow change in the grayscale image of the natural image corresponds to the low frequency part, while the region with drastic change corresponds to the high frequency part. The low frequency region represents the overall structure of the natural image, while the high frequency part focuses more on the detail edge in the natural image. Inspired by this, the multi-frequency feature theory [25] divides the relevant output feature maps of convolutional neural network into high-frequency region and low-frequency region. In order to reduce the spatial redundancy information, the low-frequency information that changes more gently is stored in a tensor with lower dimension.

On the premise of satisfying the information exchange and update between different frequencies, the low frequency region and high frequency region of the feature maps are operated separately by convolution kernel. The relevant flow of multi-frequency feature representation is shown in figure 1.

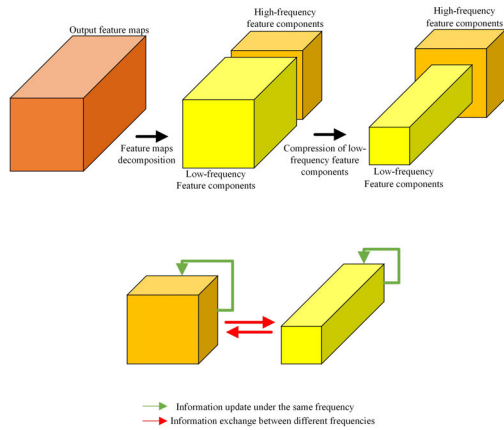


FIGURE 1. Relevant process of multi-frequency feature representation.

In order to perform correlation operations on multi-frequency feature maps, it can be seen from Figure 1 that the multi-frequency feature theory is extended to the general convolution neural network, and divides the feature maps of the general convolutional neural network into high-frequency and low-frequency feature maps. By sharing adjacent location information, the spatial features of low-frequency groups are reduced and stored in the tensor of low-resolution, so as to reduce spatial redundancy. With the reduction of spatial redundant information, the memory resource consumption and the computation cost of the convolutional neural network are greatly reduced.

### B. ADAMOD OPTIMIZER

In order to accelerate the convergence of the algorithm, the Adma [26] algorithm is widely used at present, but due to its poor convergence, not only the applicability of the algorithm is limited, but also the convergence result is not ideal. Therefore, in order to obtain better experimental results, the Stochastic Gradient Descent (SGD) [27] algorithm is still widely used in sample classification prediction. But better experimental results are at the expense of real-time detection. Therefore, the AdaMod optimizer based on the Adma algorithm is proposed. During the training process of the network model, the AdaMod optimizer does not need to warm up and is not sensitive to the learning rate of the network model. By calculating the average value of the adaptive learning rate, the abnormal learning rate in the training process is modified, thus improving the convergence of the optimizer. The AdaMod optimizer principle is as follows:

The related parameters of the optimizer are set, including step length  $\epsilon$ , moment estimation exponential decay rate  $\rho_1$ ,  $\rho_2 \in [0,1]$ , smaller constant value  $\delta$  for numerical stability, and initial parameter  $\theta$ . The first moment and second moment

variables  $s$ ,  $r$  and time step  $t$  are initialized. Randomly select  $m$  samples from the training data set,  $\{x_1, x_2, x_3, \dots, x_m\}$ , the corresponding relevant target is  $y_i$ . The gradient of the relevant sampling data set is calculated by equation (1) and the time step is updated.

$$g \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(x_i; \theta), y_i)$$

$$t = t + 1 \tag{1}$$

Through equations (2) and (3), the first moment and second moment estimates are updated.

$$s \leftarrow \rho_1 s + (1 - \rho_1) g \tag{2}$$

$$r \leftarrow \rho_2 r + (1 - \rho_2) g^2 \tag{3}$$

The deviation of the first moment and the second moment is corrected by equations (4) and (5).

$$\hat{s} \leftarrow \frac{s}{1 - \rho_1^t} \tag{4}$$

$$\hat{r} \leftarrow \frac{r}{1 - \rho_2^t} \tag{5}$$

Equations (6) and (7) are used to update the parameter  $\theta$ .

$$\Delta\theta = -\epsilon \frac{\hat{s}}{\sqrt{\hat{r}} + \delta} \tag{6}$$

$$\theta \leftarrow \theta + \Delta\theta \tag{7}$$

The above process expresses the optimization principle of the Adma optimizer. Based on the Adam optimizer, the AdaMod optimizer adds a hyper-parameter  $\rho_3$  to describe the length of memory during the model training process. Before updating the relevant parameter  $\theta$ , the relevant update operation of the smooth value  $u$  is added, as shown in equation (8).

$$v_t = \frac{\epsilon}{\sqrt{\hat{r}} + \delta}$$

$$u_t = \rho_3 u_{t-1} + (1 - \rho_3) v_t \tag{8}$$

In formula (8),  $\rho_3$  represents the measure of memory length, and  $1/\rho_3$  represents the range of exponential average sliding. The closer the value of  $\rho_3$  is to 1, the larger the memory range of the optimizer. After the smooth value  $u_t$  is calculated, it is compared with the learning rate  $v_t$  calculated by the optimizer. In order to avoid a high learning rate in the training process, the smaller value of the two is selected to update the relevant parameter  $\theta$ . The relevant description can be expressed by equation (9).

$$\theta_t = \theta_{t-1} - \min(v_t, u_t) \hat{s} \tag{9}$$

As described in the related literature of AdaBound [28], abnormal learning rate and the fluctuation of learning rate generally appear at the end of training, which is not conducive to the convergence and generalization of the optimizer.

Referring to the idea of exponential moving average, the AdaMod optimizer first calculates the low-order moment value of the gradient. Secondly, the hyper-parameter  $\rho_3$  is introduced to describe the length of memory in the training

process of the model. Through the parameter  $\rho_3$ , the long-term memory in the training process is introduced into the next step of the optimizer process, so as to avoid the optimizer falling into a bad state and trim the adaptive learning rate of excessive value. Thus the generalization and convergence of the optimizer are improved. In addition, the AdaMod optimizer can control the change of the adaptive learning rate at the beginning of training to ensure the stability of the training start and training process, eliminating the relevant “warm-up phase”, so that the convergence result of the optimizer is better, the convergence speed is faster, and the overall performance is better.

### III. ALGORITHM DESIGN

#### A. LOSS FUNCTION

As a typical single-stage object detector, SSD object detection model has the advantages of simplicity and efficiency, and has been widely used in many fields [29], [30]. However, due to the low correlation between the predicted object category score of the model and the accuracy of object positioning, which leads to the decline of the performance of the SSD model.

