


Article

# CP-SSD: Context Information Scene Perception Object Detection Based on SSD

Yun Jiang <sup>†</sup>, Tingting Peng <sup>\*,†</sup>  and Ning Tan <sup>†</sup> 

College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China

\* Correspondence: pengtingting14@gmail.com

† These authors contributed equally to this work.

Received: 20 June 2019; Accepted: 8 July 2019; Published: 11 July 2019



**Abstract:** Single Shot MultiBox Detector (SSD) has achieved good results in object detection but there are problems such as insufficient understanding of context information and loss of features in deep layers. In order to alleviate these problems, we propose a single-shot object detection network Context Perception-SSD (CP-SSD). CP-SSD promotes the network's understanding of context information by using context information scene perception modules, so as to capture context information for objects of different scales. Deep layer feature map used semantic activation module, through self-supervised learning to adjust the context feature information and channel interdependence, and enhance useful semantic information. CP-SSD was validated on benchmark dataset PASCAL VOC 2007. The experimental results show that, compared with SSD, the mean Average Precision (mAP) of the CP-SSD detection method reaches 77.8%, which is 0.6% higher than that of SSD, and the detection effect was significantly improved in images with difficult to distinguish the object from the background.

**Keywords:** object detection; deep convolutional neural network; context feature scene perception; semantic activation; parallel dilated convolution

## 1. Introduction

Object detection is one of the main tasks of image processing. Its main purpose is to be able to accurately locate and classify objects in images. It has been comprehensively used in many communities such as face recognition, road detection, and driverless car, and so forth. The traditional object detection methods such as Histogram of Oriented Gradient (HOG) [1], Scale Invariant Feature Transform (SIFT) [2], are based on hand-crafted features (e.g., RGB color, texture, Gabor filter and gradient). Hand-crafted features lack sufficient discriminative representation, perform poorly in generalization ability and are easily affected by low contrast quality. It is difficult and time-consuming to perform object detection on a large and complex dataset.

The deep convolutional neural network method promotes the understanding of dynamic objects. However, it still faces the challenge of a lack of rich semantic features and insufficient understanding of context information. In Region-Convolutional Neural Network (R-CNN) series, the selection of regional proposal and repeated convolution of feature maps greatly increases the time and complexity of object detection. R-CNN [3] is several times more accurate than the traditional algorithms based on HOG and SIFT. However, this method of obtaining regional proposals by Selective search in R-CNN will lose a lot of contextual information. In the Single Shot MultiBox Detector (SSD) [4] algorithm, the end-to-end network structure is used to remove the steps in the R-CNN for region proposals. It directly inputs the entire picture into the network structure and predicts objects of different sizes using feature maps of different scales in the Convolutional Neural Network (CNN) [5]. In the backbone network Visual Geometry Group 16 (VGG16) [6], low-level detection feature maps are generated and

then several layers of feature detection maps are constructed, so as to learn semantic information in a layered manner. However, for low-level features, there are usually no appropriate strategies to understand contextual information fully, and to capture strong semantic information successfully, which makes it difficult to be understood for complex scenarios. This may result in inaccurate detection of the object and imperfect low-level feature information can cause high-level language information to be affected. At the same time, the receptive field of each layer in the CNN is fixed and there is an inconsistency between objects of different scales in the natural image, which may impair the object detection performance. For small objects, feature extraction is not easy, which may result in a loss of information and the inability to detect small objects.

According to the above analysis, a new single-shot network model CP-SSD is designed to alleviate the problems in the above SSD. Two modules of context information sensing and semantic activation are added to the original SSD. The context information sensing module uses different convolution kernels to perceive objects of different sizes and combines their important context information. The contextual information of different regions can help to distinguish the goals and backgrounds of various categories more accurately. In addition, inspired by SE-Net [7], a semantic activation module is used on the additional detection feature map built in the SSD. The semantic activation module uses the self-attention mechanism to learn the relationship between the channel and the object, and learn the weights of different channels.

