PAPER Special Section on Deep Learning Technologies: Architecture, Optimization, Techniques, and Applications

A Novel SSD-Based Detection Algorithm Suitable for Small Object

Xi ZHANG[†], **Yanan ZHANG**[††], **Tao GAO**[†], **Yong FANG**[†], ***and*** **Ting CHEN**[†a)], ***Nonmembers***

SUMMARY The original single-shot multibox detector (SSD) algorithm has good detection accuracy and speed for regular object recognition. However, the SSD is not suitable for detecting small objects for two reasons: 1) the relationships among different feature layers with various scales are not considered, 2) the predicted results are solely determined by several independent feature layers. To enhance its detection capability for small objects, this study proposes an improved SSD-based algorithm called proportional channels' fusion SSD (PCF-SSD). Three enhancements are provided by this novel PCF-SSD algorithm. First, a fusion feature pyramid model is proposed by concatenating channels of certain key feature layers in a given proportion for object detection. Second, the default box sizes are adjusted properly for small object detection. Third, an improved loss function is suggested to train the above-proposed fusion model, which can further improve object detection performance. A series of experiments are conducted on the public database Pascal VOC to validate the PCF-SSD. On comparing with the original SSD algorithm, our algorithm improves the mean average precision and detection accuracy for small objects by 3.3% and 3.9%, respectively, with a detection speed of 40FPS. Furthermore, the proposed PCF-SSD can achieve a better balance of detection accuracy and efficiency than the original SSD algorithm, as demonstrated by a series of experimental results.

key words: *object detection, deep learning, neural networks, SSD, feature pyramid*

1. Introduction

Object detection is widely explored due to its numerous applications in computer vision fields [1]–[5], such as intelligent transport systems, computer-aided diagnosis, image retrieval, and military reconnaissance.

The traditional object detection process is generally divided into three steps: 1) selection of few candidate regions in the given images, 2) extraction of the corresponding features from these regions, and 3) classification using the trained classifiers. Traditional algorithms have two major issues: 1) when designing a region selection strategy based on the sliding window for different object images, the united optimal standard is not available. This results in redundant window scans and high temporal complexities and 2) features extracted using artificially designed algorithms are not very robust considering the diversity of object profile, varying background and illumination, and so on. Thus, the traditional algorithms are incapable of meeting the demands of reliable object detection in real time. Deep learning based on neural networks is a potential candidate because it can quickly detect objects with high accuracy.

Deep learning is a promising methodology for object detection. Thus, a series of superior algorithms has been proposed; current popular algorithms can be classified into two categories: 1) two-stage detection based on candidate regions and 2) one-stage detection based on regression. Two-stage detection algorithms based on candidate regions include R-CNN [6], Fast R-CNN [7], and Faster R-CNN [8]. These algorithms usually extract the candidate region at the predicated object position on a certain feature map. In contrast, in one-stage detection, the candidate region extraction stage is eliminated. It inputs the entire image into the network directly and outputs the object's box boundary and corresponding classified result by regression analysis. Such algorithms include single-shot multibox detector (SSD) [9] and YOLO series [10]–[12] representatively.

Nevertheless, a series of R-CNN algorithms cannot make any image scaling on the original object. In fact, in certain deeper layers of the neural network, few pixels are left for the small object itself. Therefore, considerable edge information is easily lost on the extracted feature map, significantly decreasing the detection accuracy. Meanwhile, the extracted feature map is divided into $n \times n$ blocks by the YOLO series. When a single block contains a large number of small objects, it may result in detection failures with a high probability. Different from the YOLO series, the original SSD algorithm groups multiple candidate regions into a single block, from which the bounding boxes are extracted using a multiscale feature pyramid model. It can make a better balance between efficiency and accuracy. However, only the underneath layer conv4_3 of SSD is used to detect the small object. Thus, it always has insufficient information and does not fully consider the relationships among different feature layers with different scales, leading to poor detection performance. Therefore, it is necessary to investigate methods to enhance the original SSD for small object detection. In this respect, several research and explorations have been conducted in recent years.

Based on the original SSD, several optimized algorithms have been proposed to overcome the drawbacks of small object detection. Although the original SSD algorithm uses a single feature layer of different scales to participate in object detection, consider the inherent relationships between feature maps of different scales. Therefore, several algorithms have been developed since 2017 to improve the

Manuscript received October 28, 2021.
Manuscript revised December 10, 2021.
Manuscript publicized January 6, 2022.

[†]The authors are with School of Information Engineering, Chang'an University, Xi'an, Shaanxi 710064, China.
[††]The author is with China Mobile Group Shanxi Company Limited, Taiyuan, Shanxi, 030032, China.
a) E-mail: tchenchd@126.com (Corresponding author)

DOI: 10.1587/transinf.2022DLP0037

detection accuracy [13]–[21]. Small objects are difficult to detect because they occupy tiny pixels in a picture. To solve this problem, several researchers proposed improved algorithms [22]–[26]. When considering the fusion of different feature layers, the interdependence among each feature map channel is frequently overlooked, resulting in difficult detection of small objects with poor detection performance. To solve this problem, a nonlocal channel attention block [27] is introduced into small objects detection network, which can enhance the contextual semantic information of small objects in the shallow features. Further, based on the realization principle of one-stage detectors, a small object detection algorithm guided by a dual attention model [28] is proposed. It introduces two attention models to improve the detection performance, particularly for small objects. Moreover, in a faster region convolutional neural network (Faster-RCNN), an attention mechanism [29] was added to prevent other useless information from adapting to the background of the large range of remote sensing image vision, resolving the complex problem of small objects.

In summary, despite the advancements, there are several challenges that need to be solved: 1) the dependent relationships among different feature layers in the feature pyramid model are neglected during the process of feature fusion; 2) feature fusion with the introduction of deconvolution may increase the algorithm complexity; and 3) the object detection does not meet the real-time requirement. To address the aforementioned challenges, we propose an improved SSD-based algorithm called proportional channels' fusion SSD (PCF-SSD), which includes the following enhancements: (1) to better use the relationships among feature maps with different scales, bilinear interpolation is introduced to create feature fusion of upper and lower layer maps. This can enrich feature maps' detailed and semantic information for accurate prediction; (2) during the process of feature fusion, the number of feature map channels is adjusted by convolution. The weight ratio of the fused upper and lower layers is set to 2 : 1, which can further enrich the detailed information of a small object; (3) the number of prior anchor boxes and the size ratio of the prior anchor box to the original image are adjusted, making the network module more suitable for small objection detection; and (4) an improved loss function is suggested for training the model to obtain a better training model. For further improvement, the detection performance of the model while ensuring the real-time detection.

