Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 131 (2018) 564–572

Procedia Computer Science

www.elsevier.com/locate/procedia

8th International Congress of Information and Communication Technology (ICICT-2018)

Vehicle type detection based on deep learning in traffic scene

LI Suhao[a]—, LIN Jinzhao[a], LI Guoquan[a], BAI Tong[a], WANG Huiqian[a], PANG Yu[a]

[a]*Chongqing University of Posts and Telecommunications,400065,Chongqing,China*

Abstract

Nowadays, vehicle type detection plays an important role in the traffic scene. Deep Learning algorithm has been widely used in the field of object detection. Applied the Faster RCNN framework, improved the RPN networks, which was an effective and representative of the Convolutional Neural Network of deep learning on object classification algorithm, and combined with the MIT and Caltech car dataset as well as some different types of vehicle pictures in the Internet, to detection and recognization the three types of vehicles which are commom in traffic scene. The experimental results show the effectiveness and high-efficiency of method of deep learning in the vehicle type detection.

Keywords: Deep learning, CNN, Faster RCNN, RPN, vehicle type detection;

1. Introduction

In recent years, with the rapid growth in the number of transportation vehicles, traffic regulation faces enormous challenges. Vehicle type testing is an important part of intelligent transportation systems. Its function is to detect the type of vehicle and provide information for road monitoring and traffic planning. Vehicle type detection, as a key technology to construct video surveillance of traffic conditions, has long been widely concerned by researchers at home and abroad. Target detection is an important branch of image processing and computer vision. Its research methods are mainly divided into background-based modeling and method based on apparent feature information [1]. This paper mainly focuses on the detection algorithm based on vehicle target apparent feature information, that is, detects and classifies the vehicle target in the actual traffic video or picture. Its main difficulty lies in the picture or video frame of the vehicle target will change due to lighting, angle of view and the interior of the vehicle and other

* Corresponding author. Tel.: +86-18983845720
*E-mail address:*420666399@qq.com

10.1016/j.procs.2018.04.281

changes as well as the characteristics of different types of vehicles. In view of the above difficulties, scholars at home and abroad have made many attempts using traditional machine learning methods, but the results are not satisfactory.

The traditional method of target detection is generally divided into three phases: First, select a few candidate regions on a given image, and then extract features from these regions and classify them by trained classifiers. Here we introduce each of these three stages separately.

Area selection is to locate the location of the target. As the target may appear anywhere in the image, and the size of the target, the aspect ratio is not sure, so the first sliding window strategy to traverse the entire image, and you need to set a different scale, different aspect ratio.

For feature extraction, it is not so easy to design a robust feature due to the morphological diversity of the target, the diversity of light variations, and the diversity of the background. However, the quality of extracted features has a direct impact on the accuracy of classification. (HOG [1], SIFT [2], etc. are commonly used in this stage)

The final classifier mainly uses SVM[3], Adaboost [4]and so on.

To summarize, there are two main problems in traditional target detection: one is that the region selection strategy based on sliding window is not specific, the time complexity is high, and the window is redundant; secondly, the hand-designed features are not very good for the diversity change Robustness. For now, target detection based on traditional machine learning methods has encountered bottlenecks and a more scientific approach is expected.

With the rapid development of deep learning theory and practice, the goal of machine learning based detection and classification has entered a new phase. Unlike traditional feature extraction algorithms that rely on prior knowledge, deep convolutional neural networks have some degree of invariance to geometric transformations, deformations and illumination and effectively overcome the variability of vehicle appearance and are adaptive to training data Build feature descriptions for greater flexibility and generalization. For target detection, the recognition accuracy is an indicator that researchers want to improve all the time. Speaking of recognition accuracy, we must mention the mean average pre-measurement (mAP), which measures the detection accuracy in target detection. To put it simply, in the detection of multiple categories, each category can draw a curve according to recall and precision, then AP is the area under the curve, and mAP is the average of multiple categories of AP, which is between 0 To 1, and the bigger the better.

The paper [5] mainly uses DNN to do Deep Neural Networks. The author regards target detection as a regression problem, returns to the position of the Bounding Box, finds the position of the target category and the target in a picture, it reaches 30% mAP on the VOC 2007 test set. The paper [6] improved Alex-net and tested the network on the test dataset using image scaling and sliding window methods. A method of image localization was proposed. Finally, a convolutional network was used to classify, locate and detect three the computer vision task, with 24.3% mAP in the ILSVRC2013 test suite.

