ATLANTIS PRESS

Advances in Engineering Research, volume 154

International Conference on Mechanical, Electrical, Electronic Engineering & Science (MEEES 2018)

Traffic Sign Detection Based on Faster R-CNN in Scene Graph

Wei Zhao [a], Zhiqiang Wang [b] and Hongda Yang [c]

School of Vehicle & Transportation Engineering, Henan University of Science and Technology, Luoyang Henan 471003, China

[a]zhaowei@haust.edu.cn, [b]1204410227@qq.com, [c]492385877@qq.com

Keywords: Intelligent transportation, traffic sign detection, convolutional neural network, deep learning, transfer learning.

Abstract. The use of intelligent detection and identification software for traffic signs have been an indispensable part of the advancement of transportation systems and networked cars into an intelligent system. As neural networks become more effective in image recognition and classification testing, they are being applied through the use of traffic sign detection and recognition, gradually deepening the quality of research. Founded on the thoughts of deep learning and transfer learning, this paper uses the method of Faster Region-based Convolutional Neural Networks (Faster R-CNN) and the pre-trained neural network model, Alex net, to detect traffic signs in scene graph. Building off the positive reviews of the Alex Net neural network model in image classification recognition, a target recognition neural network model was trained in the framework of Faster R-CNN neural network using scene image data sets. According to the mark on the test data set, the results indicate that using the pre-trained neural network model can quickly build a traffic sign detection model.

1. Introduction

The intelligent detection and identification of traffic signs has become a basic function for intelligent network-linked automotive systems and is also a direction for the development of intelligent traffic. However, due to the complexity of driving environments, the detection and recognition of traffic signs are often affected by various factors including: passing vehicles, buildings, and roadside vegetation. The driver nor the vehicle vision system can accurately determine the meaning of traffic signs and as such cannot make the correct judgment on the next driving operation. For this reason, researchers have put forward methods that hope to identify traffic signs in real time.

The first step in identifying a traffic sign is the ability to detect the location of a traffic sign from the image, and the second is to identify the meaning of the sign. At present, there are two common methods for detecting traffic signs. One separates the traffic sign according to its color and shape characteristics. For example, the literature [1] uses a Gaussian model to segment the image, obtaining the position of the traffic sign, and extracts the histogram of oriented gradient (HOG) features, and then finally uses support vector machines (SVM) to classify the signs. The literature [2] uses a random sample consensus (RANSAC) matching traffic sign template to distinguish the traffic sign position through the detection of Harris corners. The other method uses neural networks to learn to accurately identify traffic signs. The literature [3] constructs a hyper-pixel map model based on the color and boundary features of the scene graph and through a priori position, thresholds segmenting the regions of interest (ROI) of the traffic sign and using the convolutions of Caffeine's neural network training to identify traffic signs. The literature [4-5] uses the deep convolutional neural networks to extract the image features and identify the traffic signs through the support vector machine (SVM).

With the continuous development of neural network algorithms, deep learning methods based on neural networks have become better at image recognition problems. Based on the ideas of deep learning and transfer learning, this paper uses the method of Faster Region-based Convolutional Neural Network (Faster R-CNN) algorithm [6] and the pre-trained deep neural network model, Alex Net [7], to carry detect traffic signs in the graphed image. After testing theories, a model capable of detecting traffic signs was obtained, and its validity verified by testing images of the test set.

2. Basic Theory of Deep Learning and Convolutional Neural Networks

2.1 Convolutional Neural Network Structure.

Convolutional neural networks are feed forward neural networks whose neuron structures are a convolution process. The typical structure of a convolutional neural network is shown in Figure 1, including the input layer, convolutional layer, pooled layer, fully connected layer, and output layer. [8] Deep convolutional neural networks alternately utilize multiple convolutional layers and pooled layers. According to the design, multiple fully-connected layers are typically added to enhance the generalization ability of the network.