In terms of loss function, the improved SSD model enhances the correlation between the object classification score and the object positioning accuracy. The improved model uses Visual Geometry Group 16 (VGG-16) [31] as the basic network, and adds an IoU prediction loss branch to predict the IoU value between the default bounding box and the real bounding box. Multiply the classification score of the predicted object and the predicted IoU value, and use the result as the detection confidence of the improved model. The improved model includes classification loss branch, regression loss branch and IoU prediction loss branch. The loss structure of the improved SSD model is shown in figure 2.

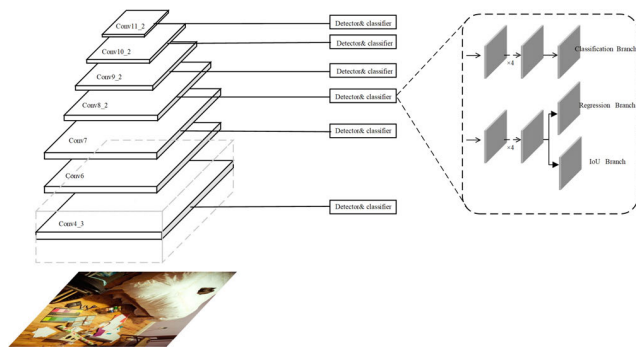


FIGURE 2. Improving the loss structure of SSD mode.

An IoU loss prediction branch was added to the improved SSD model. In order to ensure the effectiveness of the improved scheme and not affect the efficiency of the model, the head of the IoU prediction branch is similar to other branches, only including a  $3 \times 3$  convolution layer, and the sigmoid activation function layer ensures that the predicted value of IoU is at  $[0,1]$ . Since the range of IOU prediction

value is between  $[0,1]$ , the Binary Cross Entropy (BCE) loss is taken as the IOU prediction loss, which can be expressed by equation (10). In addition, the classification loss of the improved model adopts Cross Entropy (CE) loss, while the regression loss adopts smooth L1 loss, which is respectively represented by equations (11) and (12). In the training process of the model, the three loss branches participate in the training together.

$$L_{IoU} = \frac{1}{N_{Pos}} \sum_{i \in Pos} BCE(IoU_i, I\hat{o}U_i) \quad (10)$$

$$L_{cls} = \frac{1}{N_{Pos}} \left( \sum_{i \in Pos} CE(p_i, \hat{p}_i) + \sum_{i \in Neg} CE(p_i, \hat{p}_i) \right) \quad (11)$$

$$L_{loc} = \frac{1}{N_{Pos}} \sum_{i \in Pos} \sum_{m \in \{cx, cy, w, h\}} smooth_{L1}(I_i^m - \hat{g}_i^m) \quad (12)$$

The total loss of the model can be expressed by equations(13).

$$L_{all} = L_{cls} + L_{loc} + L_{IoU} \quad (13)$$

Before outputting the prediction box of the object to be detected, the final score of the default box is calculated by equation (14), in which the parameter  $\rho$  is used to control the weight of the category score and the IoU value. Compared with a single classification score, the calculation method of the default box score enhances the correlation between the detection confidence and the positioning accuracy. Applying the calculation results to the ranking in the NMS process can better suppress the poor local detection.

$$S = p_i^\rho IoU_i^{1-\rho} \quad (14)$$

#### B. MODEL STRUCTURE DESIGN

##### 1) DISASSEMBLY OF FEATURE MAPS

Based on the scale space theory, a series of Gaussian filters can be used in the processing of related images. With the change of the size of the Gaussian filter, the representation of the image at different scales can be obtained. Assuming that  $H(x, y)$  represents a two-dimensional image and the two-dimensional Gaussian function is  $G(x, y; t)$ , the linear scale space of the relevant image can be obtained by convolution of the two, as shown in equation (15).

$$\begin{aligned} L((x, y; t) &= H(x, y) * G(x, y; t) \\ &= H(x, y) * \frac{1}{2\pi t} e^{-\frac{x^2+y^2}{2t}} \end{aligned} \quad (15)$$

where  $t = \sigma^2$  represents the variance of the Gaussian filter, which is called the scale parameter. The larger the value of  $t$ , the more dramatic the smoothing of the related image. When  $t = 0$ , the image is not smoothed.

The convolution result is equivalent to the image itself. Similarly, the output feature maps of the convolutional layer can be divided into two parts, high-frequency features and low-frequency features. The low-frequency feature components of the correlation feature maps are obtained by the



Gaussian filter with  $t = 2$ . The components not processed by the Gaussian filter are called high-frequency feature components. Due to the redundancy of the low-frequency features, the low-frequency component correlation feature maps was halved to 1/2 of the high-frequency component correlation feature maps.

Suppose the input feature tensor of the convolutional layer of the improved SSD model is  $X \in \mathbb{R}^{c \times h \times w}$ , where  $c$  represents the number of feature maps, and  $h$  and  $w$  are the spatial dimensions of the input tensor.  $X$  is divided into two parts: high-frequency feature component  $X_H$  and low-frequency feature component  $X_L$ .  $X_H \in \mathbb{R}^{(1-\beta)c \times h \times w}$ ,  $X_L \in \mathbb{R}^{\beta c \times h \times w}$ ,  $\beta \in [0, 1]$  represents the proportion allocated to low frequency feature component.

### 2) CONVOLUTION OPERATION BASED ON HIGH AND LOW FREQUENCY FEATURE MAPS

Although the decomposition of the feature maps can effectively reduce the spatial redundancy, it is also accompanied by corresponding problems. Because of the difference of spatial resolution between the low-frequency feature and the high-frequency feature, the traditional convolution calculation cannot directly operate the decomposed input feature tensor. In order to act directly on the decomposed feature tensor  $X = \{X_L, X_H\}$ , so as to avoid extra computing cost and hardware overhead, the following strategies are adopted.

