A Traffic Scene Target Detection Algorithm with Dual Attention Module

Shouzhao Chen[1, a]

[1]College of Information Engineering, Shanghai Maritime University, Shanghai, China.

[a]1836939283@qq.com

Abstract

Target detection is one of the most challenging problems in the field of computer vision. In the field of intelligent transportation, the use of deep learning algorithms for target detection of vehicles and pedestrians has become a research hotspot. In the natural environment, light is complex and changeable and the interference is large. Pedestrians and vehicles in the video image have problems such as small size and scale change. In order to solve these problems, this paper proposes a detection model based on the combination of Convolutional Block Attention Module and YOLOv3. By introducing channel attention and spatial attention information into the feature extraction process, the weight distribution is learned from the feature, and then this weight distribution is applied to the original feature to change the distribution of the original feature, enhance the effective feature, and suppress the invalid The feature or noise of the network model can improve the feature extraction ability of the network model without significantly increasing the amount of calculation and parameter. At the same time, the parameters of the BN layer are merged into the convolutional layer to improve the forward inference speed of the model. The experimental results show that the detection accuracy of the improved algorithm has been significantly improved. A comparison experiment with the YOLOv3 algorithm on the PASCAL VOC2007 data set, mAP increased by 2.4% and 3.2% on the KITTI-2D data set. The experiment verifies the effectiveness and advancement of this method.

Keywords

Deep learning; Target detection; Attention Module; YOLOv3.

1. INTRODUCTION

Pedestrian and vehicle detection have good application scenarios in the fields of image analysis, unmanned driving, and intelligent robots. In the natural environment, the light is changeable and the interference is large. Pedestrians and vehicles in the video image have problems such as small size and scale change. To solve these problems, it is necessary to design a target detection network with better adaptability and stronger generalization ability.

Current target detectors based on convolutional neural networks can be divided into two categories according to whether to generate a pre-selection box, one is a model based on region candidates, and the other is a model based on regression.

The model based on region candidates can be divided into two stages: one is suggestion of candidate regions, and the other is feature extraction and classification. Typical representatives of this model are: R-CNN [1], Fast R-CNN [2], Faster R-CNN [3]. In 2014, Girshick combined CNN with the target candidate region mechanism to propose the R-CNN model; in 2015, Girshick used the region of interest pooling strategy (RoI Pooling) to improve R-CNN and proposed Fast

R-CNN; Ren proposed a network of regions Combined with the Fast R-CNN algorithm, the Faster R-CNN model is proposed, and the detection speed and accuracy are further improved. Due to the complex network structure and many training parameters of these algorithms, the detection speed is difficult to meet the real-time requirements.

Regression-based models need to predefine default boxes, and then classify objects and predict the boxes on each default box. The typical algorithms of this detection model are YOLO [4], YOLOv2 [5] and SSD [6] algorithms. Compared with the region candidate detection model, although the detection accuracy is lower, the speed is significantly improved. SSD uses a small-size convolution kernel in the higher layers of the network to detect multi-scale feature maps, losing a lot of shallow visual information, which plays an important role in the recognition of small targets. Fu [7] used the residual neural network ResNet-101 [8] to replace the original VGG16 [9] on the basis of SSD, and introduced a deconvolution block at the same time, and proposed a new target detection model DSSD. YOLOv3 [10] is a target detection algorithm proposed by Redmon et al. based on YOLO and YOLOv2. The YOLO (You Only Look Once) series only needs one end-to-end calculation to detect the target. It has the characteristics of lightweight structure and fast calculation speed, and is suitable for real-time target detection. However, the detection accuracy is low, and it is not sensitive to the detection of small targets. For dense, overlapping areas, the detection will be missed.