2. Related Work

2.1 Original SSD Model

Figure 1 presents the characteristic comparisons between feature maps with a single layer and pyramid feature map with multi-scale layers. As shown in Fig. 1 (a), only features from the deepest layer are utilized for prediction. These features are the network structures adopted by the YOLO algorithm as well as can always acquire higher detection

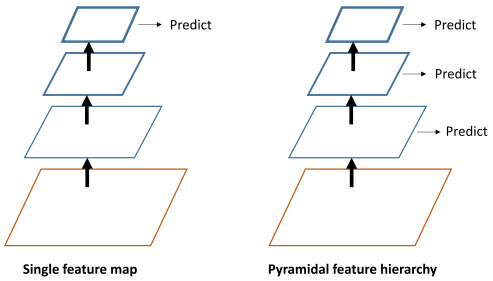


Fig. 1 (a) Single feature map prediction model. (b) Feature pyramid prediction model

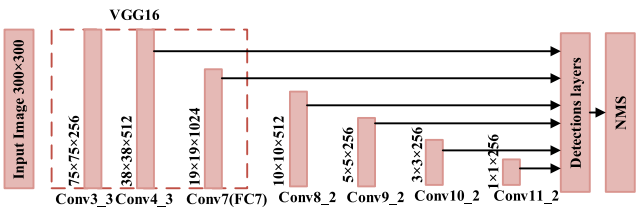


Fig. 2 The structure of SSD algorithm

speed, but relatively lower accuracy. Different from feature maps with a single layer, the pyramid feature map in Fig. 1 (b) adopted by SSD, which can utilize features from several multi-scale layers to perform softmax classification and location regression. This function is aimed to improve the detection accuracy for small objects.

SSD is based on a feed-forward convolutional network that produces a fixed-size collection of bounding boxes and scores for the presence of object class instances in those boxes. The structure of SSD is shown in Fig. 2. SSD algorithm followed by a non-maximum suppression step to produce the final detections. It utilizes VGG-16 as the backbone network after adjusting its original fully connected layer FC6 to Conv_6 by the convolution operation with the kernel size of 3 and filter depth of 1024, as well as FC7 to Conv_7 by the convolution operation with the kernel size of 1 and filter depth of 1024. Then, SSD adds the auxiliary structure to obtain multi-scale feature maps, including the cascade feature convolutional layers of Conv8_2, Conv9_2, Conv10_2, and Conv11_2, to further strengthen SSD detection accuracy. The default boxes in SSD is different. At meanwhile, SSD has different aspect ratios on each feature map cell, for multiple feature maps including Conv4_3, Conv7, Conv8_2, Conv9_2, Conv10_2, Conv11_2. The default boxes tile the feature map in a convolutional manner, so that, the position of each box relative to its corresponding cell is fixed. At each feature map cell, both of the offsets relative to the default box shapes in the cell, and the per-class scores that indicate the presence of a class instance in each of those boxes can be predicted. The idea of default boxes in SSD is very similar to the Anchor of Faster R-CNN, which can reduce the network's training complexity to some extent and make it easier to converge. Therefore, the idea of Anchor is significant to improve the accuracy of the One-stage detection algorithm. If no Anchor, SSD

directly obtains the position coordinates and width and height of the object through regression. Nevertheless, the wide and high difference of various objects on the dataset is large. If regression is directly utilized, this will make the model difficult to converge and even fall into a poor local optimal state.

2.2 SSD Algorithm Performance Drawbacks

Although SSD has achieved good results on two crucial indictors, mean average precision (MAP) and frame per second (FPS). Its recall rate for small objects is still low and lags behind algorithms with two-stage detection. In addition, occasional detection failures cannot be avoided particularly for tiny objects. If the input image contains some small objects, after convolution and pooling, these small objects only occupy very few pixels on feature map Conv4_3 with little information. These make it impossible to locate them accurately, even know whether they are background pixels or corresponding detection object pixels. Since small objects on the large-scale feature map are relatively easy to detect, SSD utilizes Conv4_3 that is far from the top layer of the feature pyramid for prediction. However, the detection ability of Conv4_3 for small objects is still insufficient. On the other hand, SSD uses six feature maps with different sizes to predict the objects independently, but information from different feature maps are not fused effectively. In fact, the high-resolution feature maps from the lower layer always contain certain key details about the small object that can help to accomplish accurate object locating. But it is not very easy to discriminate the object from the background, because the underlying feature map experiences fewer convolution operations, and fails to extract adequate advanced features without sufficient semantic information. On the contrary, the low-resolution feature maps from the upper layer undergoes many convolution operations and can extract rich semantic information, but due to over-sampling, much detail information is lost. After a series of pooling operations from lower layers to upper layers, key feature information especially small objects maybe disappeared, it leading to mismatching between the real object and its default bounding box. These are suffering the detection accuracy deterioration. Moreover, during the samples' training process, small object samples cannot be trained adequately for the reason as the following: SSD utilizes the indicator called Intersection over Union (IoU) to decide whether the samples are positive or negative. Since the small object always occupies too few pixels, if the value of IoU between it and the default bunding box is less than the threshold value of 0.5, the small object cannot be easily matched with the correct default bounding box successfully, and thus be judged as a negative one.