In 2014, RBG (Ross B. Girshick) designed the R-CNN framework [7] by using region proposal and CNN instead of sliding window add manual design features used in traditional target detection, making a great breakthrough in target detection and opening up a new algorithm based on depth Learning target detection boom. It combines traditional machine learning with deep learning to propose classical algorithms such as Selective Search [8], which increases the mAP of the VOC 2007 test suite to 66%. Followed by the optimization of target detection networks like SPP-net [9], Fast RCNN [10], Faster RCNN[11], YOLO [12] and SSD [13], but Faster RCNN puts all the target detection implementation modules To a unified deep convolutional network framework, the detection accuracy compared to other deep learning objectives detection methods are higher. To this end, this paper uses a Faster R-CNN target detection algorithm in depth learning domain and combines with vehicle images of different types of vehicles on MIT [14] and Caltech [15] vehicle databases and networks to construct a vehicle type detection system. Experimental results show that the system identifies more accurately the main types of vehicles in three types of traffic scenarios: cars, minibus and SUVs.

In addition to this, complex periodic arrays of dipolarly coupled magnetic dots are of special interest because they can support the propagation of non-reciprocal spin waves, i.e. ($w(k) \neq w(-k)$), where w is the angular frequency and k is a wave vector, which could find application in the signal transmission and information processing as well as in the design of microwave isolators and circulators..

2. Vehicle Type Detection System Architecture

This paper designs the vehicle type examination system frame like Fig. 1. First, collect the appropriate training samples (different types of vehicle images) from the different types of vehicle images on the MIT and Caltech vehicle databases and the network, and mark the different types of vehicle targets in the sample. Then, the samples are input into the improved RPN network training until the network converges. Then the convolutional network parameters trained by the RPN network are input into the Fast-RCNN network for training until the network converges. Finally, the parameters of each network layer are input into the model for different Type Vehicle Target Recognition.

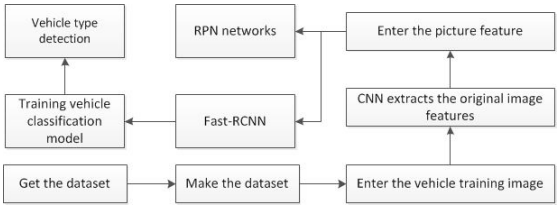


Fig. 1. Vehicle type detection system frame diagram

2.1. Faster RCNN algorithm

Faster RCNN proposed using RPN network to extract the candidate box, compared with the Selective Search method to extract fewer candidates, more efficient. Using Fast RCNN network for network parameter training and target detection.

2.1.1. Region Proposal Networks.

The RPN network takes a picture of arbitrary size as its input and outputs a batch of rectangular region candidate boxes, each region corresponding to a target's existing probability score and location information [17]. In this study, the vehicle type detection problem is mainly to improve the RPN layer in the Faster RCNN model (see Fig. 2). The traditional RPN network generates 256 characteristic maps by 2-step convolution, in which 3 × 3 convolution the nucleus produces 192 feature maps, with 5 × 5 convolution kernels producing 64 feature maps. The improved RPN network in this paper only produces 256 characteristic maps through a one-time 3 × 3 convolution kernel, which simplifies the network structure and reduces the computational complexity, as shown in Fig. 3. After the convolution operation, the last layer of the convolution feature graph is obtained, and the convolution kernel (sliding window) of 3 × 3 is used to convolve the feature graph on this feature graph. Because in this 3 × 3 region, each feature map to obtain a 1-dimensional vector, the last layer of the convolution layer 256 features, so this 3 × 3 region convolution can be a 256-dimensional feature vector, followed by classification layer cls layer and regression layer reg layer were used for classification and border (Bounding Box) regression. At the central point of the 3 × 3 convolution kernel sliding window, there are 9 region proposals corresponding to three kinds of scales (eg 128,256,512) and three aspect ratios (1:1,1:2,2:1) The mechanism of this mapping is called anchor, which generates nine candidate windows. Finally, according to region proposal score level, select the top score of 300 region proposal, as Fast RCNN input target detection.