The YOLO series of methods believe that each area in the feature map contributes the same degree to the final detection result of the model. When detecting vehicles and pedestrians in actual traffic scenes, noise information such as changes in light and adhesion of vehicles to pedestrians are not conducive to the positions of vehicles and pedestrians. Accurate extraction, if such invalid information can be effectively suppressed, and higher weights are applied to the features of the area where vehicles and pedestrians are located, the detection accuracy can be better improved.

The emergence of the attention mechanism can effectively solve such problems. The attention mechanism theory was first proposed by Bahdanau [11] and applied to the field of machine translation. The attention mechanism can effectively learn the weight distribution of different parts of the input data or feature maps, reduce the impact of background information, and only focus on the part of the regional information that is conducive to task realization when processing information, and filter out secondary information to improve the model effect. Improve the recognition ability and robustness of the model.

This paper introduces the Convolutional Block Attention Module (CBAM) [12] based on the YOLOv3 detection model, The attention model focuses on the most relevant features as needed, suppresses invalid area information, improves the accuracy of target detection, and simplifies the feature extraction network (Darknet-53) in the YOLOv3 model to improve the speed of forward chaining of the model.

2. SCHEME DESIGN OF DETECTION NETWORK

2.1. Introduction of YOLOv3 Model

YOLOv3 adopts the network structure of darknet-53(avgpool, full connection layer and softmax at the end of the network are omitted and a convolution layer is added). It combines the basic feature extractor of YOLOv2 and the idea of residual network. As shown in Figure 1. Yolov3 draws lessons from the FPN [13] (feature pyramid networks) network structure, uses multi-scale to detect targets of different sizes, and outputs three feature maps with different scales.

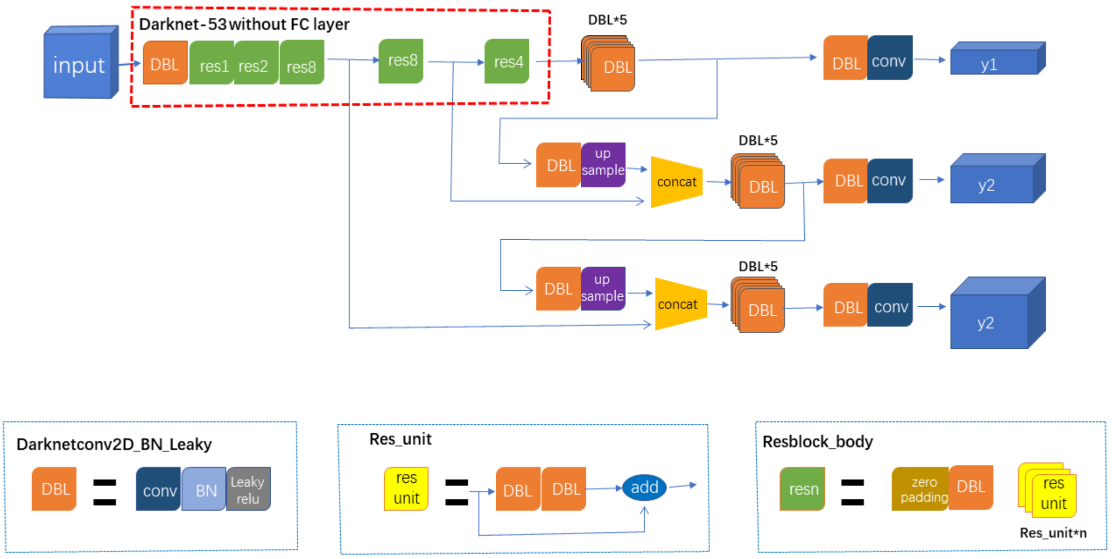

Figure 1. Structure of YOLOv3 network model

Too many layers of the straight-tube network structure will cause problems such as gradient explosion during training. By introducing a residual structure, the depth of the network architecture can be increased and the gradient explosion and disappearance of the training process can be prevented. Darknet-53 consists of 5 residual blocks. Each residual block is composed of residual units. The residual unit consists of DBL units. Two DBL units perform residual operations in the residual blocks. The DBL unit contains Convolutional layer, Batch Normalization and Leaky ReLu activation functions.