3. The Proposed PCF-SSD Algorithm

3.1 Fusion Strategy for Feature Maps

The original SSD algorithm only utilizes feature maps with different scales independently to detect objects, doesn't fully consider the relationships among layers from the feature pyramid. In fact, feature maps with relatively large scales at lower layers always contain amounts of location and other detailed information, whereas feature maps with relatively small scales at upper layers include much advanced semantic information that is extracted by multiple convolution operations layer by layer. To further develop the respective advantages of information from upper layers as well as lower layers, a fusion algorithm called PCF-SSD is proposed to accomplish feature maps' effective fusion by different proportional channels. There are two effective fusion methods [20] for image classification and detection: concatenation and element-wise summation. By combining several certain channels, concatenation inevitably increases images' feature numbers. Different from concatenation, element-wise summation can increase feature information by adding the corresponding feature maps. Both of the feature fusion methods are beneficial to the final image classification and detection. However, element-wise summation requires that the fusion feature maps should have the same scale and channel number, which seriously limits its flexible applications. Inspired by the experiments of FSSD [20], if compared with the element-wise summation, concatenation is more helpful to recognize objects and enhance detection accuracy more effectively. Therefore, our proposed algorithm adopts concatenation to accomplish the fusion of feature maps with different scales.

Since the size of feature maps at the lower layer are larger than that of maps at the upper layer, it is necessary to resize them with the same scales by sampling, which is the premise for feature fusion. The proposed fusion strategy is given as follows. By max-polling operation, the original feature map with a size of 76×76 at layer Conv3_3 is down-sampled to two new maps with sizes of 38×38 and 19×19, and then respectively fused with maps at layer Conv4_3 and Conv7. By bilinear operation, the original feature map with a size of 10×10 at layer Conv8_2 is up-sampled to a new map with a size of 19×19. Afterward, the newly fused map at Conv4_3 is used for prediction directly, whereas the newly fused map at Conv7 still needs to be further fused with the newly resized map at layer Conv8_2. As shown in Fig. 3, a series of similarly resized and fused operations for feature maps are done, including fusion at Conv8_2 and Conv9_2, as well as fusion at Conv9_2 and Conv10_2. These newly fused feature maps with multiple scales are finally combined for object prediction. Furthermore, the channel relationship among the two feature maps that will be fused is fully considered during fusion, and convolution by 1×1 are utilized to guarantee that any two fused maps have a certain feature dimension, which also means two fused maps are

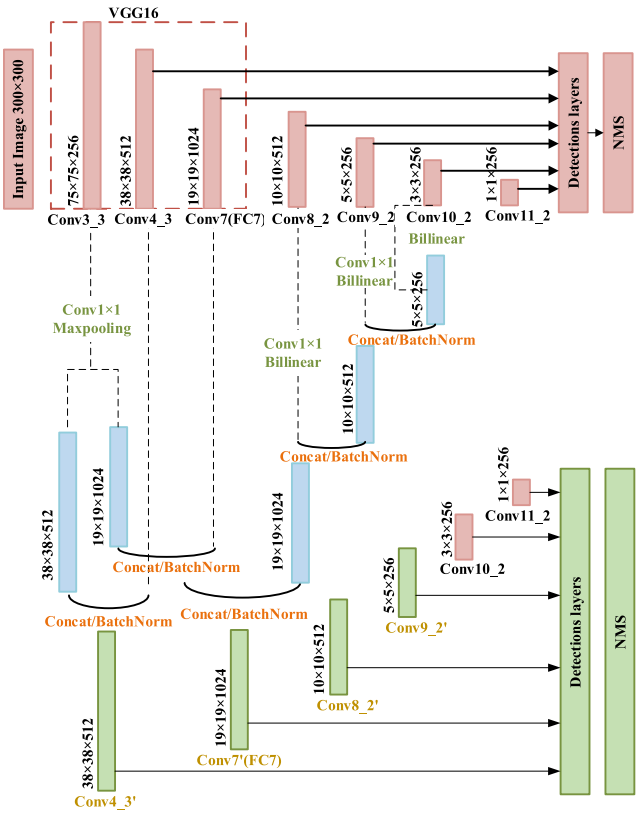


Fig. 3 The network structure of the PCF-SSD

merged according to the ratio of channel number 2 : 1. It is worth noting that a series of newly generated maps definitely contain more detailed information as well as sematic feature information, and as the input features, they are very helpful to enhance the object recognition capability in the object detector. The structure of our proposed SSD is illustrated in Fig. 3.

3.2 Scale Adjustment for Default Box

As shown in Fig. 3, there are 6 newly fused feature maps utilized for detection in our proposed PCF-SSD algorithm, including Conv4_3, Conv7, Conv8_2, Conv9_2, Conv10_2 and Conv11_2, respectively with sizes of 38×38, 19×19, 10×10, 5×5, 3×3, and 1×1, then the numbers of their corresponding default boxes are set as 9, 6, 6, 6, 4, and 4. The size change of the default box is linear, and its scale size can be calculated in Eq. (1):

$$S_k = \min_w S_w + \frac{\max_w S_w - \min_w S_w}{m-1}(k-1),\ \forall k \in \{1, \ldots, m\} \tag{1}$$

Where m is the number of feature maps, S_k represents the scale ratio between the default box and original image. $\max_w S_w$ and $\min_w S_w$ respectively represent the maximum size and the minimum size. During the training stage, the real label may not be matched with its corresponding default box easily. In order to solve that problem caused by the object's tiny size, the proposed values of $\min_w S_w$ and $\max_w S_w$ are adjusted from the original 0.2 and 0.9 to 0.1 and 0.9. This way effectively avoids the problem that the real label cannot find the corresponding prior box to match with it because the object is too small.

Table 1 shows the default frame size of each scale feature layer after the improvement what mentioned above.

Table 1 Default frame size of each scale feature layer after improvement

Feature layer	min_size	max_size
Conv4_3	15	30
Conv7	30	90
Conv8-2	90	150
Conv9-2	150	210
Conv10-2	210	270
Conv11-2	270	330

3.3 Loss Function Improvement for Object Detection

The loss function of PCF-SSD and SSD is basically similar. The defect location coordinates are output through the regression function, and the Softmax function predicts the classification confidence. The total loss function $L(x, c, l, g)$ is defined as the weighted summation of localization loss $L_{loc}(x, l, g)$ and classification loss $L_{cla}(x, c)$, shown as Eq. (2):

$$L(x, c, l, g) = \frac{1}{N}[L_{cla}(x, c) + \alpha L_{loc}(x, l, g)] \tag{2}$$

Where x indicates whether the prediction box is matched with the default box successfully, c is the confidence level, l is the location information of prediction box, g is the location information of actual box, N is the number of default boxes matched with actual boxes. $L_{cla}(x, c)$ utilizes SoftmaxLoss1 [9], and $L_{loc}(x, l, g)$ utilizes SoftmaxLoss1 [9] α is the weighed coefficient for adjusting proportion relationship between classification loss and location loss, here set to 1. $\beta_{L1}\left(l_i^m - \hat{g}_j^m\right)$ represents the smooth L_1 norm, l represents the prediction box, g represents the ground truth box, and d represents the prior box.

