Journal of Computers Vol. 33 No. 4, August 2022, pp. 69-79
doi: 10.53106/199115992022083304006

YOLO-Based Efficient Vehicle Object Detection

Ting-Na Liu, Zhong-Jie Zhu*, Yong-Qiang Bai, Guang-Long Liao, Yin-Xue Chen

Ningbo Key Lab of DSP, Zhejiang Wanli University, Ningbo 315000, China
2411828375@qq.com, zhongjiezhu@yeah.net, byq-163@163.com,
623459525@qq.com, 1161059230@qq.com

Received 16 September 2021; Revised 12 January 2022; Accepted 13 February 2022

Abstract. Vehicle detection is one of the key techniques of intelligent transportation system with high requirements for accuracy and real-time. However, the existing algorithms suffer from the contradiction between detection speed and detection accuracy, and weak generalization ability. To address these issues, an improved vehicle detection algorithm is presented based on the You Only Look Once (YOLO). On the one hand, an efficient feature extraction network is restructured to speed up the feature transfer of the object, and reuse the feature information extracted from the input image. On the other hand, considering that the fewer pixels are occupied for the smaller objects, a novel feature fusion network is designed to fuse the semantic information and representation information extracted by different depth feature extraction layers, and ultimately improve the detection accuracy of small and medium objects. Experiment results indicate that the mean Average Precision (mAP) of the proposed algorithm is up to 93.87%, which is 11.51%, 18.56% and 20.42% higher than that of YOLOv3, CornerNet, and Faster R-CNN, respectively. Furthermore, its detection speed can meet the real-time requirement of practical application basically with 49.45 frames per second.

Keywords: YOLO, vehicle object detection, depthwise convolution, K-means++

1 Introduction

As the key technology of intelligent traffic management, vehicle detection algorithm is widely used in traffic information collection and urban road safety network planning [1-3]. However, the existing algorithms still cannot meet the high requirements of accuracy and real-time for complex traffic scenes. Hence, improving the algorithm efficiency is one of the important research topics in the field of intelligent transportation.

Currently, the deep learning-based algorithms are the mainstream and show great potential in the field of vehicle detection, which can be mainly divided into two categories: two-stage algorithms and one-stage algorithms. For the former, a region proposal is generated firstly, then the candidate regions are put into classifier for classification and position correction. In detail, Girshick et al. proposed the Region-based Convolutional Neural Network (R-CNN) . A selective search method was used to extract candidate regions from the image, and then the features extracted from each candidate region are input into support vector machine for classification [4]. To avoid repeating CNN operations on the same area, Girshick et al. proposed Fast R-CNN, which can enhance the efficiency by generating regions on the feature map [5]. Based on R-CNN, Ren et al. proposed Faster R-CNN to solve the extremely time-consuming problem of candidate region generation, by using region proposal network (RPN) instead of selective search method [6].

For one-stage algorithms, such as Single-shot multi-box Detector (SSD) series and You Only Look Once (YOLO) series, they do not generate candidate regions in advance, but directly predict and regress the features for object detection. The SSD algorithm combines the ideas of regression and multi-scale detection to obtain six feature maps of different scales, and perform object detection and position regression on different receptive fields [7]. However, the detection efficiency decreases significantly with the expansion of the input image specification. Law et al. proposed CornerNet which directly predicted the upper left and lower right corners of the object to obtain the detection frame [8]. However, the redundancy of the network parameters is high, which leads to low detection efficiency. Redmon et al. proposed the YOLO to detect the object end-to-end [9], which can greatly improve the detection efficiencybut lead to low-precision positioning and low-recall of small objects. Based on the YOLO, Redmon et al. developed YOLOv2 to improve the detection accuracy of small objects, by taking Darknet-19 as the feature extraction network and fusing the shallow features and deep features.[10]. With the deep residual network, Redmon et al. proposed YOLOv3 with Darknet-53and multi-scale feature fusion to significantly improve the detection performance of the algorithm [11-12]. Subsequently, YOLOv4 was proposed by adding cross-stage partial network and path aggregation network (PANet), to further improve the accuracy of object detection [13-

* Corresponding Author

14]. In brief, YOLO model can detect the object in real time, but these algorithms still cannot well compromise the detection accuracy and speed. To address the above issues, an improved YOLO is proposed in this paper by optimizing the convolution process and reconstructing the feature extraction module and feature fusion module. Experiment results show that the mean Average Precision (mAP) of the proposed algorithm is up to 93.87%, which is 11.51%, 18.56% and 20.42% higher than that of YOLOv3, CornerNet, and Faster R-CNN, respectively. Meanwhile, the detection speed still can meet the real-time requirement of practical application basically with 49.45 frames per second. Specifically speaking, the main contributions of this paper are summarized as follows:

(1) The feature extraction network is reconstructed to reuse the extracted feature information. The skip-connection module is used to connect the deep residual network module, which reuses the feature information extracted from the input image, thereby improving the detection accuracy of vehicle object.

(2) The improved depthwise separable convolution is adopted in the skip-connection deep residual network module to reduce the computational complexity of the algorithm. One convolution kernel in depthwise convolution is responsible for one channel, and one channel is convolved by only one convolution kernel, which greatly reduces the number of parameters and computational complexity of the algorithm, thereby ensuring the detection speed of vehicle object.

