Vehicle Detection Method of Automatic Driving based on Deep Learning

Zhang Meihong

Abstract—**To improve the vehicle recognition technology in automatic driving, an improved faster region convolutional neural network (Faster RCNN) method is proposed. Firstly, a basic network of vehicle recognition based on Resnet network is designed. Secondly, multi-scale feature fusion strategy is proposed to meet the need for small target detection. Thirdly, for the regional proposal network (RPN) in Faster RCNN, the anchor box is designed by fitting the characteristics of vehicles, which obtains a better recommended area. This design further reduces the false detection and missing detection rate. Vehicle images from California Institute of Technology dataset are used for experiment. The experimental results show that the proposed algorithm can effectively reduce the training loss and RMSE. And the accuracy of vehicle recognition is improved.**

Index Terms—**Faster RCNN; neural network; automobile control; image identification; feature fusion**

I. Introduction

With the advancement of machine vision and artificial intelligence [1-3], vehicle detection and recognition technology has gradually become a research hotspot for intelligent transportation system [4-8]. Vision based vehicle detection has attracted extensive attention. Vehicle detection methods are mainly divided into the following four categories: The first category is the method based on vehicle features [9]. This kind of algorithm uses vehicle symmetry, color, shadow, corner, and other significant features to detect vehicles. However, it is vulnerable to illumination intensity, image noise and driving environment. The detection effect is unreliable. The second category is the method based on template matching [10], which matches the video stream with many vehicle templates. The area with high correlation is the vehicle area. These methods have higher accuracy than feature-based methods. However, they have poor real-time and scalability. They are also affected by illumination. The third category is the method based on optical flow field [11], which can extract moving objects by collecting images of moving process. Compared with the first two methods, the accuracy has been improved. However, the calculation amount is large. And the performance of real-time is poor. The fourth category is machine learning method [12,13], which trains vehicle samples to obtain model parameters. With the development of parallel GPU, the real-time and scalability of the algorithm are enhanced. Machine learning has gradually become the mainstream tool in the field of vehicle detection.

Manuscript received September 11, 2021; revised November 16, 2022.

Zhang Meihong is an associate professor of the Henan Industry and Trade Vocational College, Zhengzhou, Henan, China, 450000 (corresponding author, e-mail: 9224656@qq.com).

The vehicle detection algorithms based on machine learning are divided into shallow learning and deep learning. Detection algorithms based on shallow learning manually extract image features. Literature [14] proposes a vehicle detection method based on one class support vector machine. The moving vehicles are segmented from the background utilizing the Gaussian model. Then the geometric features are collected. The extracted features are used by support vector machine to classify the vehicles. However, these traditional methods rely on manual feature extraction, which may be seriously disturbed by external factors. The performance may be further reduced in complex scenes. Different from shallow learning methods, the methods based on deep learning do not manually extract feature [15-17]. Common detection algorithms based on deep learning include single-stage and two-stage model. However, the single-stage model may lose a certain amount of detection accuracy in quick detection [18]. For the two-stage model, continuous detection and classification are required. The region convolutional neural network (RCNN) model is proposed by Girshick [19]. RCNN is an important reference framework for the two-stage model of target detection. The joint search and convolutional neural network are used. Multiple candidate regions are extracted from the original input image. Finally, the image target detection is realized through the classification and regression network. RCNN uses deep convolutional network to classify the target regions, which can achieve high detection accuracy. But it has the shortcoming of high training time and slow target detection. Fast RCNN has been proposed subsequently. Fast RCNN utilizes the strategy of sharing convolutional computation. It can effectively reduce the running time of target detection. However, region recommendation algorithm of Fast RCNN is time consuming to provide the hypothesis of the target location. The Faster RCNN algorithm is proposed by Girshick [20,21]. Faster RCNN uses regional proposal network (RPN) to build target candidate areas. Such design reduces the feature number of the input image. It significantly improves the detection speed and accuracy.

However, there are still many unsolved problems in the target detection. Feature extraction is easy to be disturbed by background noise. The deep convolution operation is easy to lose spatial detail. The multi-scale attributes of the target are not considered, and so on. To solve the above problems, the Resnet module and multi-scale feature fusion are introduced. And the scale of anchor box is improved. An improved Faster RCNN method is proposed to realize vehicle recognition.

II. Faster RCNN Algorithm

The Faster-RCNN network is one of the important frameworks in the target detection.