$\beta_{L1}(l_i^m - \hat{g}_j^m)$ is given in Eq. (3):

$$\beta_{L1}(x) = \begin{cases} 0.5x^2 & if\ |x| < 1 \\ |x| - 0.5 & otherwise \end{cases} \tag{3}$$

Localization Loss $L_{loc}(x, l, g)$ can be given in Eq. (4):

$$L_{loc}(x, l, g) = \sum_{i \in N_{pos}, m \in \{c_x, c_y, w, h\}}^{N} \sum x_{ij}^k \beta_{L1}\left(l_i^m - \hat{g}_j^m\right) \tag{4}$$

Where $x_{ij}^k \in \{0, 1\}$, if the i-th default box can be matched with the j-th actual box successfully for the k-th category, $x_{ij}^k = 1$, else $x_{ij}^k = 0$. N_{pos} is the set of positive samples. c_x, c_y respectively represent the center coordinates of the bounding box, whereas w and h are respectively the width and height of the bounding box. l_i^m is the predictive value of default box. $\hat{g}_j^m$ is the location parameter of the actual box,

defined as Eq. (5):

$$\begin{cases} \hat{g}_j^{c_x} = \dfrac{g_j^{c_x} - d_i^{c_x}}{d_i^w} \\ \hat{g}_j^{c_y} = \dfrac{g_j^{c_y} - d_i^{c_y}}{d_i^h} \\ \hat{g}_j^w = \log \dfrac{g_j^w}{d_i^w} \\ \hat{g}_j^h = \log \dfrac{g_j^h}{d_i^h} \end{cases} \tag{5}$$

Classification Loss $L_{cla}(x, c)$ can be given in Eq. (6):

$$L_{cla}(x, c) = -\sum_{i \in \mathrm{POS}}^{N} x_{i,j}^p \log\left(\hat{c}_i^p\right) - \sum_{j \in \mathrm{NEG}}^{N} \log\left(\hat{c}_j^0\right) \tag{6}$$

Where $i \in$ POS and $j \in$ NEG are the prediction box of the i-th positive sample and the prediction box of j-th negative sample, respectively. $\hat{c}_i^p$ is defined in the following Eq. (7):

$$\hat{c}_i^p = \frac{\exp\left(c_i^p\right)}{\sum_q \exp\left(c_i^q\right)} \tag{7}$$

If there is relatively higher proportion between the positive samples and negative samples, the model becomes convergence. What's more, adding network layer functions may bring overfitting problems. To avoid those mentioned above, inspired by a newly improved loss function proposed by Zhang Siyu in 2019 [32]. In this paper the regularization term L_2 is introduced into our Loss Function, and then the modified $L'(x, c, l, g)$ is presented in Eq. (8):

$$L'(x, c, l, g) = L_{cla}(x, c) + L_{loc}(x, c, g) + \varphi L_2 \tag{8}$$

Where φ is the L_2 normalizing factor, we set $\varphi = 0.1$ to guarantee the penalty value close to the original loss.

3.4 Algorithm Description

The flowchart of our proposed algorithm is illustrated in Fig. 4. Firstly, the original SSD framework is improved: six feature maps with different scales are selected to resize by up-sampling or down-sampling and are further fused to generate new feature maps by channel combinations. Next, the scales of default boxes for different fused feature maps and Loss functions are adjusted, which can help to train our improved SSD algorithm quickly reaching the maximum number of iterations, and then preserving the training parameters; finally, our well-trained PCF-SSD is utilized to extract features of the input image, predict the location and category of each object, and output the recognition objects with their corresponding category labels.

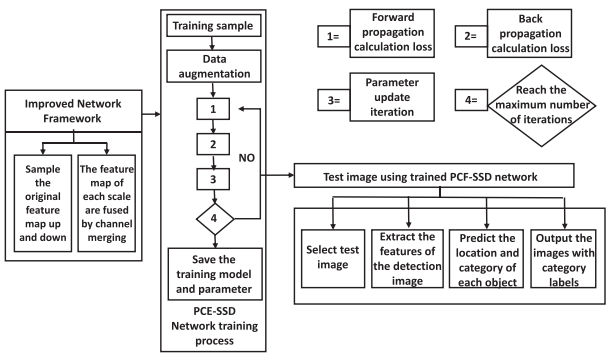


Fig. 4 PCF-SSD algorithm flowchart

4. Experiments

In order to verify the strength of our proposed PCF-SSD algorithm on small object detection, experiments are conducted on the PASCAL VOC dataset. A series of data augmentation strategies are adopted including horizontal flip, random crop, color distortion and random patch sampling.

4.1 Evaluation Indicators

Accuracy and detection efficiency are utilized to evaluate the performance of our proposed PCF-SSD algorithm. There are two key indicators for accuracy measurement, one is average precision (AP) and the other is MAP (mean Average Precision). AP is a comprehensive evaluation indicator for a certain category that can be calculated approximately according to the area enclosed by the precision and recall (PR) curve and the axis. PR curve is the precision curve about recall [33]. The bigger the area is, the higher the AP value is. MAP is the mean value of APs from all the different categories, and it can be used to evaluate the whole detection performance of the given model. The above-mentioned indicators can be calculated in the following Eqs. (9-12):

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{9}$$

$$\mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{10}$$

$$AP = \int_0^1 P(R)dR \tag{11}$$

$$MAP = \sum_{i=1}^{\mathrm{N}} \frac{AP_i}{N} \tag{12}$$

Both the precision (P) rate and recall (R) rate are two key classification evaluation indicators, calculated by several parameters listed in Table 2. True positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), are the four different possible outcomes of a single prediction for a two-class case with classes "1" ("yes") and "0"

Table 2 Classification evaluation

	Actual Class (observation)	
Predicted Class (expectation)	TP Correct result	FP Unexpected result
	FN Missing result	TN Correct absence of result

("no"). An FP is when the outcome is incorrectly classified as "yes" (or "positive"), when it is in fact "no" (or "negative"). An FN is when the outcome is incorrectly classified as negative when it is in fact positive. TPs and TNs are obviously correct classifications.