(3) The feature fusion network is reconstructed to fuse the feature information extracted from different depth network layers. Before the feature pyramid is generated, two adjacent network layers are spliced together by down-sampling to fuse the semantic information and location information extracted by different depth network layers, which combines the global features and local features and compensates for the loss of information caused by the down-sampling in the feature extraction network, thereby improving the accuracy to detect small and medium vehicle objects.

(4) The across datasets test is carried out with the recalculated anchors. For the fairness, the experiments are conducted on the public KITTI dataset and our self-built dataset. Firstly, the anchors are recalculated and the algorithm model is trained on the KITTI dataset. And then, the performance of our algorithm is verified on both datasets with the trained model. The results validate the effectiveness and generalization ability of the proposed algorithm.

2 Related Works

Vehicle detection is one of the key techniques of intelligent transportation system with high requirements for accuracy and real-time. To improve the detection accuracy, more complex and deeper feature extraction networks are often used to extract more effective features; Or new network modules are added to enhance the feature extraction ability of the network.

Nguyen et al. improved Faster R-CNN and proposed using context-aware pooling instead of ROI pooling to improve accuracy in detecting small and occluded vehicles [15]. Zhou et al. combined the adversarial network with cascaded Fast R-CNN, which achieved better robustness for small and occlusion objects [16]. The advantage of the two-stage algorithm is that it can fully extract image features and achieve accurate classification and positioning. However, due to that the candidate regions generated by them are numerous and easy to repeat, and a lot of time is consumed in the processing of inputting each candidate region into convolutional neural network to obtain the prediction confidence and accurate position of objects, which limits the detection speed. Zhang et al. added three residual blocks to the bottom of the residual network of the original YOLOv3 and designed six multiscale convolutional feature maps for prediction to formulate the DF-YOLOv3, thus improving vehicle detection accuracy in complex scenes [17]. Xu et al. proposed Attention-YOLO, in which channel and spatial attention mechanisms are added to the feature extraction network of YOLOv3 to improve the positioning accuracy of the algorithm [18]. Mao et al. used the inverted residuals technique to improve the convolutional layer and added three SPP-blocks to solve the multiscale vehicle object detection problem [19]. The above work improves the detection accuracy of vehicle object detection algorithm from different angles. However, adding modules to the feature extraction network makes the network deeper and more complex, which increases the computational complexity and reduces the speed of vehicle object detection; And the inverse convolution technique is used to expand the receptive field, which may increase the influence of the background area on the feature point information, resulting in inaccurate object location or false detection of the background as the object. In this paper, an improved YOLO is proposed in this paper by reconstructing the feature extraction network and feature fusion network, and optimizing the convolution process.

3 Proposed Algorithm

In this section, the proposed algorithm is introduced in detail, mainly including feature extraction network, feature fusion network and convolution process in training stage. And the diagram of the proposed algorithm is shown in Fig. 1.

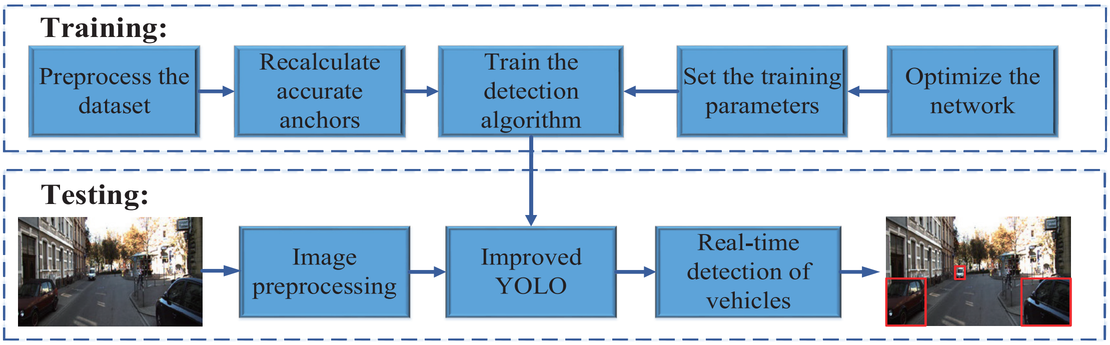

Training:
Preprocess the dataset
Recalculate accurate anchors
Train the detection algorithm
Set the training parameters
Optimize the network
Testing:
Image preprocessing
Improved YOLO
Real-time detection of vehicles

Fig. 1. Diagram of the proposed algorithm

3.1 Network Architecture

The proposed YOLO is mainly divided into three modules: the feature extraction network module based on the improved Darknet-53 convolutional neural network, the feature fusion network module and the feature map prediction module. The network structure is shown in Fig. 2. The feature extraction network in this paper is obtained by adding skip-connection modules on the basis of five groups of residual network modules of Darknet-53 network. It is composed of five large skip-connection deep residual network modules, each of which contains a different number of residual units and convolution modules. Residual unit is constructed by inputting the residual operations with two convolution modules. The introduction of the residual units makes the network deeper and prevents the gradient from disappearing. And the convolution module is composed of convolution layer, batch normalization (BN) layer and Mish activation function layer. The Mish activation function allows negative values to produce better gradient flow, and the smoothing feature can make the feature information penetrate into the network more deeply [20]. The comparison between Leaky ReLU activation function [21] and Mish activation function is shown in Fig. 3. The whole feature extraction network involves five down-sampling layers, and the multi-scale feature map is predicted and recognized in the last three ones.