## 2. Related Work

In recent years, the CNN has achieved great success in computer vision tasks, such as image classification [8–12], segmentation [13,14] and object detection [3,4,15–21]. Among them, object detection is a basic task that has been extensively studied. There are two frameworks for the network proposed by the research community for object detection—the two-stage framework and the a single-stage framework. In the two-stage framework, such as the R-CNN series, region proposals with different scales and aspect rates are predicted by extracting feature maps using CNN. Then, classification and regression are carried out based on features extracted from the region proposals. With the help of R-CNN [3], the deep learning mechanism was introduced into object detection for the first time in Reference [3], and the algorithm for adaptive object detection was proposed. It generates region proposals which define the set of candidate detections available to the detector. Each region proposal is input into a large CNN to extract features and then use category-specific linear Support Vector Machines (SVMs) for classification. Finally, the object is detected by regression correction. In SPP-Net [17], the spatial pyramid pooling layer is added into the network behind the R-CNN network structure, which overcomes the shortcomings of R-CNN requirements for the size of the input region proposals and hence it helps to improve detection accuracy. For Fast R-CNN [18], a region of interest (RoI) pooling layer based on SPP-Net is introduced, which reduced the number of convolution layers and greatly improved the detection speed. For Faster R-CNN, the regional proposal network (RPN) is introduced in the process of extracting region proposals by Fast R-CNN. The regional proposal is generated on the convolution feature diagram of the last layer of RPN module and they are input into the RoI pooling layer of Fast R-CNN, so as to optimize the selection of the regional proposal, to reduce repeated feature extraction and to improve the accuracy of regional extraction and network training speed. The RoI pooling layer of Faster R-CNN [19] to RoIAlign by Mask R-CNN [20] improved and the bilinear interpolation method is adopted to reduce the position error of the boundary regression box. Meanwhile, a Mask generation task was added to improve the detection accuracy to some extent.

In single-stage frameworks, such as the You Only Look Once (YOLO) [21] series and SSD, the object classifiers and regressions are applied in a dense manner, without the need for object-based pruning. They all classify and regress a set of pre-computed anchors. In the YOLO algorithm, the end-to-end training mode is used to reduce the network structure. Although the accuracy is slightly worse than that of the R-CNN series, the speed is much faster than that of the R-CNN series. It reduces the error rate of background image detection and enhances the global information of the

image. SSD also has a great improvement in speed, and uses a backbone network (for example, VGG16) to generate a low-level detection feature map. Based on this, it constructs several layers of object detection feature maps to learn semantic information in a layered manner, with lower layers detecting smaller objects and higher layers detecting larger objects, so as to eliminate region proposals and subsequent pixel resampling stages.

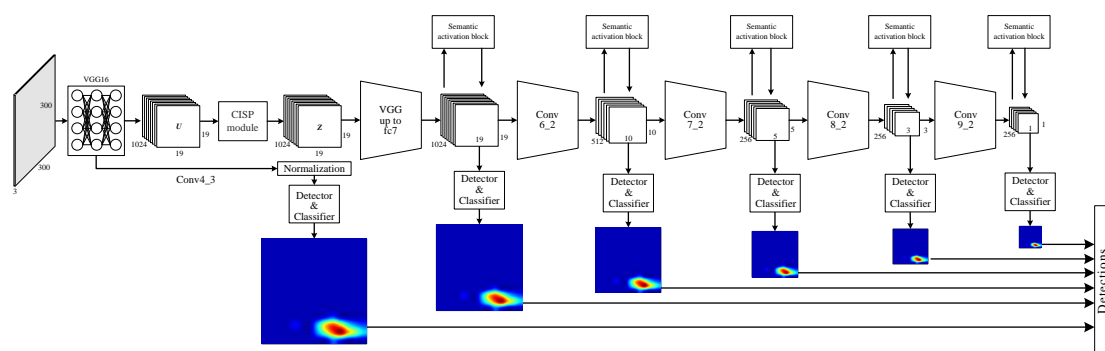
### 3. CP-SSD

CP-SSD (Context Perception SSD) is a single-shot object detection network based on SSD, which consists of three main parts,—the SSD model, context information scene perception module—which is used to capture local context information of different sizes, and the semantic activation module, which enriches the semantic information in a self-supervised manner. Please refer to Figure 1 for the structure of CP-SSD.

In SSD, VGG16 is used as the backbone network to generate the low-level detection feature map  $U$ . Based on that, the feature map  $U$  is continuously downsampled through a series of convolutional layers with a stride of 2 (i.e., fc7 to conv9\_2) by applying anchors of different sizes and aspect rates in a hierarchical manner, so as to detect objects of small to large sizes.

In the context information scene perception module, we used multiple dilated rate convolution layers in parallel and each dilated convolution layer has different dilated rates. The larger the dilated rate is, the larger the receptive field of the convolution kernel is. The context information sensing module performs feature extraction on the feature map  $U$  through convolution kernels of different receptive fields so that the model can perceive changes in the context information between different scales and different sub-regions. In this way, the loss of feature information is reduced, and the image is understood more comprehensively.

In the deeper detection layer, a higher level of detection feature map is enhanced using a semantic activation module. In order to detect objects of different sizes, the feature map is downsampled in the fc7 to conv9\_2 layer, which reduces the resolution of feature maps and increases the receptive field of the model. However, semantic information and location information are lost in each downsampling. Therefore, the semantic activation module is used on fc7 to conv9\_2 to learn the relationship between the channel and the object by self-supervised learning, so as to adjust and enrich the semantic information.