It is assumed that  $W \in \mathbb{R}^{c \times k \times k}$  represents the  $k \times k$  convolution kernel of the improved model, and  $X, Y \in \mathbb{R}^{c \times h \times w}$  represent the input and output of the correlation convolution calculation. Based on section 3.2.1, the input characteristic tensor  $X$  of convolution calculation is divided into two parts: high and low frequency characteristic components,  $X = \{X_H, X_L\}$ , and the corresponding output  $Y = \{Y_H, Y_L\}$ . Suppose that the input and output of convolution calculation have the same dimension  $c$ , that is,  $c_{in} = c_{out} = c$ . In order to obtain the convolution result  $Y$ , the convolution kernel  $W$  is divided into  $W_H$  and  $W_L$ , as shown in Figure 3, and corresponding convolution calculation process is represented by equations (16) and (17).

$$Y_H = f(X_H; W_{H \rightarrow H}) + \text{upsample}(f(X_L; W_{L \rightarrow H}), 2)$$

$$= Y_{p,q}^{H \rightarrow H} + Y_{p,q}^{L \rightarrow H}$$

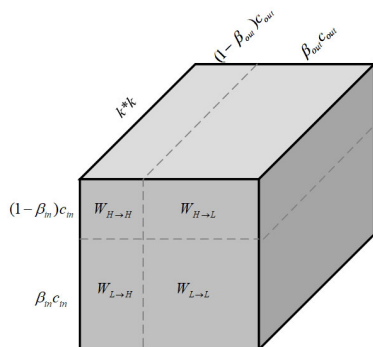


FIGURE 3. Decomposition of convolution kernel.

$$= \sum_{i,j \in M_k} W_{i+\frac{k-1}{2}, j+\frac{k-1}{2}}^{H \rightarrow HT} X_{p+i, q+j}^H$$

$$+ \sum_{i,j \in M_k} W_{i+\frac{k-1}{2}, j+\frac{k-1}{2}}^{L \rightarrow HT} X_{\lfloor \frac{p}{2} \rfloor + i, \lfloor \frac{q}{2} \rfloor + j}^L \quad (16)$$

$$Y_L = f(X_L; W_{L \rightarrow L}) + f(\text{pool}(X_H, 2); W_{H \rightarrow L})$$

$$= Y_{p,q}^{L \rightarrow L} + Y_{p,q}^{L \rightarrow H}$$

$$= \sum_{i,j \in M_k} W_{i+\frac{k-1}{2}, j+\frac{k-1}{2}}^{L \rightarrow LT} X_{p+i, q+j}^L$$

$$+ \sum_{i,j \in M_k} W_{i+\frac{k-1}{2}, j+\frac{k-1}{2}}^{H \rightarrow LT} X_{2^*p+0.5+i, 2^*q+0.5+j}^H \quad (17)$$

For the  $Y_{H \rightarrow H}$  part in equation (16), the information is updated using normal convolution calculation. While the  $Y_{L \rightarrow H}$  part, the low frequency feature component  $X_L$  is up-sampled, and then the corresponding convolution operation is performed. Similarly,  $Y_{H \rightarrow L}$  in equation (17) is processed by average pooling. In equations (16) and (17),  $(p, q)$  represents the position coordinate,  $M_k = \{(i, j): (i = \{-\frac{k-1}{2}, \dots, \frac{k-1}{2}\}, j = \{-\frac{k-1}{2}, \dots, \frac{k-1}{2}\})\}$ ,  $Y_{H \rightarrow H}$  and  $Y_{L \rightarrow L}$  represent the internal information update of the high and low frequency feature components.  $Y_{H \rightarrow L}$  and  $Y_{L \rightarrow H}$  represent the information exchange between the high and low frequency feature components.

### 3) THE CONVOLUTION OPERATION MODULE AND COMPATIBILITY PROCESSING

The traditional SSD object detection model takes VGG16 as the basic network and replaces the fc7 of the basic network with conv7. By adding Conv8\_2, Conv9\_2, Conv10\_2, Conv11\_2 detection layers to increase the convolution depth. The detection model combines Conv4\_3, Conv7, Conv8\_2 and other convolutional layers to detect and identify the objects.

To enhance the detection efficiency of the detection model, reduce the calculation overhead and hardware overhead of the model, the improved model processed the ordinary convolution layers of the traditional SSD detection algorithm, decomposed the relevant input feature tensor, and compressed the spatial resolution of the low-frequency feature component. Then the purpose of reducing model calculation overhead and related memory overhead is achieved. In addition, through the setting of the switch control parameter  $\beta$ , the processed ordinary convolution layers has good compatibility with the convolution layers participating in the object detection, and the relevant process is shown in figure 4.

In the related improved SSD detection algorithm, in order to convert ordinary features into multi-frequency feature components for representation, the algorithm sets  $\beta_{out} = 1$  and  $\beta_{in} = 0$  in the Conv1\_1 layer. Except for the relevant convolutional layers used for object detection, the remaining convolutional layers are all set to  $\beta_{out} = \beta_{in} = \beta$ . In order to ensure the compatibility of the multi-frequency feature component convolution modules and the object detection layers, it is necessary to transform the multi-frequency feature representation outputs from the general convolution layers

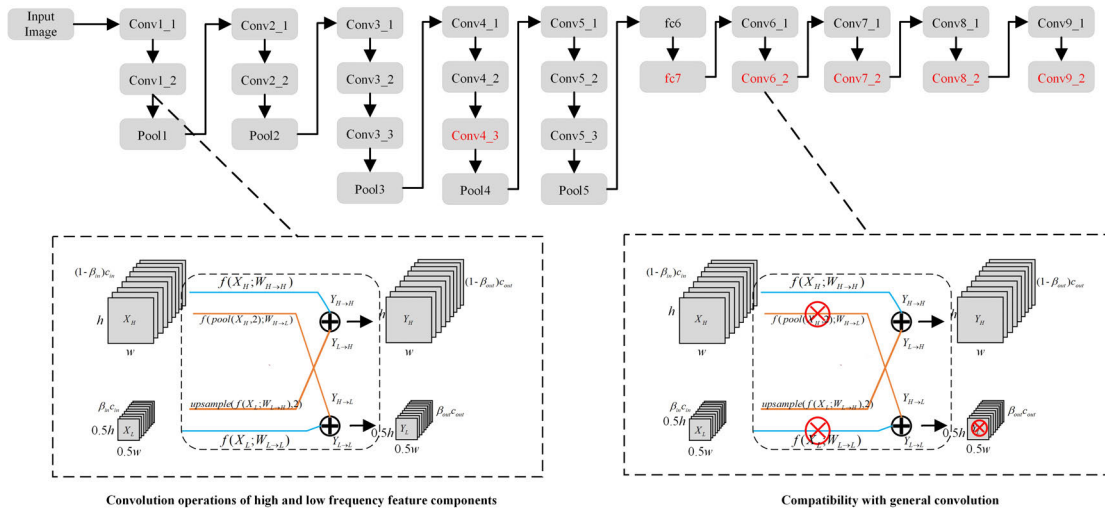


FIGURE 4. Convolution operation and compatibility processing of the improved model.

into the common feature representation. Therefore, before relevant object detection, set  $\beta_{out}$  to 0. At this time, the convolution path related to the output of the low-frequency feature components will be disabled, so as to generate the output of a single full-feature layer for the detection of related objects in the input image.