In the feature fusion network module, a multi-scale feature fusion network is designed to fuse the feature information of different depth network layers. SPP module is added to increase the receiving range of feature more effectively [22]. And on the premise of increasing less computation, the feature map with strong low-resolution semantic features and the feature map with weak high-resolution semantic features but rich spatial features are fused. The output tensor of the detection layer is $S \times S \times B \times(4 + 1 + C)$, where B represents the number of detection objects per grid; 4 represents the width and height of the prediction boundary box and the horizontal and vertical coordinates of the center point; 1 represents the confidence score (the confidence of the object at this location) and C represents the total number of categories. To input the image size 608 × 608 as an example, the size of three multi-scale detection layers is $19 \times 19 \times 3 \times (5 + C)$, $38 \times 38 \times 3 \times (5 + C)$, $76 \times 76 \times 3 \times (5 + C)$ to detect large, medium and small objects respectively. The objects of the vehicle detection task are divided into four categories: Car, Truck, Van, Others; there are no background categories, so $C = 4$.

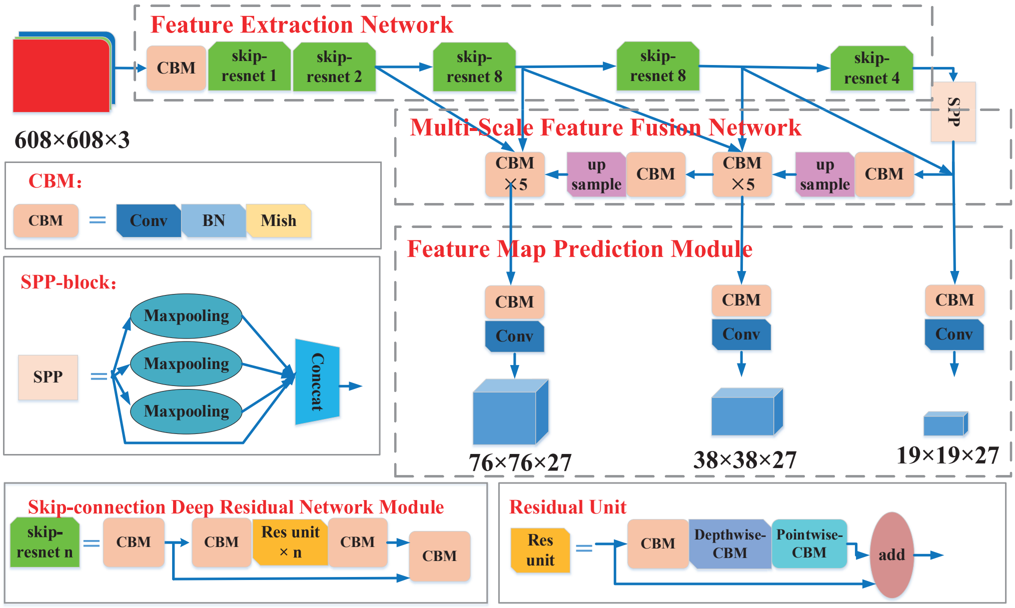
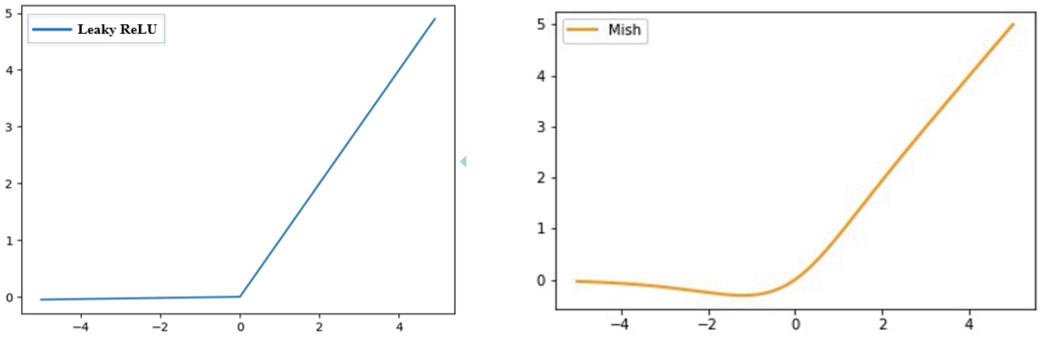

Feature Extraction Network
CBM
skip-resnet 1
skip-resnet 2
skip-resnet 8
skip-resnet 8
skip-resnet 4
SPP
608×608×3
Multi-Scale Feature Fusion Network
CBM ×5
up sample
CBM
CBM ×5
up sample
CBM
CBM:
CBM = Conv BN Mish
Feature Map Prediction Module
CBM
Conv
CBM
Conv
CBM
Conv
76×76×27
38×38×27
19×19×27
SPP-block:
SPP = Maxpooling Maxpooling Maxpooling Concat
Skip-connection Deep Residual Network Module
skip-resnet n = CBM CBM Res unit × n CBM CBM
Residual Unit
Res unit = CBM Depthwise-CBM Pointwise-CBM add

Fig. 2. Network structure of the improved YOLO



(a) The Leaky ReLU activation function

(b) The Mish activation function

Fig. 3. Comparison between (a) and (b)

3.2 Efficient Feature Extraction Module

In the feature extraction network, the introduction of residual unit solves the problem of gradient disappearance when the network deepens [23]. Fig. 4 shows the residual unit network structure. The core expression of the residual unit is:

$$x_n = H_n(x_{n-1}) + x_{n-1} \ , \tag{1}$$

where x_n represents the output of the n layer; H_n represents the nonlinear transformation including the convolutional layer, the BN layer and the activation function layer.

