YOLO-RFB: An Improved Traffic Sign Detection Model

Zhongqin Bi[1], Fuqiang Xu[1], Meijing Shan[2(✉)], and Ling Yu[1]

[1] School of Computer Science and Technology, Shanghai University of Electric Power, Shanghai 200090, China

[2] Department of Information Science and Technology, East China University of Political Science and Law, Shanghai 201620, China

5805831@qq.com

Abstract. With the development of intelligent transportation system, the detection method of traffic signs plays an important role in unmanned driving. However, due to the real-time and reliability characteristics of the automatic driving system, each traffic sign needs to be processed in a specific time interval to ensure the precision of the test results. Automatic driving is developing rapidly and has made great progress. Various traffic sign detection algorithms are proposed. Especially, convolutional neural network algorithm is concerned because of its fast execution and high recognition rate. But in the real world of complex traffic conditions, those algorithms still have problems such as poor real-time detection, low precision, false detection and high missed detection rate. To overcome those problems, this paper proposed an improved algorithm named as YOLO-RFB based on YOLO V4 network. Based on YOLO V4 network, the main feature extraction network is pruned, and convolution layer is replaced by RFB structure in two output feature layers. In the detection results of GTSDB data sets, the mAP of improved algorithm achieves 85.59%, 4.76% points higher than the original algorithm, and the FPS reaches 48.72, which is slightly lower than that of the original YOLO V4 algorithm 50.21.

Keywords: Unmanned driving · Traffic sign detection · GTSDB · YOLO V4

1 Introduction

In recent years, with the application of deep learning technology in the field of unmanned driving, the commercialization of unmanned vehicles has gradually become the focus and trend. Car companies and Internet companies are rushing to enter the self-driving field. The automobile industry is a special industry, because it involves the safety of passengers, any accident is unacceptable, so there are almost strict requirements for safety and reliability. Therefore, in the process of studying unmanned driving, the precision, real-time performance and robustness of the algorithm are highly required [1]. Traffic

S. Deng et al. (Eds.): MobiCASE 2021, LNICST 434, pp. 3–18, 2022.
https://doi.org/10.1007/978-3-030-99203-3_1

sign detection is one of the important parts of unmanned driving system, which plays an important role in reducing safety accidents and assisting drivers in driving. Due to the high requirement of real-time detection of traffic signs in the process of vehicle driving, and the influence of light, weather and shooting Angle, it is difficult to detect traffic signs in real time. In addition, due to the large number of traffic signs in the image, the small target, the pixel value contains few features, which increases the difficulty of traffic sign detection. Currently, there are four main detection methods for traffic signs, which are color-based method, shape-based method, multi-feature fusion based method and deep learning based method [2]. Among them, the detection method based on color is susceptible to the influence of illumination and other factors, and its robustness is poor under complex illumination conditions. The factors such as color difference and complex background will also cause the loss of effective information. The shape-based detection method has good robustness and strong anti-noise ability. However, due to the large amount of computation and high requirements on hardware, it cannot meet the requirements of real-time performance. The methods based on multi-feature fusion often need to extract target features manually. The traffic sign detection method based on deep learning can automatically extract target features and has good model generalization ability, which has been widely used.

In this paper, on the basis of analyzing the problems existing in the application of YOLO V4 network in traffic sign detection, the optimized YOLO V4 network can not only quickly classify traffic signs, but also effectively improve the precision of traffic sign detection. Firstly, this paper pruned the backbone network in YOLO V4 network to reduce convolution operations and effectively improve the speed of the model without losing too much performance. Secondly, a new module is added to the network to enhance the feature extraction ability of the network by simulating human receptive field.

2 YOLO V4

As shown in Fig. 1, the network of YOLO V4 [3] consists of three parts: trunk network, neck network and head network. Among them, the backbone network is CSPDarkNet53, CSP [4] network is a new backbone network that can enhance the learning ability of neural networks, and maintain precision, reduce computing bottlenecks and memory costs while being lightweight. Darknet53 is the backbone network used by YOLOv3 [14], which combines the ideas of CSP network to form the CSPDarkNet53 network. The neck network was SPP [5] and PAN [6]. SPP network is the same size of features obtained by convolution of CSPDarknet53. PAN network has the structure of repeated feature extraction, which improves the precision of small object detection. The head network is responsible for the final prediction task.