2.2. Simplification of Darknet-53 Model

The Batch Normalization (BN) layer [14] is introduced in the YOLOV3 network. By adding a BN layer after each convolutional layer, the network convergence can be accelerated during the network training period and over-fitting can be prevented, and parameter tuning can be simplified to make the network more stable.

When training a deep network model, the BN layer can accelerate the network convergence and prevent over-fitting. It is generally placed after the convolutional layer. After the BN layer normalizes the data, it can effectively solve the problem of gradient disappearance and gradient explosion. Although the BN layer has played an active role in training, however, there are more layers of operations in the forward inference of the network, which affects the performance of the model and takes up more memory or video memory space. Because BN is a linear operation during inference, we can superimpose this linear operation on the previous convolutional layer, and merge the parameters of the BN layer into the convolutional layer to improve the forward chaining speed of the model.

In YOLOv3, the BN calculation process is as follows:

$$x_{out} = \frac{\gamma(x_{conv}-\mu)}{\sqrt{\sigma^2}} + \beta \tag{1}$$

Where $\gamma$is the scaling factor, $\mu$is the mean, σ^2 is the variance, β is the offset, x_{conv} is the convolution calculation result:

$$x_{conv} = \sum_{i=0}^{n}(x_i * w_i) \tag{2}$$

Where x_{out} is the calculation result of the BN layer, and x_{conv} is the convolution calculation result before the BN.

During training, the scaling factor γ, mean μ, variance σ^2, and bias β are always updated, but during forward chaining, the above four parameter values are fixed, that is, during forward inference, the mean and variance come from the training sample Data distribution.

Merging the parameters of the BN layer into the convolutional layer, the mathematical principle is as follows:

$$x_{out} = \frac{\gamma(\sum_{i=0}^{n}(x_i * w_i) - \mu)}{\sqrt{\sigma^2 + 0.000001}} + \beta \tag{3}$$

Which is:

$$x_{out} = \sum_{i=0}^{n}(x_i * \frac{\gamma * w_i}{\sqrt{\sigma^2 + 0.000001}}) - \frac{\gamma * \mu}{\sqrt{\sigma^2 + 0.000001}} + \beta \tag{4}$$

After the merger, the weight parameters become:

$$w^{\prime} = \frac{\gamma * w^{\prime}}{\sqrt{\sigma^2 + 0.000001}} \tag{5}$$

The offset becomes:

$$\beta^{\prime} = \beta - \frac{\gamma * \mu}{\sqrt{\sigma^2 + 0.000001}} \tag{6}$$

The combined calculation expression becomes:

$$x_{out} = \sum_{i=0}^{n}(x_i * w_i^{\prime}) + \beta^{\prime} \tag{7}$$

During training, the BN layer is added directly after the convolutional layer. After the training is completed, the BN layer in the network is discarded, and the original volume integration weight and bias and the four parameters of the BN layer (γ, μ, σ^2, β) are retained. , According to the combined calculation expression, replace the weight of the convolution kernel and update the bias.

2.3. YOLOv3 Detection Model Integrated with Convolutional Block Attention Module

The method in this paper introduces the CBAM module in YOLOv3 to independently learn the weights of pixels between different channels and the weights of pixels at different locations in the same channel, thereby enhancing key features and suppressing redundant features. Compared with the original YOLOv3 model, it improves the accuracy of target detection. The improved detection model is shown in Figure 2. At the same time, because of the simplification of the Darknet model, the speed of forward inference of the model is improved, and the detection performance is further improved.

