Real-time vehicle detection and computer intelligent recognition through improved YOLOv4

Zhenhao Ni\textsuperscript{1,2}, Tingna Liu\textsuperscript{1,\ast}, Ke Li\textsuperscript{1,2}, Yongqiang Bai\textsuperscript{1}, Zhongjie Zhu\textsuperscript{1}

\textsuperscript{1}Ningbo Key Lab of DSP, Zhejiang Wanli University, Ningbo 315000, China
\textsuperscript{2}Zhejiang CRRC Electric Vehicle Co., LTD, Ningbo 315100, China

\textsuperscript{\ast}Corresponding author: 2019882017@zwu.edu.cn

Abstract. Vehicle detection is one of the key techniques of intelligent transportation system with high requirements for real-time and accuracy. To better balance the requirements, a vehicle detection algorithm based on the You Only Look Once (YOLO) v4 is proposed in this paper. On the one hand, the improved depthwise separable convolution is adopted to ensure the real-time performance. On the other hand, a novel feature fusion network is designed to gather more original feature information of different depth network layer. Experimental results show that the proposed algorithm can reduce the detection time by half while ensuring the accuracy, compared with the pristine YOLOv4.

Keywords: YOLOv4, deep learning, object detection.

1. Introduction

Vehicle object detection is an important branch of computer vision and the basis of intelligent transportation, vehicle tracking and other fields. Accurately detecting vehicle objects in real time has always been a research focus of interdisciplinary computer vision and transportation research, which attracts much attention from scholars at home and abroad.

Currently, vehicle detection algorithms can be divided into two categories, i.e., traditional algorithms and deep learning-based algorithms. Traditional algorithms adopt the similar exhaustive sliding window method or image segmentation technology to generate a large number of candidate regions, and then extract image features from each candidate region a classifier to determine the category of the candidate region. However, the traditional algorithms have the limitations of high computational complexity and low detection accuracy. The deep learning-based algorithms overcome these above shortcomings. There are two categories of object detection algorithms based on deep learning: object detection algorithms based on candidate regions and object detection algorithms based on regression. The object detection algorithms based on candidate regions, also known as two-stage algorithm, divide the object detection problem into two phases: one is to generate a region proposal, the other is to put the candidate regions into the classifier for classification and location correction. Girshick et al. successfully applied Convolutional Neural Networks (CNN) in the field of object detection, proposed R-CNN [1] algorithm. He et al. proposed the Spatial Pyramid Pooling Network (SPP-Net) [2] detection algorithm. Girshick et al. proposed Fast R-CNN [3] algorithm, which they received inspired by the SPP-Net algorithm, the SPP layer is simplified into a single scale ROI-
Pooling layer to unify the size of the candidate area features. Ren et al. proposed the Faster R-CNN [4] algorithm and designed RPN substitution selective search algorithm to assist in sample generation. The advantage of the above algorithms is that the image features can be fully extracted to achieve accurate classification and location. However, two-stage structure has the disadvantages of a slow running speed and low efficiency. The object detection algorithms based on regression directly transform the problem of object classification and location into a regression problem. Joseph et al. proposed the You Only Look Once (YOLO) series algorithms [5-7], each has a simple network structure and higher real-time performance.

YOLOv4 [8] integrates various advanced technologies based on YOLOv3 and provides the good trade-off in terms of speed and accuracy. We have conducted in-depth research on the various sub-modules of YOLOv4, so that the improved algorithm can accurately recognize the vehicles in the images in real time, and its performance is better than that of YOLOv4.