**Figure 1.** Context Perception-Single Shot MultiBox Detector (CP-SSD) network structure.

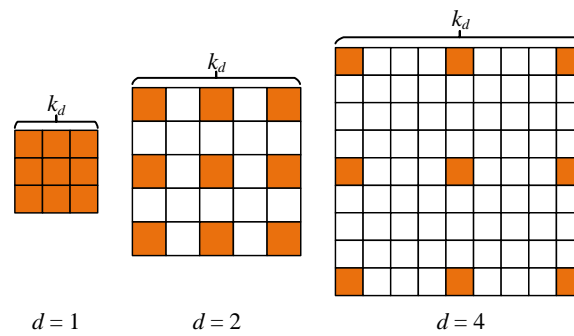
#### 3.1. Dilated Convolution and Receptive Field

The dilated convolution [13] increases the receptive field of the convolution kernel without introducing extra parameters. The formula for the 1-D dilated convolution is defined as follows:

$$y[i] = \sum_{k=1}^K x[i + r \cdot k]w[k] \quad (1)$$

Here,  $x[i]$  denotes the input signal,  $y[i]$  denotes the output signal,  $i$  denotes the dilated rate,  $w[k]$  denotes the  $k$ -th parameter of the convolution kernel, and  $K$  is the size of the convolution kernel. In the standard convolution,  $r = 1$ .

The 2-D dilated convolution is constructed by inserting 0 between each weight of the convolution kernel. For a convolution kernel of size  $k \times k$ , the size of the resulted dilated convolution kernel is  $k_d \times k_d$ , where  $k_d = k + (k - 1) \times (r - 1)$ . Therefore, the larger the dilated rate  $r$  is, the larger the receptive field of the convolution kernel is. For example, for a convolution kernel of  $k = 3$ , when  $r = 4$ , the corresponding receptive field size is 9. Figure 2 shows the dilated convolution kernel for different dilated rates. In Figure 2 the dark portion denotes the effective weight, and the white portion denotes the inserted zero.



**Figure 2.** Dilated Convolution with Different Dilated Rates.

### 3.2. Context Information Scene Perception Module

In the object detection, the objects to be detected usually have a different scale, so the feature map must contain feature information of receptive fields at different scales. In deep learning, the size of the receptive field can be roughly expressed as the degree of utilization of the context information by the model. But at a high level, the previously important semantic information usually could not be combined by the network. Inspired by PSPNet [22], a contextual information scene perception module was designed, which achieves this goal by parallel dilated convolution of different dilated rates. The same feature map is input to these convolutional layers and different dilated rates  $d$  is used to make the convolution kernels have different receptive fields. Then, the feature information of different sizes is sampled. Finally, the feature maps of these outputs are concatenated together. The structure of the context information scene perception module is shown in Figure 3. Firstly, a  $1 \times 1$  convolution is used to reduce the number of channels of the feature map  $\mathbf{U} \in \mathbb{R}^{W \times H \times 512}$ , so as to obtain a feature map  $\mathbf{U}' \in \mathbb{R}^{W \times H \times 512}$ . Then, the dilated convolution  $(d_1, d_2, d_3, d_4) = (1, 2, 4, 6)$  with four different dilated rates is used in parallel to carry out feature sampling on the feature map  $\mathbf{U}'$ , and the feature map  $\mathbf{V}_1 \in \mathbb{R}^{W \times H \times 256}$ ,  $\mathbf{V}_2 \in \mathbb{R}^{W \times H \times 256}$ ,  $\mathbf{V}_3 \in \mathbb{R}^{W \times H \times 128}$ , and  $\mathbf{V}_4 \in \mathbb{R}^{W \times H \times 128}$  is obtained. Finally, the feature map is spliced to obtain the final feature map  $\mathbf{Z} \in \mathbb{R}^{W \times H \times 1024}$ ,  $\mathbf{Z} = [\mathbf{U}', \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3, \mathbf{V}_4]$ .