Frames per second (FPS) is a key indicator of detection efficiency, which is defined as the number of pictures that can be recognized in one second. The more FPSs are, the smoother the detection speed will be. When the frame rate is generally above 24, it can be considered to be basically smooth.

4.2 PCF-SSD Model Training

4.2.1 Experimental Environment

The experimental environment used in this paper is Windows 10 operating system, Lenovo 30BGA0N400. The CPU model is Intel Xeon E3-1225, the GPU model is NVIDIA Quadro P40000, whose memory size is 8G, and the RAM is 16GB. The proposed PCF-SSD model is trained in the framework of TensorFlow [35].

4.2.2 Dataset of PASCAL VOC

Our proposed PCF-SSD algorithm is compared with other object detection algorithms based on the public dataset of Pascal VOC. As one of the very commonly used public datasets, Pascal VOC [36] contains 20 object categories with abundant samples, especially with large number of small samples, and it has been an authoritative dataset of a comprehensive evaluation for small objects. Most object detection algorithms are tested and compared on Pascal VOC to verifying their detection performances for small objects. Since adopting random initial learning features may increase the training difficulty of the algorithm model, we utilize the pre-trained VGG16 model with default settings to overcome that problem. Moreover, we use stochastic gradient descent (SGD) [34] to further optimize the loss function and then seek the optimum solution. After conducting a series of parameter adjusting experiments, the optimum network super parameters are concluded, which are more suitable for our proposed algorithm model. The corresponding super parameters are listed in Table 3.

After the model is constructed and the parameters are determined, the PCF-SSD network model is trained. The process of network training is recorded in the following figure. Figure 5 records the whole process of network model training, as can be seen from the figure, the loss reduction of the verification set is very close to that of the training set. During the first 5000 steps of training, the loss value of the model decreased rapidly. From 5000 to 40000 steps, the model tends to be stable gradually; After 40000 steps, the model began to stabilize.

Table 3 The corresponding superparameters utilized in PCF-SSD model traing

Parameter iteration method	Momentum SGD
Momentum	0.9
Max_iter	120000
Learning rate decay	Polynomial_decay
Initial learning rate	1e-3
Weight_decay	5e-4
Global_step	1000, 80000, 100000
lr_decay_factors	0.1, 1, 0.1, 0.01
Batch size	32

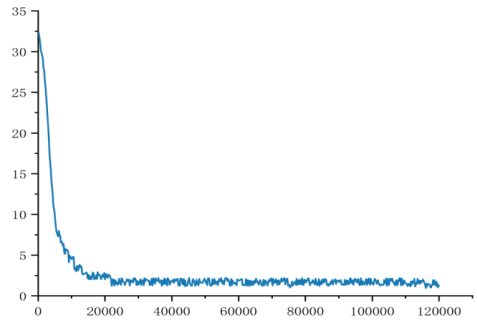


Fig. 5 Model training loss value

4.3 Experimental Results and Analysis

To verify the performance of PCF-SSD, we compare it with other existing excellent object detection algorithms by conducting a series of experiments. Following the regular practice of most researchers, all the tests are based on the datasets of Pascal VOC, where PASCAL VOC2007 trainval and PASCAL VOC2012 trainval are utilized as the training set, while the PASCAL VOC2007 test is utilized as the testing set. There are 20 object categories in PASCAL VOC dataset, where the categories of "Boat", "Bottle", "Plant", "Chair", "Table", "Bird", "Sheep", "Tv" have few object samples, and the size of the object is small, resulting in relatively poor detection. Therefore, our proposed PCF-SSD algorithm is evaluated by calculating the MAP value of those samples from the above-mentioned 20 object categories. During the following testing process, the value of intersection over union (IOU) is set as 0.5.

4.3.1 Visual Comparison and Analysis

Figure 6 shows a series of visual detection comparisons between the original SSD algorithm and our proposed PCF-SSD algorithm. In Fig. 6, there are 6 groups of detection results for different small objects, where the left image of each group is the detection result of the original SSD algorithm, and the right of each group is the result of our proposed PCF-SSD algorithm. It can be seen that PCF-SSD can detect more small objects if compared with the original SSD algorithm under the same conditions. Furthermore, PCF-SSD can identify the object category more accurately.

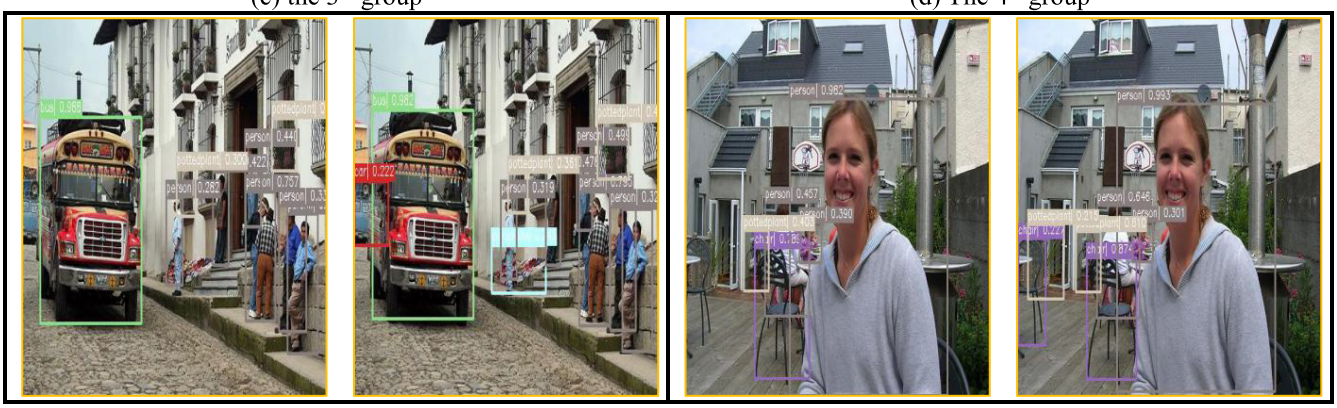
Table 4 shows the comparisons of the identified object numbers between SSD and PCF-SSD. It can be seen that