C. OPTIMIZATION OF TRAINING PROCESS

To further improve the real-time detection of the SSD model and accelerate the convergence speed of the model. Instead of using the traditional SGD algorithm, the improved model uses the AdaMod optimizer to optimize the real-time performance of model detection. The AdaMod optimizer is used to adjust the abnormal adaptive learning rate during the training of the improved model, which guarantees the stability of the training process. The AdaMod optimizer introduces a hyper-parameter to describe the length of memory during the model training, which improves the generalization and convergence of the SSD model and accelerates the model’s convergence speed.

IV. EXPERIMENT

A. EXPERIMENTAL DATA SETS AND EVALUATION INDICATORS

The related experiments are based on the MS COCO data set, which contains approximately 118,000 training images, 5000 verification images, and 20,000 unlabeled test images. It contains 500,000 labeled objects from 80 object categories. In addition, in order to verify the generality of the improved SSD algorithm, the relevant experiments based on the PASCAL VOC2012 data set were expanded, which is composed of 17125 training images and 5138 test images.

The evaluation indicators of the improved model are carried out from two aspects: On the one hand, the detection accuracy of the improved SSD model is measured by the following four types of AP values:  $AP_{0.9}$  value when the IoU

threshold is set to 0.9,  $AP_{0.75}$  value when the IoU threshold is set to 0.75,  $AP_{0.5}$  value when the IoU threshold is set to 0.5, and the average AP value (Average for  $AP_{0.5}$ ,  $AP_{0.75}$  and  $AP_{0.9}$ ). On the other hand, the FPS value is used to evaluate the real-time performance of the improved model.

B. RELATED RESEARCH

1) RESEARCH ON SETTING OF HYPER-PARAMETER  $\rho$  IN IOU PREDICTION BRANCH

In order to explore the impact of the IoU prediction loss branch on the improved SSD model, a series of studies have been carried out on the relevant parameters  $\rho$ . The relevant IoU prediction loss branch uses binary cross-entropy loss, and the detection confidence of the improved SSD model is calculated by the formula  $S = p_i^\rho IoU_i^{1-\rho}$ . The results of the exploration are shown in tables 1 and 2.

TABLE 1. Effect of hyper-parameter  $\rho$  on AP values of SSD 300 (based on MS COCO data set).

$\rho$	AP	$AP_{0.5}$	$AP_{0.75}$	$AP_{0.9}$
SSD algorithm	26.47	43.1	25.8	10.5
$p_i * IoU_i$	28.33	45.3	27.3	12.4
1.0	27.87	46.2	26.4	11.0
0.9	28.60	47.5	26.8	11.5
0.8	29.17	48.1	27.3	12.1
0.7	29.63	48.6	28.0	12.3
0.6	29.83	49.0	28.3	12.2
0.5	30.27	49.4	28.6	12.8
0.4	30.83	49.7	29.4	13.4
0.3	30.77	50.1	29.1	13.1
0.2	28.80	44.7	28.1	13.6
0.1	26.70	42.3	24.9	12.9
0	0.37	0.4	0.4	0.3

The detection confidence of the model depends on the two parts of the category score and the IoU value, and the relevant contribution of the category score and the IoU value to the model detection confidence depends on the parameter  $\rho$ . It can be seen from table 1 that when the value of  $\rho$

**TABLE 2. Effect of hyper-parameter  $\rho$  on AP values of SSD 300 (based on PASCAL VOC2012 data set).**

$\rho$	AP	AP <sub>0.5</sub>	AP <sub>0.75</sub>	AP <sub>0.9</sub>
SSD algorithm	52.77	75.8	58.2	24.3
$p_i * IoU_i$	53.83	76.4	59.4	25.7
1.0	53.63	76.8	58.9	25.2
0.9	54.30	77.2	59.8	25.9
0.8	54.73	77.4	60.5	26.3
0.7	55.17	77.8	60.9	26.8
0.6	55.60	78.3	61.4	27.1
0.5	56.00	78.5	61.7	27.8
0.4	56.63	79.1	62.3	28.5
0.3	56.80	79.4	62.1	28.9
0.2	51.17	74.6	56.4	22.5
0.1	46.37	69.8	50.7	18.6
0	0.6	0.8	0.6	0.4

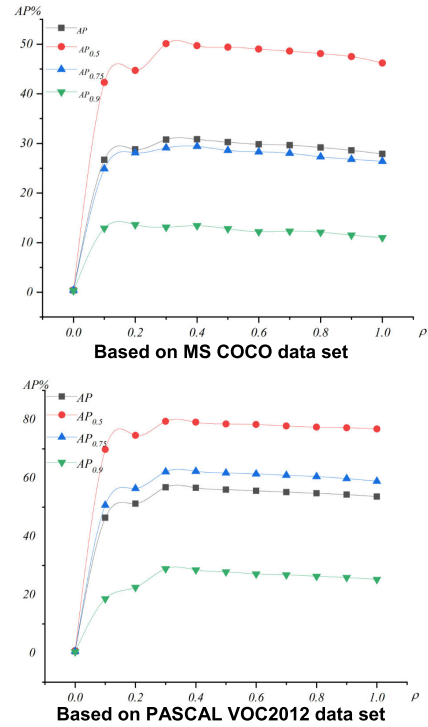
is set to 1, the average accuracy value AP of the model is slightly increased by 1.4% compared with the original SSD algorithm, which indicates that IOU prediction loss branch of the improved SSD model is beneficial to improve model performance. When the value of  $\rho$  is set to 0.4, the average accuracy value AP of the improved model reaches the best, which is 4.36% higher than the original SSD algorithm. In addition, according to the experimental data in table 2, it can be seen that when  $\rho$  is 1, the average accuracy value AP is slightly improved by 0.86% compared with the original algorithm. When  $\rho$  is 0.3, the average accuracy value AP is 56.8% and the accuracy is improved by about 4%.