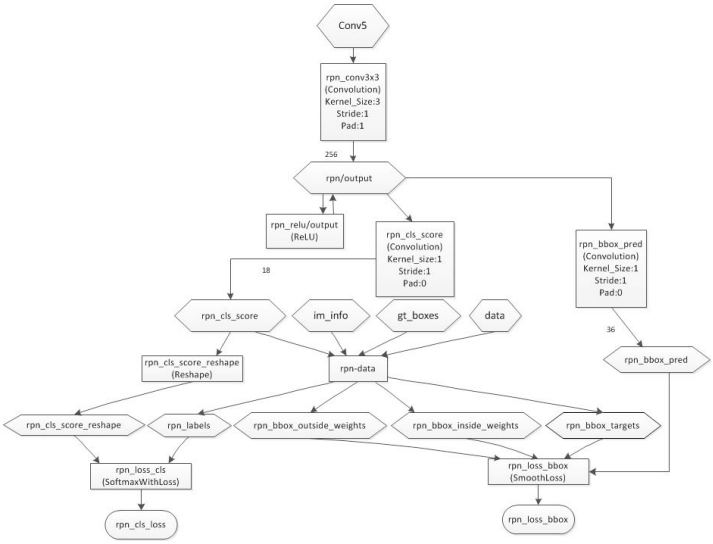


Fig. 2. Traditional Region Proposal Networks network structure diagram.

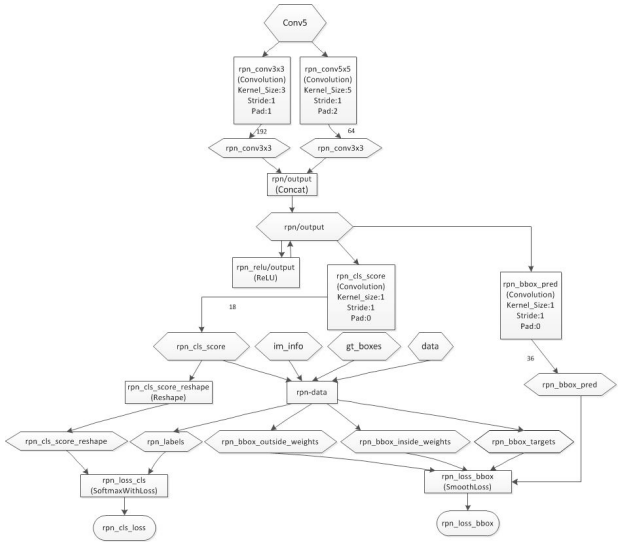


Fig. 3. Improved Region Proposal Networks network structure diagram.

2.1.2. Region Proposal Networks Loss Function.

To train RPNs, this article assigns each anchor a binary label (whether or not it is a type of vehicle target). If this anchor and a Ground Truth GT (Ground Truth) overlap ratio IoU (Intersection-over-Union) greater than 0.7, recorded as a positive sample; if it and any one of the calibration overlap ratio is less than 0.3, denoted as negative samples. With these definitions, this paper uses the objective function minimizing multitasking loss in Fast R-CNN.

For each anchor, first connect to a two-category softmax followed by two score outputs that represent the probability that it is an object and the probability that it is not an object, followed by a bounding box's regressor output representing the anchor 4 coordinate positions, so the overall RPN loss function can be defined as

$$L(\{p_i\},\{t_i\}) = \frac{1}{\mathrm{N_{cls}}}\sum_{\mathrm{i}} L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(\mathrm{t}_i, \mathrm{t}_i^*) \tag{1}$$

Where i represents the anchor, = 1 when the anchor is a positive sample, and = 0 if it is a negative sample. (2) is the logarithmic loss of both categories (target and non-target); the regression loss is calculated using equation (3), where R is defined in Fast RCNN the robust loss function is given by Eq (4).

$$L_{cls}(p_i, p_i^*) = log[p_i^* p_i + (1 - p_i^*)(1 - p_i)] \tag{2}$$

$$L_{reg}(t_i, t_i^*) = R(t_i - t_i^*) \tag{3}$$

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 & if\ |x| < 1 \\ |x| - 0.5 & otherwise \end{cases} \tag{4}$$

2.1.3. Region Proposal Networks And Fast RCNN.

In the last step, we have described how to propose a training network for generating regions without considering how region-based object detection CNN uses these proposed frames. For the purpose of detection, Fast R-CNN is used in this paper. How to train both of them in the same network structure to train a multi-task network model with shared convolution. We know that if the network model is trained separately for two different tasks, even if their structure and parameters are exactly the same, convolution kernels in their respective convolutional layers will also change in different directions, resulting in that the network weights cannot be shared. Therefore, we use the following methods of training (steps shown in Fig. 4)

