Vehicle Detection in Complex Scenes based on YOLOV5

Hongbo Zhang[1] and Junhan Mu[2, *]

[1]Department of Tianjin University of Science & Technology, Tianjin, China

[2]School of Northeastern University, Qinhuangdao, China

*Corresponding author: 1911411135@mail.sit.edu.cn

Abstract. The purpose of the algorithm is to effectively detect vehicles in different conditions, especially in scenes recorded by vehicle recorders. The proposed method is designed to handle complex scenarios with high accuracy and efficiency. Through the use of YOLOV5, the algorithm is able to identify and locate vehicles despite occlusions, density, and low lighting. The intermediate hidden layer's activation function is achieved by using the Leaky ReLU, and data enhancement strategies are utilized to enhance detection performance. By recognizing 100 images of traffic vehicles in complex scenes, the validation results show that the recognition rate of this recognition method is 78.87%, 89.11% and 91.88% for occluded vehicles, crowded vehicles and independent vehicles, respectively. Furthermore, the recognition speed reached 0.012 s/a, which was reduced by 0.005s compared to the original time. All the results demonstrate that the proposed algorithm has a high recognition rate and real-time speed for the complex scenes, indicating that the convolutional neural network has a promising future in the vehicle detection of complex application scenes.

Keywords: Complex scenes; Vehicle detection; YOLOV5; Deep learning.

1. Introduction

China is a country with a large population base and many automobiles. With the continuous development and progress of technology, the use of artificial intelligence for vehicle detection is more significant and practical. Since the detection will appear in a variety of complex scenarios, including dark nights, rainy days, or multi-vehicle with overlapping. Therefore, accurate detection of vehicles in complex environments is a key problem to be solved, which is also the bottleneck of limiting the development of artificial intelligence algorithms in the application of vehicle detection. In recent years, researchers have conducted in-depth studies on the accurate detection of vehicles in complex environments in recent years[1]. Li et al. [2] developed a method to extract the features of shadows beneath the vehicles using the Histogram ofed Gradients (HOG) technique. They then trained a classifier using Support Vector Machines (SVM) to eliminate the interference caused by shadows in vehicle recognition. Shi et al. [3] combines the Faster RCNN and VGG-16 to carry the vehicle detection, which can realize the part of the layer sharing as well as the weight sharing to reduce the training time significantly. However, the training dataset ignores the existence of complex scenes and the condition of vehicle occlusion, which lacks generalization. Ye et al. [4] proposes a dataset for lane line detection, determining the road area of interest and labeling the oncoming vehicles. The dataset is utilized to train another convolutional neural network to adapt it to the vehicle binary classification recognition task, which has a good accuracy in complex environments while is time-consuming since it needs a large number of images in the preliminary stage. Wang et al. [5] projects the vehicle features onto other relevant feature spaces to generate a principal component analysis (PCA) weight vector and an independent component analysis (ICA) coefficient vector, which are adopted to estimate the vehicle jointly. This method greatly reduces the amount of computation and improves the accuracy, while there is less training for complex scenes such as black night.

The rapid advancement of deep target detection algorithms has led to the growing popularity of single-stage detection techniques like YOV5, which is currently emerging as the preferred choice for many applications. The YOLOV5 model is a superior method in terms of detection speed, recognition accuracy, and data characterization ability for images, compared to existing methods. The YOLOV5 model's application to vehicle detection has been the subject of research by many scholars.

Nevertheless, the investigation of the effectiveness of these algorithms in complex scenes is still at an early stage. To address these challenges, this study aims to enhance the dataset by including diverse car images captured in various complex scenes. This augmentation allows for more recognition in the presence of different complex scenarios, greatly enhancing the generalization capability of vehicle detection. In order to enhance the data, we converted some images into black and white images and mixed them into the dataset to improve accuracy and speed of recognition. Upon completion of training, it was discovered that the speed of vehicle detection was slightly improved and the accuracy of vehicle detection was higher.

2. Methods

2.1. Original Image Acquisition

Selected traffic monitoring videos are used to collect the original images for model training and testing. A total of 700 original images were collected, 350 for each of the two different time periods: daytime and nighttime on a sunny day, in JPEG format and with a resolution of 2,352×1,568 pixels, as shown in Figure 1.