To improve the completeness of feature extraction and enhance the transfer of object features, the skip-connection module is used to connect the deep residual network module. The network structure is shown in Fig. 5. In the skip-connection deep residual network module, the way that the features are transferred layer by layer between the network layers is changed by the skip-connection module, that part of the network layer can be skipped, so that the features can be directly transferred to the following network layer. All input information is converted through two branches of the residual unit, the feature information extracted from the input feature map is reused, improving the integrity of feature information extraction.

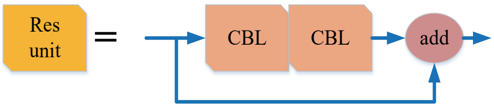


Fig. 4. Network structure of the ordinary residual unit

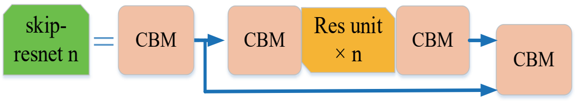
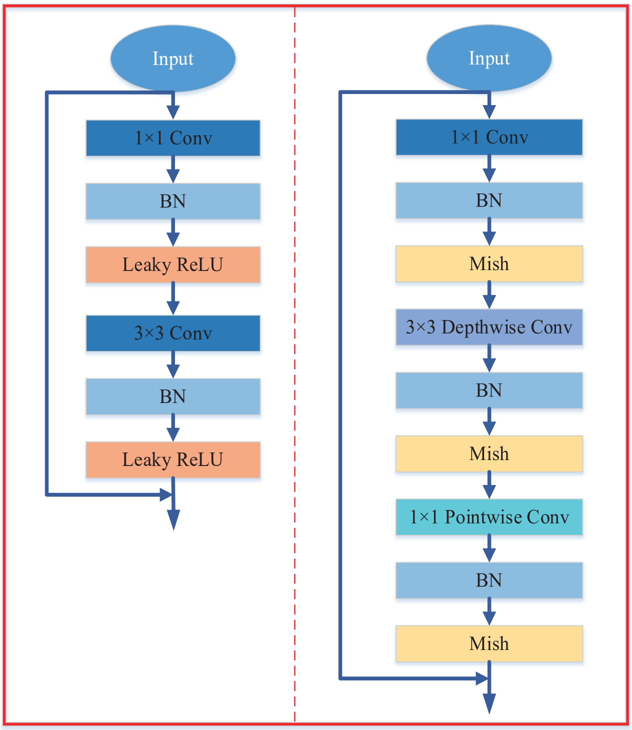


Fig. 5. Network structure of the skip-connection deep residual module
n represents the number of residual unit in the skip-connection deep residual module

Convolution layer is an important part of the whole neural network, which can automatically extract complex feature information from the image. However, with the increase of convolution kernel, the computation amount of ordinary convolution increases exponentially. To further improve the transmission speed of features in the network and ensure the real-time performance of algorithm detection, the improved depthwise separable convolution is used to replace the ordinary convolution in the skip-connection deep residual network module, and without increasing the receiving field, 1 × 1 convolution is added to increase the nonlinearity. The network structure comparison of ordinary convolution and improved depthwise separable convolution in residual unit is shown in Fig. 6. Among them, the depthwise convolution performs layered convolution on the input multi-channel vector, that is, one convolution is responsible for one channel, and one channel is convoluted by only one convolution. The channel dimension of the result of the depthwise convolution is expanded by the subsequent pointwise convolution, obtaining the same result as the traditional convolution operation [24].



(a) The ordinary convolution (b) Depthwise separable convolution
Fig. 6. Comparison of (a) and (b) in residual unit

In Fig. 6, the number of input channels is set to 3, and the number of output channels is set to 256. The ordinary convolution is directly connected with a 3 × 3 × 256 convolution kernel, so the number of parameters is 3 × 3 × 3 × 256 = 6912. The improved depthwise separable convolution proposed in this paper is completed in three steps: 1 × 1 convolution, 3 × 3 depthwise convolution and 1 × 1 pointwise convolution; therefore, the number of parameters is 1 × 1 × 1 × 256 + 3 × 3 × 3 + 3 × 1 × 1 × 256 = 1051, which is much less than the number used in the ordinary convolution, which greatly improves computational efficiency.

3.3 Multi-Scale Feature Fusion Module

The feature extraction network contains five down-sampling layers. Each time the feature map undergoes down-sampling, the feature information of different resolutions will be lost, resulting in less feature information extraction of small-scale objects. Therefore, a new multi-scale feature fusion network is proposed in this paper, the core expression of which is:

$$D=\{P_{n-k}(f'_{n-k}),\ldots P_n(f'_n)\}\ , \tag{2}$$

where $f'_n = f_n + s_n + s_{n-1}$, $f'_{n-1} = f_{n-1} + s_{n-1} + s_{n-2}$, … , $f'_{n-k} = f_{n-k} + s_{n-k} + s_{n-k-1}$, in the formula, $0<k<n$, represents the feature map extracted by the current network layer, represents the feature map output by the current skip-connection deep residual module, represents the feature map output by the previous skip-connection deep residual module.