Step 1: Initialize with ImageNet model, train one RPN network independently;

Step 2: Still using ImageNet model initialization, but using the proposal from the previous RPN network as input, training a Fast-RCNN network, thus, the parameters of each layer of the two networks are not shared at all;

Step 3: Initialize a new RPN network using the Fast-RCNN network parameters of Step 2 but set the learning rate for those convolutional layers shared by RPN and Fast-RCNN to 0, not update, just update RPN-specific network layer, re-training, this time, the two networks have shared all the common convolutional layer;

Step 4: Those network layers that are still fixedly shared are also joined by the network layer unique to Fast-RCNN to form a network and continue to train and fine-tune the network layer unique to Fast-RCNN. At this time, the network has realized our imagine the goal, the internal network to predict proposal and achieve the detection function.

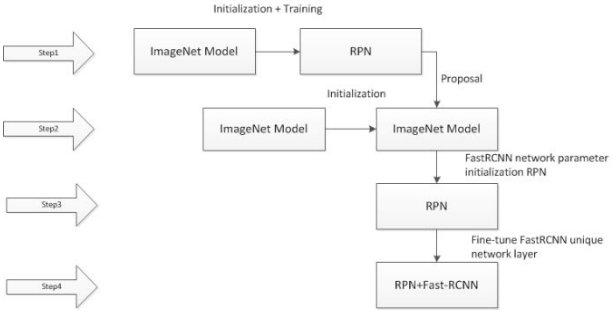


Fig. 4. RPN network and Fast R-CNN training flow chart.

3. Experiment

3.1 Dataset

The classic Faster RCNN achieves high accuracy for classifying and testing 1000 objects. This paper transforms the detection problem of the target into the classification detection problem (including different types of vehicle targets). The training original image set comes from the MIT and Caltech vehicle database and the pictures of different types of vehicles on the network, including the car, minibus, SUV 3, using VOC 2007 data set format and production tools.

The vehicle sample data set produced in this article: Image: 5042/JPG, Labels: 5042/XML, Bounding Boxes: Car 5038/ Rectangles, Minibus 987/Rectangles, Suv 1207/Rectangles. Among them, different types of vehicle data sets selected in Table 1.

Table 1. Frequency of Special Characters

Model	Total dataset	Training set	Test set
car	5038	3358	1680
minibus	987	658	329
Suv	1207	804	403

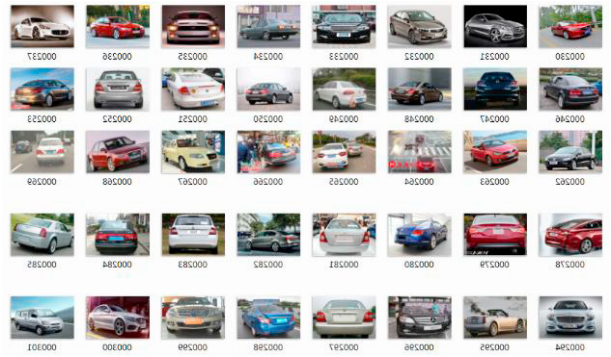


Fig. 5. Original image samples of different types of vehicles.

3.2. Experiment Environment

The main software and hardware configuration of the experiment is as follows: The computer operating system is Windows 7 64-bit, compile environment MATLAB R2014b; The video card is NVIDIA GeForce GTX 1080 Ti, 11G memory, Intel ® Core i5-4590 CPU @ 3.30GHz processor; Open CV 3.0 version; CUDA 8.0; caffe installation.

3.3. Experimental results and analysis

The In this paper, the ZF model (Zeiler and Fergus) [18] and VGG16 model [19] of Faster R-CNN algorithm are used to train and fine tune the parameters in the network model. Firstly, it is initialized with the ImageNet [20] model to train an RPN network independently. Then the Fast-RCNN network is initialized with the network weight of this RPN and the proposal of the previous RPN is used as the input of the Fast-RCNN to train the Fast R- CNN network; re-initialize the parameters of the RPN network using the parameters of the trained Fast R-CNN network and fine-tune the RPN network; and then fix the convolution layer of the Fast R-CNN and use the candidate box RPN to extract Fast R The CNN network is fine-tuned; afterwards, this process is iteratively training the RPN, Fast-RCNN, and eventually converging the network.