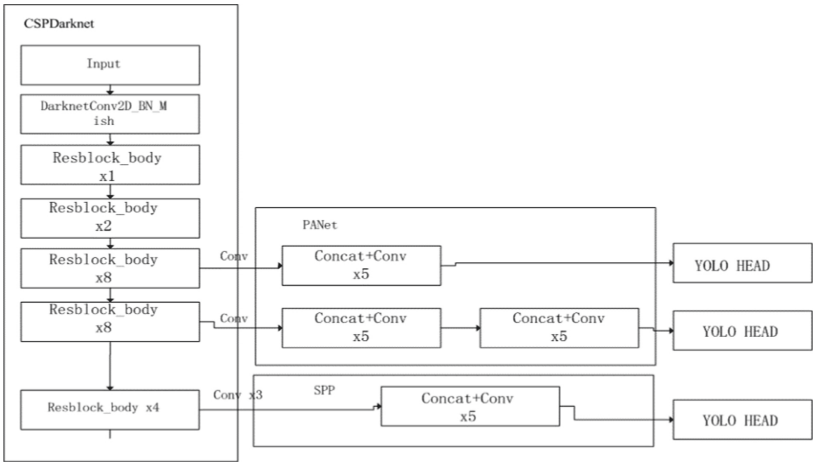


Fig. 1. Network structure of YOLO V4

3 Improved YOLO V4

In this paper, the backbone network and feature extraction network are improved based on YOLO V4 network model.

3.1 RFB

RFB [8] network is used for object detection, which can achieve good results while taking into account speed. This network mainly introduces Receptive Field Block (RFB) into SSD [15] network. The purpose of introducing RFB is to enhance the feature extraction capability of the network by simulating the Receptive Field of human vision. In terms of structure, RFB draw on the idea of Inception Mainly, dilated convolution [17] is added on the basis of Inception to effectively increase the receptive field. The overall improvement is based on SSD network to improve the detection speed while ensuring the precision.

RFB introduced the concept of initial frame, that is in the center of the cell of each characteristic graph set a series of scales and different initial box size, they will reverse mapping to one of the original position, if the initial frame's position at the right moment and the location of the true target box overlap degree is high, then predict the initial frame's category by loss function and fine-tune the shape of these initial boxes to make it match the actual target box of the tag [18].

The initial box has two main parameters: scale S and aspect ratio a. Assuming that m feature graphs are used for prediction [20], the initial box calculation formula of each feature graph is as follows:

$$S_k = S_{\min} + \frac{S_{\max} - S_{\min}}{m - 1}(k - 1), k \in [1, m] \tag{1}$$

With the deepening of network layers (smaller feature graphs), the scale of initial frames increases linearly. The minimum initial box scale is 0.2, and the maximum is 0.9. It is the general scale design principle of RFB. The details are shown in Table 1.

Table 1. The default boxes size on each feature map

Index	Feature map	Feature map size	Default box scale	True size
1	Conv4_3	38	0.2	60
2	Conv7	19	0.34	102
3	RFB stride 2	10	0.48	144
4	RFB stride 2	5	0.62	186
5	Conv10_2	3	0.76	228
6	Conv11_2	1	0.9	270

The width to height ratio of the initial box is set as follows:

$$a_r \in \{1, 2, 3, \frac{1}{2}, \frac{1}{3}\} \tag{2}$$

The initial frame width and height on each feature graph can be obtained by the following formula:

$$w_k^a = S_k \sqrt{a_r} \tag{3}$$

$$h_k^a = \frac{S_k}{\sqrt{a_r}} \tag{4}$$

Added a square initial box for the initial box with an aspect ratio of 1:

$$S_k^{'} = \sqrt{S_k + S_{k+1}} \tag{5}$$

The details of the feature map are shown in Table 2.

3.2 YOLO-RFB

YOLO V4 has higher detection precision than previous algorithms, but it will take a long time if the detection is carried out on the terminal with poor hardware performance. In order to improve the detection speed and precision of traffic signs, this section studies the YOLO V4 network and makes corresponding improvements [19].

The original YOLO V4 model replaced the trunk extraction network by DarkNet53 with CSPDarkNet53, using a CSPNet structure to enhance CNN's learning ability and maintain precision while being lightweight. At the output end, there are three characteristic layers of different sizes, P3, P4 and P5, whose sizes are 1/8, 1/16 and 1/32 of the original input size respectively. After a convolution operation, the feature images

Table 2. Characteristic graph and its size used in RFB network

Feature map	Size
Conv4_3	38 * 38
Conv7	19 * 19
RFB stride 2	10 * 10
RFB stride 2	5 * 5
Conv10_2	3 * 3
Conv11_2	1 * 1

output by the two feature layers P3 and P4 enter the PANet structure for feature fusion. Feature layer P5 needs to conduct a convolution operation, then enter SPP structure for maximum pooling, separate context features, conduct another convolution operation, and finally enter PANet structure for feature fusion. These operations enable the model to extract the features of different scales and types effectively to a certain extent [7].