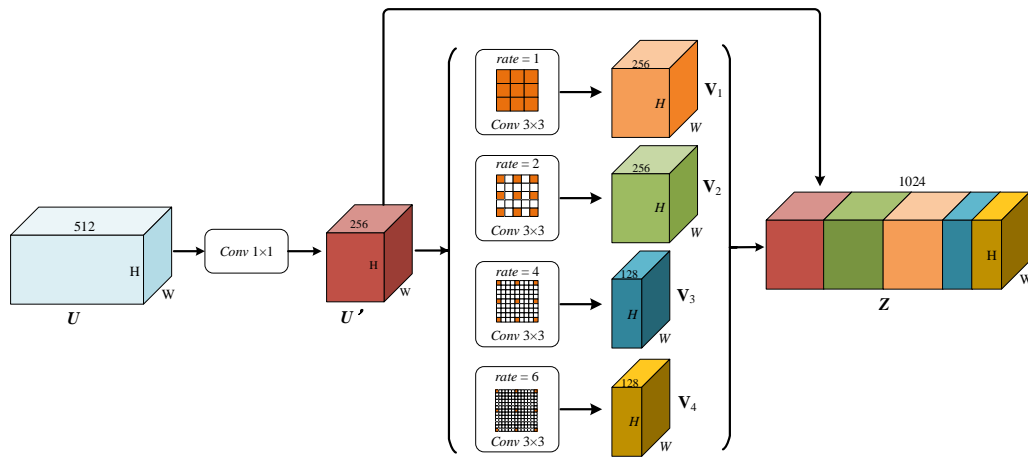


Figure 3. Context information scene perception module.

### 3.3. Semantic Activation Block

The semantic activation module is used to adjust the interdependence between contextual feature information and channels by self-supervised learning, and to selectively enhance useful semantic information according to the self-attention mechanism and suppress harmful feature information.

The semantic activation module is shown in Figure 4, which consists of three steps: spatial pooling  $f_{gap}(\cdot)$ , channel-wise attention learning  $f_{fcl}(\cdot, \theta)$ , and channel weights adaptive  $f_{fuse}(\cdot, \cdot)$ .

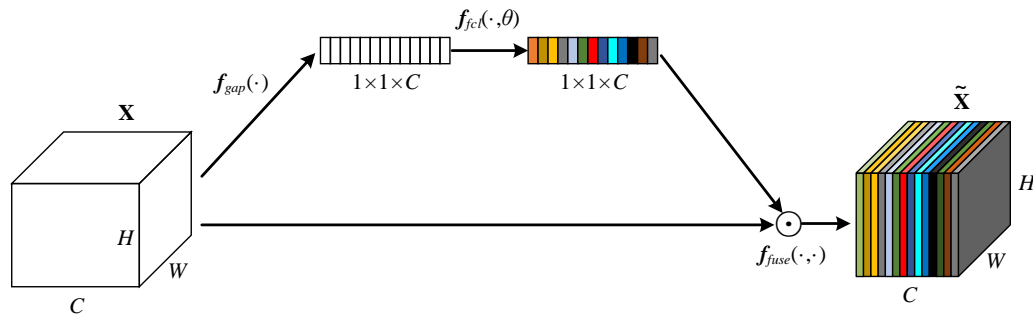


Figure 4. Semantic activation block.

**Spatial pooling:** For a given input  $\mathbf{X} \in \mathbb{R}^{H \times W \times C}$ , by globally pooling  $\mathbf{X}$  with  $f_{gap}(\cdot)$  to generate  $\mathbf{V} \in \mathbb{R}^C$ , the  $i$ -th element in  $\mathbf{V}$  is obtained as following:

$$v_i = f_{gap}(\mathbf{X}_C) = \frac{1}{W \times H} \sum_{i=1}^W \sum_{j=1}^H x_c(i, j) \quad (2)$$

**Channel-wise attention learning:** In order to make full use of the information summarized in  $\mathbf{V}$ , the  $f_{fcl}(\cdot, \theta)$  operation is used to capture the direct correlation of the channel. To do this, a gating mechanism and a sigmoid activation function are used as follows:

$$\mathbf{S} = f_{fcl}(\mathbf{V}, \theta) = \sigma(\mathcal{G}(\mathbf{V}, \theta)) = \sigma(\theta_2 \varphi(\theta_1 \mathbf{V} + b_1) + b_2) \quad (3)$$

Here  $\varphi$  denotes the ReLU activation function and  $\sigma$  denotes the *Sigmoid* activation function,  $\theta_1 \in \mathbb{R}^{C' \times C}$ ,  $\theta_2 \in \mathbb{R}^{C \times C'}$ . In order to reduce the complexity of the model, we use two fully connected methods to form the bottleneck layer. That is, firstly the dimension is reduced to  $C'$ , and then it is upgraded to  $C$ . In the experiment, we set  $C' = \frac{1}{2}C$  in all modules.

**Channel weights adaptive:** The final output selects the relevant semantic features by using  $f_{fuse}(\cdot, \cdot)$ , to make sure that the related semantic information is assigned a larger weight, and the

unrelated semantic information is assigned a smaller weight for generating the final feature map  $\tilde{\mathbf{X}}$ . Here, the  $c$ -th channel in  $\tilde{\mathbf{X}}$  is defined as:

$$\tilde{\mathbf{x}}_c = \mathbf{f}_{fuse}(\mathbf{x}_c, \mathbf{s}_c) = \mathbf{x}_c \cdot \mathbf{s}_c \quad (4)$$

Here  $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \dots, \tilde{\mathbf{x}}_C]$ ,  $\mathbf{x}_c \in \mathbb{R}^{H \times W}$ .