In this paper, the original image data of the training samples are from the MIT and Caltech vehicle databases and the pictures of different types of vehicles on the network. The original data samples are then processed according to the experimental requirements and then trained. At last, we choose all kinds of scenes to detect the images of all

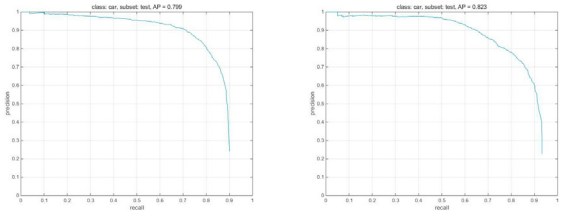
kinds of vehicles, and get the accuracy of the target detection of the three kinds of vehicles such as cars, minibus and SUVs with different networks and different numbers of samples.

For different depth learning network models (including ZF and VGG16) and different numbers of samples, the recall curves of the experimental results of three different types of vehicles are shown in Fig. 6. Table 2 shows the results of different types of vehicles in different sample numbers and different the average accuracy of the network. Accuracy in the curve precision, recall recall, and area average accuracy AP (Average Precision) formed by the curve are commonly used evaluation indicators in the field of target detection. Accuracy means that all "correct tests" account for all "detected" ratios. Recall rates represent all "correctly detected" proportions for all "should be detected," and the average accuracy is the area of the curve consisting of accuracy and recall. Three in an ideal state of 1, the closer the value of 1 to prove the better the test results. It can be seen from Fig. 6 that as the number of samples increases, the effect of vehicle type detection also increases. In the case of the same number of samples, network detection with more convolutional layers is better.

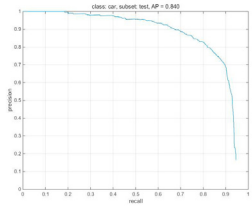
Fig. 7 is a car, minibus and SUV three different types of vehicles in different sample sizes and different models of the vehicle type detection effect diagram. According to the result analysis, if the number of samples is increased moderately, the recognition accuracy of all kinds of vehicles can be increased. However, if the sample size is increased, there will be over fitting, and the detection target will be more or the position of the frame will be inaccurate. Depth networks have more convolution layers and more features to be extracted. By increasing the convolution depth of the model (changing from ZF to VGG16), the number of extracted features can be increased. The experimental results show that the depth network model (VGG16) detects more different types of vehicle targets more and more accurately.

Table 2. Average accuracy of different types of vehicles under different sample sizes and different networks

Model	3052(ZF)	3052(VGG16)	5042(ZF)	5042(VGG16)
car	79.9%	82.3%	84.0%	84.4%
minibus	73.9%	74.8%	81.0%	83.8%
Suv	68.3%	70.1%	73.7%	78.3%

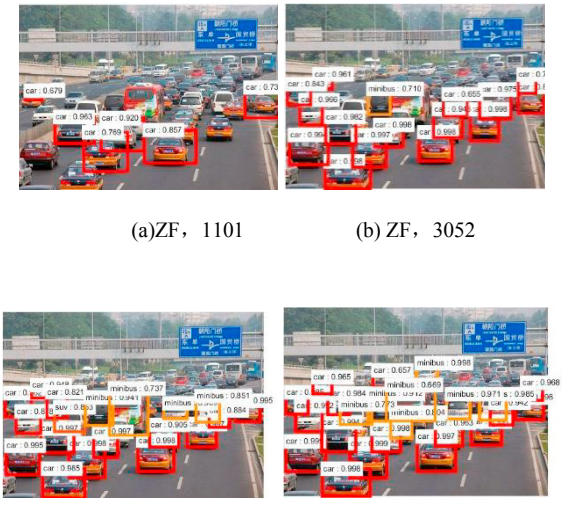


(a) ZF，3052(car) (b) VGG16，3052(car)



(C) ZF，5042(car)

Fig. 6. Precision-Recall curves for training with different models and numbers of samples.



(a)ZF，1101 (b) ZF，3052

(c) ZF，5042 (d) VGG16，5042

Fig. 7. Different models, the number of vehicles under the type of test results.

Table 3. Test under the same conditions compared with time

Model	Fast RCNN	FasterRCNN	This Paper
Time cost /ms	320	120	81

In this paper, the detection model under the same conditions with the Fast RCNN and Faster RCNN in the above data set for the detection time comparison, as shown in Table 3. It can be seen that under the same conditions, by improving the RPN network and reducing the complexity of the algorithm of the model, this method can obviously improve the vehicle type detection time without reducing the effect of Faster RCNN in detection and meeting the requirements of traffic scenarios Vehicle target detection.

