Vehicle Detection Based on Improved YOLO-LITE Algorithm

Shuai Zhao[a], Fucheng You[b] and Shaomei Wang[c]

Beijing Institute of Graphic Communication No.1, Xinghua Street (two section), Daxing District, Beijing, China

[a]17839192463@163.com, [b]13245649069@163.com, [c]1437128206@qq.com

Keywords: YOLO-LITE, YOLO, deep learning, vehicle detection

Abstract: With the increasing traffic congestion in urban roads, people have higher and higher requirements for real-time monitoring of vehicles, but the traditional vehicle detection algorithms are too demanding on computer hardware. Like the YOLOv3 algorithm, although the performance aspect is more objective, it requires high hardware equipment. Therefore, based on such problems, the YOLO-LITE algorithm is proposed. This algorithm achieves the goal of introducing vehicle detection into a non-GPU computer, and experiments have shown that the YOLO-LITE algorithm is not satisfactory in terms of mAP, but the FPS aspect is very good, and the YOLO-LITE algorithm is 8.8 times faster than Tiny-YOLO algorithm.it is very helpful for the lack of GPU computers to achieve vehicle detection.

1. Introduction

With the development of urban road traffic, vehicle detection has received extensive attention in the field of intelligent transportation and intelligent driving. In

recent years, computer vision-based vehicle detection has received more and more attention, Compared with previous detection methods, computer vision-based vehicle detection can provide more information. With the rapid development of deep learning, various vehicle detection algorithms have been proposed, and gradually replaced the past detection methods. Convolutional neural networks [1-2] are a very important deep learning method, which can automatically extract features from video or images and is widely used in computer vision.

With the opening of the proposition based on the deep learning target detection algorithm, the R-CNN [3] algorithm becomes a rookie in the field of deep learning target detection. The algorithm uses the selective search method in the process of generating candidate regions, which creates a new idea of deep learning target detection based on candidate regions for the subsequent emergence of SPP-net [4], Fast R-CNN [5], Faster R CNN [6], R FCN [7]. Among them, R-CNN uses convolutional neural network to extract the target image features, and uses the selective search method [8] to extract the suggestion box, which is greatly improved compared with the traditional machine learning method. However, R-CNN also has some disadvantages, such as fixed size image input, cumbersome training and slow speed. Subsequent SPP-Net and Fast R-CNN have solved the problems of slow speed and tedious training to different extents, but they cannot be monitored in real time. Even though the R-FCN, which has a good performance ratio, has improved, it still cannot be monitored in real time. It still faces huge challenges in real-time detection.

Redmond J proposed the YOLO [9] detection algorithm in 2015.YOLO is an end-to-end detection algorithm. Although it belongs to CNN, YOLO can complete the detection task directly and quickly in the detection process. YOLO is able to guarantee both accuracy and detection rate. The standard YOLO can detect 45 images per second, while the Fast YOLO detection speed reaches 155f/s. YOLOv2[10] optimizes the model structure of YOLO, significantly improves the detection speed of the target, and the detection accuracy is the same as the SSD, but the network of YOLOv2 is simple and does not improve the detection accuracy. YOLOv3 [11] uses the depth residual network to extract image features and achieve multi-scale prediction, and obtains the best balance between detection accuracy and speed. Although the YOLO algorithm has good performance for target

detection, the performance requirements of external computer equipment are relatively high, which is difficult for many beginners to accept, so it is necessary to find an algorithm that requires low computer requirements.