#### 4. Analysis and Discussion of Experimental Results

We implemented the proposed model CP-SSD with help of the pytorch [23] deep learning framework. The server configuration of the training model was: Intel(R) Xeon(R) E5-2620 v3 2.40GHz CPU, Tesla K80 GPU and Ubuntu64 system.

##### 4.1. Data Sets and Data Enhancements

PASCAL VOC [24] is a benchmark dataset for visual object classification recognition and detection, which includes 20 categories. The VOC2007 test section (testing dataset of VOC2007) is widely used by the research community for validating the performance of object detection models. In our training process, all the samples of train and val of VOC2007 and VOC2012 are used as the training set. The training set contains 16,551 pictures with 40,058 objects and the testing set contains 4952 pictures with 12,032 objects. In this dataset, smaller objects account for a large proportion of the objects.

In order to make the model more robust to various input object sizes and shapes, each training image is randomly sampled in one of the following ways:

(1) The original image without any further processing; (2) The original image with overlap of 0.1, 0.3, 0.5, 0.7 or 0.9 is selected; (3) A portion of the original image is cropped randomly.

After the above sampling step, each sampling area was resized to a fixed size ( $300 \times 300$ ) and flipped at a probability level of 0.5.

##### 4.2. Experimental Parameter Settings

In order to compare the effectiveness of the CP-SSD network model with SSD, we used the same training settings as SSD. For the model, first we set  $lr = 10^{-3}$  to train for 80k iterations, then we set  $lr = 10^{-4}$  for 20k iterations and finally we set  $lr = 10^{-5}$  for another 20k iterations. The momentum was fixed to be 0.9 and the weight decay was set to be 0.0005,  $batchsize = 32$ , and the backbone structure of the model was initialized using pre-trained VGG16 weights.

##### 4.3. Ssd with Context Information Scene Perception Module

In Table 1, we validate the SSD with and without the context information scene perception module(CISP) for detection performance. In terms of general object detection, the overall performance of the model reached 77.6% after applying the context information sensing module to the SSD and the performance improved by 0.4% compared with the original SSD. Especially for samples with similar backgrounds and objects, the original SSD cannot detect some objects because it cannot understand the context information. Using the context information sensing module to perceive and fuse the local context information at different scales, it is possible to understand some complex scenes and detect the objects from the background.

**Table 1.** Test results of SSD and SSD+CISP.

Method	mAP	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow
SSD [4]	77.2	78.8	85.3	75.7	71.5	49.1	85.7	86.4	87.8	60.6	82.7
SSD+CISP	77.6	80.5	85.1	76.0	71.1	52.9	86.1	86.4	87.1	61.3	81.8
Method	mAP	Table	Dog	Horse	Mbike	Person	Plant	Sheep	Sofa	Train	tv
SSD [4]	77.2	76.5	84.9	86.7	84.0	79.2	51.3	77.5	78.7	86.7	76.2
SSD+CISP	77.6	76.7	84.5	86.4	85.0	79.0	53.0	76.5	80.9	85.5	77.3

#### 4.4. Ssd with Semantic Activation Block

In Table 2, we show the detection performance of the SSD with and without semantic activation block (SAB). For high-level low-resolution feature maps, self-supervised adjustment of channel weights to enhance useful feature information can better distinguish between object and background. From the table, we can see that the semantic activation module can improve the performance of the model by 0.4%, which indicates the effectiveness of the semantic activation module. Compared with the original SSD, although the addition of the semantic activation module increases the amount of parameters and computation, the cost of the increased parameters and computation on the running time required by speed of the model is negligible.

**Table 2.** Test results of SSD and SSD+semantic activation block (SAB).

Method	mAP	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow
SSD [4]	77.2	78.8	85.3	75.7	71.5	49.1	85.7	86.4	87.8	60.6	82.7
SSD+SAB	77.6	81.3	84.9	75.7	72.0	50.7	85.4	86.4	87.9	61.8	82.3
Method	mAP	Table	Dog	Horse	Mbike	Person	Plant	Sheep	Sofa	Train	tv
SSD [4]	77.2	76.5	84.9	86.7	84.0	79.2	51.3	77.5	78.7	86.7	76.2
SSD+SAB	77.6	77.7	85.6	87.7	81.9	79.1	52.4	77.5	81.6	84.7	76.3