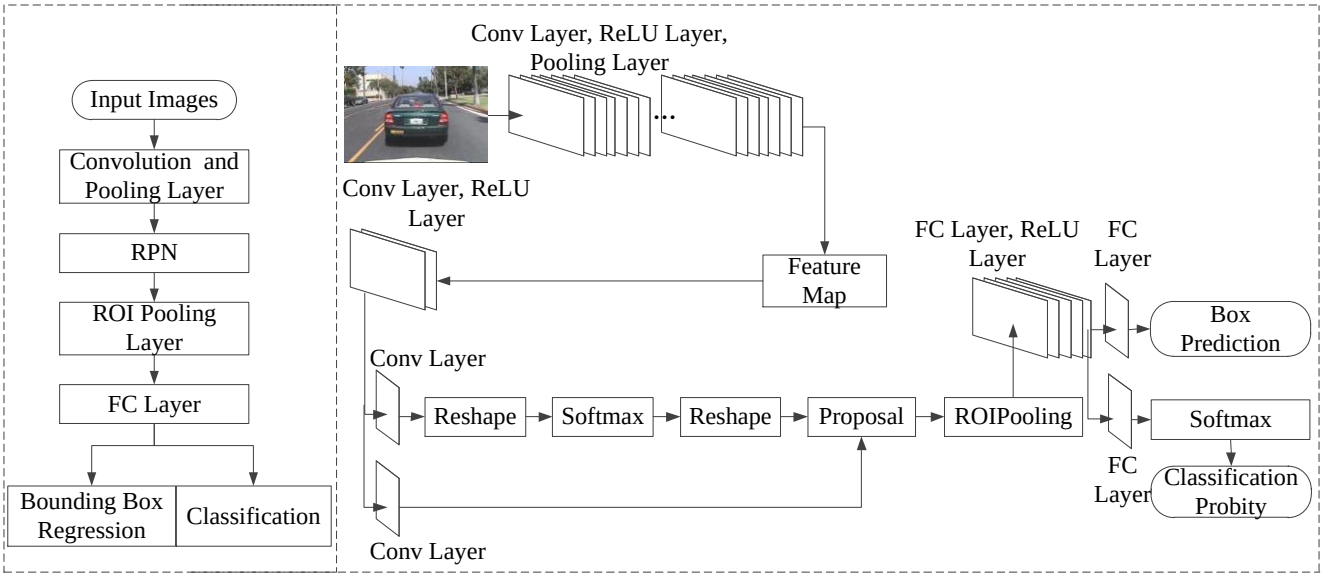


Fig. 1. Faster RCNN detection flow

The detection flow of Faster-RCNN is shown in Figure 1, which mainly includes the following four parts.

(1) Backbone network: The image feature is extracted. Then the feature is provided for RPN network.

(2) RPN network: The feature is received from the backbone network. The candidate area of the detection target is constructed.

(3) Pooling layer of the region of interest (ROI): The candidate regions are mapped to the feature. The feature is pooled into a unified scale for subsequent detection and classification.

(4) Classification and regression: The target category is classified in the ROI area. The boundary box is regressively used to modify the target box. As a result, the accurate position offset is obtained.

RPN network mainly learns two parts of information, namely category information and location information. The category information can be calculated according to the characteristic map. The result is judged as the foreground class if containing the target. Otherwise, it is judged as the background class. The location information is not directly trained by RPN. RPN learns the offset of the candidate area. The sliding window method is used for traditional candidate box selection. The sliding window method costs much time for each image. The parameters of each step cannot be shared. RPN integrates the object detection process into a neural network, which improves the detection efficiency. The specific detection steps are shown in Figure 2. Each point on the RPN characteristic generates k anchors. The input characteristic map is operated with $n \times n$ convolution. The output dimension remains unchanged. Then the output is divided into two lines. One is to distinguish whether k anchors are foreground or background, i.e., 2 k scores. The other line is to calculate the anchor location.

To train the RPN, a binary category label should be assigned to each anchor point. After constructing the positive and negative anchor points, the training process can be transformed to minimize a multi-task loss function. For the case of one anchor point, the classification loss L_{cls} and the regression loss L_{reg} can be defined as follows.

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

$$\begin{aligned} L_{reg}(t_i, t_i^*) &= f(t_i - t_i^*) \\ &= f(x_i - x_i^*) + f(y_i - y_i^*) + \\ &\quad f(w_i - w_i^*) + f(h_i - h_i^*) \end{aligned} \tag{2}$$

$$f(x) = \begin{cases} 0.5x^2, & |x| < 1 \\ |x| - 0.5, & else \end{cases} \tag{3}$$

where p_i is the probability that anchor point i is predicted as a target. p_i^* is set to be 1 for the positive anchor point. p_i^* is set to be 0 for the negative anchor point. $t_i = (t_{i,x}, t_{i,y}, t_{i,w}, t_{i,h})$ represents 4 parameterized coordinates of the predicted border. $t_i^* = (t_{i,x}^*, t_{i,y}^*, t_{i,w}^*, t_{i,h}^*)$ represents the true border coordinates of a positive anchor point.