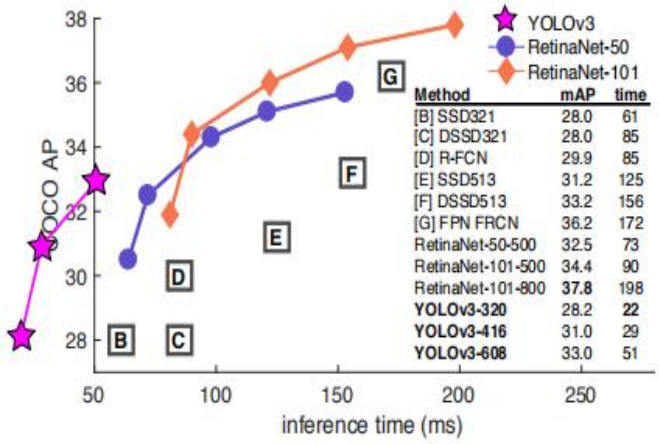
2. Algorithm introduction

2.1 algorithmic thinking

The basic idea of the YOLO-LITE algorithm is consistent with the YOLO algorithm. Firstly, the feature extraction network is used to extract the features of the target image, and the feature map of the corresponding size is obtained, the common size is 13*13.Then the input target image is divided into 13*13 grid cells, and the bounding box predicted by each grid cell is different. The bounding box predicted in the YOLO algorithm and the YOLOv2 algorithm is 2; the bounding box predicted in the YOLOv3 algorithm is 3. Then, when the center coordinate of a target falls into which grid cell, the target cell is predicted by the grid cell. Only the largest bounding box of the IOU in these corresponding bounding boxes is used to predict the target. On the other hand, each grid cell outputs an array of B*5+C dimensions, where C refers to the number of categories and B refers to the number of prediction boxes, such as S=7, B=2 in YOLO algorithm, 20 classes in PASCAL VOC data set, so C=20, B*5+C=30.

2.2 algorithm advantages

The YOLO algorithm is a regression-based detection method, which is a new end-to-end algorithm that can quickly complete detection tasks. Especially in the YOLOv3 algorithm, the deep residual network is used, which greatly reduces the time consumption compared with the past, and the experimental results show that the YOLO algorithm greatly improves the accuracy of target detection. Figure 1 below is a comparison of YOLOv3 with other algorithms:



Method	mAP	time
[B] SSD321	28.0	61
[C] DSSD321	28.0	85
[D] R-FCN	29.9	85
[E] SSD513	31.2	125
[F] DSSD513	33.2	156
[G] FPN FRCN	36.2	172
RetinaNet-50-500	32.5	73
RetinaNet-101-500	34.4	90
RetinaNet-101-800	**37.8**	198
YOLOv3-320	28.2	**22**
YOLOv3-416	31.0	29
YOLOv3-608	33.0	51

Figure 1. Algorithm comparison result

3. YOLO-LITE algorithm principle

3.1 Network structure

The network structure in the YOLO algorithm is roughly divided into two parts, the first part is used for feature extraction network, mainly for extracting the general features of the target; the second part is used for post-processing network, which is mainly used to regress the coordinates and categories of the objects to be detected. In YOLO-LITE algorithm, only 7 convolution layers are used, totaling 749 filters. Compared with the 9 convolution layers of Tiny-YOLOv2, a total of 3181 filters are much simpler. The overall structure of YOLO-LITE is similar to YOLOv2. It uses a universal CNN for vehicle detection. Its activation function is leaky reluctant, and the maximum pooling in the YOLOv2 algorithm is retained to adjust the size of the tensor. The following Figure 2 is the network structure of the YOLO-LITE algorithm:

Figure 2. The network structure

3.2 coordinate prediction

The way of predicting the coordinates of the bounding box continues the practice in YOLOv2, where the output predicted by the model is tx, ty, tw, th, and the coordinates of the grid cell are represented by cx and cy. For example, if the size of a convolution layer's characteristic graph is 13*13, so its grid cell has 13*13, then the coordinate cx of the grid cell in the first column of the second row is equal to 2, and cy is equal to 1.pw and ph represent the size of the predicted bounding box. bx, by, bx and bx are the coordinates and dimensions of the center of the predicted bounding box. The specific calculation formula is as follows and as shown in the Figure 3 and Figure 4:

$$b_x = \sigma(t_x) + c_x$$
$$b_y = \sigma(t_y) + c_y$$
$$b_w = p_w e^{t_w}$$
$$b_h = p_h e^{t_h}$$