#### 4.5. Comparison of Methods

In Table 3, we compared the R-CNN, YOLO, and SSD methods on the VOC 2007 test dataset. For RCNN based algorithms, RCNN [3] is the first algorithm to use CNN for object detection. It has great shortcomings in the selection of region proposals. Too many region proposals are selected by the algorithm, which requires a lot of memory and the normalization process of the input network makes the algorithm lose a lot of context information and features, resulting in only 50.2% positioning accuracy. In order to cope with solving the feature loss problem of R-CNN in image normalization, Fast-RCNN [18] inputs the whole image into the network and extracts fixed-length feature vectors from the feature map through the region of interest (RoI) pool layer. The resulting classification and coordinate information eventually increased the accuracy to 70.0%. However, Faster-RCNN still does not solve the problem caused by 2000 regional proposals generated by selective search. Therefore, in Faster-RCNN [19] algorithms, the RPN module is proposed, which utilizes 9 kinds of anchors with different area ratios and applying CNN to complete the object detection completely. The mAP reached 76.4%. In YOLO [21] algorithms an end-to-end network is used to remove the selection of regional proposal individually, combining the selection of regional proposal with the object detection network. Due to the simple network structure, the speed of object detection is much higher than that of the RCNN based algorithms but there are great restrictions to the position and size of the object. The detection effect of mAP is only 57.9%, especially for small objects. In SSD [4], low-level feature maps are separated to improve the detection effect of small objects but there are shortcomings of insufficient semantic information. The additional detection layer features use the downsampling method to increase the receptive field but the resolution of the downsampling reduction feature map



causes a large loss of feature information and the mAP on the testing dataset is only 77.2%. The mAP of the CP-SSD on the testing dataset reached 77.8%, which is 0.6% higher than the original SSD.

In CP-SSD, we use the CISP module to fuse context information and prior information between different scales and different sub-regions from feature maps  $U$ . In the context information perception module, convolution with different dilated rates can be used in parallel to capture different sizes of objects, which makes the model understand local context information more comprehensively. It alleviates the problem that SSD lacks understanding of semantic scene and context information. In the higher level feature map, we proposed the semantic activation module to enhance the semantic information. In the semantic activation module, the global average pooling method is used to remove the spatial information. It learns the relationship between channels and objects in a self-supervised way, promotes the useful feature information, restrains the irrelevant feature information and adjusts and enriches the semantic information. At the same time, SSD uses ResNet101 instead of VGG16 as the backbone network in the experiment. The network structure of ResNet101 is deeper than that of VGG16 and its feature extraction ability is stronger. However, the results of our proposed method using VGG16 (77.8%) perform significantly better than those using the SSD model of ResNet101 (77.1%), which highlights the effectiveness of the CP-SSD method.

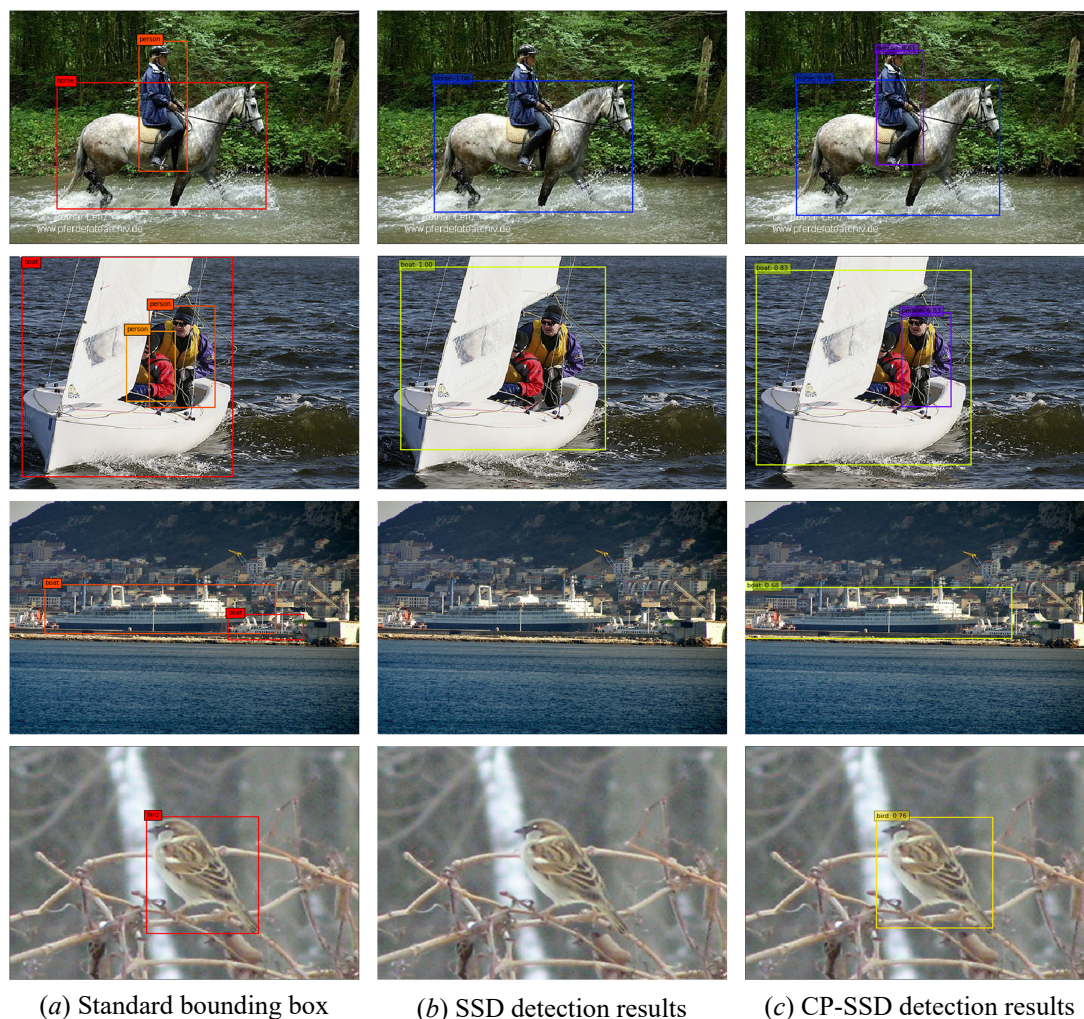
**Table 3.** Test results of CP-SSD in PASCAL VOC2007.