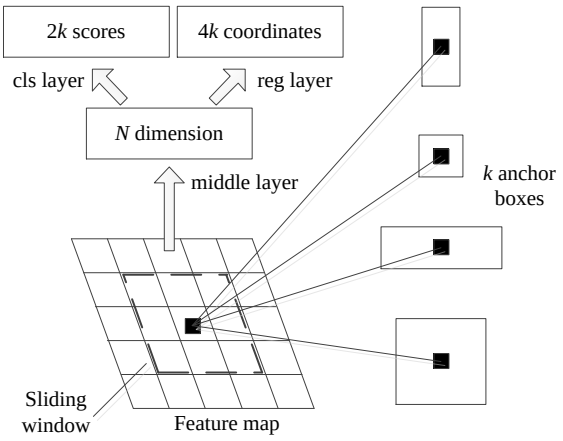


Fig.2. RPN structure

III. The Improved Faster RCNN for Vehicle Identification

A. Network structure design

Faster RCNN uses VGG as its basic neural network. The VGG network has a relatively simple structure. But VGG can not solve the gradient dispersion caused by deep network. VGG is hard to improve the information expression for

high-dimension image. To solve this problem, the network structure of Faster RCNN is combined with Resnet.

The Faster RCNN based on Resnet is shown in Figure 3. Resnet is widely used in detection, segmentation, recognition, and other fields. The common Resnet structures are Resnet50 and Resnet101. The value 50 and 101 represent the number of network layers. If deep layers are used, the high precision is obtained. And the slow speed is carryout. Considering the overall speed and precision, Resnet50 is selected as the basic convolutional neural network model for Faster RCNN.

The network structure of Resnet50 is divided into five parts: conv1, conv2_x, conv3_x, conv4_x and conv5_x. conv1 is the convolution layer. conv2_x, conv3_x, conv4_x and conv5_x have 3, 4, 6 and 3 building blocks respectively. Each building block contains 3 convolution layers.

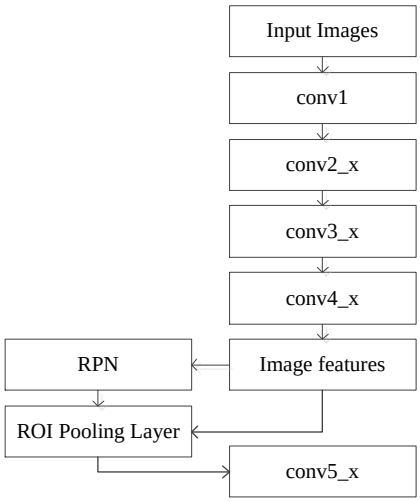


Fig. 3. Faster RCNN network structure based on Resnet50

The transfer learning method is used to train the deep convolutional neural network. The pre-trained parameters from another dataset are used to initialize the network. Then the vehicle image set is used to tune the parameters.

B. Adaptive adjustment for regional suggestion

For the regional suggestion network of RCNN, many candidate regions are generated to detect the target. If these candidate regions are used for training, the training time would greatly increase. Therefore, NMS algorithm is used to screen the candidate boxes. *M* candidate boxes with high confidence are selected for the final training. The loss function is divided into two parts: classification loss and regression loss. The loss function can be expressed as follows.

$$L = \frac{1}{N_{cls}} \sum_j L_{cls}(p_j, p_j^*) + \alpha \frac{1}{N_{cls}} \sum_j p_j^* L_{reg}(y_j, y_j^*) \quad (4)$$

$$p_j^* = \begin{cases} 0, & \text{negative label} \\ 1, & \text{positive label} \end{cases} \quad (5)$$

where j is the jth anchor in each training batch. p_j is the probability that the jth anchor is predicted as the prospect. p_j^* is the actual label of the sample. $p_j^*=0$ is the negative label. $p_j^*=1$ is the positive label. y_j is the coordinate of the candidate area for network prediction. y_j^* is the true coordinate for the corresponding positive sample. $L_{cls}(p_j, p_j^*)$ and $L_{reg}(y_j, y_j^*)$ of equation (4) are the classification loss and regression loss separately. α is the weight factor.

The classification loss function $L_{cls}(p_j, p_j^*)$ is defined as follows.

$$L_{cls}(p_j, p_j^*) = -\log_2[p_j^* p_j + (1-p_j^*)(1-p_j)] \quad (6)$$

The regression loss function $L_{reg}(y_j, y_j^*)$ is defined as follows.

$$L_{reg}(y_j, y_j^*) = R(y_j - y_j^*) \quad (7)$$

where $R()$ represents the robust loss function, showed in equation (8).

$$R(y) = \begin{cases} 0.5y^2 & |y| < 1 \\ |y| - 0.5 & else \end{cases} \quad (8)$$

The traditional Faster-RCNN uses NMS algorithm to screen *M* candidate frames in the region suggestion network layer. Large candidate frames cost a lot of training time. Appropriately reducing the number of candidate frames can improve the detection speed. Therefore, the ARP layer is introduced into the region suggestion network layer. This method can reduce the training time. The number of ARP can be expressed as follows.

$$N_{ARP} = \begin{cases} B(1+\delta_1), & L_i \geq \beta L_{i-1} \\ B, & \beta L_{i-1} > L_i > \gamma L_{i-1} \\ B(1-\delta_2), & L_i \leq \gamma L_{i-1} \end{cases} \quad (9)$$

where i is the serial number of each Q training. B is the number of the candidate boxes during $Q \times i$ time instance to $(Q+1) \times i$ time instance. L_i is the average loss. δ_1 and δ_2 are the penalty factor.