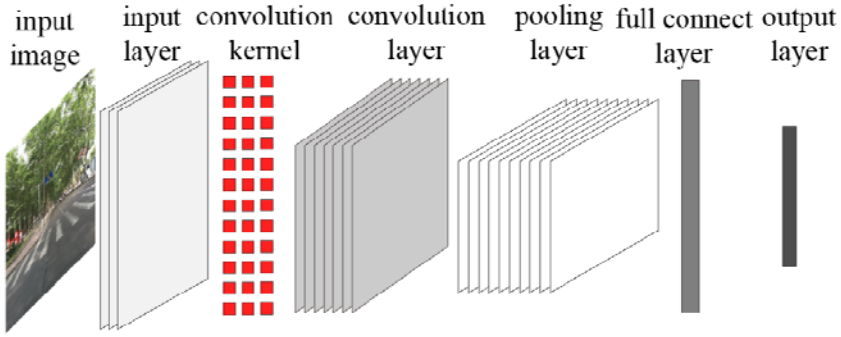


Fig 1. Simple convolutional neural network structure

The convolutional neural network training process consists of the forward-propagation process of information and an error back-propagation process. The information is forward-propagated after being calculated by each layer of neurons. In the training process, the error is back-propagated (BP) to determine the weight and offset of the neuron.

The forward propagation of information begins with the initial input image. The convolution calculation computes the weighted sum of data that corresponds to the convolution kernel. The mobile convolution kernel performs convolution calculations to obtain the feature map. The calculation of the convolutional layer can be simplified as:

$$a^k = f\left(\sum_{i,j=1}^{3} w_{ij}^k \cdot x_{ij} + b^k\right) \tag{1}$$

Where, w_{ij}^k is the weight value of the convolution kernel; x_{ij} the input pixel value corresponding to the weight of the convolution kernel; the offset corresponding to the first convolution kernel; b^k is the bias; $f(x)$ is the activation function. Convolutional neural network commonly used activation functions as a rectified linear unit (RLU), $f(x) = \max(0, x)$.

The pooling layer is used to reduce the dimensions of the feature map obtained by the convolution layer. The pooling operation takes the statistical value of the neighborhood; the calculation of the maximum value of the neighborhood is called the maximum pooling, and the calculation of the average value of the neighborhood is respectively called the average pooling. The fully connected layers take all the values of the feature map as input, resulting in a fixed number of outputs. It is a further abstraction of the information, and the resulting output can be classified because of the output layer. The output layer is the layer that obtains the classification results. It uses classifiers to categorize the input and output the results, such as Soft ax classifier, SVM classifier, etc. When there is a label, it is associated with a corresponding label to represent the classification.

The purpose of training the neural networks is to find the optimal weights and form a convolution kernel that can extract image features. In the training process, to generate results of the output layer as close to the actual value of the sample, using the gradient descent method the loss function is optimized. One commonly used loss function is the squared loss function:

$$\min\ E = \frac{1}{2} \cdot \sum_{i=1}^{K} (a_i - y_i)^2 \tag{2}$$

Here, E is loss function; a_i is the output result; y_i is truth result; K is the total number of output layer elements.

The gradient descent method is also commonly used to minimize the objective function. Since the direction of the negative gradient falls in the direction of where the function falls the fastest, the gradient is obtained by solving the first derivative of weight and the current value, and the learning rate η is introduced to make it follow the direction of the gradient drops. The calculation process is shown in formulas (3)-(5):

$$w \leftarrow w + \Delta w \qquad b \leftarrow b + \Delta b \tag{3}$$

$$\Delta w_i = -\eta \frac{\partial E}{\partial w_i} \qquad \Delta b = -\eta \frac{\partial E}{\partial b_i} \tag{4}$$

$$\begin{aligned} \frac{\partial E}{\partial w_i} &= \frac{\partial E}{\partial h}\frac{\partial h}{\partial w_i} = (h-y)\, f'(xw+b) = (h-y)\, x \\ \frac{\partial E}{\partial b_i} &= \frac{\partial E}{\partial h}\frac{\partial h}{\partial b_i} = (h-y)\, f'(xw+b) = h-y \end{aligned} \tag{5}$$

In equation (5), the activation function is a rectified linear unit, and the derivative is:

$$f'(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{6}$$

2.2 Deep Learning and Transfer Learning.

With the in-depth study of machine learning, the use of more neural network structure levels is to compensate for the incomplete feature extraction has become the main feature of machine learning. In this way, the image features can be extracted layer by layer and can be abstracted continuously until the content of the images can be identified by the classifier. Transfer learning uses the pre-trained neural network models to train personal data. The pre-training model has demonstrated success in image classification. According to the need to retain some of the weight parameters, one neural network model can be built for special application scenarios. The redesigned network structure will have the ability to quickly learn data sets and reducing the number of data sets. The pre-training neural network model can also utilize the Faster R-CNN algorithm, and further use Alex Net to abstract the scene image, by designing a special classifier to detect traffic signs in scene image.