Method	backbone	mAP	Aero	bike	bird	boat	bottle	bus	car	cat	chair	cow
RCNN [3]	AlexNet	50.2	67.1	64.1	46.7	32.0	30.5	56.4	57.2	65.9	27.0	47.3
Fast [18]	VGG16	70.0	77.0	78.1	69.3	59.4	38.3	81.6	78.6	86.7	42.8	78.8
Faster [19]	VGG16	73.2	76.5	79.0	70.9	65.5	52.1	83.1	84.7	86.4	52.0	81.9
Faster [8]	ResNet101	76.4	79.8	80.7	76.2	68.3	55.9	85.1	85.3	89.8	56.7	<b>87.8</b>
RON384++ [25]	VGG16	77.6	<b>86.0</b>	82.5	76.9	69.1	<b>59.2</b>	86.2	85.5	87.2	59.9	81.4
Shrivastava et al. [26]	VGG16	76.4	79.3	80.5	76.8	<b>72.0</b>	58.2	85.1	<b>86.5</b>	<b>89.3</b>	60.6	82.2
YOLO [21]	Darknet	57.9	77.0	67.2	57.7	38.3	22.7	68.3	55.9	81.4	36.2	60.8
SSD321 [4]	ResNet101	77.1	76.3	84.6	79.3	64.6	47.2	85.4	84.0	88.8	60.1	82.6
SSD300 [4]	VGG16	77.2	78.8	85.3	75.7	71.5	49.1	85.7	86.4	87.8	60.6	82.7
CP-SSD(SSD+CISP+SAB)	VGG16	<b>77.8</b>	83.9	<b>86.3</b>	<b>80.1</b>	69.9	50.6	<b>86.5</b>	85.6	88.4	<b>62.8</b>	79.4
Method	backbone	mAP	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
RCNN [3]	AlexNet	50.2	40.9	66.6	57.8	65.9	53.6	26.7	56.5	38.1	52.8	50.2
Fast [18]	VGG16	70.0	68.9	84.7	82.0	76.6	69.9	31.8	70.1	74.8	80.4	70.4
Faster [19]	VGG16	73.2	65.7	84.8	84.6	77.5	76.7	38.8	73.6	73.9	83.0	72.6
Faster [8]	ResNet101	76.4	69.4	<b>88.3</b>	<b>88.9</b>	80.9	78.4	41.7	<b>78.6</b>	79.8	85.3	72.0
RON384++ [25]	VGG16	77.6	73.3	85.9	86.8	82.2	79.6	52.4	78.2	76.0	86.2	<b>78.0</b>
Shrivastava et al. [26]	VGG16	76.4	69.2	87.0	87.2	81.6	78.2	44.6	77.9	76.7	82.4	71.9
YOLO [21]	Darknet	57.9	48.5	77.2	72.3	71.3	63.5	28.9	52.2	54.8	73.9	50.8
SSD321 [4]	ResNet101	77.1	76.9	86.7	87.2	<b>85.4</b>	79.1	50.8	77.2	<b>82.6</b>	<b>87.3</b>	76.6
SSD300 [4]	VGG16	77.2	76.5	84.9	86.7	84.0	79.2	51.3	77.5	78.7	86.7	76.2
CP-SSD(SSD+CISP+SAB)	VGG16	<b>77.8</b>	<b>77.9</b>	83.1	88.1	84.5	<b>80.0</b>	<b>53.5</b>	74.1	77.1	86.4	77.0