C. Multi-scale feature fusion

After kernel convolution of each layer, the traditional RPN obtains a series of feature maps, called feature layers. A total number of four feature layers are generated, named L_1, L_2, L_3 and L_4 respectively. The width, height and thickness of each feature layer are different. L_4 is the final layer of candidate extraction. Traditional RPN only takes the deepest feature layer as the candidate layer of frame extraction. Such method leads the loss of image and target information. The improved RPN structure is shown in Figure 4.

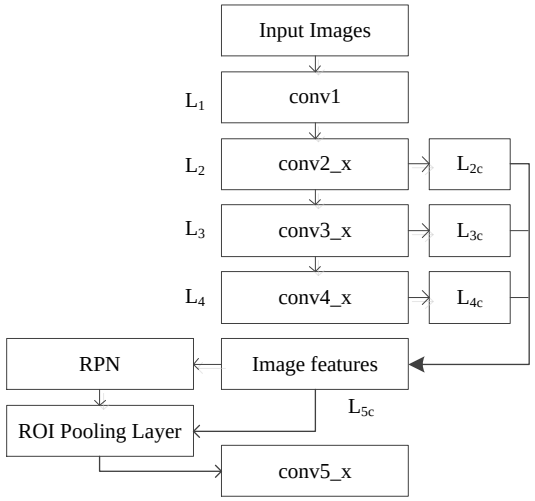


Fig. 4. Network structure of multi-scale feature fusion

Firstly, the L_2 and L_3 feature layers are used for further feature extraction. In addition, the L_2 regularization is used to obtain two new feature layers L_{2c} and L_{3c}. Then the feature layer L_{4c} is obtained by the convolution kernel. A new feature layer L_{5c} is obtained by information fusion. L_{4c}, L_{2c} and L_{3c} are fused as the final candidate layer of frame extraction. The thickness of L_5 is the same as L_{5c}. This method can be used without increasing the thickness of the candidate extraction layer. Low feature layer contains detail image information. High feature layer contains abstract feature information. Using the detail and abstract information, the reliability of target recognition and location is improved.

D. L2 normalization

Generally, the channel number and map size are different for each layer of ResNet50. The feature map scale is small in the deep convolution layer. Therefore, direct fusion of ROI pooling feature may not be effective. The scale difference is large among different feature maps. The features with large scale may weak the algorithm robustness. To solve this problem, L2 normalization is performed on the characteristic map after each ROI pooling. After normalization, the scaling operation is applied to each feature independently. For a d-dimensional input $\boldsymbol{x} = (x_1, x_2, \ldots, x_d)$, L2 norm is applied to standardize it, as shown in equation (10).

$$\hat{x} = \frac{x}{\| x \|_2} \tag{10}$$

The L2 norm of x is defined as follows.

$$\| x \|_2 = (\sum_{j=1}^{d} | x |^2)^{0.5} \tag{11}$$

where x is the original pixel vector. $\hat{x}$ is the normalized pixel vector. And d is the number of channels of each ROI pooling feature. The scaling factor λ_i is introduced. The normalized value is scaled by equation (12).

$$y_i = \lambda_i \hat{x}_i \tag{12}$$

In the training phase, the scaling factor λ and input data x are calculated by back propagation.

$$\frac{\partial l}{\partial \hat{x}} = \frac{\partial l}{\partial y} \cdot \lambda \frac{\partial l}{\partial x} = \frac{\partial l}{\partial \hat{x}} (\frac{I}{\| x \|_2} - \frac{x x^{\mathrm{T}}}{\| x \|_2^3}) \frac{\partial l}{\partial \lambda_i} = \sum_{y_i} \frac{\partial l}{\partial y_i} \hat{x}_i \tag{13}$$

E. Anchor box improvement

RPN uses 3×3 sizes sliding window, which convolutes each point on the image features of shared layer. The center point of the sliding window is called anchor. Each anchor is equipped with a group of rectangular anchor boxes. The anchor box is mapped to the image. The intersection and union ratio are calculated between the anchor box and the ground truth. Then the image is determined whether containing the target. After convolution, the anchor box is passed through two fully connected layers: the classification and regression layer. The output of the former layer is the result of the foreground or background. The output of the latter layer is 4k regression offsets. The objective function of RPN can be expressed as follows.

$$D(\{p_j\},\{q_j\}) = \frac{1}{M_c} \sum_j D_c(p_j, p_j^{\#}) + \frac{\delta}{M_r} p_j^{\#} \sum_j D_r(q_j, q_j^{\#}) \tag{14}$$

where j represents the anchor serial number in a batch training. p_j represents the probability that the anchor point j is a vehicle. $p_j^{\#}$ indicates whether the label is a positive sample 1 or a negative sample 0. q_j represents four coordinate parameters of the prediction frame. $q_j^{\#}$ represents the coordinate parameter of the real position. D_c indicates the classification loss for foreground or background. D_r represents the regression loss. δ, M_c and M_r are adjusting parameters.