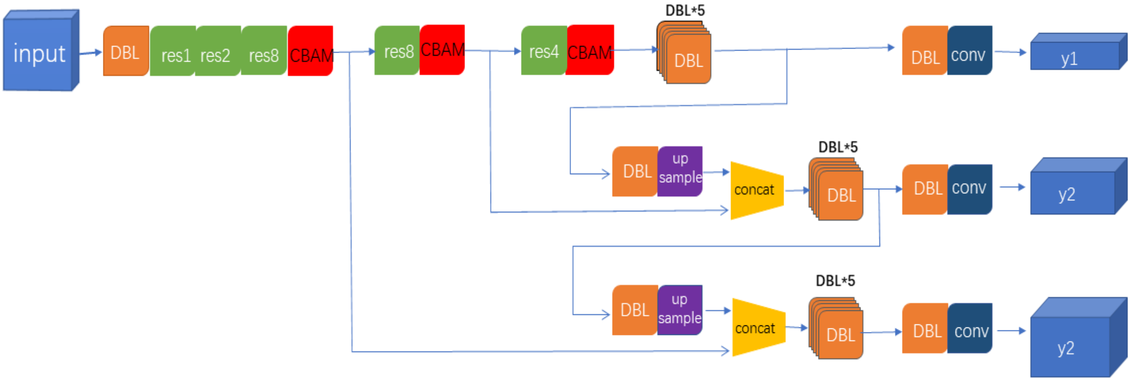

Figure 2. YOLOv3 network structure with convolution block attention module

The CBAM module is mainly composed of two parts, one is the channel attention module, and the other is the spatial attention module. As shown in Figure 3. CBAM is an attention mechanism that extracts space and channel weights through global maximum pooling and global average pooling, and combines the two weights. The network model with the addition of the CBAM module will more accurately focus on the correct objects to be classified in the inference process.

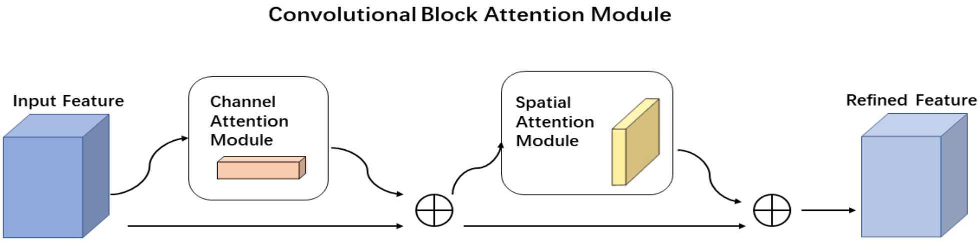

Figure 3. The overview of CBAM

The channel attention module draws on the idea of SENet (Squeeze-and Excitation Networks) [15], first compresses the spatial dimension of the input feature map, and then calculates the channel attention map. Different from SENet, when performing compression, the channel attention module not only considers average pooling, but also additionally introduces maxpooling as a supplement. The specific structure of the channel attention module is shown in Figure 4.

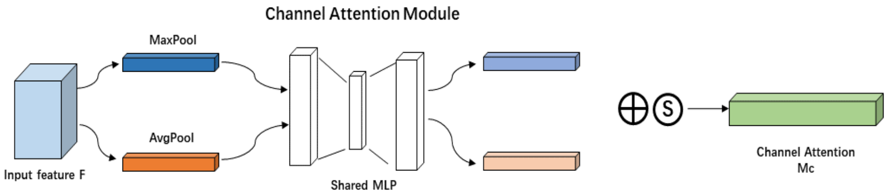

Figure 4. Diagram of channel attention module

The calculation formula of the channel attention mechanism is as follows:

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \quad (8)$$

$$M_c(F)=\sigma(W_1(W_0(F_{avg}^c))+W_1(W_0(F_{\max}^c))) \tag{9}$$

Frepresents the input feature map, F_{avg}^candF_{max}^crepresent the feature maps after average pooling and max pooling, W_0 and W_1 represent the 2-layer parameters in the multilayer perceptron (MLP). The neurons of the two-layer neural network use the ReLU function as the activation function, σ is the sigmoid function. In the calculation, F_{avg}^candF_{max}^cshare the two-layer parametersW_0andW_1in the multilayer perceptron model.