3. Faster R-CNN Algorithm

The Faster R-CNN algorithm consists of two modules: Region Proposal Network (RPN) and Fast R-CNN (Fast Region-based Convolutional Neural Networks) modules [6]. The process flow of Faster R-CNN is demonstrated in Figure 3.

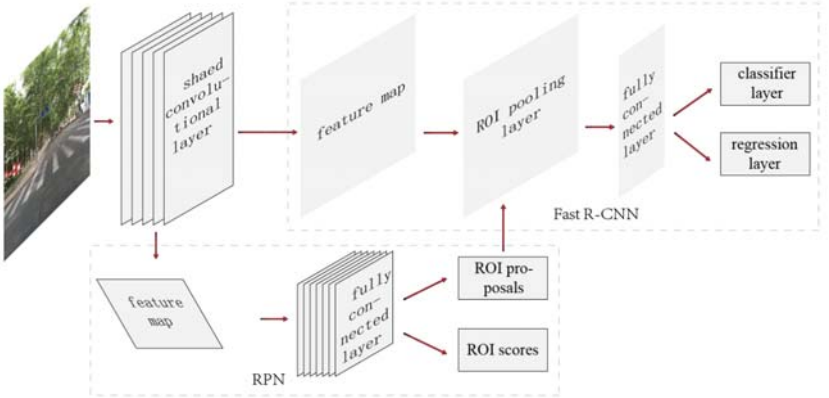


Fig 2. Process flow of faster R-CNN

The RPN module obtains the approximate area where the target is located. When the location of the target is unknown, the image is further divided into smaller pieces. By comparing the similarities between pieces, all the pieces can be merged, and the recommended target area can be obtained. The

RPN module first uses a sliding window to extract the features again on to an extracted feature map to obtain a 256-dimensional feature vector, which is then inputted it into two parallel full-connected layers, a classification and a regression layer. The RPN can understand whether the ROI is a traffic sign or not.

Fast R-CNN is a region-based convolutional neural network image recognition model. The original module first needs to divide the input picture into several candidate regions. It is common to use a selective search method based upon a graph theory and a minimum spanning tree to divide the image into 2000 candidate regions. These regions not only have redundancy elements, but also separate convolution calculations for each region. In the Faster R-CNN, this calculation process is improved, and the recommended area obtained by the RPN is used as a reference point to avoid excessively dividing the candidate area and having redundant regions. At the same time, sharing the convolutional layer has greatly improved the calculation speed of the Fast R-CNN module. Fast R-CNN outputs results through two peer output layers. One output layer uses the Soft ax classifier to estimate probability; the other uses bordering box regression to fine-tune the border position.

During training, RPN and Fast R-CNN are trained separately. The first step being to train the RPN. This process uses the pre-trained Alex Net model to initialize the shared convolutional layer parameters. The RPN unique layer is initialized with random data that satisfies the Gaussian distribution. Optimizing the joint loss function, fine-tuning the global parameters of the network, and training the RPN network. The second step uses the suggestion box, generated by RPN after training in the first step, to train a separate detection network by Fast R-CNN. The network parameter initialization is the same as the first step. At this point, the two networks have yet to share the convolutional layer. The third step utilizes the second-step detection network to initialize the shared convolutional layer parameters, retrain the RPN, and fine-tune the RPN-specific layer again. The fourth step is to maintain the fixed shared convolutional layer and fine tune the unique layer of Fast R-CNN. Through the repeated training, the two modules form a unified network, known as Faster Convolutional Neural Network (Faster R-CNN).