In the actual scene, traffic sign images often occupy a small area of the image, and their appearance is also greatly affected by the environment. If the original YOLO V4 model is directly used for training and detection, the result is not ideal. And a large number of experimental analysis show that the number of large target samples is often more than small one in network model training, which leads to the lack of information obtained by network model in feature extraction of small target samples, and small targets cannot be fully trained. Therefore, the improvement of network model should start with how to strengthen the ability of network model to extract small target samples and increase the feature information.

In view of the fact that traffic signs are small size targets in images, as well as environmental factors, the ideas of YOLO V4 model are referenced and modified in this section to obtain the improved YOLO V4 network structure, YOLO-RFB, as shown in Fig. 2. The main aspects of improvement of YOLO V4 network are shown as follows:

- The CSPDarkNet53 trunk extraction network was pruned to reduce the number of Resblock_body in the three output feature layers from 8, 8, 4 to 2, 4, 4 respectively, and reduce part of the convolution operation to improve the speed effectively while the performance of the model is guaranteed as much as possible.
- The convolution operation required by P3 and P4 output feature layers is replaced with RFB structure. RFB adds a wormhole convolution layer to Inception to enhance the feature extraction capability of the network by simulating the receptive field of human vision, so as to obtain higher semantic level and more global feature information. In this way, the characteristic information of traffic signs, especially small target samples, can be better extracted in the training process, and the performance of the network model in traffic sign detection can be improved more effectively.

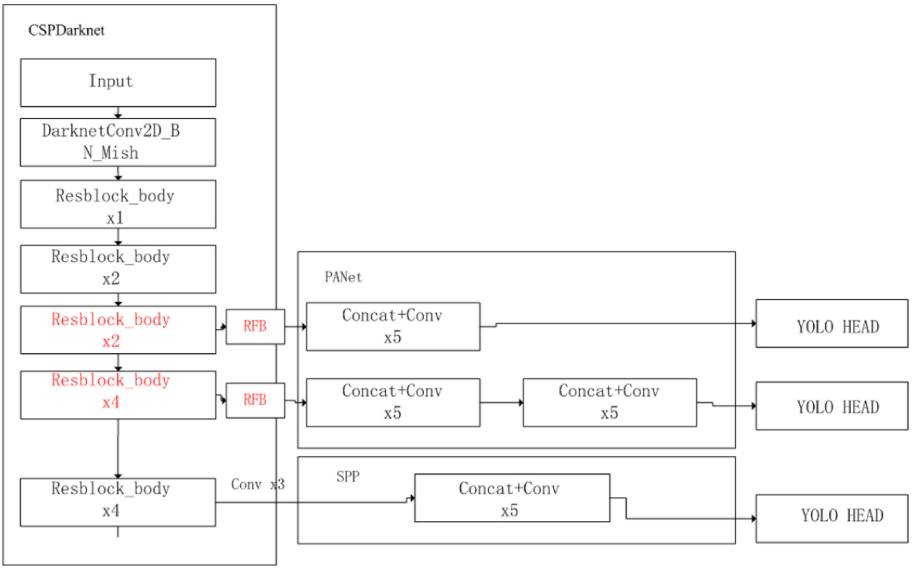


Fig. 2. Network structure of YOLO-RFB

4 Experiment and Analysis

Experiments are carried out on GTSDB data set to prove the effectiveness of the improved algorithm.

4.1 Common Data Sets

In the early 1990s, scholars from all over the world began to study traffic sign detection. But until 2011, the lack of publicly available traffic sign data sets was a major problem. In 2011, Traffic sign Detection and Recognition Competition was held in Germany, which was based on German Traffic Sign Benchmark Data Set (GTSBD) [9], indicating that traffic sign detection and recognition received high attention from the world. The proposed public benchmark data set makes the traffic sign detection algorithm has a unified evaluation standard, so as to compare the performance of the proposed network model or algorithm, also promotes the research progress of traffic sign detection and recognition.