SSD PCF-SSD SSD PCF-SSD

(a) The 1st group (b) The 2nd group

SSD PCF-SSD SSD PCF-SSD

(c) the 3rd group (d) The 4th group



SSD PCF-SSD SSD PCF-SSD

(e) The 5th group (f) The 6th group

Fig. 6 Visual detection comparisons between SSD and PCF-SSD

Table 4 Comparisons of the detected object numbers between SSD and PCF-SSD

Group NO.	Actual object number		Detected object number		Detected object number for small objects		Detection performance for small objects	
	SSD	PCF-SSD	SSD	PCF-SSD	SSD	PCF-SSD	SSD	PCF-SSD
1	10	10	4	5	2	3	Partial detected	Partial detected
2	3	3	2	3	0	1	None detected	All detected
3	4	4	3	4	2	3	Partial detected	All detected
4	6	6	5	6	0	1	None detected	All detected
5	11	11	8	10	5	6	Partial detected	Partial detected
6	7	7	5	7	4	6	Partial detected	All detected

Table 5 Comparison results of different algorithms on VOC2007 dataset

Algorithm	Basic network	Input	Training dataset	Testing dataset	MAP (%)	Speed (fps)
Fast R-CNN [7]	VGG16	224	VOC2007+VOC2012	VOC2007	70.0	0.5
Faster R-CNN [8]	VGG16	448	VOC2007+VOC2012	VOC2007	73.2	7
Faster R-CNN [8]	ResNet-101	~600×1000	VOC2007+VOC2012	VOC2007	76.4	2.4
YOLOv2 [11]	Darknet-19	352	VOC2007+VOC2012	VOC2007	73.7	81
DSSD [13]	ResNet-101	321	VOC2007+VOC2012	VOC2007	78.6	9.5
FPEF-SSD [15]	VGG16	300	VOC2007+VOC2012	VOC2007	73.2	41
C-SSD [18]	VGG16	300	VOC2007+VOC2012	VOC2007	78.2	40
ION [30]	VGG16	~600×1000	VOC2007+VOC2012	VOC2007	75.6	1.25
DFSSD [31]	VGG16	300	VOC2007+VOC2012	VOC2007	78.0	39
SSD	VGG16	300	VOC2007+VOC2012	2007	74.8	46
PCF-SSD	**VGG16**	**300**	**VOC2007+VOC2012**	**2007**	**78.3**	**40**

Table 6 MAP (%) comparisons on VOC2007 dataset

Method	Fast R-CNN [7]	Faster R-CNN [8]	Faster R-CNN [8]	YOLOv2 [11]	DSSD [13]	FPEF-SSD [15]	C-SSD [18]	ION [30]	DFSSD [31]	SSD	**PCF-SSD**
Aero	77	76.5	79.8	86.3	81.9	75.8	78.5	79.2	80.5	76.5	**83.5**
Bike	78.1	79	80.7	82	84.9	82.7	86.7	79.2	85	80.5	**82.7**
Bird	69.3	70.9	76.2	74.8	80.5	71.7	75.8	77.4	76.6	73.8	**79.1**
Boat	59.4	65.5	68.3	59.2	68.4	63.4	70.6	69.8	66.6	66.4	**71.6**
Bottle	38.3	52.1	55.9	51.8	53.9	48.7	52.7	55.7	52.2	48.8	**52.8**
Bus	81.6	83.1	85.1	79.8	85.6	81.6	86.5	85.2	85.4	83	**85.5**
Car	78.6	84.7	85.3	76.5	86.2	83.3	87.3	84.2	86.3	84.2	**85.4**
Cat	86.7	86.4	89.8	90.6	88.9	80.5	87.5	89.8	86.2	86.1	**87.8**
Chair	42.8	52	56.7	51.1	61.1	51.3	63	57.5	61.2	54.7	**61.8**
Cow	78.8	81.9	87.8	78.2	83.5	66.5	82.6	78.5	83.5	79.4	**84.9**
Table	68.9	65.7	69.4	58.5	78.7	74.8	76.5	73.8	75.9	74.9	**77.6**
Sofa	74.8	73.9	79.8	62.4	80.9	73.9	80.6	74.6	81.4	76	**78.9**
Dog	84.7	84.8	88.3	89.3	86.7	85.9	85.9	87.8	84.8	84.5	**86.3**
Horse	82	84.6	88.9	82.5	88.7	86.7	88	85.9	87.3	85.3	**86.6**
Train	80.4	83	85.3	83.8	87.2	85.3	87.3	85.2	97.2	83.9	**87.9**
Plant	31.8	38.8	41.7	49.1	51.7	52.6	54.2	49.7	53.5	50.7	**54.9**
Tv	70.4	72.6	72	68.7	79.4	72.5	76.5	82.1	78.4	74	**76.2**
Person	69.9	76.7	78.4	81.3	79.7	78.6	80.3	75.3	79.1	75.9	**80.2**
m-bike	76.6	77.6	80.9	83.4	86.7	79.3	86.7	81.3	86.2	82.8	**83.1**
Sheep	70.1	73.6	78.6	77.2	78	67.9	77.3	76.9	79	74.9	**79.2**