2. Proposed Algorithm

2.1. YOLOv4

YOLOv4 is the fourth iteration of YOLO. Compared with YOLOv3, YOLOv4 has better detection performance for occluded and overlapped objects. Since the main difficulty of vehicle detection is the overlapping and occlusion of vehicles, YOLOv4 is more suitable as the vehicle detection algorithm embedded in vehicle detection than YOLOv3. The network structure of YOLOv4 is shown in Fig. 1. The backbone network of YOLOv4 is based on the experience of YOLOv3 backbone network and cross-stage partial network (CSPNet) [9]. CSPNet solves the problem of gradient information duplication in the backbone of other large-scale convolutional neural network frameworks, and integrates gradient changes into the feature map from beginning to end. Therefore, the parameters and FLOPS values of the module are reduced, which not only ensures the speed and accuracy of inference, but also reduces the module size. The Neck module is mainly used to generate feature pyramids, enhance the module's ability to detect targets of different scales, and identify the same target of different scales and sizes. Based on the idea of Path Aggregation Network for Instance Segmentation (PANET) [10] and the framework of Mask R-CNN [11] and FPN [12], the Neck of YOLOv4 enhances information dissemination. YOLOv4 uses YOLOv3's Head in the Head module. It applies anchor on the feature map and generates the final output vector with class probabilities, object values, and bounding boxes. These feature vectors with different scales are used to detect objects of different sizes.

2.2. Optimization of YOLOv4

To balance the detection speed and detection accuracy, the convolution process and feature fusion network are optimized. The structure of the proposed algorithm is shown in Fig. 2.

Convolution layer is an important part of the whole neural network. However, there are too many parameters, and the calculation is complex in traditional convolution. To speed up the detection speed and reduce the computational cost of the vehicle detection algorithm, depthwise separable convolutions are used to replace the traditional convolutions in residual unit of the backbone network. A depthwise separable convolution, commonly called “separable convolution” in deep learning frameworks such as TensorFlow and Keras, consists in a depthwise convolution, i.e. a spatial convolution performed independently over each channel of an input, followed by a pointwise convolution, i.e. a 1×1 convolution, projecting the channels output by the depthwise convolution onto a new channel space [13]. The structure comparison is shown in Fig. 3. And using the Mish activation function makes the information feature penetrate deeper into the neural network, so the smallest component of the proposed algorithm is constituted by Darknet conv2d + BN + Mish[14] (CBM) units. The comparison of Leaky ReLU activation function and Mish activation function is show in Fig. 4.
To fully utilize the information features extracted from the backbone network, before generating the feature pyramid, the features of both layers are fused together by downsampling to gather more original information features. As shown in Fig. 5 (a), the Neck of the YOLOv4 network structure uses...
The PANET structure to fuse the deep feature information with the shallow feature information. Multiscale location and category prediction is carried out by fusing the semantic information obtained from high-level sampling and the low-level location information. The top-down structure for transferring strong semantic features from the high-level to enhance the whole pyramid. And the bottom-up pyramid network is added after the top-down structure, so the strong low-level location features are transferred, information dissemination is enhanced. As shown in Fig. 5 (b), before the neck is generated, the features of each two layers are fused together by downsampling to collect more semantic feature information and location feature information and enhance the local features and features between the global and local domains.

Figure 3. Comparison of (a) the ordinary convolution and (b) depthwise separable convolution in residual unit.

Figure 4. Comparison of (a) Leaky ReLU activation function and (b) Mish activation function.
The Neck in YOLO v4 The Neck in proposed algorithm

(a)                                                                               (b)

Figure 5. Comparison of (a) the PANET and (b) the proposed feature fusion network.

3. Experiments Results and Analysis

The experimental conditions are as follows: CPU: Intel(R) Core(TM) i9-9920X@3.5GHz. RAM: 16G. GPU: NVIDIA GeForce RTX2080 Ti.

3.1. Experimental Configuration

In general, the dataset used to train neural networks must involve at least thousands of images. An excessively small dataset may have adverse effects on the accuracy and reliability of detection. To make the dataset more consistent with the vehicles in natural scenes, 5000 relevant images are collected through online collection and offline shooting, including 3500 as the training set and 1500 as the test set. Our self-build dataset covers three common types of vehicles: cars, trucks and buses, and it reflects various complex situations, such as multiple trucks and cars in the same image, or multiple trucks, buses and cars in the same image. The images in our self-build dataset include different lighting conditions, vehicle shooting angles, resolutions, and road environments. Fig. 6 shows some of the images in our dataset.

Figure 6. Partial samples in our self-built dataset.

In the training phase, small batch random gradient descent method is used for optimization. The momentum parameter of the algorithm is set to 0.9, the batch size is 64, and the initial learning rate is set to 0.001.