GTSBD contains two data sets, German Traffic Sign Detection Benchmark (GTSDB) [10] and German Traffic Sign Recognition Benchmark (GTSRB) [11]. The GTSDB data set is used for the detection task of traffic signs, which contains 43 categories, which are divided into three categories in the detection task: For Prohibitory, Mandatory and Danger, a total of 900 images of driving scenes are available, and total of 852 traffic signs are available. The training set contains 396 samples of prohibitory traffic signs (59.5%), 114 samples of mandatory traffic signs (17.1%) and 156 samples of danger traffic signs (23.4%), and the test set contains 161 images of prohibitory traffic signs, 49 images of mandatory traffic signs and 63 images of danger traffic signs. Figure 3 shows a sample image from the GTSDB dataset.

Fig. 3. Sample image in GTSDB dataset

4.2 Experimental Environment and Hyperparameter Setting

The experiment is completed under Ubuntu16.04 operating system. The hardware used are as follows: CPU: Intel Core I9-10900K; GPU: NVIDIA GTX 3080 independent graphics card, 10G video memory. Python3.6 is the programming language used, and Pytorch1.7 is the development framework for deep learning.

The GTSDB dataset was used in the experiment (see Sect. 4.1 for details). Using the prior box of the original model cannot directly achieve the desired effect, since the GTSDB data set is much different from the COCO data set tested by the original YOLO V4. Therefore, k-means clustering method was used to obtain the prior boxes matching the GTSDB data set before the experiment.

Since the results of multiple experiments have little influence on the experimental results, Mosaic data enhancement and learning rate cosine annealing decay method are used instead of Label Smoothing. Before the training and detection of the improved YOLO V4 model, the original YOLO V4 model was first used to conduct experiments on the GTSDB data set, and the results were recorded as experimental comparison data. In the training process, the input size is set as 608 * 608, the batch size is 16, and the initial learning rate is 0.001. The optimizer uses Adam with step_size = 1 and gamma = 0.95. The total number of training steps (Epoch) was 400. In the first 200 steps, the learning rate dynamic decline method was used. Starting from the 201th step, the initial value of the learning rate was set as 1e−5. The cosine annealing attenuation method was used to change the learning rate, and the value of the minimum learning rate was set as 1e−8.

4.3 Evaluation Indicators

Mean Average Precision (mAP), Average Precision (AP), Precision and Recall are often used to evaluate the detection effect of the model in the field of object detection, Frames Per Second (FPS) and Giga Floating-point Operations Per Second (GigaFLOPS) were used to evaluate the detection performance.

Formula (6) and Formula (7) are the Precision and Recall calculation formulas. TP represents the input of the positive sample that the model thinks is positive, FP represents the input that the model misjudged as a positive sample, and FN represents the input that the model misjudged as a negative sample.

$$presicion = TP/(TP + FP) \tag{6}$$

$$recall = TP/(TP + FN) \tag{7}$$

A PR curve can be obtained by using Precision and Recall as horizontal and vertical coordinates, and the area under the PR curve is called AP index. The results of Precision and Recall are comprehensively evaluated. The definition is as follows:

$$AP_i = \sum_{k=1}^{N} p(k) \triangle r(k) \tag{8}$$

In Formula (8), $p(k)$ is the precision corresponding to the change point k of recall rate; $\Delta r(k)$ is the change of recall rate corresponding to change point K. N is the number of change points of recall rate; Different categories have different AP, i indicates the index of the category.

mAP averages the AP of all classes. The definition is as follows:

$$mAP = \frac{1}{m} \sum_{i=1}^{m} AP_i \tag{9}$$

The loss function is generally used to evaluate the error between the predicted value and the real of the model. It plays a key role in the speed of network learning and the final prediction effect of the model. IOU [12] is a commonly used indicator in object detection, used to reflect the detection effect between the prediction box and the target box. It is defined as:

$$IOU = \frac{|A \cap B|}{|A \cup B|} \tag{10}$$

YOLO V4 uses CIOU [13] to replace the regression loss function of the original BBOX. For object detection, it is necessary to encode the real frame after obtaining it and transform it into the form of prediction frame, and then compare the prediction result of the real frame with that of the network to optimize the network structure and make the prediction of the frame more accurate. CIOU is an improved version of IOU with the following formula:

$$CIOU = IOU - \frac{\rho^2(b, b^{gt})}{c^2} - \alpha v \tag{11}$$

Where, b and b^{gt} represent the central point of the prediction frame and the real frame respectively; ρ^2 is the Euclidean distance of two central points; c represents the diagonal

distance of the smallest closure region that can contain both the prediction box and the real box. Among them:

$$\alpha = \frac{v}{1 - IOU + v} \tag{12}$$

$$v = \frac{4}{\pi^2}(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h})^2 \tag{13}$$

