2019 11th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)

Vehicle direction detection based on YOLOv3

Fang Miao[1], Yiyang Tian[2], Libiao Jin[3]
School of information engineering
Communication University of China
Beijing, 100024, China
E-mail: mfcuc06@cuc.edu.cn

Abstract—With the popularity of traffic monitoring systems and traffic intelligence management, a large number of fixed and mobile cameras are installed in the city. However, manual monitoring of the camera is not efficient. The target detection method can monitor the screen and analyze the direction in which the vehicle drives and parks, helping to reduce the labor burden and improve efficiency. In this paper, we add a fine classification network after YOLOv3 to detect and discriminate the vehicle's driving and parking directions. YOLOv3 detects and extracts vehicle location information, and from which the latter network is responsible for finely classifying directions. We achieved a 91% classification accuracy on a test database of 400 images. This modified system can increase the practicality of the YOLOv3 target detection in the traffic scene.

Keywords—Convolutional neural networks;yolov3;feature extraction;object detection;data enhancement

I. INTRODUCTION

Analysis of vehicle driving and parking directions is an important issue in the field of traffic monitoring systems, because it can indicate the driving performance of the vehicle and help the monitoring personnel to distinguish whether the vehicle is parked illegally. However, with the development of the traffic monitoring system, the number of traffic surveillance cameras in the city has increased rapidly, and the analysis of the screens only by manpower is not efficient. Automatically analyzing the direction of the vehicle in the screen by means of artificial intelligence can reduce the workload, enabling efficient judgment and supervision of the vehicle's driving and parking directions, and probably reducing the number of accidents. Our purpose is to locate and classify vehicle objects captured in traffic scenes accurately so that their driving and parking directions can be discriminated. The traditional target detection algorithms need to select the region of interest in advance, and then extract features from the region of interest and send it to the classifier, which usually suffers from slow response time. As a regression-based target detection system, YOLO [1] [2] has good detection accuracy and faster detection time than other target detection methods. When an image is input, the YOLO model can output the position and classification of all objects of the image accurately and quickly. The YOLO model is implemented as a convolutional neural network and evaluated on the Pascal VOC test data set, which is excellent for detecting 20 types of objects. In this paper, we chose YOLO v3 [3] as the basis and added a neural network to the YOLO v3 for vehicle direction classification. The modified system can detect and classify the vehicle better and accurately. We also collected vehicle pictures for various scenes, sorting and marking them. And we used data enhancement to expand our training set.

The paper is organized as follows. In the Section II, the YOLO v3 neural network and related work are introduced. The Section III describes the system design scheme. The Section IV presents the achieved results. In the final sections, we summarize presented work and give directions for the future work.

II. RELATED WORK

A traditional target detection algorithm selects a region of interest (ROI) on a given image as a candidate region, and extracts features for the candidate region, such as a histogram of oriented gradient (HOG) or a scale-invariant feature transform (SIFT), and classifies the area through the training classifier. Popular vehicle detection methods in recent years have used HOG features (or Haar-like features) and SVM (or Adaboost) classifiers, optical flow methods, and the like. For example, [4] examines in detail how to use a set of HOG-based classifiers to distinguish the direction of a vehicle in an image. In the field of vehicle identification, traditional methods have limited ability to extract features and generalization. The training has higher requirements on the angle of view, the resolution and size of the picture. Although it can also achieve a good recognition accuracy, the effect is not as good as the neural network. So naturally, the deep learning model is used to solve this problem.