The shallow network in the algorithm extracts detailed features such as the edge and texture of the object, and the deep network extracts the contour features of the object. The closer *k* is to *n*, the shallower the network layer depth of the feature map, the higher the resolution, the fewer semantic features and the fuzzier the location information of the object; the closer *k* is to 0, the deeper the network layer depth of the feature map, the lower the spatial resolution, and the more obvious the location information. Formula (2), "+" represents a feature fusion operation, after the spatial and semantic features extracted from different depth network layers are fused, the information loss caused by the down-sampling layer is compensated, so as to improve the detection performance of small-scale objects. As shown in Fig. 7, the feature information extracted by the shallow skip-connection depth residual module is input into the jump connection depth residual module of the next layer through the down sampling operation, while the feature information extracted by the deep feature extraction network is transmitted to the shallow layer through the up-sampling layer. Therefore, compared with the algorithm of only specific scale object detection in each layer feature map, the feature information of different depth network layers is effectively fused by the improved YOLO proposed in this paper to achieve accurate multi-scale target detection.

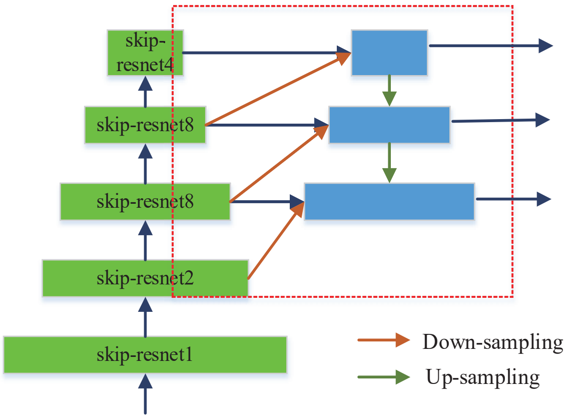


Fig. 7. Network structure of the multi-scale feature fusion proposed in this paper

3.4 Anchor Boxes Recalculation

The detection basis of YOLO is to generate a series of anchor boxes on the input image. Anchor box is the initial candidate box, which is the premise of generating prediction box. The selection of anchor will directly affect the position accuracy of the algorithm. YOLO draws lessons from the anchor box mechanism in the Faster R-CNN, but instead of manually setting the parameters of the anchor box, the dimension clustering method is used to determine the parameter values. The anchor boxes in the original YOLO is calculated by the author using K-means algorithm on the COCO dataset. K-mean is sensitive to the selection of initial clustering centers, and the clustering results vary with different initial clustering centers [25]. Aiming at the problems of K-means selecting initial clustering centers, K-means++ algorithm is used to recluster the anchor boxes according to the inherent shape of vehicle objects [26]. The basic steps of the algorithm are as follows:

1. A sample is randomly selected as the initial cluster center in the dataset.
2. Calculate the shortest distance between each sample and the existing cluster center.
3. Calculate the probability p that each point becomes the next cluster center, and select the next cluster center according to the roulette rule, the formula for calculating the probability is:

$$\frac{D(x)^2}{\sum_{x\in X} D(x)^2} . \tag{3}$$

4. Repeat steps 2 and 3 to select all cluster centers.
5. Screen out the K cluster centers.

The process of selecting cluster centers by K-means++ solves the problem of selecting cluster centers at the initial stage, so the performance of detecting vehicle objects is improved by benefiting from the anchor boxes more in line with the characteristics of vehicle objects clustered by K-means++.

4 Experimental Results and Analysis

To test the performance of the proposed algorithm, comparative experiments and across datasets testing experiments are carried out in this section. The experimental environment is as follows: The CPU is Intel (R) core (TM) i9-9920X @3.5GHz, the RAM is 16G, the GPU is NVIDIA GeForce RTX2080 Ti, the operating system is Win10, and CUDA10.0 and CUDNN7.4.2 are installed to support the GPU.

4.1 The Dataset and Evaluation Index

Deep learning-based vehicle object detection algorithms need to learn features from data samples, and the dataset must be representative. The KITTI dataset contains real image data collected from scenes such as urban areas, rural areas, and highways. Each image contains up to 15 cars and 30 pedestrians, with various degrees of occlusion and truncation [27]. The images reflect a variety of complex situations, such as multiple trucks and cars appearing in the same image or multiple trucks, vans and cars appearing in the same image, and include different lighting conditions, road environments and road conditions. For the peculiar application of vehicle object detection, the Car, Van, Truck, Pedestrian, Pedestrian (sitting), Cyclist, Tram, and Misc classes in the KITTI dataset are converted into Car, Van, Truck and Others classes. And the 7481 labeled images in KITTI dataset are divided into training set and testing set according to the ratio of 7:3 in this paper. To validate the generalization ability of the algorithm, a vehicle dataset is established through online collecting and real scene shooting. Fig. 8 shows some samples from the KITTI dataset.

Fig. 8. Partial samples from the KITTI dataset

In this paper, precision, recall, average precision (AP) and frames per second (FPS) are used to evaluate the detection performance of the algorithm. AP is the area surrounded by precision and recall curves and coordinate axes. The mAP is the average value of AP for each class. The calculation formulas are as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} , \tag{4}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} , \tag{5}$$

$$AP = \int_0^1 p(r)dr , \tag{6}$$

$$mAP = \frac{1}{Q} \sum_{q=1}^{Q} AP_q , \tag{7}$$

where TP is the number of positive samples correctly identified as positive samples by the algorithm, FN is the number of positive samples incorrectly identified as negative samples by the algorithm, FP is the number of negative samples incorrectly identified as positive samples by the algorithm, p is the precision, r is the recall, and Q is the total number of class.