αv is positively correlated with the frame height difference between the real frame and the prediction frame. CIOU refers to the deviation degree between the real box and the prediction one, then the LOSS function is:

$$LOSS_{CIOU} = 1 - IOU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v \tag{14}$$

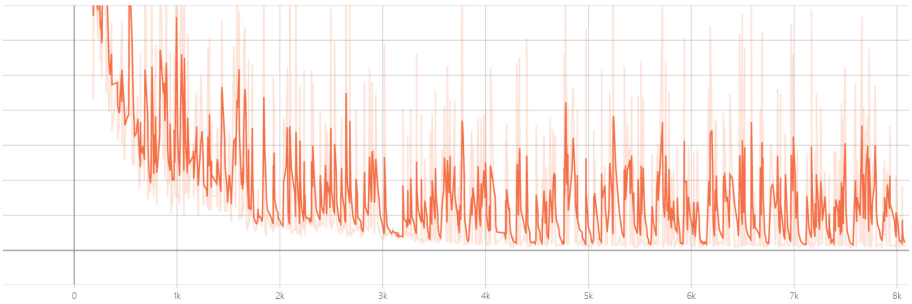
In addition to detection precision, detection speed is another important index of the object detection model. Real-time detection can be realized only when the speed reaches a certain level. The current metric used to assess detection speed is FPS, which is the number of images that can be processed per second. Generally speaking, when the FPS of the model is greater than 30, it is considered that the model meets the standard of real-time monitoring [16].

4.4 Experimental Results and Comparative Presentation

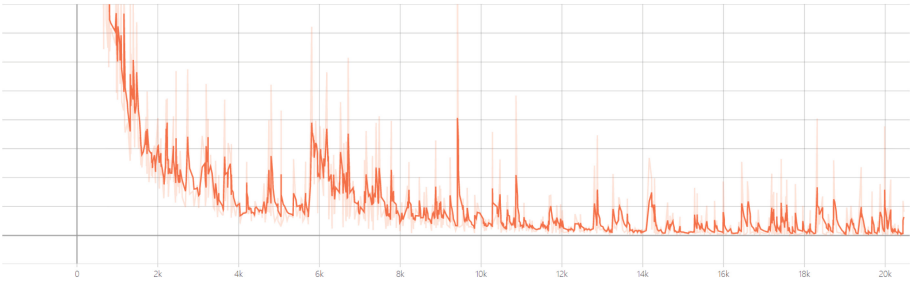
Detection precision and speed are very important in traffic sign detection tasks since vehicles in the real world are usually running at high speed, the size and cost of the network model in the vehicle system should also be considered. Therefore, the precision, speed and Parameter of the detection model must be evaluated.

The dimension of prior box of the original YOLO V4 model is (12, 16), (19, 36), (40, 28), (36, 75), (76, 55), (72, 146), (142, 110), (192, 243), (459, 401). The dimension of this group of prior frames is obtained by K-means clustering for COCO data set, while the data set tested in this paper is GTSDB data set, so the dimension of prior frames needs to be determined again. Therefore, before the experiment, K-means clustering method was used to cluster the GTSDB data set to generate the dimensions of new prior frames and obtain the dimensions of nine prior frames. (9, 15), (10, 18), (12, 20), (13, 22), (14, 25), (17, 29), (20, 34), (27, 45), (40, 66).They were assigned to the three feature maps with different scales, and the smaller the prior frames were assigned to the feature maps with larger scales.

The loss curves of training sets and test sets of the original YOLO V4 model and the improved YOLO V4 model during training are shown in Fig. 4. As can be seen from the figure, the fluctuation of loss function on the training set and test set of YOLO V4 model is particularly severe during training, while the fluctuation of the model proposed is relatively gentle and its curve drops faster, indicating that the improved model performs better in the training process. Because YOLO-RFB model uses RFB module according to the characteristics of traffic sign data set, which strengthens the semantic features of feature graph, thus improving the generalization ability of network, making network model very excellent in training effect.