The evaluation indexes selected in this paper are precision, recall, mean average precision (mAP) and detection time. Average precision (AP) is the average of precision values on the precision recall (PR) curve, also, it is the area enclosed by the PR curve and the coordinate axis. The mAP is the average value of AP for each category.
\[ \text{Precision} = \frac{TP}{TP + FP} \]  
\[ \text{Recall} = \frac{TP}{TP + FN} \]

Where true positive (TP) is the number of positive samples correctly identified as positive samples, false negative (FN) is the number of positive samples wrongly identified as negative samples, and false positive (FP) is the number of negative samples wrongly identified as positive samples. Therefore, higher values of precision and recall mean better performance of the algorithm.

### 3.2. Comparison of the experimental results

Table 1 lists the comparison of the data obtained by YOLOv4 and the proposed improved YOLOv4. The important data with the number of iterations ranging from 7000 to 12,000 are introduced. These data include precision, recall and mAP and show the advantages of the improved YOLOv4. As shown in Table 1, under the same experimental environment, YOLOv4 obtained the best detection result with a precision of 92.00%, recall of 91.00%, and mAP of 94.99%. And the improved YOLOv4 obtained the best detection result with a precision of 93.00%, recall of 91.00%, and mAP of 95.26%. The experimental results show that the improved YOLOv4 is superior to the YOLOv4, the mAP of the improved YOLOv4 is higher than that of the original YOLOv4. The red mAP curve shown in Fig. 7 indicates that the mAP of the improved YOLOv4 is higher than that of the original YOLOv4.

| Number of Iterations | YOLOv4 Precision | YOLOv4 Recall | YOLOv4 mAP (%) | Proposed Precision | Proposed Recall | Proposed mAP (%) |
|----------------------|------------------|---------------|----------------|-------------------|-----------------|------------------|
| 7000                 | 0.92             | 0.90          | 94.23          | 0.89              | 0.93            | 93.89            |
| 8000                 | 0.87             | 0.94          | 92.86          | 0.92              | 0.91            | 94.56            |
| 9000                 | 0.87             | 0.93          | 94.63          | 0.91              | 0.92            | 94.79            |
| 10,000               | 0.92             | 0.91          | 94.48          | 0.93              | 0.91            | 94.42            |
| 11,000               | 0.92             | 0.89          | 93.99          | 0.92              | 0.92            | 94.05            |
| 12,000               | 0.94             | 0.89          | 94.40          | 0.93              | 0.91            | 94.49            |
| best                 | 0.92             | 0.91          | 94.99          | 0.93              | 0.91            | 95.26            |

Table 2 lists the AP values of the three classes of test objects. The improved YOLO v4 has a higher mAP, and the AP values of all three objects are greater than YOLO v4. The results show that the improvements of the YOLO v4 in this paper is successful, and these improvements are not limited to the improvement of single-class object detection.

| Detection algorithm | AP (%) | mAP (%) |
|---------------------|--------|---------|
| YOLOv4              | 94.36  | 94.88   |
| Proposed            | 94.41  | 95.51   | 95.86 | 95.26 |
Figure 7. Comparison of the loss and mAP obtained by (a) original YOLOv4 and (b) improved YOLOv4.

3.3. Performance comparison of the improvement parts

This paper considers two indicators to verify the performance of vehicle detection algorithms: detection speed and detection accuracy. To speed up the detection speed, need to change the convolution process in the algorithm. To improve the detection accuracy, need to make better use of the features extracted from the backbone network.