The experimental data in table 1 and table 2 are respectively based on the experimental results of MS COCO and Pascal VOC 2012 data sets, which verify that the IoU prediction loss branch has good generalization on different data sets and the effectiveness of the model accuracy performance improvement. In addition, in the process of  $\rho$  value continuously decreasing from 1 to 0.3, the contribution of the predicted IoU value to the detection confidence is continuously increasing, and the AP values of the relevant improved SSD model tend to increase, which obviously indicates the relationship between the prediction loss branch of IOU and the positioning accuracy of the model, and effectively improves the performance of the model. The change of AP value is shown in Figure 5.

2) RESEARCH ON SETTING HYPER-PARAMETER  $\beta$  IN MULTI-FREQUENCY CONVOLUTION

When decomposing the output feature maps of the convolutional layers, the calculation cost of the improved SSD model and the related memory consumption are closely related to the parameter  $\beta$ . With the change of parameter  $\beta$ , the optimal setting of parameter  $\beta$  is explored based on the PASCAL 2012 data set, and the parameter  $\rho$  of the IoU prediction loss branch is set to 0.3. The calculation cost and memory consumption proportion change of the improved model are shown in table 3

It can be seen from table 3 that the increase of parameter  $\beta$  makes the relevant low-frequency feature components of the improved model increase continuously,



**FIGURE 5. Relationship between hyper-parameter  $\rho$  and SSD 300 AP values.**

**TABLE 3. The influence of hyper-parameter  $\beta$  on calculation cost and memory loss.**

$\beta$	0	0.125	0.25	0.5	0.75	0.875	1
calculation cost	100%	84%	69%	48%	32%	26%	23%
memory loss	100%	93%	82%	65%	46%	37%	28%

**TABLE 4. The influence of hyper-parameter  $\beta$  on the performance of the improved mode.**

$\beta$	AP <sub>0.5</sub> %	Time(ms)	FLOPs( $\times 10^9$ )
0	79.4	126	4.3
0.125	79.8	117	3.7
0.25	79.2	102	3.1
0.5	76.3	79	2.6
0.75	74.5	65	2.1

resulting in more low-frequency feature components being compressed, and the calculation cost and memory loss are significantly reduced. With the continuous compression of the low-frequency feature space, the relevant accuracy changes of the improved SSD model are shown in Table 4. When  $\beta$  is 0.125, the detection accuracy of the model is improved by 0.4%, and the computational cost is significantly reduced. Experimental data show that the compression of related low-frequency features will not cause the loss of important features in the image. Continuing to improve the proportion of low-frequency feature components. Before  $\beta$  reaches 0.75, the detection accuracy of the improved model is still improved compared with that of the original SSD object detection model (Table 2: the test result of ap0.5 of the original SSD model is 75.8%). When the proportion of low-frequency feature component is 75%, the accuracy rate

drops by only 1.3%, but other related performance is greatly improved. The improved SSD model effectively reduces the relevant spatial redundancy information, improves the model efficiency, and shows the effectiveness of the improved model. Figure 6 shows the effect of the hyper-parameter  $\beta$  on various indicators of the improved model.

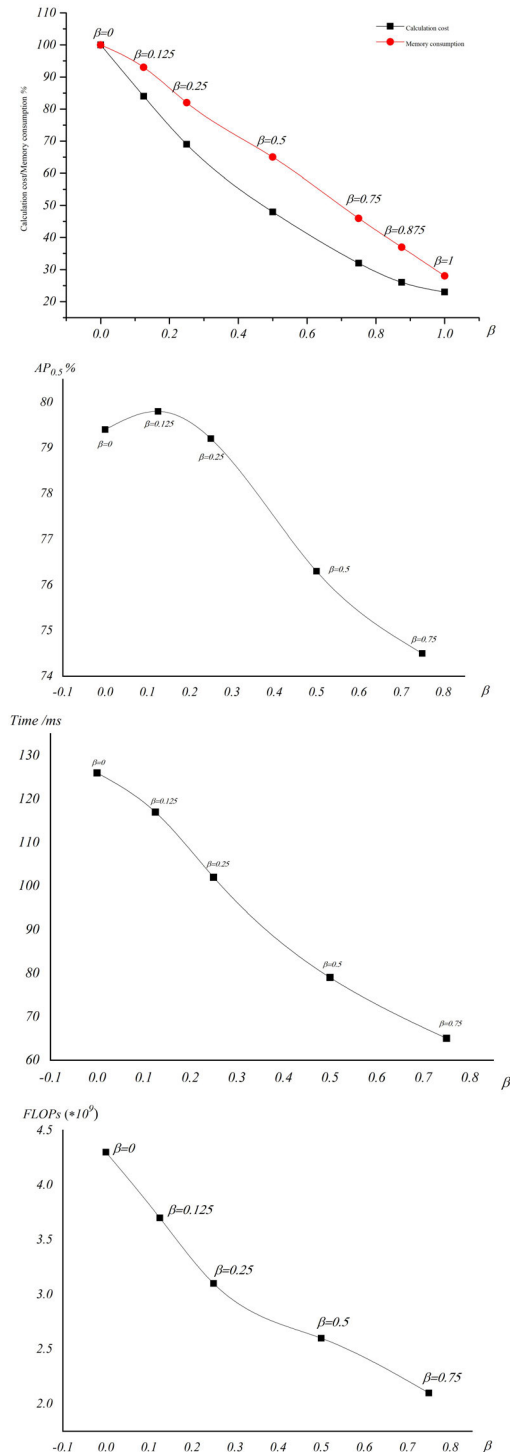


FIGURE 6. The influence of hyper-parameter  $\beta$  on various indicators of the improved model.

$Y_{H \rightarrow L}$ ,  $Y_{L \rightarrow H}$  indicate two information communication paths between high and low frequency feature components, which have an important impact on the accuracy of the improved SSD model. The deletion of any information communication path will reduce the performance of the improved model. When  $\beta$  is 0.25, the relevant experimental results are shown in Table 5.

TABLE 5. The influence of information communication path on the performance of the improved mode.