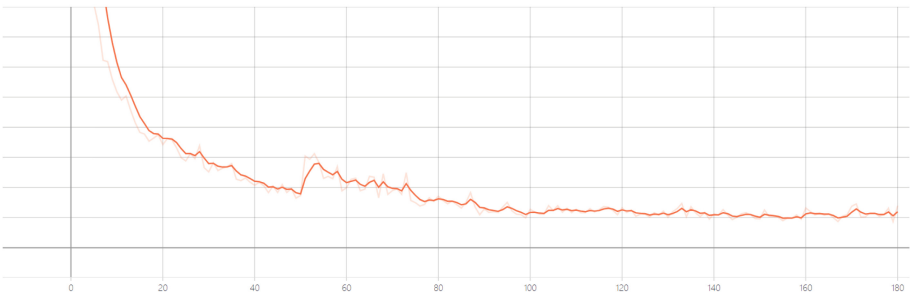


Loss curve of YOLO V4 model on training set



Loss curve of YOLO -RFB model on training set

Loss curve of YOLO V4 model on test set



Loss curve of YOLO-RFB model on test set

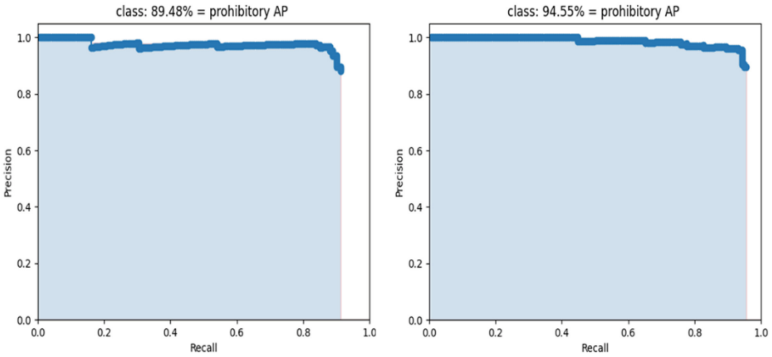
Fig. 4. Comparison of loss curve between YOLO V4 and YOLO-RFB

Figure 5 shows the comparison of precision and recall curves (PR curves) of YOLO V4 model and YOLO-RFB in the GTSDB data set of three traffic signs. The PR curves of YOLO V4 model on three different traffic signs are shown on the left, and the PR curves of improved YOLO V4 model on three different traffic signs are shown on the right. The area formed by PR curve is the AP of the corresponding traffic sign type. The higher AP is, the better the detection performance is. As can be seen from the data in the figure, the improved YOLO V4 model has good detection performance, and the detection results of each type of traffic signs are superior to YOLO V4. Because the improved YOLO V4 model adds the RFB module to the original YOLO V4 feature output layer, which can enhance the feature extraction ability of the network by simulating the receptive field of human vision, obtain higher semantic level and more global feature information, and detect traffic sign information more effectively. Thus, the performance of network model in traffic sign detection task is improved.

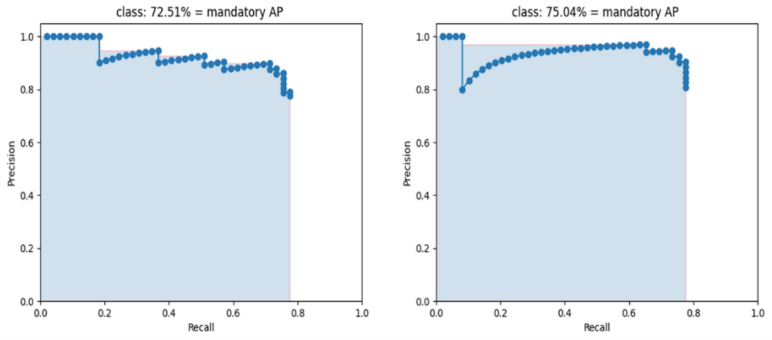
Table 3 shows the detailed precision results of each traffic sign superclass, as well as the Precision, Recall and average precision (AP) obtained by each detection model. As can be seen from the test results in the table, the AP of mandatory traffic signs is the lowest in almost all model test results, while there are obvious differences among other types of APs. The prohibitory traffic signs achieved the best results in each model, reaching 99.37% in the Cascaded model. These results are related to the distribution of sample numbers in the GTSDB dataset (59.5% for prohibitory traffic signs and 17.1% for mandatory traffic signs). Compared with the traditional two-stage traffic sign detection model Faster R-CNN Resnet 50, YOLO-RFB model still has a certain gap in the detection of compulsory traffic signs. YOLO-RFB model has significantly improved the detection results of each superclass compared with the existing one-stage model. AP of prohibitory traffic signs has increased by more than 5% points, and AP of danger traffic signs has also been greatly improved.