4. Conclusions

In this paper, the application of convolution neural network in vehicle target type detection and recognition is studied. The Faster RCNN model is applied to the actual traffic environment. By designing an improved model and using self-built data including MIT and Caltech vehicle data sets and network pictures Focus on training ZF and VGG16 networks to detect vehicle types in traffic scenarios. The experimental results show that compared with the traditional machine learning methods, the model used in this paper has been improved both in average target detection accuracy and detection rate. The classification test result of this article is also suitable for vehicle type detection of three types of cars, minibus and suv in different scenarios and has achieved good results.

References

1. Taigman Y, Yang Ming, Ranzato M A, et al. Deepface: closing the gap to human-level performance in face veri-fication[C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. [S. 1.]: IEEE Press, 2014: 1701-1708
2. Ma Xiaoyu, Grimson W E L. Edge-based rich representa-tion for vehicle classification[C]//Proc of the 10th IEEE International Conference on Computer Vision. [S. 1.]: IEEE Press, 2005: 1185-1192
3. Kazemi F M, Samadi S, Poorreza H R, et al. Vehicle recognition using Curvelet transform and SVM[C]//Proc of the 4th International Conference on Information Technology. [S. 1.]: IEEE Press, 2007: 516-521
4. Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting[C]//Proc of the Computational Learning Theory. [S. 1.]: IEEE Press, 2014: 23-27
5. C. Szegedy, A. Toshev, D. Erhan. Deep Neural Networks for Object Detection. [C]// Advances in Neural Information Processing Systems. [S. 1.]: NIPS Press, 2013: 1673-1675.
6. Sermanet P, Eigen D, Zhang X, et al. OverFeat: integrated recognition, localization and detection using convolutional networks. [C]// Advances in Neural Information Processing Systems. [S. 1.]: ICLR Press, 2014: 1055-1061.
7. Girshick R, Donahue J, Darrell T, et al. Rich feature hier-archies for accurate object detection and semantic seg-mentation[C]//Proc of ImageNet Large-Scale Visual Recognition Challenge Workshop. [S. 1.]: ICCV Press, 2013:10-15.
8. Uijlings J. R, van de Sande K. E, Gevers T, et al. Selective search for object recognition [C]//Proc of International Journal of Computer Vision [S. 1.]: IJCV Press, 2013: 115-117.
9. He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[C]//Proc of the 14th European Conference on Computer Vision. [S. 1.]: ECCV Press, 2014: 865-973.
10. irshick R. Fast R-CNN [C]//Proc of IEEE International Conference on Computer Vision. [S. 1.]: ICCV Press, 2015: 10-15
11. Ren S，He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal net-works[C]//Proc of Conference on Neural Information Processing Systems. [S. 1.]: NIPS Press, 2015: 1-15.
12. Redmon J, Divvala S, Girshick R, et al. You only look once: unified real-time object detection. [C]// Proc of Conference on Computer Vision and Pattern Recognition. [S. 1.]: CVPR Press, 2016: 13.
13. Liu Wei，Anguelov D，Erhan D, et al. Berg. SSD: single shot multibox detector[C]//Proc of IEEE Conference on Com-puter Vision and Pattern Recognition. [S. 1.]: CVPR Press, 2016: 13-17
14. MIT. MIT Pedestrian Data [EB/OL]. (2000-01-01) [2014-01-01]. http://cbcl. mit.edu/software-datasets/PedestrianData.html
15. Caltech. Caltech Pedestrian Detection Benchmark [EB/OL]. (2009-01-01) [2014-01-01]. http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/.
16. Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. [S. 1.]: CVPR Press, 2015: 1109-1123.
17. Felzenszwalb P F, Girshick R B, McAllester D, et al. Object detection with discriminatively trained part-based models [C]//Proc of IEEE Transactions on Pattern Analysis and Machine Intelligence. [S. 1.]: TPAMI Press, 2010:201-205
18. Zeiler M D, Fergus R. Visualizing and understanding convolutional networks[C]. Springer International Publishing,2014,8689:818-833.
19. Simonyan K, Zisserman A. Very Deep Convolutional networks for large-scale image recognition[C]. International Conferenceon on Learning Representations, 2015.
20. Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge [C]//Proc of International Journal of Computer Vision. [S. 1.]: IJCV Press, 2015: 115-201.