There are two kinds of object detection based on deep learning: one is a region-based two-stage target detection algorithm, such as the Fast-RCNN series [5] [6] [7], and the other is a regression-based one-stage target detection algorithm, such as YOLO, SSD [8]. R-CNN uses the selective search algorithm to extract about 2000 region proposal from top to bottom in the image, extracts features for each region proposal and classifies them with SVM, and performs border regression on region proposal. This process is divided into several stages, and it takes too much time to extract the region proposal. Faster-RCNN improved the R-CNN and used RPN (region proposal Network) instead of the original selective search method to generate region proposals, which reduced the number of region proposals from about 2,000 to 300 and improved the quality of the region proposals. The Faster-RCNN also shares convolutional layers for RPN and Fast R-CNN to reduce operations and achieves an increase in speed. [9] used the Faster-RCNN to detect vehicles, and then used the full convolutional network (FCN) to achieve vehicle taillight detection. Training different neural networks to detect

978-1-7281-1859-8/19/$31.00 ©2019 IEEE
DOI 10.1109/IHMSC.2019.10157

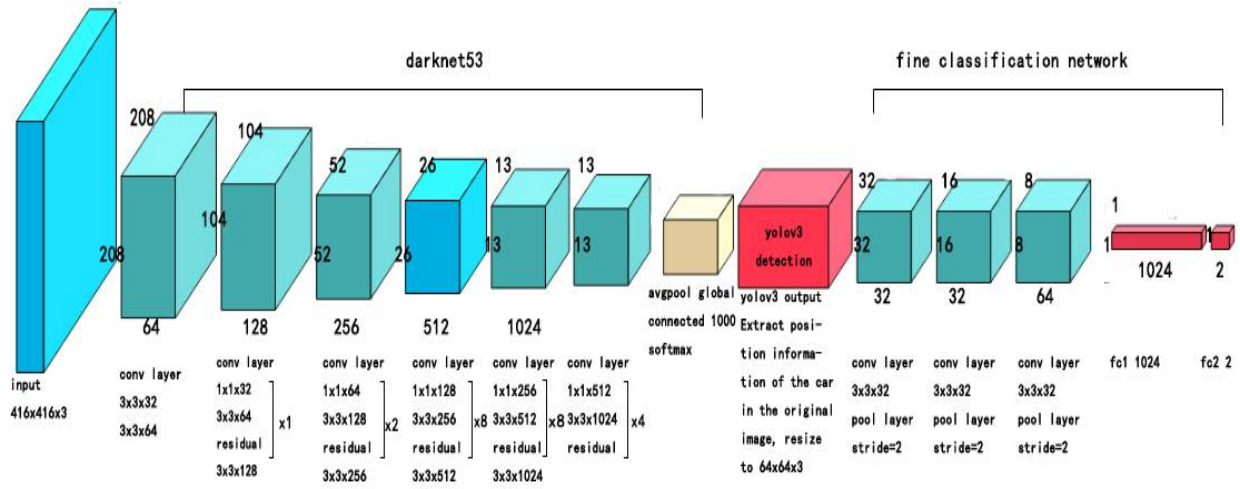

Figure1. The system structures. Two network cascades constitute the vehicle direction detection and recognition system

vehicles, such as YOLO, can accurately extract the bounding boxes of multiple object categories from the image and then proceed to the next step. YOLO is a single end-to-end network, which performs feature extraction, location and classification in a network. Therefore, the YOLO series are very fast and can be used for real-time detection. At 320x320 YOLO v3 runs in 22 ms at 28.2mAP, as accurate as SSD but three times faster.

This paper introduces a YOLO v3-based vehicle direction detection system. We added a direction classification network after the YOLO v3 network. The modified system can quickly detect and identify the direction of the vehicle. We hope that this system can be used for vehicle direction detection in traffic scenarios in the future.

III. SYSTEM DESIGN

This paper proposes a system based on the YOLO v3 model and combined with a fine classification network for vehicle direction detection. The system training process is divided into two steps. First, we use the PASCAL VOC data set to train YOLO v3 to obtain the initial weight of the YOLO v3 neural network. Then, we use the vehicle dataset to train the fine classification neural network to obtain the weighting file of the fine classification network. Finally load weights for testing. The network structure of the system is showed in Figure1.