(a) Daytime (b) Nighttime

Figure. 1 Picture of vehicles in the urban transportation environment

2.2. Complex Scenes Selection

The experiment selects different traffic scenes as the application scenarios for testing, where the traffic video is taken through the car recorder with a driving perspective, from which the images are intercepted in various scenarios including dense vehicles, night driving, etc. These scenarios may have an impact on the detection accuracy of the model. This paper randomly selects 600 (300 images each during the daytime and nighttime) images and intercepts the region with complete vehicles as the target region. The invalid image regions will be removed and then manually screen the original collected images thus avoiding the wrong selection and singularity of data samples. The final experimental data consists of positive samples (5000 images) and negative samples (3750 images), evenly distributed over two time periods (day and night) [6]. The datasets are all used for the training of convolutional neural network and parameter optimization validation, respectively. 80% of the samples are randomly selected from the positive and negative samples to construct the training set, and 20% is used as the validation set.

This paper categorizes the targets in the image captured by the surveillance into three types based on the completeness of their contours: The first type is referred to as a blocked target, which has incomplete contours due to obstruction of part of its area, as shown in Figure 2a. The second type is referred to as a neighboring target, which has two or more connected contours, as shown in Figure 2b. The third type is called an independent target, which has complete and separate contours of the target, as shown in Figure 2c [7]. It should be noted that these targets are not all independent of each other.



(a) Occluded vehicle (b) Adjacent vehicle (c) Independent vehicle

Figure. 2 Examples of vehicles under different complex scenes

After the model training is completed the remaining 100 original vehicle images (50 each for daytime and nighttime) are used as a test set for model effectiveness validation. Finally, this paper will be compared and analyzed with the existing vehicle recognition methods. Since the number of samples in two different time periods in the test data set has a balanced distribution, the average accuracy of the test results can be used as the evaluation index of the recognition effect of the model in this paper.

2.3. Construction of Vehicle Detection Model

The YOLOV5 algorithm is adopted as the baseline method of vehicle detection, which is an object detection algorithm based on convolutional neural networks and its principle can be simply summarized in the following steps. As shown in Figgure 3, the original image is fed into a convolutional neural network to obtain feature maps of different sizes through multi-layer convolution and down-sampling operations. After the last convolutional layer, a full layer is introduced to map the feature map into a fixed-size feature vector. For each grid cell, a set of Bounding Boxes is predicted along with the class of objects within each box. Here each bounding box contains four parameters: the coordinates of the center point of the bounding box, the width of the bounding box, the height of the bounding box, and the probability of the presence of an object. Each bounding box corresponds to the probability of an object category. A final object detection result is obtained by synthesizing the feature maps of the different layers. This process usually includes a Non-Maximum Suppression operation to find the box with the highest confidence level among all the detection results and to eliminate other highly overlapping bounding boxes.

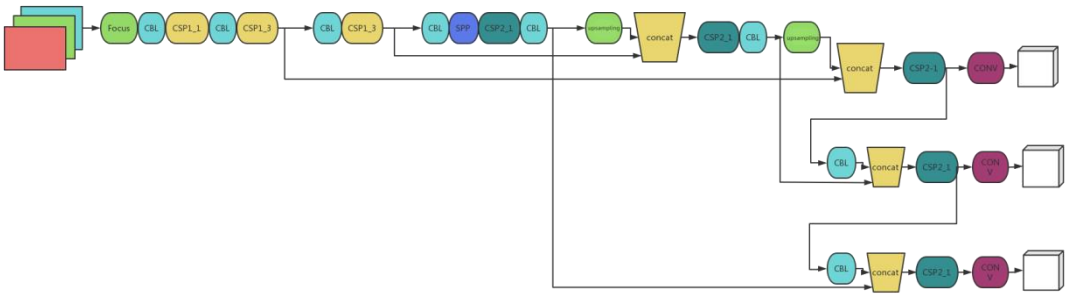


Figure. 3 The framework of proposed vehicle detection algorithm

2.4. Data Enhancement

For detection tasks, RGB images are commonly used as inputs for processing. However, the color information of some targets in scenarios, such as dark night or foggy sky, may negatively affect the detection performance. In such cases, shape and other features become more important for achieving the detection goal. To alleviate this issue, grayscale conversion can be applied to the images by using the "cvtColor" feature in OpenCV. Furthermore, rotation and mirror flip can be used to enhance data. These are traditional methods for target detection tasks [8]. In this experiment, the rotation angle is randomized between [-10, 10] degrees, and the image is restored to a grayscale image after rotation. mirror flip, both the image and the label receive the operation, and the position of the labeling box is

updated. To train and test the detection model, we use YOLOV5 which is a commonly used target detection model in deep learning. Prior to the model training, some specific adjustments are made to refine the model by using the COCO dataset.