Table 3 Comparison of the performance of each structure in the algorithm.

| Detection algorithm  | Structure                        | Precision | Recall | mAP(%) | Total BFLOPS | Average detection time (ms/frame) |
|----------------------|----------------------------------|-----------|--------|--------|--------------|----------------------------------|
| YOLOv4               | CSPDarkenet53+PANET              | 0.92      | 0.91   | 94.99  | 127.263      | 56.71                            |
| Variant of YOLOv4    | CSPDarkenet53(Depthwise-Conv)+PANET | 0.92      | 0.90   | 94.46  | 90.875       | 25.91                            |
| Proposed             | CSPDarkenet53(Depthwise-Conv)+PANET(improved) | 0.93      | 0.91   | 95.26  | 91.679       | 26.49                            |

To prove that each part of the improved YOLOv4 network is effective, ablation experiments are conducted. Table 3 shows that modifying the convolution process in the network structure can reduce parameters and reduce computational complexity. And making better use of the features extracted from the backbone network can improve the detection accuracy of the algorithm. As shown in Table 3, Traditional convolution is replaced by depthwise separable convolution, therefore, the number of parameters and the complexity of the algorithm are reduced. The total billion float operations per second (BFLOPS) is reduced from 127.263 to 90.875; the average detection time per frame is reduced from 56.71ms to 25.91ms; that is, the detection time is reduced by half, thus ensuring the detection speed of the algorithm. Semantic information and location information of different depth network layers are fused, thus improving the mAP value of the algorithm, however, the BFLOPS increases.
only by 0.804. Therefore, the improvement of the convolution process and the optimization of the network structure greatly improve the detection performance of the vehicle detection algorithm.

To further clearly show that the detection accuracy of the improved YOLOv4 is higher than that of the original YOLOv4, two sets of experimental results are shown in Fig. 8. The detection results show that the original YOLOv4 missed the detection objects (as shown in Fig. 8(a)). Compared with the original YOLOv4, the improved YOLOv4 has higher accuracy rate (as shown in Fig. 8(b)).

![Figure 8. Comparison of the detection results obtained by (a) YOLOv4 and (b) improved YOLOv4. The red, yellow and green frames correspond to cars, buses and trucks, respectively. The blue circle is the object missed by YOLOv4.](image)

4. Conclusion

Based on the YOLOv4, a novel vehicle detection algorithm is proposed in this paper to meet the real-time and high accuracy requirements of vehicle detection. Firstly, depthwise separable convolutions are employed to replace the ordinary convolutions in the algorithm, which reduces the computational complexity, and thus ensuring the real-time performance. Secondly, the feature fusion network is designed to gather semantic information and location information of different depth network layers, which enhances the local features and features between the global and local domains, enhances the integrity and effectiveness of feature extraction, and thus improving the detection accuracy of the proposed algorithm. The experimental results show that the proposed algorithm has better detection accuracy and speed.

Acknowledgments

This work was funded by the National Natural Science Foundation of China (No. 61671412), Zhejiang Provincial Natural Science Foundation of China (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, and General Scientific Research Project of Zhejiang Education Department (No. Y201941122).
References

[1] R. Girshick et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. CVPR IEEE (2014).

[2] K. M. He et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 37: 1904-1916 (2015).

[3] R. Girshick. Fast R-CNN. 2015 IEEE International Conference on Computer Vision (ICCV): 1440-1448 (2015).

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

[5] R. Joseph et al. You Only Look Once: Unified, Real-Time Object Detection. Computer Vision & Pattern Recognition IEEE (2016).

[6] R. Joseph and A. Farhadi. YOLO9000: Better, Faster, Stronger. IEEE Conference on Computer Vision & Pattern Recognition IEEE: 6517-6525 (2017).

[7] R. Joseph and A. Farhadi. YOLOv3: An Incremental Improvement. CoRR (2018).

[8] B. Alexey et al. YOLOv4: Optimal Speed and Accuracy of Object Detection. ArXiv abs/2004.10934: n. pag (2020).

[9] C. Y. Wang et al. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW): 1571-1580 (2020).

[10] S. Liu et al. Path Aggregation Network for Instance Segmentation. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition: 8759-8768 (2018).

[11] K. M. He et al., Mask R-CNN. 2017 IEEE International Conference on Computer Vision (ICCV): 2980-2988 (2017).

[12] T. Y. Lin et al., Feature Pyramid Networks for Object Detection. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR): 936-944 (2017).

[13] F. Chollet, Xception: Deep Learning with Depthwise Separable Convolutions. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR): 1800-1807 (2017).

[14] D. Misra, Mish: A Self Regularized Non-Monotonic Neural Activation Function. ArXiv abs/1908.08681 (2019): n. pag.