A. Yolo v3 Algorithm

The YOLO v3 algorithm is an improvement to the YOLO algorithm. YOLO v3 divides the input image into S*S grids (s=13), each of which is responsible for detecting objects that fall in the grid. Each grid has three bounding boxes with different initial size. The initial sizes of each anchor box are generated by k-means clustering. YOLO v3 has 5 predictions for each bounding box: tx, ty, tw, th and confidence. (tx, ty, tw, th) is the location information of the bounding box, the confidence reflects the objectness score. YOLO v3 uses logistic regression to predict objectness scores for each box. If the bounding box prior overlaps a ground truth object and is more than the other bounding box prior, its score is set to 1. If the bounding box prior is not the best but does overlaps a ground truth object by more than a threshold, we ignore it. Each ground truth object is assigned a bounding box prior. At last, YOLO v3 predicts a 3-d tensor encoding bounding box, objectness, and class predictions (N×N×[3×(4+1+C)] for the 4 bounding box offsets, 1 objectness prediction, and C class predictions). The YOLO v3 network structure is shown below.

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

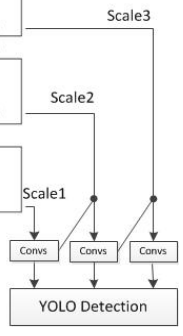

Figure2. The structure of YOLO v3

The feature extraction network is darknet-53, which has 53 convolutional layers. The darknet-53 uses 3×3 and 1×1 convolutional layers and adds some shortcut connections. After performing a series of convolution and batch normalization operations on the input image, YOLO v3 predicts boxes at three different scales. When the input image size is 416 × 416, the detection is performed on the

scales 13×13, 26×26, and 52×52. YOLO v3 merges three scales with a concept similar to FPN. Multi-scale detection can obtain the features of different resolutions, which is beneficial for detecting small target objects.

B. *Fine classification neural network*

The network consists of three convolutional layers and two fully connected layers. There is a maximum pooling layer behind each convolutional layer. The classification and location information of the vehicle target is obtained from the original image through the YOLO v3 network. We extract the location of the car mapped to the original image, regard it as a region proposal, and resize the region proposal to 64×64 as the input to the direction classification network. And finally, the network output the classification result. The structure of the fine classification network is as follows.

TABLE I. THE STRUCTURE OF CLASSIFICATION NEURAL NETWORK

type	filters	size	output
conv	32	3x3	64x64
pool		2x2	32x32
conv	32	3x3	32x32
pool		2x2	16x16
conv	64	3x3	16x16
pool		2x2	8x8
fc1			1024
fc2			2
softmax			

The training process for the vehicle direction classification network is as follows.

(1) Initialize the network, and randomly initialize the parameters to be trained in the network according to the number of neurons by using a truncated normal distribution with a standard deviation of 0.05.

(2) Normalize the image sample to 64×64 size, input the network, and send the vehicle direction label to the SoftMax classifier. Perform a forward propagation, output vehicle direction discrimination probability, and calculate the average cross entropy of the vehicle discriminating direction and the actual direction as loss.

(3) Using the backpropagation algorithm, finely adjust the parameter values of the network following the direction of loss reduction according to the gradient descent algorithm.

(4) Replace the image sample, repeat the above steps iteratively, and stop training until the loss falls to a suitable level.

IV. IMPLEMENTATION AND RESULT

In this section, we introduce implementation details and experiment-related settings. We then present the results of the vehicle direction detection and analyze the proposed method.

A. *Data Composition*

The vehicle dataset contains a total of 4,400 pictures. The training set consists of 4000 images and the test set consists of 400 images. We marked the dataset with the front label and the side label of the car. At first, there were 1000 original pictures (500 fronts and 500 sides). We expanded the training set to 4000 by horizontal flipping, cropping and Gaussian blur.

B. *Training*

This system is based on the YOLO v3 target detection network. We designed our system by adding a fine classification network after the YOLO v3 network. The system training steps are as follows:

(1) Using the YOLO v3 network to train on the Pascal VOC datasets, the 20 categories of target detection weight files are obtained.

(2) Resizing images to a size of 64x64, and then Using the classification network to train on the resized datasets, the weights of direction classification network are obtained.