3. Experiments and Performance Analysis

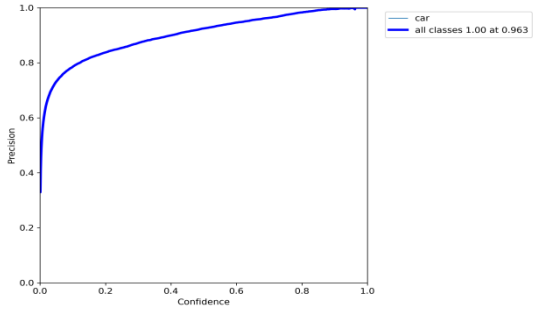
3.1. Experimental Software and Hardware

This paper was processed on a laptop computer with an Inter(R) Core(TM) i7 processor at 2.20 GHz, 32 GB of RAM, a 1TB hard drive, an NVIDIA GeForce RTX 3080 Laptop GPU for graphics, and running on Windows 11 64-bit, PyCharm Community Edition 2023.1.4.

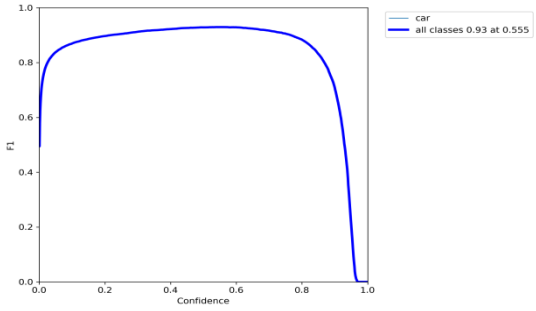
3.2. Training Accuracy and Robustness

The YOLOV5 structure described above is used to train YOLOV5 using the training set samples, the initial weights of the network are initialized with a Gaussian distribution with the standard deviation of 0.01 and the mean value of 0, 100 sample iterations are set, the initial learning rate is set to 0,001, and the momentum factor is set to 0.9. 100 iterations of the above training set are performed, and the variation curves are shown in Figure 4a.

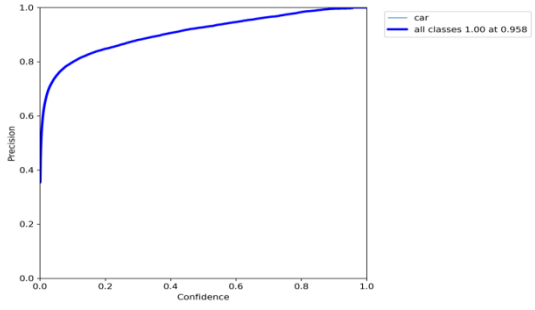
The results show that with the increasing number of iterations, the accuracy of the training set and validation set is increasing [9], and the accuracy of vehicle detection reaches 96.3% and tends to stabilize as the threshold value of the execution degree (confidence) increases. At the same time, as the threshold of execution (confidence) increases, F1 first rises and gradually shrinks to zero, and F1 reaches 93% when the threshold reaches 0.55. It indicates that the sum accuracy and recall rate of vehicle detection is more accurate. In Figure 4b, with the threshold (confidence) from 0.24 to 0.93, F1 is more stable indicating that YOLOV5 is more robust in vehicle detection. Overall the expected training effect is achieved.



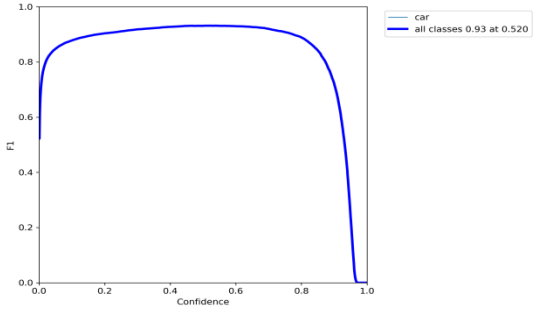
(a) Accuracy after black and white processing



(b) F1 after black and white processing



(c) Accuracy before black and white processing



(d) F1 before black and white processing

Figure. 4 Model performance after/before black and white processing

3.3. Post-training Feature Map Visualization and Analysis

Following the network structure, the trained model is used to detect vehicles in complex scenes[10]. We improve the accuracy and speed of detection through data enhancement and enrichment of the environment by converting color pictures into black and white images to be mixed into the dataset

and adding many pictures from different scenes. Figure 5 shows the output results produced after inputting the photographs after converting them into black and white images, which can be seen that pictures in various complex situations are detected using YOLOV5 after converting them into black-and-white images with high accuracy.