Different from the channel attention, the spatial attention focuses on 'where' is an informative part, which is complementary to the channel attention. The spatial attention module first performs compression at the channel level, and performs average pooling and max pooling on the input feature map in the channel dimension, and then obtains a feature map with two channels. Then a hidden layer containing a single convolution kernel is used to convolve it. The spatial attention module structure is shown in Figure 5.

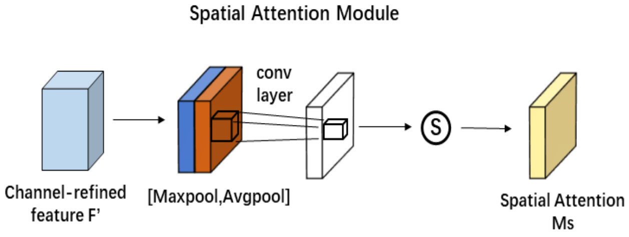

Figure 5. Diagram of spatial attention module

3. EXPERIMENTAL METHODS AND RESULTS

3.1. Data Preparation

This paper mainly studies the possible application of target detection network in unmanned driving scenarios, mainly used to identify other vehicles, pedestrians, non-motor vehicles and other targets in the past. The data set uses two data sets that are frequently used in object detection comparative experiments: Pascal VOC2007 and KITTI-2D. As one of the target detection benchmark data, KITTI includes real image data collected in urban, rural, and highway scenes. Each image can contain up to 15 cars and 30 pedestrians, with various degrees of occlusion and truncation. Pascal VOC2007 is a standard data set to measure the ability of image classification and recognition. It contains 5011 images in the training set, 4952 images in the test set, a total of 9963 images, and a total of 20 types.

3.2. Evaluation Standard

In this paper, the Mean Average Precision (mAP) in the field of target detection is used as the evaluation standard to measure the model's performance in detecting vehicles and pedestrians. The P-R curve reflects the changes in the relationship between different recall rates and the maximum precision accuracy rate under the corresponding recall rates. Average Precision (AP) refers to the area under the P-R (precision-recall) curve, and mAP refers to the average AP of the same model for the detection category. The specific expressions of each indicator are as follows:

$$Pr\,e\,cision=\frac{TP}{TP+FP} \tag{10}$$

$$Re\,c\,all=\frac{TP}{TP+FN} \tag{11}$$

$$AP = \int_0^1 PRdr \tag{12}$$

$$mAP = \frac{1}{C}\sum_{C_i \in C} AP_{(C_i)} \tag{13}$$

TP represents the number of samples that the model predicts to be positive and actually is positive, FP represents the number of samples predicted by the model to be positive and actually negative, FN indicates the number of samples predicted by the model to be negative and actually positive, P, R represent precision rate and recall rate, When calculating the precision rate and recall rate, it is used to judge whether the predicted frame and the real label are equipped with Intersection over Union(IoU), In this paper, it is assumed that the coincidence rate of the label frame and the prediction frame is $\alpha \geq 0.5$ as a positive case, and the rest are negative cases.

$$\alpha = \frac{B_{pred} \cap B_{gt}}{B_{pred} \cup B_{gt}} \tag{14}$$

In the above expression, B_{pred} is the label frame of network prediction, B_{gt} is the label frame of label information, $B_{pred} \cap B_{gt}$ is the area of intersection, and $B_{pred} \cup B_{gt}$ is the area of merging.