For classification loss, the loss function is defined as follows.

$$D_c(p_j, p_j^{\#}) = -\log_2 \left(p_j^{\#} p_j + (1 - p_j^{\#})(1 - p_j) \right) \tag{15}$$

For regression loss, the loss function is defined as follows.

$$D_r(p_j, p_j^{\#}) = L(q_j - q_j^{\#}) \tag{16}$$

where $L()$ represents the robust loss function, which can be expressed as follows.

$$L(y) = \begin{cases} 0.5y^2 & | y | < 1 \\ | y | - 0.5 & \text{else} \end{cases} \tag{17}$$

In the RPN of traditional Faster RCNN, the anchor box is composed of nine types: three sizes of 8, 16, 32 pixels and three aspect ratios of 0.5, 1.0, 2.0. In the task of vehicle identification, large anchor box size may lead to image information loss. Small anchor box size may lead to incomplete image information. The recommended area generated by RPN can contain more vehicle information. The detection rate of small target vehicle is improved. And the prediction accuracy of target location is increased.

F. Model training flow

The model training steps are as follows:

①Prepare training and testing sets. Make samples into standard target detection data set.

②The pre-trained Resnet50 network is used to initialize the RPN network parameters. The back propagation and random gradient descent are used to tune the RPN network.

③Initialize the Faster-R-CNN network parameters with the pre-trained Resnet50 network. Extract the candidate area with the RPN network in step ②. Train the target detection network.

④ Reinitialize the RPN network parameters with the trained target detection network. Use the adaptive adjustment for regional suggestion. Then perform multi-scale feature fusion. Fine tune the RPN network separately.

⑤ Use the RPN network tuned in step ④ to extract candidate regions. And tune the target detection network with L2 normalization and the improved anchor box.

⑥Repeat steps ④ to ⑤ until the network converges or reaches the maximum training time.

IV. Experiment and Result Analysis

A. Experiment dataset

295 images from California Institute of Technology are selected to test the proposed algorithms. The dataset is composed of two parts. One part is the pictures of cars on urban roads in Southern California. The other part is the pictures of cars in parking lots or on roadside parking space. The resolution of the images is 640×480. 60% images are

selected as the training data. And the rest 40% images are chosen as the testing data. To increase the number of training samples, the images are randomly flipped. As a result, the number of training samples is 708.

TABLE I
THE NUMBER OF EXPERIMENT DATASET

Num of original data	Num of training data with flipping	Num of testing data
295	708	118

B. *Evaluating indicator*

The loss function curve, precision recall (P-R) curve and average accuracy (AP) are used to analyze the algorithm performance. The loss function curve changes with the number of iterations, reflecting the convergence speed and similarity between predicted and real value. The P-R curve can describe the relationship between accuracy and recall. It can directly reflect the detection performance. The AP represents the percentage of correct prediction. The recall rate means the prediction probability of a positive sample among all the positive samples. The expressions of accuracy and recall are as follows.

$$P = \frac{y_{\mathrm{TP}}}{y_{\mathrm{TP}} + y_{\mathrm{FP}}} \tag{18}$$

$$R = \frac{y_{\mathrm{TP}}}{y_{\mathrm{TP}} + y_{\mathrm{FN}}} \tag{19}$$

where y_{TP} indicates the number of correctly detected vehicles. y_{FP} refers to the number of false vehicles. y_{FN} indicates the number of undetected vehicles.

AP is the area enclosed by the P-R curve, which is the quantitative form of the detection accuracy. The expression of AP is as follows.

$$AP = \int_0^1 P(R)dR \tag{20}$$

where AP is composed of precision and recall for each category. The calculation of precision and recall is shown in formula (18) and formula (19) respectively.