Figure. 5 YOLOV5 processing results

3.4. Validation of Model Effectiveness

In order to verify the reliability and stability of the model, 100 images of cars in different complex scenes from the test set are recognized, and the accuracy is chosen as an effective evaluation index in this paper. The recognition results are shown in Figure 4c and 4d. By comparing Figures 4a and 4c, we found that the vehicle detection rate reached 95.8% before black and white processing, and increased to 96.3% after black and white processing, improving by 0.5%. In Figure 4d, as the threshold changes, the F1 value fluctuates less, showing higher robustness. Although the recognition accuracy of the vehicle improves in different complex scenarios, there are still some inaccurate recognition problems in certain images (such as in Figure 6). This indicates that the car images in different scenes in the dataset are still not rich enough. Thus, more car images in complex environments are needed for expanding the dataset and further improving the recognition accuracy. Overall, we have achieved the expected results.



Figure. 6 Vehicle detection in complex scenes

4. Conclusion

The aim of this paper is to explore methods of data enhancement using image black-and-white preprocessing to optimize the performance of target detection algorithms. We convert color images into grayscale images and process them with operations such as adding noise and deformation to increase the diversity of data. The experimental results show that using image black-and-whitening preprocessing method for data enhancement can effectively improve the accuracy and robustness of the model on target detection. In addition to using the image black-and-whitening preprocessing to increase the diversity of data, we also used some other data enhancement techniques to increase the amount and diversity of data. One of the methods is to use a random cropping technique to produce different spatial scales and locations of the images by cropping the input images at different locations and scaling and rotating the cropped images to increase the data dithering and model robustness. Another approach is to use the random perturbation technique, which introduces random perturbations such as scaling, rotating, and deforming in the input image, making the model better adapted to the complex and variable image situations in real application scenarios. Finally, we also use GPU-based parallel computing techniques to reduce the computation time during the training process and improve the model training efficiency and speed. For future research directions, we will further explore how to utilize additional image processing techniques to increase data diversity to improve model performance. We will also investigate how to improve the accuracy and speed of the model while keeping the model parameters low to better adapt to real-world application scenarios.
In conclusion, by employing a variety of data enhancement techniques and efficient model training techniques, our target detection algorithms have made great progress in terms of accuracy and speed, and have demonstrated exceptional performance and reliability in real-world application scenarios. We believe that our algorithms will become more and more intelligent, efficient and appropriate with the continuous development and innovation of computer vision technology.

References

[1] Fu Longsheng, Feng Yali, Elkamil Tola, Liu Zhihao, Li Rui, & Cui Yongjie. Multicluster kiwi image recognition in field based on convolutional neural network.

[2] Li Xing, Guo Xiaosong, & Guo Jinbin. (2013). A forward vehicle identification method based on hog features and svm. Computer Science, 40 (11A), 4.

[3] Shi Kaijing, Bao Hong, Xu Bingxin, Pan Weiguo, & Zheng Ying. (2018). The vehicle detection method based on faster rcnn. Computer Engineering, 44 (7), 6.

[4] Ye Yunsheng. (2019). A vehicle detection method based on a convolutional neural network. Agricultural Equipment and Vehicle Engineering, 57 (2), 5.

[5] Wang, C. C. R. , & Lien, J. J. J. . (2008). Automatic vehicle detection using local features—a statistical approach. IEEE Transactions on Intelligent Transportation Systems, 9(1), 83-96.

[6] Siqi Zhao. (2021). Research on Cross-site Scripting Attack Detection Based on Deep Learning (Master's thesis, Liaoning University of Science and Technology). https://kns.cnki.net/KCMS/detail/detail.aspx?dbname=CMFD202201&filename=1021169114.nh.

[7] Yan Zhang. (2016). Research on Target Tracking Algorithm Based on Detection and Graph-cuts (Master's thesis, Nanjing University of Aeronautics and Astronautics). https://kns.cnki.net/KCMS/detail/detail.aspx?dbname=CMFD201701&filename=1016925789.nh

[8] Xing Wang. (2022). Research on low-light image enhancement algorithm for wellbore (Master's thesis, Anhui University of Science and Technology). https://kns.cnki.net/KCMS/detail/detail.aspx?dbname=CMFDTEMP&filename=1023543790.nh

[9] Sun Ge. Study on the detection method of rebar surface defects based on machine vision. (Doctoral dissertation, Shandong Jianzhu University).

[10] Zhu Lei. (2020). Research and implementation of vehicle detection, tracking and ranging method based on binocular vision. (Doctoral dissertation, Hunan University).