4. Experiments and Results Analysis

4.1 Data Sets.

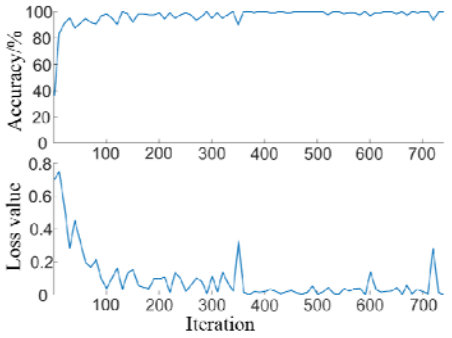
The data set used in this experiment include the foreign traffic sign detection data set [9] (GTSDB) and the scene image data set of domestic urban traffic signs. The foreign traffic sign detection data set consists of a total of 900 scene images with a pixel size of 1360×800 with the given the location data of traffic signs in the scene images. The data set was divided into 600 training images and 300 test images, with a total collected amount of 125 pictures of different scenes and five common traffic signs in a typical city. To increase the number of dataset images, the images were extended by rotating the dataset by 5 degrees and 10 degrees, and then the images were set to two types of pixel sizes of 1920×1080 and 800×600, and 750 images were obtained. The evaluation of training effects uses the leave-out method [10], which divides the data set into two mutually exclusive sets, one for training and one for testing. Due to the small number of datasets in the domestic cities, the training set used all the pictures, and the test set consisted of 225 randomly selected images.

The set of images for domestic cities are manually tagged with the Training Image Labeler application in MATLAB. Drawing a rectangular border around the scene image helps to enclose the traffic sign. The application records the rectangular frame's coordinates and the length and width.

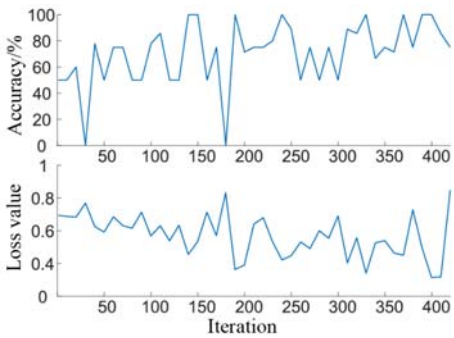
4.2 Optimization and Testing of Training Parameters.

For the training of foreign traffic sign detection data sets, the training period is set to 10, the batch size commonly used the value 128, and the learning rate is a fixed value of 2.48×10-5. At the same time each time the parameters are updated, the effect of the previous parameter value should be reduced, i.e. β takes 0.82, each iteration is trained once using 128 samples, and one cycle uses all the samples to retrain.

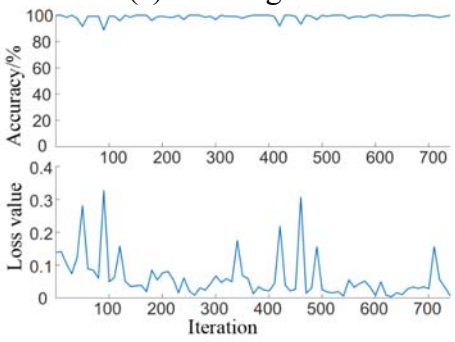
Figure 3 illustrates the changes in the accuracy and loss values of each batch of samples during the training phase of the foreign traffic sign detection data set. In the figure, a set of relevant data is obtained for each iteration 10 times.