$\beta$	$Y_{H \rightarrow L}$	$Y_{L \rightarrow H}$	AP <sub>0.5</sub> %
0.25	delete	delete	76.0
	retain	delete	78.1
	delete	retain	77.9
	retain	retain	79.2

### C. COMPARISON OF RELATED MODELS

Based on MS COCO and PASCAL VOC2012 data sets, the improved SSD model and related comparison models are trained and tested. In the training stage of the improved model, in order to accelerate the convergence of the model, the traditional SGD algorithm is no longer used, and the relevant training process of the model is optimized using the AdaMod optimizer. In addition, based on the exploration results of relevant experiments, the correlation parameter  $\rho$  of the IoU prediction loss branch is set to 0.3, and the related hyper-parameter  $\beta$  in the multi-frequency convolution operation is set to 0.25. Set the relevant parameters of the AdaMod optimizer, where the step size  $\varepsilon$  is 0.001, the moment estimation exponential decay rates  $\rho_1$ ,  $\rho_2$  are 0.9 and 0.999, and the smaller constant value  $\delta$  used for numerical stability is set to  $10^{-8}$ , the measurement parameter  $\rho_3$  of the memory length is set to 0.99. Perform sufficient iterative training on all models involved in the experiments, and use the test sets of the relevant data sets to test the trained model. The test results and experimental analysis are as follows:

#### 1) COMPARISON OF EXPERIMENTAL RESULTS BASED ON MS COCO DATA SET

Based on the MS COCO data set, the improved model and related comparison models are trained and tested. The comparison results are shown in Table 6.

Table 6 shows the test results of the improved SSD detection algorithm and related comparison algorithms on the MS COCO data set. It can be seen from the data in the table that the improved model has better real-time detection and higher detection accuracy than Faster RCNN, YOLO v2, SSD, FSSD and DSSD detector models. On the related test set of MS COCO, the average accuracy of AP<sub>0.5</sub> and AP<sub>0.75</sub> on the improved SSD 300 algorithm can reach 39.55%. Compared with the FSSD model, the average accuracy of our algorithm is improved by 1.8%. Compared with the traditional SSD detection model, the accuracy of the improved model is increased by 5.1%, and the original SSD model is significantly improved. In terms of real-time detection, the FPS value of the improved SSD 300 model can reach 61, which is



TABLE 6. Comparison of related algorithms based on MS COCO data set.

algorithm	data set	basic network	FPS	$(AP_{0.5}+AP_{0.75})/2$	$AP_{0.5}$	$AP_{0.75}$
Faster RCNN <sup>[6]</sup>	MS COCO	VGG16	7	30.45	42.7	18.3
Faster RCNN <sup>[6]</sup>	MS COCO	ResNet101	2.4	32.65	45.2	20.1
YOLOv2 <sup>[32]</sup>	MS COCO	Darknet19	76	31.60	44.0	19.2
SSD300 <sup>[5]</sup>	MS COCO	VGG16	49	34.45	43.1	25.8
SSD512 <sup>[5]</sup>	MS COCO	VGG16	22	39.40	48.5	30.3
FSSD300 <sup>[14]</sup>	MS COCO	VGG16	45	37.75	47.7	27.8
FSSD513 <sup>[14]</sup>	MS COCO	VGG16	19	43.15	52.8	33.5
DSSD321 <sup>[12]</sup>	MS COCO	ResNet101	12	37.65	46.1	29.2
DSSD513 <sup>[12]</sup>	MS COCO	ResNet101	8	44.25	53.3	35.2
Our SSD300	MS COCO	VGG16	61	39.55	49.8	29.3
Our SSD512	MS COCO	VGG16	39	44.90	54.1	35.7

TABLE 7. Comparison of related algorithms based on PASCAL VOC 2012 data set.

algorithm	data set	basic network	FPS	$(AP_{0.5}+AP_{0.75})/2$	$AP_{0.5}$	$AP_{0.75}$
Faster RCNN <sup>[6]</sup>	VOC 2012	VGG16	7	65.05	73.4	56.7
Faster RCNN <sup>[6]</sup>	VOC 2012	ResNet101	2.4	67.80	76.4	59.2
YOLOv2 <sup>[32]</sup>	VOC 2012	Darknet19	74	67.95	78.6	57.3
SSD300 <sup>[5]</sup>	VOC 2012	VGG16	48	67.00	75.8	58.2
SSD512 <sup>[5]</sup>	VOC 2012	VGG16	22	70.65	79.5	61.8
FSSD300 <sup>[14]</sup>	VOC 2012	VGG16	46	68.55	78.8	58.3
FSSD513 <sup>[14]</sup>	VOC 2012	VGG16	18	71.70	80.9	62.5
DSSD321 <sup>[12]</sup>	VOC 2012	ResNet101	12	68.35	78.6	58.1
DSSD513 <sup>[12]</sup>	VOC 2012	ResNet101	8	71.80	81.5	62.1
Our SSD300	VOC 2012	VGG16	60	70.45	79.2	61.7
Our SSD512	VOC 2012	VGG16	38	73.05	82.7	63.4

enough to meet the needs of real-time detection. It is believed that the improvement of model performance can be described from the following two aspects. On the one hand, the introduction of the IoU prediction branch can more accurately locate the objects in the input image, the positioning effect of the objects are improved. The missed detection of small and medium-sized objects has been improved. On the other hand, the AdaMod optimizer makes the model convergence faster. In addition, the improved algorithm performs convolution operation based on multi-frequency feature maps, compresses the low-frequency feature components of the relevant convolution layers output, reduces the spatial redundant information of the improved SSD algorithm and reduces the interference of irrelevant information. In the end, the accuracy and the real-time detection of the improved model have been well improved. The relevant experiments fully demonstrated the advantages of the improved model and the effectiveness of the algorithm improvement. The  $AP_{0.5}$  iterative training changes of the relevant models are shown in Figure 7.

2) COMPARISON OF EXPERIMENTAL RESULTS BASED ON PASCAL VOC 2012 DATA SET

In order to verify the generality of the improved detection algorithm, the improved model and related comparison models were trained and tested again based on the PASCAL VOC 2012 data set. The experimental results are shown in Table 7.