4.2 Comparison and Analysis of Experimental Results

In the training phase, small batch random gradient descent is used for optimization, and a set of values that can improve the network quality are selected, in which the momentum parameter is set to 0.9, the weight attenuation coefficient is set to 0.0005, the initial learning rate is set to 0.001, and the input image size is set to 608 × 608. Due to the limitation of memory, batch is set to 64 and subdivision is set to 64.

In this paper, two indicators to verify the performance of vehicle detection algorithm are considered: detection accuracy and detection efficiency. To prove that each part of the modified YOLO is effective, comparative experiments are conducted. As shown in Table 1, after the feature extraction network of the original YOLOv3 is replaced by the skip-connection deep residual feature extraction network, the recall and mAP are increased by 11% and 9.9% respectively, because the feature extraction network proposed in this paper can reuse the features extracted from the input feature map. After replacing the FPN with the multi-scale feature extraction network proposed in this paper, the feature information of different depth network layers is spliced, and the semantic feature information and spatial feature information with different weights are adaptively learned. Under the condition of less loss of computing speed, the mAP of the algorithm is further improved by 1.61%. Comparative experiments show the effectiveness of the feature extraction module and feature fusion module proposed in this paper, that is, on the premise of ensuring the detection efficiency, it effectively improves the accuracy of vehicle detection algorithm. The red mAP curve shown in Fig. 9 indicates that the mAP of the improved YOLO is significantly higher than that of the original YOLOv3.

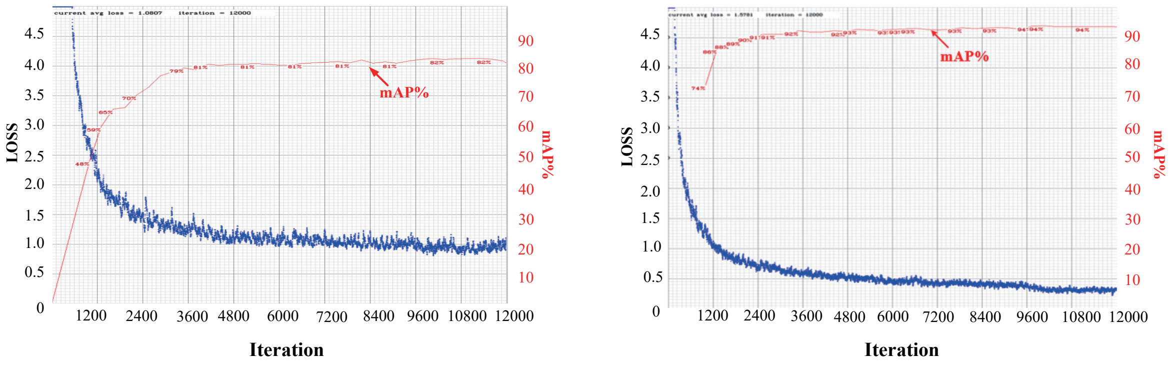
Table 1. Comparison of the performance of each function structure in the modified YOLO

Detection algorithm	Structure	Precision	Recall	mAP (%)	FPS
YOLOv3	Darkenet53 + FPN	0.92	0.79	82.36	53.67
Proposed 1	skip-connection deep residual network + FPN	0.92	0.90	92.26	52.08
Proposed 2	skip-connection deep residual network + multi-scale feature fusion network	0.94	0.91	93.87	49.45

Table 2. Comparison of the performance with classic algorithms

Detection algorithm	AP (%)				mAP (%)	FPS
	Car	Truck	Van	Others		
CornerNet	80.69	81.03	80.31	59.21	75.31	28.85
Faster R-CNN	78.85	78.78	75.43	60.74	73.45	20.05
YOLOv3	89.11	88.68	87.17	64.51	82.36	53.67
YOLOv4	95.92	98.88	98.41	84.87	94.52	30.66
Proposed	95.74	98.83	97.69	83.23	93.87	49.45