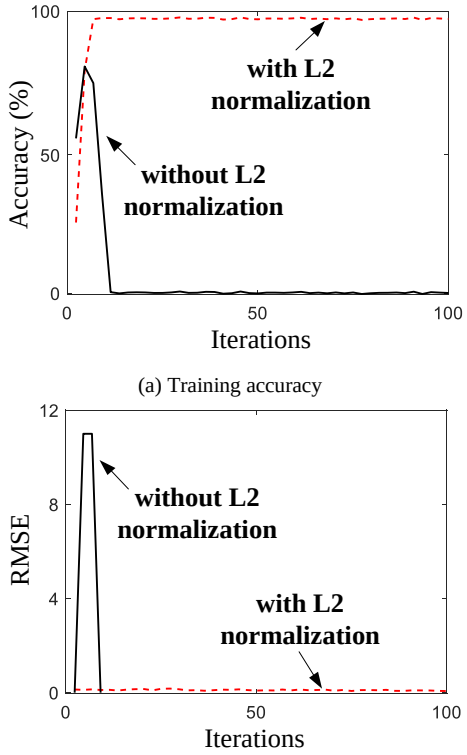
C. *Analysis of parameter adjustment*

1. Different learning rate

In the process of model training, three different learning rates are used to train the model separately, which are 0.01, 0.001 and 0.0001. The model iteration is fast while the training time is short for learning rate 0.01. But the detection performance is not ideal. When 0.001 is used as the learning rate, the iteration time of the model increases. The final detection effect is ideal. When the learning rate 0.0001 is used, the model iteration is slow. The training time is greatly long, while the convergence is slow. The detection result with the learning rate 0.0001 is almost the same as the learning rate 0.001. The result indicates that the properly small learning rate is effective for Faster RCNN.

2. Without L2 normalization

The channel number and map size are different for each layer. The scale difference is large among the feature maps. The features with large scale may weak the algorithm robustness. If the feature map is directly fused after ROI pooling, the model may fail to converge to the ideal state. The detection effect is poor without L2 normalization. L2 normalization is performed on the characteristic map after each ROI pooling. After normalization, the scaling operation is applied to each feature independently. L2 normalization is used for multi-scale feature map fusion. Different feature map scale is normalized before transmitting into the subsequent network, which ensures the training stability.



(a) Training accuracy

(b) Training RMSE

Fig. 5. Training accuracy and RMSE with and without L2 normalization

The training process without L2 normalization is shown in Figure 5, while the training process with L2 normalization is also shown in Figure 5. For the training process without L2 normalization, the training accuracy increases during the initial training stage. But the training accuracy decreases to zero soon. Meanwhile the training RMSE also increases rapidly during the initial training stage. It indicates the model divergence. For the training process with L2 normalization, the training accuracy continues to increase during the whole training process. Meanwhile the training RMSE vibrates near a small value.

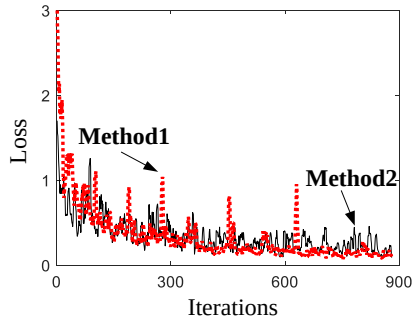
3. β and γ for training loss

ARP is introduced to adaptively adjust the number of candidate frames. The average value of regression loss is calculated at the same intervals. The value of (β , γ) is (3.0,1.0), (2.0,0.8) and (1.2,0.6) respectively. The total loss under different intervals is recorded. When the value of (β , γ) is (2.0,0.8), the training loss is the smallest. Therefore, set the current average loss greater than 0.8 times of the previous average loss. And set the current average loss less than 2 times of the previous average loss. If it is not in this interval, follow formula (9) to adjust the number of the candidate areas. To avoid the accuracy reduction caused by the low number of candidate frames, the value of the penalty factor δ_2 should not be large. The detection accuracy is reduced if $B(1-\delta_2)$ is small. Similarly, δ_1 should not be

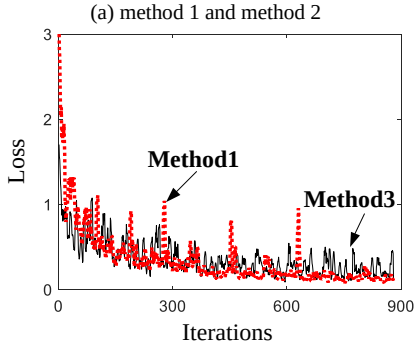
small. The detection accuracy is reduced with small $B(1+\delta_1)$.

D. Comparison with other vehicle identification models

The improved Faster RCNN (method1) is compared with two other methods. One comparison method uses Faster RCNN with traditional VGG network (method2). The other comparison method uses the Faster RCNN with Resnet network (method3). The training loss and RMSE of the three algorithms are shown in Figure 6 and Figure 7.