3.3. Parameters of the Experimental Platform

The experiment was performed on a server with GPU model NIVIDAGeForceGTX1080Ti(x2), and the operating system was 64-bit Ubuntu16.04, CUDA8.0, cuDNN6.0. Based on the above configuration, use the deep learning framework PyTorch to build the operating environment. The loss function is consistent with Redmon, and the adaptive moment estimation algorithm (Adam) [16] is used to optimize the model. The batchsize is set to 8, the number of iterations is 90k, the initial learning rate is 0.001, and the weight attenuation coefficient is 0.0001. Each time the weight is updated, batch normalization is used for regularization. Training is complete Use the Darknet-53 simplified model for forward chaining.

3.4. Experimental Results and Analysis

In the same data set, the pixel sizes of pedestrians and vehicles in each picture are different, and the number of vehicles and pedestrians contained in the data set is also different. Therefore, in the same data set, the detection accuracy of pedestrians and vehicles is different.

The author's method was trained and verified on the PASCAL VOC2007 data set, and compared with the YOLOv3 algorithm. The experimental results are shown in Table 1:

Table 1. Experimental results of each algorithm on PASCAL VOC2007

Detection model	Vehicle/%	Pedestrian/%	mAP/%	fps
YOLOv3	84.3	76.5	80.4	58
ours	86.4	79.2.	82.8	56

The author's method was trained and verified on the KITTI-2D data set, and compared with the YOLOv3 algorithm. The experimental results are shown in Table 2:

Table 2. Experimental results of each algorithm on KITTI-2D

Detection model	Vehicle/%	Pedestrian/%	mAP/%	fps
YOLOv3	80.4	66.8	73.6	54
ours	84.1	69.5	76.8	51

Analyze the experimental results in Table 1 and Table 2: The mAP of pedestrians and vehicles on the PASCAL VOC2007 and KITTI-2D datasets of this method is increased by 2.4% and 3.2% compared with the original YOLOv3, and the frame per second performed well. It shows that the author's algorithm can better recognize vehicles and pedestrians in traffic scenes. Some experimental results are shown in Figure 6:

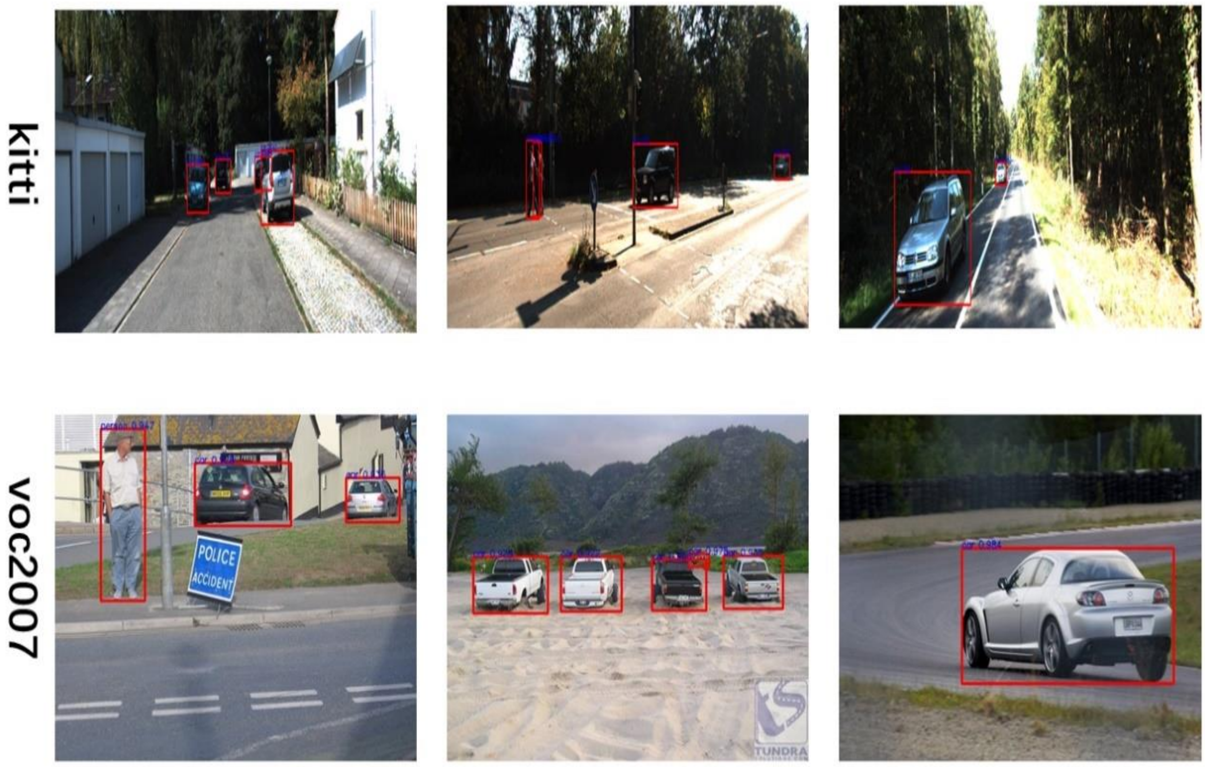

Figure 6. Partial comparison chart of experimental results

4. CONCLUSION

In this paper, a detection model based on the combination of the Convolutional Block Attention Module and the YOLOv3 algorithm is proposed. By increasing the CBAM module, the selective fusion of deep and shallow features enables the model to adaptively find the best weight on the scale and reduce manual labor. Intervention can improve the feature extraction capabilities of the model without significantly increasing the amount of calculations and parameters, and achieve end-to-end training. At the same time, the parameters of the BN layer are merged into the convolutional layer to improve the speed of forward inference of the model. Compared with the original YOLOv3 detection model, the algorithm effectively improves the detection performance of pedestrians and vehicles, and provides an end-to-end solution to the problem of target detection in the field of intelligent driving traffic. There is still room for improvement in the detection accuracy and detection speed of the model in this paper. The next step will be to improve the network for more in-depth research.

REFERENCES

[1] GIRSHICK R B, DONAHUE J, DARRELL T, et al. Region-based convolutional networks for accurate object detection and segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(1): 142-158.

[2] GIRSHICK R B. Fast R-CNN[C]//International Conference on Computer Vision. 2015:1440-1448.

[3] REN S Q, HE K M, GIRSHICK R B, et al. Faster R-CNN: Towards real-time object detection with region proposal networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.

[4] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016: 779-788.

[5] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multi-box detector[C]//Proceedings of 2016 European Conference on Computer Vision, 2016: 21-37.

[6] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition, 2017: 7263-7271.

[7] FU C Y, LIU W, RANGA A, et al. DSSD: Deconvolutional single shot detector [J] 2017: arXiv: 1701.06659.

[8] He K, Zhang S, Ren S, et al. Deep residual learning for image recognition[C]//IEEE Conference on Computer Vision and Pattern Recognition. Boston, USA, 2015: 770-778.

[9] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image Recognition[C]//Proceedings of 2015 International Conference on Learning Representations, 2015: 1-14.

[10] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.

[11] BAHDANAUD, CHO K, BENGIO Y. Neural machine translation by jointly learning to align and translate[J]. Computation and Language, 2014: arXiv: 1409.0473.

[12] Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European Conference on Computer Vision (ECCV),2018: 3-19

[13] Lin T Y, Dollar P, Girshick R, et al. Feature pyramid networks for object detection[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, 2017: 936-944.

[14] Ioffe S , Szegedy C . Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift[C]//International Conference on International Conference on Machine Learning. JMLR.org, 2015.

[15] HU J, SHEN L, SUN G. Squeeze-and-Excitation Networks[C]//Proceedings of the 2017 International Conference on Computer Vision and Pattern Recognition (CVPR), Piscataway: IEEE, 2018:7132-7141.

[16] Kingma D P, Ba J. Adam: A method for stochastic optimization[J/OL]. [2019-07-20]. https://arxiv.org/ abs/1412.6980