#### 4.6. Detection Examples

In Figure 5, we visualize some of the images. The localization results of CP-SSD were compared with the original SSD. As shown in Figure 5, in the upper two rows of images, SSD cannot locate people on horseback and boat, while CP-SSD can. CP-SSD uses the semantic activation module to capture more prior information before downsampling, so it can understand the image more accurately. In the lower two rows of images, the boats and buildings in the image are similar in shape and color and the color of the bird is similar to the surrounding environment. SSD cannot accurately detect the position of ships and birds because of the lack of understanding of the scene. CP-SSD can more fully understand the contextual prior information so that it can better distinguish the background and the detected objects, and determine the location of the ship and the bird through the contextual information.





(a) Standard bounding box

(b) SSD detection results

(c) CP-SSD detection results

Figure 5. Partial detection example.

## 5. Conclusions

In this paper, we proposed a single-shot object detection method, CP-SSD, to alleviate the problem of insufficient understanding of contextual scene information in SSD. We introduced a context information scene perception module and captured different scales of contextual information by parallel dilated convolution of different dilated rates, so as to improve the model's ability to understand the scene. Meanwhile, the semantic activation module was used to enrich the semantic information of the feature map in the deep detection feature map. We validated CP-SSD on the PASCAL VOC 2007 benchmark dataset. The experimental results showed that, compared with SSD, YOLO, Faster R-CNN and other methods, our proposed CP-SSD method had better performance on the test set and the mAP was 0.6% higher than that of SSD. In future research, we will work on how to balance the global information feature extraction and improve the accuracy of small object detection.

**Author Contributions:** Y.J. contributed towards the algorithms and the analysis. As the supervisor of Y.J., she proofread the paper several times and provided guidance throughout the whole preparation of the manuscript. T.P. and N.T. contributed towards the algorithms, the analysis, and the simulations and wrote the paper and critically revised the paper. All authors read and approved the final manuscript.

**Funding:** This work was supported by the National Natural Science Foundation of China (No. 61163036), The program of NSFC Financing for Natural Science Fund in 2016 (No. 1606RJZA047), The institutes and Universities Graduate Tutor Project in Gansu (No. 1201-16), The Third Period of the Key Scientific Research Project of Knowledge and Innovation Engineering of the Northwest Normal University (No. nwnu-kjcxgc-03-67).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the International Conference on Computer Vision & Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; pp. 886–893.
2. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
3. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
4. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.
5. LeCun, Y.; Bengio, Y. Convolutional networks for images, speech, and time series. In *The Handbook of Brain Theory and Neural Networks*; The MIT Press: Cambridge, MA, USA, 1995; p. 3361.
6. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
7. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake, UT, USA, 18–22 June 2018; pp. 7132–7141.
8. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
9. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
10. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
11. Lee, J.; Kim, E.; Lee, S.; Lee, J.; Yoon, S. FickleNet: Weakly and Semi-supervised Semantic Image Segmentation using Stochastic Inference. *arXiv* **2019**, arXiv:1902.10421.
12. Wang, Y.; Xie, L.; Liu, C.; Qiao, S.; Zhang, Y.; Zhang, W.; Tian, Q.; Yuille, A. Sort: Second-order response transform for visual recognition. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1359–1368.
13. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv* **2014**, arXiv:1412.7062.
14. Shelhamer, E.; Long, J.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 343–3440.
15. Dai, J.; Li, Y.; He, K.; Sun, J. R-fcn: Object detection via region-based fully convolutional networks. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 379–387.
16. Tang, P.; Wang, X.; Bai, X.; Liu, W. Multiple instance detection network with online instance classifier refinement. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2843–2851.
17. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [[CrossRef](#)] [[PubMed](#)]
18. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
19. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
20. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.

21. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
22. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
23. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in pytorch. In Proceedings of the NIPS 2017 Autodiff Workshop, Long Beach, CA, USA, 9 December 2017.
24. Everingham, M.; Eslami, S.A.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vis.* **2015**, *111*, 98–136. [[CrossRef](#)]
25. Kong, T.; Sun, F.; Yao, A.; Liu, H.; Lu, M.; Chen, Y. Ron: Reverse connection with objectness prior networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5936–5944.
26. Shrivastava, A.; Gupta, A. Contextual priming and feedback for faster r-cnn. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 330–348.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).