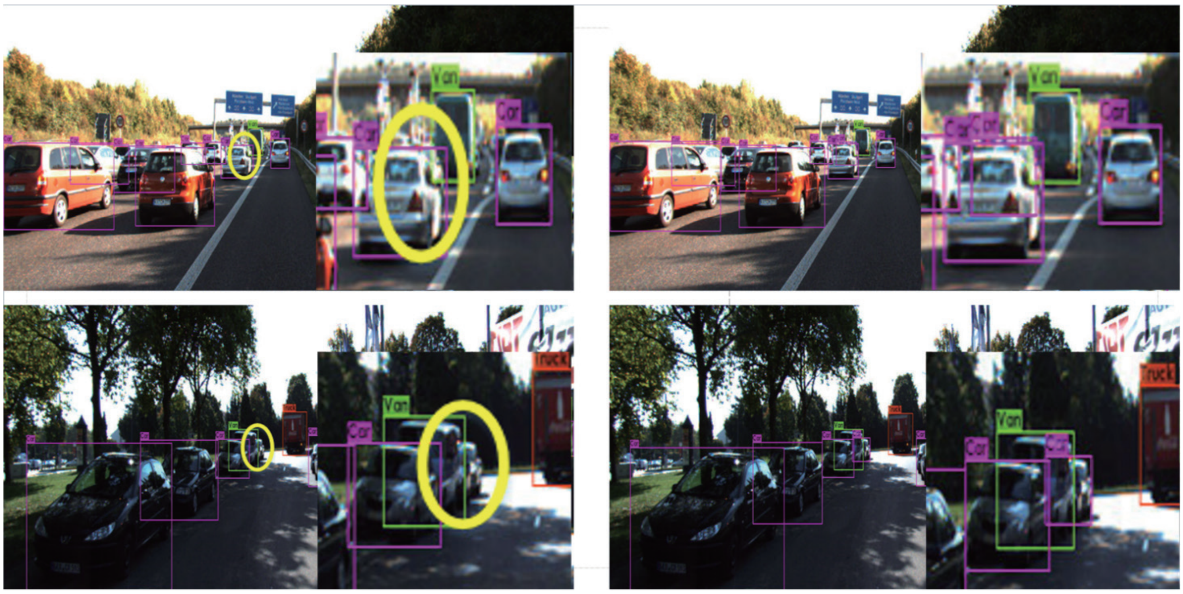
To test the performance of the improved YOLO, a comparative experiment with other classical object detection algorithms is carried out on KITTI dataset. Table 2 lists the comparison of AP values of various detection algorithms for different classes. It can be seen from the table that the mAP of modified YOLO is 11.51% higher than YOLOv3, 20.42% higher than Faster R-CNN, and 18.56% higher than CornerNet. Furthermore, compared with CornerNet, Faster R-CNN, and YOLOv4, the improved YOLO also has a great detection speed advantage.



(a) YOLOv3 (b) Improved YOLO

Fig. 9. Comparison of the loss and mAP obtained by (a) and (b)

To further clearly show that the detection precision of the modified YOLO is higher than that of the original YOLOv3, two groups of experimental results on KITTI dataset are shown in Fig. 10. The detection results show that the original YOLOv3 has missed detection of vehicle objects (as shown in Fig. 10(a)). Compared to the original YOLOv3, the proposed improved YOLO has a lower missed detection rate (as shown in Fig. 10(b)).



(a) YOLO v3 (b) Improved YOLO

Fig. 10. Comparison of the detection results obtained by (a) and (b)
(The red, green, and orange frames correspond to cars, vans and trucks, respectively.
And the content circled in yellow are the objects of a missed detection by YOLO v3.)

4.3 Analysis of the Generalization Ability

To test the generalization ability of the improved YOLO, the optimal algorithm model trained on the KITTI dataset is verified on the self-built vehicle dataset. The detection results are shown in Fig. 11. The detection results show that the occluded vehicles can still be detected by the modified YOLO, which proves that the algorithm has a certain generalization ability. However, the algorithm model is not trained in the self-built vehicle dataset, so the detection performance is not good enough, and there are the cases of missing detection and false detection.

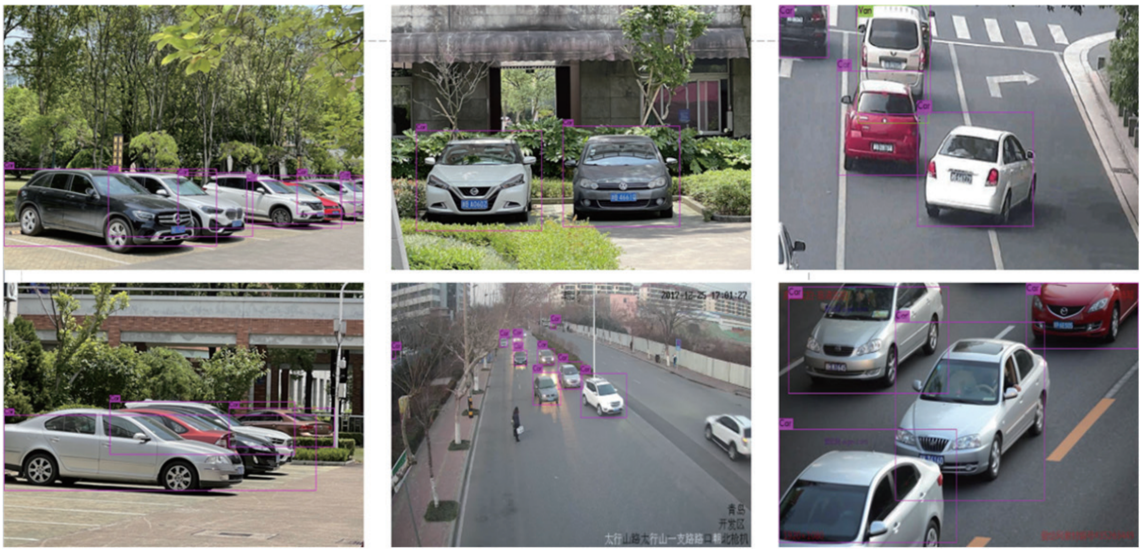


Fig. 11. Test results on self-built vehicle dataset

5 Conclusion

Based on the YOLO series, a novel vehicle object detection algorithm is proposed in this paper, to meet the high efficiency requirements of vehicle detection. Firstly, the feature extraction network is reconstructed with skip-connection deep residual structure, which greatly enhance the integrity and effectiveness of feature extraction, and then improve the detection accuracy of the proposed algorithm. Secondly, the improved depthwise separable convolution is employed to replace the ordinary convolution in the skip-connection deep residual modules, which can reduce the computational complexity, and thus ensuring the real-time performance. Finally, the multi-scale feature fusion network is designed to make full use of the semantic information and location information of different depth network layers, and further improves the detection accuracy. Experimental results prove that the improved algorithm shows high detection accuracyto meet the requirements of real-time detection. In addition, cross-dataset tests prove the generalization ability of the proposed algorithm in the field of vehicle object detection.