Finally, the execution time of YOLO-RFB was compared with that of traditional detection methods. Table 4 shows the mAP, FPS, GFLOP and Parameter obtained by various detection models. According to the data in the table, the mAP of the two-stage detection model is significantly higher than that of the one-stage detection model, but its FPS is the lowest. This is because all the two-stage detection models are formed into a series of candidate boxes as samples, and then the samples are classified through the convolutional neural network. Automatic driving technology has high requirements for real-time detection, and the detection speed of two-stage model is often not up to the requirements. Although the mAP of the one-stage detection model is slightly lower, the FPS is all over 40, which can achieve a stable real-time detection effect. The original YOLO V4 model was a fast and accurate choice with a mAP of 80.83% and an FPS of 50.21. The performance of YOLO-RFB model in FPS is not significantly decreased, but the performance of mAP is greatly improved, it can be seen that the improved model has better performance, stronger robustness and better generalization ability.

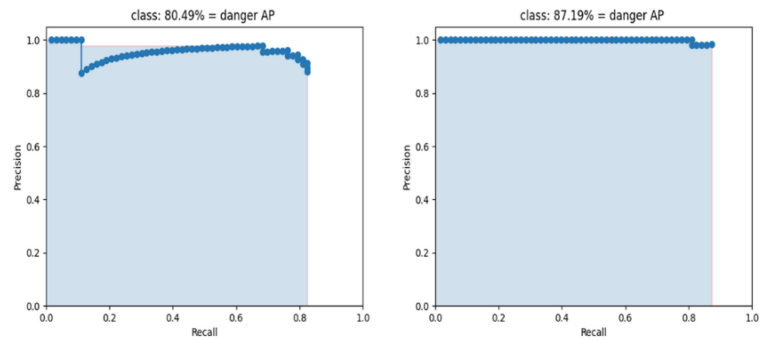
As shown in Fig. 6, YOLO-RFB model is used to detect images of GTSDB data set. The left image is the detection result of YOLO-RFB model, and the right is the detection result of original YOLO V4 model. The score of visual detection is greater than the threshold of 0.3. These images show three common scenarios: the first is a normal driving scenario on the road, with an extra detection box on the right side of



PR curves of YOLO V4 (left) and YOLO-RFB (right) on Prohibitory signs



PR curves of YOLO V4 (left) and YOLO-RFB (right) on Mandatory signs



PR curves of YOLO V4 (left) and YOLO-RFB (right) on Danger Signs

Fig. 5. Performance comparison chart of YOLO V4 and YOLO-RFB

Table 3. The detection results of each traffic sign detection model on GTSDB

Model	Class	Precision (%)	Recall (%)	AP (%)
Faster R-CNN Resnet 50	Prohibitory	91.38	98.75	98.62
	Mandatory	70.00	85.71	85.15
	Danger	79.45	92.06	90.78
Cascaded R-CNN	Prohibitory	84.66	99.38	99.37
	Mandatory	76.67	93.88	92.58
	Danger	86.76	93.65	93.52
MST-TSD	Prohibitory	96.95	78.88	78.77
	Mandatory	90.00	55.10	54.46
	Danger	93.18	65.08	65.05
Yolov3	Prohibitory	92.31	89.44	88.73
	Mandatory	79.07	69.39	65.70
	Danger	94.55	82.54	82.06
YOLO V4	Prohibitory	88.02	91.30	89.48
	Mandatory	77.55	77.55	72.51
	Danger	88.14	82.54	80.49
YOLO-RFB	Prohibitory	89.53	95.43	94.55
	Mandatory	80.85	77.55	75.04
	Danger	98.21	87.30	87.19

Table 4. Performance of traffic sign detection models on GTSDB

Model	mAP (%)	FPS	GFLOP	Parameter (10^6)
Faster R-CNN Resnet 50	95.77	2.26	1837.54	59.41
Cascaded R-CNN	95.15	11.70	269.90	64.59
MST-TSD	66.10	42.12	7.59	13.47
Yolov3	78.83	46.55	62.78	50.59
YOLO V4	80.83	50.21	63.84	63.94
YOLO_RFB	85.59	48.72	81.06	71.89

the image, which mistakenly identifies the background building as a traffic sign, while the YOLO-RFB model does not detect; In the second scenario, driving on the street with multiple traffic signs of different categories, the model can only detect one traffic sign without improvement, while the improved model can well avoid the occurrence of missed detection. The third is the scenario with multiple traffic signs on both sides of the road. It can be observed that the unimproved model missed one traffic sign, while the improved model can detect all traffic signs well. From the comparison of detection results, it can be concluded that YOLO-RFB model performs well in the task of traffic sign detection.