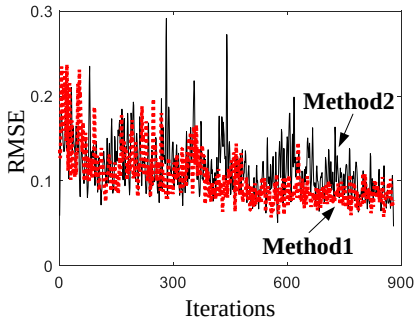


(a) method 1 and method 2

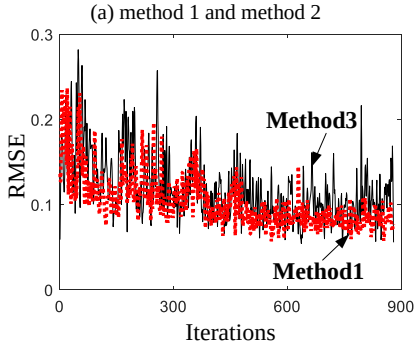


(b) method 1 and method 3

Fig. 6. Training loss function cures



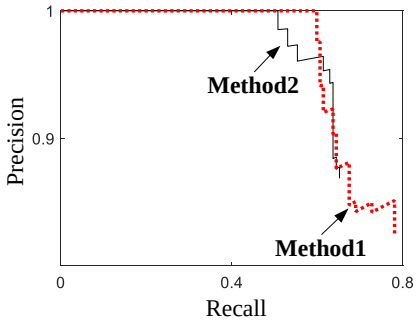
(a) method 1 and method 2



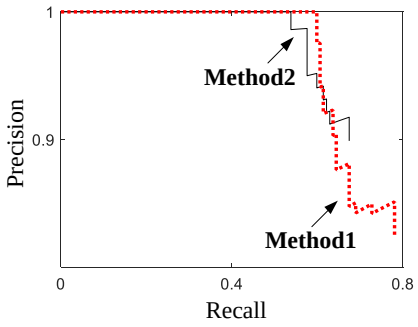
(b) method 1 and method 3

Fig. 7. Training RMSE

Seen from Figure 6, the random initialization of parameter leads to a large loss during the early training stage. The loss value of method1 is more than 1 during the initial training stage. Method2 and method3 converge fast due to their simple network structure. As the iteration increasing, the loss value of mehtod1 is gradually lower than method2 and method3. The loss value of method1 is close to 0.2 after training stability. Seen from Figure 7, the RMSE of method1 is similar to the RMSE of method2 and method 3 during the initial training stage. After training stability, the RMSE of method1 is lower than the RMSE of method2 and method3. It demonstrates that method1 has better robustness in the task of vehicle identification. The adaptive adjustment of regional suggestion provides a new strategy to solve the multi-scale problem. The new strategy can refer the anchor frames with multiple scales and aspect ratios. Then the classification and regression are performed. Due to the anchor based multi-scale design, the features can be computed on a single-scale image. The strategy uses shared features without increasing the training time. It can also solve the multi-scale problem, which improves the model recognition performance. Theoretically, the accuracy and recall rate should be high. The P-R curves of the three methods are shown in Figure 8.



(a) method 1 and method 2



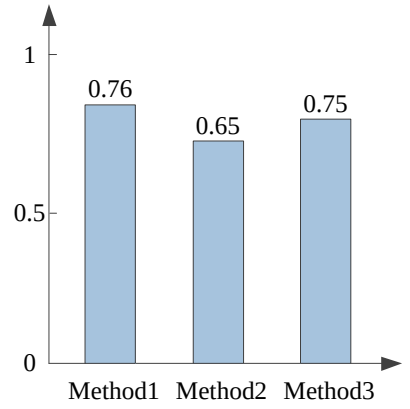
(b) method 1 and method 3

Fig. 8. P-R curves

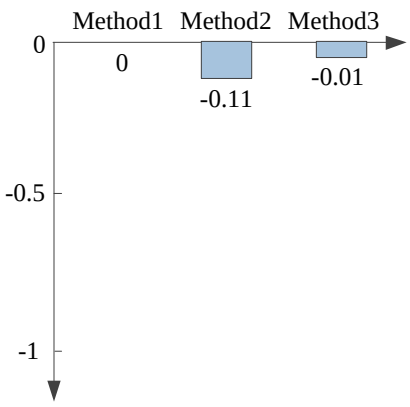
Seen from Figure 8(a) and Figure 8(b), the improvement of basic network structure and the design of multi-dimensional feature fusion have a positive impact on vehicle detection. The proposed anchor frame is more in line with the size and proportion characteristics. The detection accuracy and recall rate have been significantly improved.

The actual detection results of the three methods are shown in Figure 9. Seen from Figure 9, the anchor box of method1 is the most accurate. And method1 can accurately identify the vehicles. Figure 10 shows the comparison results before and after the improvement. Figure 10(a) shows the change of average detection accuracy of different methods. Figure 10(b) shows the change of detection accuracy. Negative value represents the reduction of detection accuracy. The detection accuracy of method1 is 1% higher than method2. And the detection accuracy of method1 is 2% higher than method3.

Fig.9. Comparison of three methods for actual pictures



(a) Average precision



(b) Precision comparison

Fig. 10. Comparison of different methods for AP

V. CONCLUSIONS

In this study, Faster RCNN model is applied to vehicle recognition. Combined with the advantages of Resnet network structure in target recognition, the network structure is improved. To use the information contained in image features, a multi-scale feature fusion strategy is introduced. The selection method of anchor box is improved to achieve vehicle recognition. The experimental results show that the proposed method can adapt to different backgrounds. The recognition results show that the proposed method can better meet the requirements of vehicle recognition, which lays a good foundation for vehicle recognition in automatic driving.