Figure 3. Calculation formula

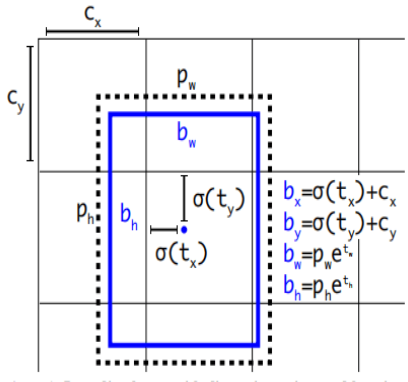


Figure 4. Calculation diagram

3.3 Loss function

In the YOLO-LITE algorithm, the target detection is learned as a regression problem, so the mean square error function is used, but different weight values are used for different parts. Generally, it can be divided into two categories, ie positioning error and first, a larger weight is used, and then it distinguishes between the bounding box that does not contain the target and the confidence of the bounding box containing the target. For the former, a smaller weight value is used, and other weight

the values are set to 1. Then use the mean square error, which treats the bounding boxes of different sizes equally. The specific calculation formula of the loss function is shown in Figure 5.

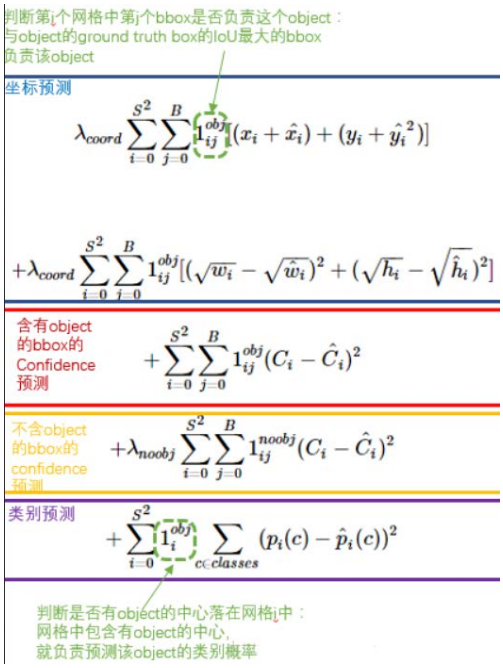


Figure 5. Loss function

4. Algorithm improvement

4.1 Change input size

YOLO-LITE algorithm replaces the input size from 416 * 416 to 224 * 224, changing the size of the input, so that the amount of FLOP also changes, the smaller the input size, the smaller the amount of FLOP required.

4.2 Simplified convolution layer

YOLO-LITE algorithm uses only 7 convolutional layers, fewer layers and a shallower network; simple and simple, which greatly reduces the number of weights.

4.3 Remove the BN layer

Remove the BN layer. Yolo-tiny retains the BN layer, while YOLO-LITE completely removes the BN layer. The BN layer can speed up convergence during training and improve training performance, but the BN layer adds a lot of computational overhead.

5. Analysis of experimental results

5.1 Experimental data set

The data set used in the experiment was created from aerial images, and the vehicle targets in the images were labeled using LableImg. There are a total of 2,500 images, and 2,000 of them are randomly selected as training sets and 500 images are used as test sets.

5.2 Experimental platform

The hardware configuration processor is a 32G CPU (E5-2609), the software development environment is Windows10 + CUDA 9.1 + CUDNN7, and the application language is mainly C++.

5.3 Experimental results

By comparing the original YOLO-LITE algorithm with the original Tiny-YOLOv2 algorithm, the speed has been greatly improved, and the speed of the YOLO-LITE algorithm is 8.8 times that of Tiny-YOLOv2.The detection effect of the YOLO-LITE algorithm is shown in Figure 6, 7, 8:

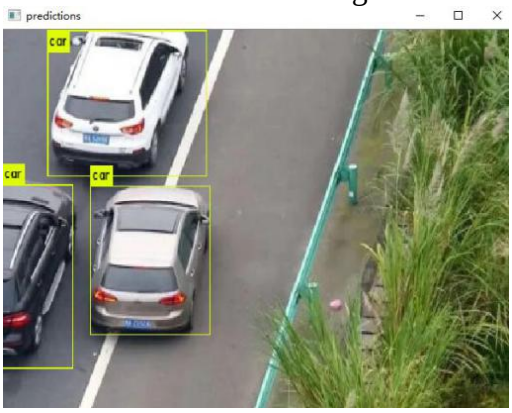


Figure 6. Detection effect

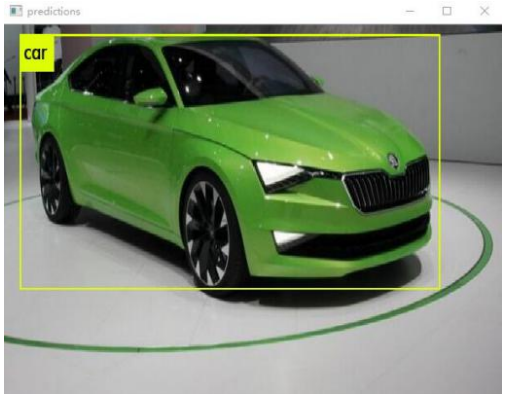


Figure7. Detection effect

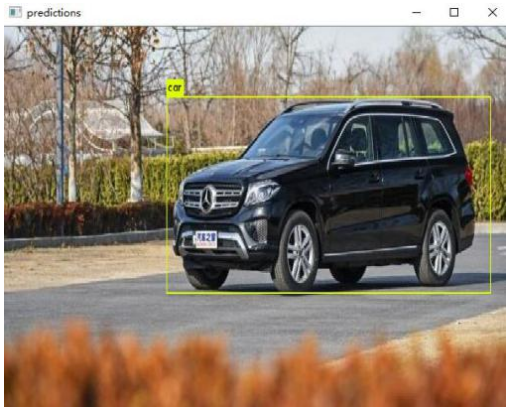


Figure 8. Detection effect

Acknowledgments

This work was partially supported by Joint Funding Project of Beijing Municipal Commission of Education and Beijing Natural Science Fund Committee (KZ201710015010), Project of National Scientific Found (No.61370188), Project of Beijing Municipal College Improvement Plan (PXM2017_014223_000063) and New Project of Green Printing and Publishing Technology by Cooperative Creating Center (PXM_014223_000025) and BIGC Project (Ec201803 Ed201802 Ea201806).

References

[1] BENSRHAIR A, BERTOZZI M, BROGGI A, etal, A cooperative approach to vision-based vehicle detection [C] // Intelligent Transportation Systems. IEEE, 2001: 207-212.

[2] Linjiang Xie, Guishu Ji, Qing Peng, et al. Application of Improved Convolutional Neural Network in Pedestrian Detection [J]. Computer Science and Exploration, 2018, 12 (5): 708-718.

[3] Girshick R, Donahue J, Darrell T, et alRich feature hier-archies for accurate object detection and semantic seg-mentation [C] //CVPR, 2014:580-587.

[4] He K M, Zhang X Y, Ren S Q,et al.Spatial pyramid pooling in deep convolutional networks for visual rec-ognition [J]. arXiv: 1406.4729v4, 2015.

[5] Girshick R .Fast R_CNN [C] //ICCV, 2015: 1440_1448.

[6] Ren S Q, He K M, Girshick R, et al.Faster R-CNN: towards real- time object detection with region proposal networks [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39 (6): 1137-1149.

[7] Dai J F, He K M, Sun J.R-FCN; object detection via region-based fully convolutional networks [J]. arXiv: 1605.06409, 2016.

[8] Uijlings J R R,Sande K E A V D,Gevers T,et al.Selec-tive search for object recognition [J].International Journal of Computer Vision ,2013, 104 (2): 154-171.

[9] Redmon J, Divvala S, Girshick R, et al.You only look once: unified, real-time object detection [C]// CVPR, 2016.

[10] Redmon J, Farhadi A.YOLO9000; better, faster, stronger [J]. arXiv: 1612.08242v1, 2016.

[11] Redmon J, Farhadi A.YOLOv3: an incremental improve-ment [J], arXiv: 1804.02767v1, 2018.