Fig. 6. The detection results of YOLO-RFB (left) and YOLO V4 (right) in GTSDB

5 Conclusion

This paper study the detection of traffic signs. The original YOLO V4 model is first used for experiments. It is found that the original network model has a good performance

in the detection speed, but the mAP is not enough. Therefore, a traffic sign detection model YOLO-RFB based on YOLO V4 network was proposed. Based on YOLO V4 network, the main feature extraction network was pruned, and the convolution operation was replaced by RFB structure in two output feature layers. In the detection results of GTSDB, mAP of YOLO-RFB model reached 85.59%, higher than 80.83% of original YOLO V4 model. Experimental results show that the proposed model can improve the detection precision obviously while the detection rate remains stable.

While this article on the traffic sign detection task had certain research results, but there are still some problems deserves further research, the model still has a lot of optimization and improvement in future work, the model will be looking for more suitable for traffic sign detection feature extraction of the network and data sets, further enhance the speed of the model test and precision.

References

1. Zhang, X.Y., Gao, H.B., Zhao, J.H., Zhou, M.: Overview of autonomous driving technology based on deep learning. J. Tsinghua Univ. (Nat. Sci. Ed.) **58**(04), 438–444 (2018)
2. Han, S., Kang, J., Min, K., Choi, J.: DiLO: direct light detection and ranging odometry based on spherical range images for autonomous driving. ETRI J. **43**(4), 603–616 (2021)
3. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: YOLO V4: optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020)
4. Wang, C.Y., Liao, H.Y.M., Wu, Y.H., et al.: CSPNet: a new backbone that can enhance learning capability of CNN. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 390–391 (2020)
5. He, K., Zhang, X., Ren, S., et al.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1904–1916 (2015)
6. Mei, Y., Fan, Y., Zhang, Y., et al.: Pyramid attention networks for image restoration. arXiv preprint arXiv:2004.13824 (2020)
7. Yun, S., Han, D., Oh, S.J., et al.: CutMix: regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6023–6032 (2019)
8. Liu, S., Huang, D.: Receptive field block net for accurate and fast object detection. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 385–400 (2018)
9. Stallkamp, J., Schlipsing, M., Salmen, J., et al.: The German traffic sign recognition benchmark: a multi-class classification competition. In: The 2011 International Joint Conference on Neural Networks, pp. 1453–1460. IEEE (2011)
10. Houben, S., Stallkamp, J., Salmen, J., et al.: Detection of traffic signs in real-world images: the German traffic sign detection benchmark. In: The 2013 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2013)
11. Stallkamp, J., Schlipsing, M., Salmen, J., Igel, C.: Man vs. computer: benchmarking machine learning algorithms for traffic sign recognition. Neural Netw. **32**, 323–332 (2012). https://doi.org/10.1016/j.neunet.2012.02.016
12. Yu, J., Jiang, Y., Wang, Z., et al.: UnitBox: an advanced object detection network. In: Proceedings of the 24th ACM International Conference on Multimedia, pp. 516–520 (2016)
13. Zheng, Z., Wang, P., Liu, W., et al.: Distance-IoU loss: faster and better learning for bounding box regression. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 07, pp. 12993–13000 (2020)

14. Redmon, J., Farhadi, A.: YOLOV3: an incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
15. Liu, W., Anguelov, D., Erhan, D., et al.: SSD: Single Shot MultiBox Detector. Springer, Cham (2016)
16. Yao, Z., Cao, Y., Zheng, S., et al.: Cross-iteration batch normalization (2020)
17. Yin, Q., Yang, W., Ran, M., Wang, S.: FD-SSD: an improved SSD object detection algorithm based on feature fusion and dilated convolution. Signal Process. Image Commun. **98** (2021)
18. Li, F., et al.: Decoding imagined speech from EEG signals using hybrid-scale spatial-temporal dilated convolution network. J. Neural Eng. **18**(4), 0460c4 (2021). https://doi.org/10.1088/1741-2552/ac13c0
19. Kong, D., Li, J., Zheng, J., Xu, J., Zhang, Q.: Research on fruit recognition and positioning based on you only look once version4 (YOLOv4). In: Journal of Physics: Conference Series, vol. 2005, no. 1 (2021)
20. Sharma, M., Bansal, R.K., Prakash, S., Asefi, S.: MVO algorithm based LFC design of a six-area hybrid diverse power system integrating IPFC and RFB IETE. J. Res. **67**(3), 394–407 (2021). https://doi.org/10.1080/03772063.2018.1548908