REFERENCES

[1] Chin Poo Lee, and Kian Ming Lim, "MFRD-80K: A Dataset and Benchmark for Masked Face Recognition," *Engineering Letters*, vol. 29, no.4, pp1595-1600, 2021

[2] Yi-Qin Zhou, Xin-Yu Ouyang, Nan-Nan Zhao, Hai-Bo Xu, and Hui Li, "Prescribed Performance Adaptive Neural Network Tracking Control of Strict-Feedback Nonlinear Systems with Nonsymmetric Dead-zone," *IAENG International Journal of Applied Mathematics*, vol. 51, no.3, pp444-452, 2021

[3] Shumpei Takezaki, and Kazuya Kishida, "Construction of CNNs for Abnormal Heart Sound Detection using Data Augmentation," *Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists* 2021, 20-22 October, 2021, Hong Kong, pp18-23

[4] Pavani K, "Novel Vehicle Detection in Real Time Road Traffic Density Using Haar Cascade Comparing with KNN Algorithm based on Accuracy and Time Mean Speed," *Revista Gestão Inovação e Tecnologias*, vol. 1, no.2, pp897-910, 2021.

[5] Chen X Z, Chang C M, Yu C W, "A Real-Time Vehicle Detection System under Various Bad Weather Conditions Based on a Deep

Learning Model without Retraining," *Sensors*, vol. 20, no. 20, pp5731, 2020.
[6] Chen Y, Qin R, Zhang G, "Spatial Temporal Analysis of Traffic Patterns during the COVID-19 Epidemic by Vehicle Detection Using Planet Remote-Sensing Satellite Images," *Remote Sensing*, vol. 13, no.2, pp208, 2021.
[7] Dominguez J, Al-Tam F, Sanguino T, "Vehicle Detection System for Smart Crosswalks Using Sensors and Machine Learning," *18th International Multi-Conference on Systems, Signals & Devices (SSD)*, 2021.
[8] Kowol K, Rottmann M, Bracke S, "YOdar: Uncertainty-based Sensor Fusion for Vehicle Detection with Camera and Radar Sensors," *13th International Conference on Agents and Artificial Intelligence*, 2021.
[9] Lee C H, Lim Y C, Kwon S, "Stereo vision-based vehicle detection using a road feature and disparity histogram," *Optical Engineering*, vol. 50, no. 2, pp4-23, 2011.
[10] Sebdani F M, Pourghassem H, "Vehicle Detection Based on Template Matching in Traffic Surveillance System," *International Review on Computers & Software*, vol. 7, no. 3, pp1114-1121, 2012.
[11] Jain A M, Tiwari N, "Airborne vehicle detection with wrong-way drivers based on optical flow," *International Conference on Innovations in Information,Embedded and Communication Systems (ICIIECS)*, 2015.
[12] Akshobhya K M, "Machine learning for anonymous traffic detection and classification," *11th International Conference on Cloud Computing, Data Science & Engineering*, 2021.
[13] Xiang Y, Fu Y, Huang H, "Global Topology Constraint Network for Fine-Grained Vehicle Recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol.21, no. 7, pp2918-2929, 2020.
[14] Roxana V P, Alberto S R, Deni T R, "Vehicle Detection with Occlusion Handling, Tracking, and OC-SVM Classification: A High Performance Vision-Based System," *Sensors*, vol.18, no.2, pp374, 2018.
[15] Ammar A, Koubaa A, Ahmed M, "Vehicle Detection from Aerial Images Using Deep Learning: A Comparative Study," *Electronics*, vol. 10, no. 7, pp820, 2021.
[16] Yang X, Wang F, Bai Z, "Deep Learning-Based Congestion Detection at Urban Intersections," *Sensors*, vol. 21, no. 6, pp2052, 2021.
[17] Ligayo M, Costa M T, Tejada R R, "An Augmented Deep Learning Inference Approach of Vehicle Headlight Recognition for On-Road Vehicle Detection and Counting," *International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, 2021.
[18] H Wang, Yu Y, Cai Y, "Soft-Weighted-Average Ensemble Vehicle Detection Method Based on Single-Stage and Two-Stage Deep Learning Models," *IEEE Transactions on Intelligent Vehicles*, vol. 99, pp1, 2020.
[19] Girshick R, Donahue J, Darrell J, Malik J, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp580-587.
[20] Girshick R, "Fast R-CNN," *Proc. of IEEE International Conference on Computer Vision*, 2015, pp1440-1448.
[21] Ren S, He K, Girshick R, Sun J, "Faster R-CNN:Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.39, no.6, pp1137-1149, 2017.

Zhang Meihong is from Dongming, Shandong. She acts as an associate professor in Henan Industry and Trade Vocational College. Her research interests are automotive power and control technology, neural network, etc.