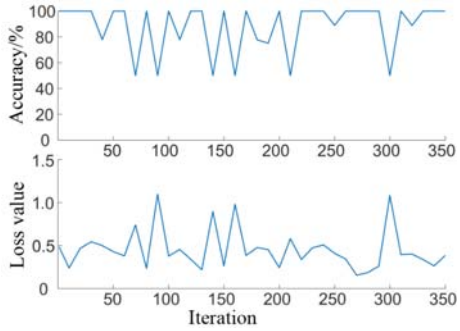
(a) Training RPN alone



(b) Training Fast R-CNN alone



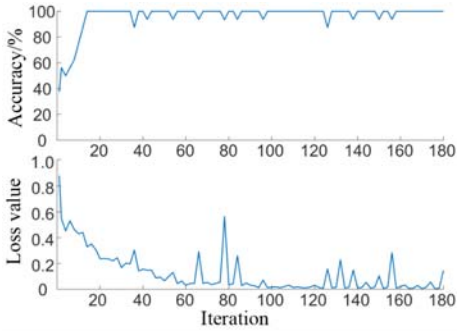
(c) Retraining the RPN



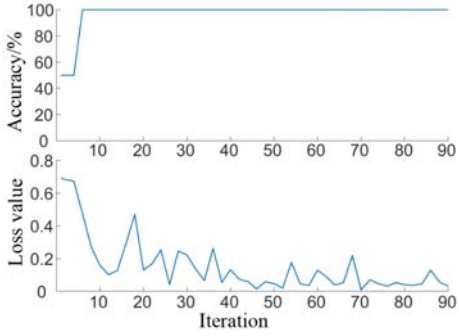
(d) Retraining Fast R-CNN again

Fig 3. Training of foreign data sets

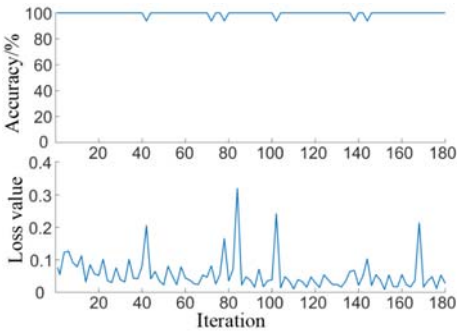
The same network model training was used for domestic urban image sets. The training period was set to 30, the batch size was 256, the fixed learning rate was 7.14×10-5, and the β was 0.6. Figure 4 demonstrates the changes in accuracy and loss values of each batch of samples in the four phases of the training of domestic traffic sign data sets. A set of relevant data is obtained for each iteration.



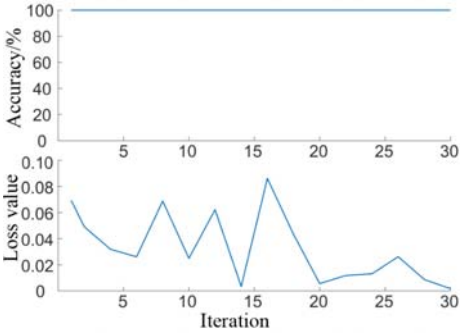
(a) Training RPN alone



(b) Training Fast R-CNN alone



(c) Retraining the RPN



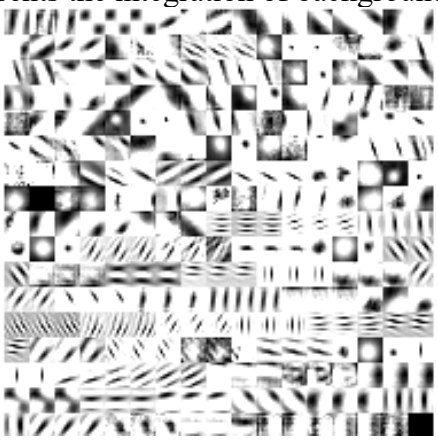
(d) Retraining Fast R-CNN again

Fig 4. Training of domestic data sets

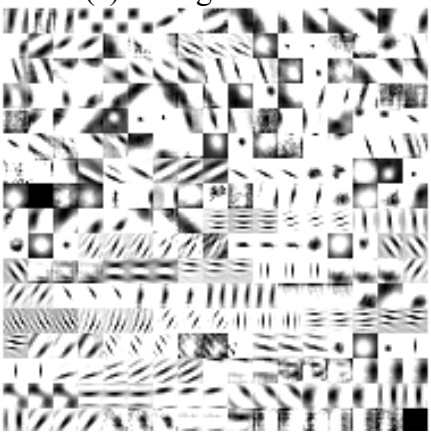
4.3 Analysis of Experimental Results.

Alex Net is a pre-trained deep convolutional neural network model capable of classifying and recognizing 1000 objects, therefore, the information extracted by the model is very close to the target during the initial training, so the loss value is minimal. During continuous training, the parameters in the network gradually changed, and the ability to detect traffic signs improved quickly.

Figure 5 (a) and (b) are the weight features of the first convolutional layer of the shared convolutional network after the training of foreign datasets and domestic datasets, respectively. The graph mainly displays the effect of the neural network on the learning of underlying features such as colors and edges. Since the underlying features are simply an overview of the image, the weights do not change much. The lines in the weight feature map represent the extraction of textures, and different lengths and directions represent the size and direction of the extracted texture features. The slurred weight feature map represents the integration of background features.