According to the experimental data in table 7, the  $AP_{0.5}$  of the improved SSD 300 algorithm on the PASCAL VOC 2012 data set can reach 79.2%, which is 3.4% higher than the detection accuracy of the original SSD 300 algorithm. Compared with DSSD, FSSD and other algorithms,

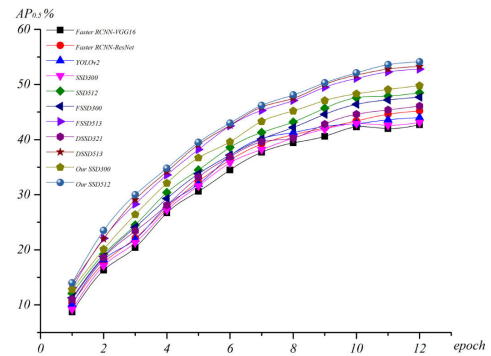


FIGURE 7.  $AP_{0.5}$  iterative training changes of related models (based on MS COCO data set).

the detection accuracy of the improved algorithm still has obvious advantages on the Pascal VOC 2012 data set. Analysis of the real-time detection of the improved model, compared to Faster RCNN, SSD, FSSD and DSSD, the detection speed of our improved model is much faster. In this paper, it is considered that convolution operation based on multi-frequency feature maps and compression of correlation low-frequency feature components play a key role in improving the speed of model detection. The  $AP_{0.5}$  iterative training changes of the related models are shown in Figure 8. Combining the experimental results of MS COCO and PASCAL VOC data sets, it can be considered that the improved model has good generality on different data sets.

D. DETECTION EFFECT ANALYSIS

The advantages of the improved SSD algorithm are mainly reflected in three aspects. Firstly, our algorithm improves

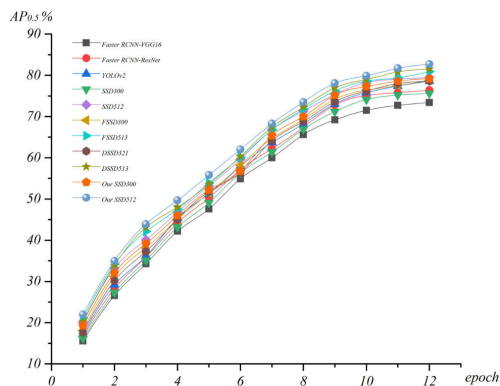


FIGURE 8. AP<sub>0.5</sub> iterative training changes of related models (based on PASCAL VOC 2012 data set).

the situation of repeatedly detecting multiple parts of the same object and taking multiple objects as the same detection object. For example, in Figure 9 (a<sub>1</sub>), the traditional SSD detection model repeatedly detects the same object, which is obviously incorrect. From the comparison figure(a<sub>2</sub>), the improved algorithm can significantly improve this phenomenon. In addition, in Figure 9 (c<sub>1</sub>), the traditional SSD object detection algorithm detects multiple objects as the same object, and the real situation should be two objects. Secondly, compared with (d<sub>1</sub>) and (d<sub>2</sub>) in Figure 9, the improved algorithm has better detection effect on small and medium-sized objects than the traditional SSD object detection algorithm. By introducing the IoU prediction loss branch, the improved model can more accurately locate the objects in the input image. Compared with the traditional SSD object detection algorithm, our algorithm can

successfully detect more small and medium-sized objects. Finally, the traditional detection algorithm relies on the regression of the bounding box to complete the positioning of the objects, without considering the fuzzy situation of the real bounding box. Generally speaking, the bounding box regression with higher classification score should be more accurate, but the real situation is not the case. As shown in figure (d<sub>1</sub>), the first person on the left outputs two prediction boxes, and the positioning effect of the bounding box with higher score (0.97) is not as good as that of the bounding box with lower score (0.91). To this end, the improved algorithm effectively improves this situation by exploring the optimal value of the hyper-parameter  $\rho$  in the IoU detection branch.

### V. CONCLUSION

The improved SSD multi-object detection model has been improved in terms of detection rate and efficiency, and reduced the calculation cost and related hardware cost of the model. Its contributions are mainly reflected in the following aspects:

- (1) Aiming at the defect that the correlation between the predicted object category score and the object positioning accuracy of the traditional SSD model is weak, the improved model enhances the correlation between the object score and the positioning accuracy by adding the IoU prediction loss branch, so as to improve the detection accuracy of the model.
- (2) In order to improve the real-time performance of the algorithm and reduce the spatial redundancy of the model, the convolution correlation module of multi-frequency feature components is designed for the traditional SSD object detection model, which reduces

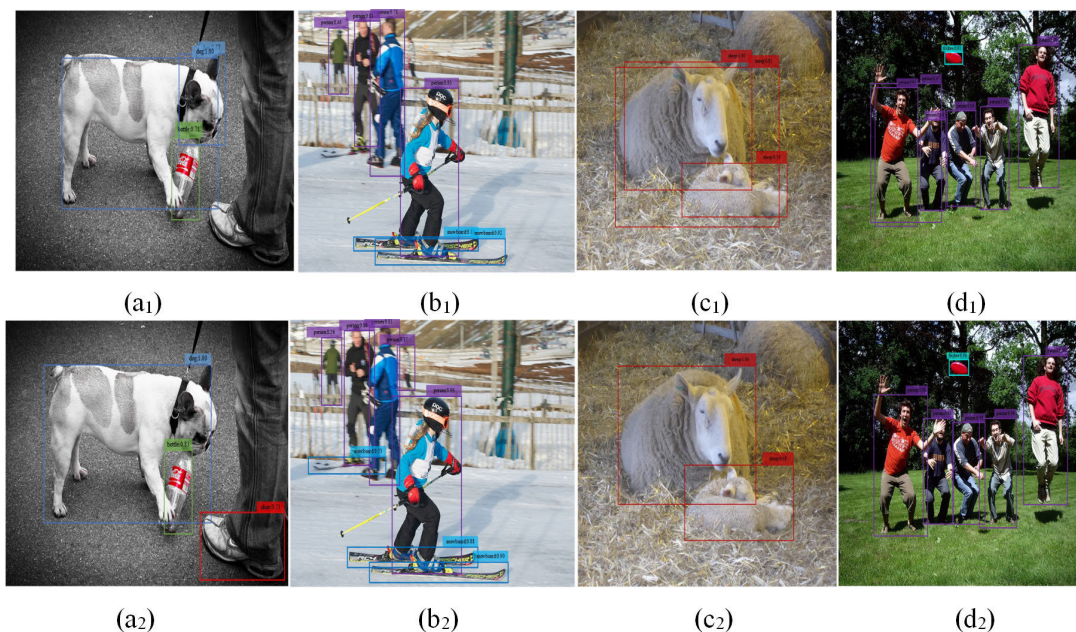


FIGURE 9. The detection results of the original SSD model ( $\alpha_1 - d_1$ ) and the improved model ( $\alpha_2 - d_2$ ).

the calculation cost and related hardware cost of the traditional model.