Vehicle detection is one of the key techniques of intelligent transportation system with high requirements for accuracy and real-time. In the future work, under the condition of meeting the real-time detection of vehicle objects, we will further improve the detection accuracy of small and medium-sized vehicle objects, to meet higher requirements and contribute to the development of intelligent transportation system.

6 Acknowledgement

This research was funded by the National Natural Science Foundation of China (No. 61671412), Zhejiang Provincial Natural Science Foundation (No. LY19F010002, No. LY21F010014), Natural Science Foundation of Ningbo, China (No. 202003N4323), Ningbo Municipal Projects for Leading and Top Talents (No. NBLJ201801006), School-level Research and Innovation Team of Zhejiang Wanli University, Fundamental Research Funds for Zhejiang Provincial Colleges and Universities, General Scientific Research Project of Zhejiang Education Department (No. Y201941122), and Innovation Training Program.

References

[1] F. Yang, Automobile Fine-Grained Detection Algorithm Based on Multi-Improved YOLOv3 in Smart Streetlights, Algorithms 13(2020) 114.

[2] L. Chen, Deep Neural Network Based Vehicle and Pedestrian Detection for Autonomous Driving: A Survey, IEEE Transactions on Intelligent Transportation Systems 22(2021) 3234-3246.

[3] P. Sharma, H. Liu, A Machine-Learning-Based Data-Centric Misbehavior Detection Model for Internet of Vehicles, IEEE Internet of Things Journal 8(2021) 4991-4999.

[4] R. Girshick, Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation, in: Proc. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014.

[5] R. Girshick, Fast R-CNN, in: Proc. 2015 IEEE International Conference on Computer Vision, 2015.

[6] S.-Q. Ren, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, IEEE Transactions on Pattern Analysis and Machine Intelligence 39(2015) 1137-1149.

[7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, SSD: Single Shot Multi Box Detector, in: Proc. 2016 Europe Conference on Computer Vision, 2016.

[8] H. Law, J. Deng, CornerNet: Detecting Objects as Paired Keypoints, International Journal of Computer Vision 128(3) (2020) 642-656.

[9] J. Redmon, You Only Look Once: Unified, Real-Time Object Detection, in: Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[10]J. Redmon, A. Farhadi, YOLO9000: Better, Faster, Stronger, in: Proc. 2017 IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[11]T.-Y. Lin, Feature Pyramid Networks for Object Detection, in: Proc. 2017 IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[12]J. Redmon, A. Farhadi, YOLOv 3 : An Incremental Improvement, <https://arxiv.org/abs/1804.02767>, 2018 (accessed 18.04.08)

[13]S. Liu, Path Aggregation Network for Instance Segmentation, in: Proc. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[14]B. Alexey, YOLOv4: Optimal Speed and Accuracy of Object Detection, <https://arxiv.org/abs/2004.10934>, 2020 (accessed 20.04.23).

[15]H. Nguyen, Improving Faster R-CNN Framework for Fast Vehicle Detection, Mathematical Problems in Engineering (2019) 1-11.

[16]T. Zhou, Enhance the recognition ability to occlusions and small objects with Robust Faster R-CNN, International Journal of Machine Learning and Cybernetics 10(2019) 3155-3166.

[17]F.-K. Zhang, F. Yang, C. Li, Fast Vehicle Detection Method Based on Improved YOLOv3, Computer Engineering and Applications 55(2)(2019) 12-20.

[18]C.-J. Xu, X.-F. Wang, Y.-D. Yang, Attention-YOLO: YOLO Detection Algorithm That Introduces Attention Mechanism, Computer Engineering and Applications 55(6)(2019) 13-23.

[19]Q.-C. Mao, Finding every car: a traffic surveillance multi-scale vehicle object detection method, Applied Intelligence 50(3)(2020) 1-12.

[20]D. Misra, Mish: A Self Regularized Non-Monotonic Neural Activation Function, <https://arxiv.org/abs/1908.08681>, 2019 (accessed 19.10.02).

[21]V. Nair, E.-H. Geoffrey, Rectified Linear Units Improve Restricted Boltzmann Machines, in: Proc. 2010 International Conference on Machine Learning, 2010.

[22]K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 37(2015) 1904-1916.

[23]K.-M. He, Deep Residual Learning for Image Recognition, in: Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[24]F. Chollet, Xception: Deep Learning with Depthwise Separable Convolutions, in: Proc. 2017 IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[25]J. MacQueen, Some methods for classification and analysis of multivariate observations, in: Proc. 1967 Berkeley Symposium on Mathematical Statistics and Probability, 1967.

[26]D. Arthur, V. Sergei, k-means++: the advantages of careful seeding, in: Proc. 2007 Annual ACM-SIAM Symposium on Discrete Algorithms, 2007.

[27]G. Andreas, Vision meets robotics: The KITTI dataset, The International Journal of Robotics Research 32(2013) 1231-1237.