(a) Foreign data sets



(b) Domestic data sets

Fig 5. Attribute weights of the first convolutional layer of a shared convolutional network

After the training is completed, the data of the test set is tested using the trained model. The test results will give all the borders with a regional score greater than 0.5. Generally, the border with a higher score will be able to detect the target more accurately. The evaluation index of the model detection is the area overlap ratio Iowa, representing the ratio of the intersection area and the union area of the real target area and the detection area in the image. The results are correct if the value of Iowa is greater than 0.5. Table 1 shows the test results of the two test models.

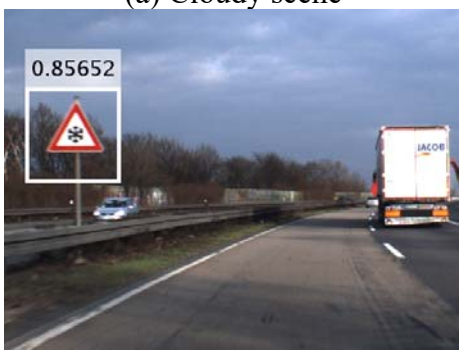
Tab 1. Average accuracy of two test sets

data sset	Average accuracy
1	52%
2	65%

Figure 6 and Figure 7 are the detection results of foreign and domestic test sets, respectively. The white border indicates the detection result of the detection model. The digital display box image contains the probability of the traffic sign.

(a) Cloudy scene

(b) Sunny day scene



(c) Highway scene

(d) Ordinary road scene

Fig 6. Test results of foreign traffic sign test sets

(a) Open road scenario

(b) Greenway scene

(c) Ordinary road scenes

(d) Greenway scene

Fig 7. Test results of the domestic urban traffic sign test set

5. Conclusion

Based on the idea of migration learning, Faster R-CNN target detection algorithm uses pre-trained neural network models to extract image features and is suitable for training a target detection model system using a small amount of data. The RPN module proposed by the algorithm greatly optimizes the selection process of the target area suggestion box. The algorithm also improves the speed of detection by improving Fast R-CNN, avoiding the repeated extraction scene image feature. In this

paper, training and testing of traffic sign scene graphs at home and abroad demonstrate that the use of Faster R-CNN training target detection model demonstrates a significant improvement, and its adaptability to different scenes is better. The next step is to collect more scene images, improve the model's detection accuracy, and add traffic class labels so that the target detection model can recognize the traffic sign symbol semantics.

References

[1]. Chang Failing, Huang Cui, Liu Chengdu, et al. Traffic sign detection based on Gaussian color model and SVM [J]. Chinese Journal of Scientific Instrument, 2014, 35(1): 43-49.

[2]. Ge Xia, Yu Feng in, Chen Ying. Ransack Algorithm with Harris angular point for detecting traffic signs [J]. Transducer and Microsystem Technologies, 2017, 36(3): 124-127.

[3]. Liu Hansen, Zhao Xiangmo, Li Qian, et al. Traffic sign recognition method based on graphical model and convolutional neural network [J]. Journal of Traffic and Transportation Engineering, 2016(3): 124-127.

[4]. Wang Xiaoping, Huang Janie, Liu Wensum. Traffic sign recognition based on optimized convolutional neural network architecture [J]. Jamal of Computer Applications, 2017, 37(2): 530-534.

[5]. Xin Jin, Cain Fixing, Deng Haiti, et al. Traffic Sign Classification Based on Deep Learning of Image Invariant Feature [J]. Journal of Computer-Aided Design & Computer Graphics, 2017, 29(4): 632-640.

[6]. Ren S, He K, Airsick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks [C]// International Conference on Neural Information Processing Systems. MIT Press, 2015:91-99.

[7]. Krizhevsky A, Sutskever I, Hinton G E. Image Net classification with deep convolutional neural networks [C]// International Conference on Neural Information Processing Systems. Curran Associates Inc. 2012:1097-1105.

[8]. Zhou Freidan, JIN Limping, DONG Jun. Review of Convolutional Neural Network [J]. Chinese Journal of Computers, 2016, 36(9): 2508-2515.

[9]. Hoban S, Stall Kamp J, Salman J, Et al. Detection of traffic signs in real-world images: The German traffic sign detection benchmark [C]// International Joint Conference on Neural Networks. IEEE, 2013:1-8.

[10]. Zhou Zhuhai. Machine learning [M]. Beijing: Tsinghua University Press, 2016: 23-47.