Table 7 Comparison of the MAP of small object detection in VOC2007 dataset (MAP%)

algorithm	Bottle	Plant	Chair	Boat	TV	Table	Bird	Sheep	MAP
Fast R-CNN [7]	38.3	31.8	42.8	59.4	70.4	68.9	69.3	70.1	56.4
Faster R-CNN [8]	52.1	38.8	52.0	65.5	72.6	65.7	70.9	73.6	61.4
Faster R-CNN [8]	55.9	41.7	56.7	68.3	72.0	69.4	76.2	78.6	64.9
YOLOv2 [11]	51.8	49.1	51.1	59.2	68.7	58.5	74.8	77.2	61.3
DSSD [13]	53.9	51.7	61.1	68.4	79.4	78.7	80.5	78.0	69.0
FPEF-SSD [15]	48.7	52.6	51.3	63.4	72.5	74.8	71.7	67.9	62.9
C-SSD [18]	52.7	54.2	63.0	70.6	76.5	76.5	75.8	77.3	68.3
ION [30]	55.7	49.7	57.5	69.8	82.1	73.8	77.4	76.9	67.9
DFSSD [31]	52.2	53.5	61.2	68.1	78.4	75.9	76.6	79.0	68.1
SSD	48.8	50.7	54.7	66.4	74	74.9	73.8	74.9	64.8
PCF-SSD	**52.8**	**54.9**	**61.8**	**71.6**	**76.2**	**77.6**	**79.1**	**79.2**	**69.2**

our proposed PCF-SSD can detect more objects than original SSD. Moreover, small objects can be recognized more easily by PCF-SSD if compared with SSD. Therefore, PCF-SSD has better detection performance, especially for small objects.

4.3.2 Numerical Comparison and Analysis

In order to further verify PCF-SSD has better detection performance on the training dataset with small samples, we trained our algorithm model as well as other algorithm models based on VOC2007 and VOC2012 datasets and conduct a series of testing experiments based on the VOC2007 dataset. The comparison results between PCF-SSD and some other representative algorithms are listed in Table 5, and our proposed algorithm has verified its own superiority. It can be seen that the MAP of PCF-SSD is 8.30%, 5.1%, 1.9% and 2.7% higher than that of Fast R-CNN, Faster R-CNN (VGG), Faster R-CNN (Residual-101) and Fast R-CNN-based ION respectively. Meanwhile, the detection speed of PCF-SSD also presents a superiority because of its relatively few time cost. Although PCF-SSD detection speed is slightly lower than that of the regression-based YOLOv2, its MAP is improved by 4.6%. MAP of the original SSD and PCF-SSD achieves 74.8% and 78.3% respectively. Obviously, PCF-SSD MAP is greatly improved

by 3.5% at the cost of only 6 fps what losing on the detection speed, which can definitely meet the real-time detection requirement completely. PCF-SSD also has good performance if compared with some other SSD-based algorithms including DSSD, FPEF-SSD, C-SSD, DFSSD. The MAP of PCF-SSD is slightly lower than that of DSSD, but it really has a relatively much faster detection speed. The detection accuracy comparisons of different algorithms are also given in Table 6.

As for SSD algorithm there are 8 categories of objects whose recognition accuracy is lower than 75% in the above experiments and are viewed as small objects that are difficult to detect correctly. Table 7 lists comparisons of the detection accuracy between several algorithms with our proposed PCF-SSD, and the corresponding MAPs show that our proposed PCF-SSD can greatly improve small objects' recognition capability. MAP of PCF-SSD is 4.4% higher than that of the original SSD. Compared with other region-based object detection algorithms, PCF-SSD has great advantages in terms of both detection accuracy and speed. If compared with the regression-based one-stage algorithms such as YOLOv2, PCF-SSD has much higher detection accuracy at the cost of losing some detection speed. If compared with other SSD-based small objects detection algorithms, PCF-SSD has verified its superior for almost all the 8 categories of small objects in terms of MAP.

5. Conclusions

In this study, we propose an improved algorithm PCF-SSD for improving the detection performance of the original SSD-based algorithms. The upper layers of the feature pyramid typically contain abstract and rich semantic information, whereas the lower layers typically contain high resolution and information that is more detailed. The proposed PCF-SSD algorithm uses a novel feature fusion method to fuse specific upper layers with specific lower layers in appropriate proportions, resulting in fused feature maps with rich semantic and detailed information. Furthermore, the prior box size is adjusted to improve the detection capability of PCF-SSD for small objects. Finally, an enhanced loss function is suggested to train the network model for accelerating its rapid convergence. A series of experiments are performed to validate the proposed algorithm. A significant improvement with a relatively high value of MAP for object recognition is obtained. The detection speed of PCF-SSD meets real-time requirements. Although PCF-SSD has proven to be superior, there is still room for advancement. In following future works, we will perform super-resolution reconstruction of the upper feature layers and optimize a series of network parameters for obtaining better recognition accuracy and fast recognition speed.

Acknowledgments

This project is supported by national key research and development plan project (2019YFE0108300); national Natural Science Foundation of China (52172379, 62001058); Shaanxi Province Key R&D Program Project (2019GY-039) and the special funded project of basic scientific research operation fees of central universities of Chang'an University (300102241201).

References

[1] M. Feniche and T. Mazri, "Lane detection and tracking for intelligent vehicles: A survey," 2019 International Conference of Computer Science and Renewable Energies (ICCSRE), IEEE, 2019.
[2] G.M. Lingani, D.B. Rawat, and M. Garuba, "Smart traffic management system using deep learning for smart city applications," 2019 IEEE 9th annual computing and communication workshop and conference (CCWC), IEEE, 2019.
[3] N. Arunkumar, M.A. Mohammed, M.K.A. Ghani, D.A. Ibrahim, E. Abdulhay, G. Ramirez-Gonzalez, and V.H.C. de Albuquerque, "K-means clustering and neural network for object detecting and identifying abnormality of brain tumor," Soft Computing, vol.23, no.19, pp.9083–9096, 2019.
[4] W. Zhiming and Z. Hang, "A fast image retrieval method fusion of multilayer convolutional neural network features," Journal of Computer-Aided Design & Computer Graphics, vol.31, no.8, pp.1410–1416, 2019.
[5] P. Hao, "Military object recognition based on deep learning," Hangzhou Dianzi University, 2018.
[6] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) IEEE, 2015.
[7] R. Girshick, "Fast R-CNN," Computer Science, pp.1440–1448, 2015.
[8] 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, 2017.
[9] W. Liu, et al., "SSD: Single shot multiBox detector," Lect. Notes Comput. Sci., pp.21–37, Springer, Amsterdam, Netherlands, 2016.
[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," Computer Vision & Pattern Recognition, 2016.
[11] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," IEEE Conference on Computer Vision & Pattern Recognition IEEE, pp.6517–6525, 2017.
[12] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv preprint, arXiv:1804.02767, 2018.
[13] C.Y. Fu, et al., "DSSD: Deconvolutional single shot detector," arXiv preprint, arXiv:1701.06659, 2017.
[14] L. Cui, R. Ma, P. Lv, X. Jiang, Z. Gao, B. Zhou, and M. Xu, "MDSSD: multi-scale deconvolutional single shot detector for small objects," Sciece China, Information Sciences, vol.63, no.2, 2020.
[15] P. Qin, C. Li, J. Chen, and R. Chai, "Research on improved algorithm of object detection based on feature pyramid," Multimedia Tools & Applications, vol.78, pp.823–927, 2019.
[16] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "DSOD: Learning Deeply Supervised Object Detectors from Scratch," 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, 2017.
[17] P. Luo and L.I. Ming, "Vehicle detection algorithm of small target with feature fusion," Journal of Henan University of Science and Technology (Natural Science), vol.40, no.2, pp.40–44+6–7, 2019.
[18] D. Yang, C. Bi, L. Mao, and R. Zhang, "Contour feature fusion SSD Algorithm," 2019 Chinese Control Conference (CCC), pp.40–44, 2019.
[19] S. Qu, K. Huang, A. Hussain, and Y. Goulermas, "MPSSD: Multi-Path Fusion Single Shot Detector," 2019 International Joint Confer-