- (3) In order to improve the real-time performance of SSD model and accelerate the convergence speed of the model. By introducing the AdaMod optimizer, the adaptive learning rate of the abnormal value of the improved model was modified, and the generalization and convergence of the traditional SSD model are improved.
- (4) Through a large number of experiments, the optimal settings of the hyper-parameter  $\rho$  in the IoU detection branch and the hyper-parameter  $\beta$  in the multi-frequency convolution operation are explored. Based on MS COCO and PASCAL VOC2012 authoritative data sets, it is verified that the improved model has good performance in different data sets.

## REFERENCES

- [1] J. Ju and J. Xing, "Moving object detection based on smoothing three frame difference method fused with RPCA," *Multimedia Tools Appl.*, vol. 78, no. 21, pp. 29937–29951, Nov. 2019.
- [2] S. Hussein, P. Kandel, C. W. Bolan, M. B. Wallace, and U. Bagci, "Lung and pancreatic tumor characterization in the deep learning era: Novel supervised and unsupervised learning approaches," 2018, *arXiv:1801.03230*. [Online]. Available: <http://arxiv.org/abs/1801.03230>
- [3] S. Kim, W.-J. Song, and S.-H. Kim, "Robust ground target detection by SAR and IR sensor fusion using AdaBoost-based feature selection," *Sensors*, vol. 16, no. 7, pp. 1117–1134, 2016.
- [4] H. Xu, Z. Yang, G. Chen, G. Liao, and M. Tian, "A ground moving target detection approach based on shadow feature with multichannel high-resolution synthetic aperture radar," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 10, pp. 1572–1576, Oct. 2016.
- [5] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 21–37.
- [6] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
- [7] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.
- [8] L. Hui-Lan, T. Kang, and K. Fan-Sheng, "The progress of human action recognition in videos based on deep learning: A review," *Acta Electron. Sinica*, vol. 47, no. 5, pp. 1162–1173, 2019.
- [9] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, Sep. 2013.
- [10] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
- [11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.
- [12] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," 2017, *arXiv:1701.06659*. [Online]. Available: <https://arxiv.org/abs/1701.06659>
- [13] J. Jeong, H. Park, and N. Kwak, "Enhancement of SSD by concatenating feature maps for object detection," 2017, *arXiv:1705.09587*. [Online]. Available: <http://arxiv.org/abs/1705.09587>
- [14] Z. Li and F. Zhou, "FSSD: Feature fusion single shot multibox detector," 2017, *arXiv:1712.00960*. [Online]. Available: <http://arxiv.org/abs/1712.00960>
- [15] J. Ding, X. Ren, R. Luo, and X. Sun, "An adaptive and momental bound method for stochastic learning," 2019, *arXiv:1910.12249*. [Online]. Available: <http://arxiv.org/abs/1910.12249>
- [16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [17] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2961–2969.
- [18] S. Han, J. Pool, S. Narang, H. Mao, E. Gong, S. Tang, E. Elsen, P. Vajda, M. Paluri, J. Tran, B. Catanzaro, and W. J. Dally, "DSD: Dense-sparse-dense training for deep neural networks," 2016, *arXiv:1607.04381*. [Online]. Available: <http://arxiv.org/abs/1607.04381>
- [19] J.-H. Luo, H. Zhang, H.-Y. Zhou, C.-W. Xie, J. Wu, and W. Lin, "ThiNet: Pruning CNN filters for a thinner net," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 10, pp. 2525–2538, Oct. 2019.
- [20] F. Tung and G. Mori, "CLIP-Q: Deep network compression learning by in-parallel pruning-quantization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7873–7882.
- [21] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1492–1500.
- [22] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: <http://arxiv.org/abs/1704.04861>
- [23] Y. Chen, Y. Kalantidis, J. Li, S. Yan, and J. Feng, "Multi-fiber networks for video recognition," in *Proc. 15th Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 352–367.
- [24] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.
- [25] Y. Chen, H. Fan, B. Xu, Z. Yan, Y. Kalantidis, M. Rohrbach, S. Yan, and J. Feng, "Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution," 2019, *arXiv:1904.05049*. [Online]. Available: <http://arxiv.org/abs/1904.05049>
- [26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [27] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, vol. 22, no. 3, pp. 400–407, 1951.
- [28] L. Luo, Y. Xiong, Y. Liu, and X. Sun, "Adaptive gradient methods with dynamic bound of learning rate," 2019, *arXiv:1902.09843*. [Online]. Available: <http://arxiv.org/abs/1902.09843>
- [29] H. Q. Xing, Z. Q. Du, and B. Su, "Pedestrian detection method based on modified SSD," *Comput. Eng.*, vol. 44, no. 11, pp. 228–233, 2018.
- [30] L. Bao-Qi, H. Yu-Yao, Q. Wei, and H. Ling-Jiao, "SSD with parallel additional feature extraction network for ground small target detection," *Acta Electron. Sinica*, vol. 48, no. 1, pp. 84–91, 2020.
- [31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: <http://arxiv.org/abs/1409.1556>
- [32] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7263–7271.



**JINLING LI** was born in Shanxi, China. She received the Bachelor of Literature degree from Shanxi University, the master's degree from Beijing Foreign Language University, and the Ph.D. degree from Xi'an Jiaotong University. Her main research interests include smart city, computer vision, and computing intelligence.



**QINGSHAN HOU** was born in Yangquan, Shanxi. He received the Bachelor of Engineering degree with the Department of Computer Science and Technology, Shanxi Datong University, from September 2014 to July 2018. He is currently pursuing the master's degree with Shanxi Normal University. His research interests include computer vision and data mining.



**JIANGUO JU** is currently pursuing the Ph.D. degree with the School of Information Science and Technology, Northwest University, Xi'an. His research interests include deep learning, data mining, and computer vision.

...



**JINSHENG XING** was born in Taiyuan, China. He received the Bachelor of Science degree from the Mathematics Department, Shanxi Normal University, in July 1985, the master's degree in computer science from the Mathematics Department, Beijing Normal University, in July 1988, and the Ph.D. degree from the System Institute, Xi'an Jiaotong University, in November 2000.

From December 2000 to February 2001, he was a Visiting Scholar with the University of Reading, U.K. From March 2001 to July 2004, he was with Xi'an Jiaotong University. His main research interests include intelligent control, data mining, artificial neural networks, and uncertain pulse hybrid systems and applications.