ence on Neural Networks (IJCNN), 2019.
[20] Z. Li and F. Zhou, "FSSD: Feature fusion single shot multibox detector," arXiv preprint, arXiv:1712.00960, 2017.
[21] G. Chen, et al., "Research on multi-target parts recognition system based on improved SSD," New Technology & New Process, 2019.
[22] J.W. Wen, Y.W. Zhan, and C.H. Lietal, "Design of atrous filter to strengthen small object detection capability of SSD," Application Research of Computers, vol.36, no.3, pp.861–865, 2019.
[23] C. Tang, Y.S. Ling, K.D. Zheng, et al., "Object detection method of multi-view SSD based on deep learning," Infrared and Laser Engineering, vol.47, no.1, pp.290–298, 2018.
[24] J. Zhang, C. Xu, M. Tang, et al., "Research on improved object detection method based on SSD," Laser and Infrared, vol.49, no.8, p.7, 2019.
[25] Q. Chang, et al., "Object detection algorithm based on image super-resolution network," Modern Computer, vol.25, p.4, 2019.
[26] J. Leng and Y. Liu, "An enhanced SSD with feature fusion and visual reasoning for object detection," Neural Computing & Applications, vol.31, no.10, pp.6549–6558, 2018.
[27] Y. Liang and J. Li, "Small objects detection method based on multi-scale non-local attention network," Journal of Frontiers of Computer Science and Technology, vol.14, no.10, pp.1744–1753, 2020.
[28] Z. Ji, Q. Kong, and J. Wang, "Object detection algorithm guided by dual attention models," Laser & Optoelectronics Progress, vol.57, no.6, 061008, 2020.
[29] H. Li, C. Li, J. An, et al., "Remote sensing image object detection based on improved attention mechanism by convolutional neural network," Journal of Image and Graphics, vol.24, no.8, pp.1400–1408, 2019.
[30] S. Bell, C.L. Zitnick, K. Bala, and R. Girshick, "Inside-Outside Net: Detecting Objects in Context with Skip Pooling and Recurrent Neural Networks," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2016.
[31] Z. Ye and J. Zheng, "object detection algorithm DFSSD based on autonomous driving scenario," Computer Engineering and Applications, pp.1–10, 2020.
[32] Z. Siyu and Z. Yi, "Small object pedestrian detection based on multi-scale feature fusion," Computer Engineering and Science, vol.9, pp.1627–1634, 2019.
[33] H. Li, K. Lin, J. Bai, A. Li, and J. Yu, "Small object detection algorithm based on feature pyramid-enhanced fusion SSD," Complexity, vol.2019, 2019.
[34] G.P. Wang, M. Duan, and C.Y. Niu, "Stochastic gradient descent algorithm based on convolution neural network," Computer Engineering and Design, vol.39, no.2, p.6, 2018.
[35] M. Abadi, et al., "Tensorflow: A system for large-scale machine learning," 12th USENIX symposium on operating systems design and implementation (OSDI 16), 2016.
[36] M. Everingham, S.M. Ali Eslami, L. Van Gool, C.K.I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes Challenge: A Retrospective," International Journal of Computer Vision, vol.111, no.1, pp.98–136, 2015.

Xi Zhang got the M.S. degree in Advanced Computer Science from University of Birmingham, UK, in 2017. From 2017 to now, she stayed in Chang'an University, school of information engineering, China, as a research fellow.

Yanan Zhang got M.Eng. degree in Information and Communication Engineering from Chang'an University, Xi'an City, Shannxi Province, China, in 2021. She is working at China Mobile Group Shanxi Company Limited, Taiyuan, Shanxi, China now.

Tao Gao received the B.Sc. degree, M.Sc. and Ph.D. degree in School of Electronics and Information, Northwestern Polytechnical University in 2002, 2016 and 2010, respectively. He is working in School of Information Engineering, Chang'an University, Xi'an City, Shannxi Province, China. He current research interests include image processing and computer vision.

Yong Fang received the B.Eng., M.Eng., and Ph.D. degrees in communications engineering from the School of Communications Engineering, Xidian University, Xi'an, Shaanxi, China, in 2000, 2003, and 2005, respectively. Dr. Fang was a recipient of the EAI IoTaaS Best Paper Award in 2020 and the AMIA TBI Best Student Paper Award in 2016. He served as an Area Editor for International Journal of Electronics and Communications from 2016 to 2017. He was a (leading) guest editor for Taylor&Francis Journal of Intelligent Transportation Systems, Springer Journal of Mobile Networks and Applications, Elsevier Journal of Sustainable Cities and Society. He is also an Associate Editor of IEEE Transactions on Communications and IEEE Communications Letters.

Ting Chen received Ph.D. degrees in Information and Communication Engineering from Xi'dian University in 2011. She is an associate professor in School of Information Engineering, Chang'an University, Xi'an, China. Her research interests include signal processing, wireless networks, etc.