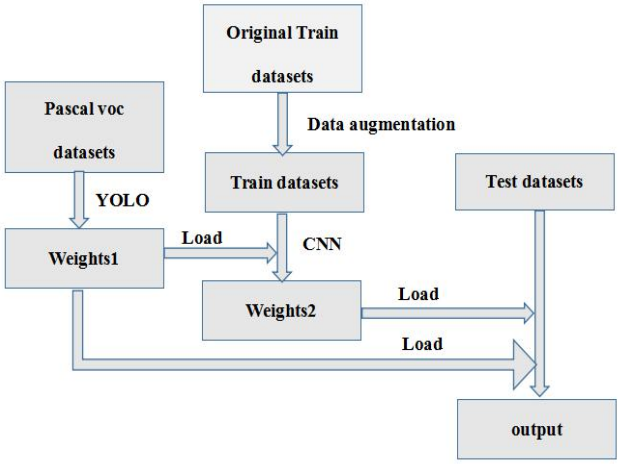

Figure 3. System training flow

As shown in the figure above, the two networks are cascaded. We train the two networks separately. After output by the first network, the image needs to be normalized to 64×64 and then input to the second network.

C. Tests and Results

The test image passes through the YOLO v3 network firstly to obtain the classification and positioning information of all objects detected in the image. The system stores the positioning information and classification information detected by YOLO v3 network and extracts the position information of the targets which are classified as vehicle. Then the system resizes the extracted region proposal to 64× 64 and sends them to the fine classification network. The system ultimately outputs the location and direction of the vehicle object in the picture. We set the initial learning rate to 0.00001. The system test results are as follows.

Figure 4. Test results

The test sample set contains vehicle images of various ambient illumination. After testing 400 test samples, the vehicle direction recognition accuracy is over 90%. As shown in the figure above, in a simple scenario, when the vehicle on the screen is relatively large, the system detects good results. When the detection scene is too complicated, that is, there are multiple vehicles on the image, the detection effect is not good. The reason is that the sizes of vehicles are too small and these vehicles are occluded from each other, resulting in insufficient resolution of the classified network, and the features are not well characterized. Next, we quantify the relationship between the number of iterations and the performance of the test.

TABLE II. EFFECT OF EPOCHS ON ACCURACY AND AVERAGE PRECISION

Epoch	Accuracy	AP
500	0.7350	0.7420
1000	0.8325	0.8456
2000	0.8575	0.8716
4000	0.8900	0.9035
6000	0.9075	0.9175
7000	0.9100	0.9194
8000	0.9125	0.9202
10000	0.9150	0.9221

As shown in the table above, we set the number of iterations from 500 to 10000 at reasonable intervals. The experimental results show that the detection accuracy increases with the number of iterations. When the number of iterations is increased to 6000, the accuracy is above 90%. When the number of iterations is less than 2000, the training effect is not very good because the training is not convergent.

V. CONCLUSIONS

In this paper, we propose to build a vehicle direction detection system based on the YOLO v3 algorithm combined with fine classification neural network. For small sample data sets, it is a very effective method to pre-train the Pascal VOC dataset to obtain the target detection weight file and then use the YOLOv3 network for training and prediction. The added fine-category network increases the classification level of target detection, enabling it to be applied to scenarios where vehicle violations driving and parking are detected. The results of this experiment show that it is an effective and robust detection method. In the future work, the accuracy of the algorithm can be improved by training larger and more diverse data sets including different lighting conditions and occlusions. We also hope to add more angles to the classification training in the future, so that the system can discriminate the driving and parking direction of the vehicle more accurately.

ACKNOWLEDGMENT

This work was supported by “National Key R&D Program of China´ (Grant No.2017YFB1402203) and “the Fundamental Research Funds for the Central Universities”

REFERENCES

[1] Redmon J, Divvala S, Girshick R, et al. You Only Look Once: Unified, Real-Time Object Detection[J]. 2015.

[2] Redmon J, Farhadi A. YOLO9000: Better, Faster, Stronger[C]// IEEE Conference on Computer Vision & Pattern Recognition. 2017.

[3] Redmon J, Farhadi A. YOLOv3: An Incremental Improvement[J].2018.

[4] Rybski P E, Huber D F, Morris D D, et al. Visual Classification of Coarse Vehicle Orientation using Histogram of Oriented Gradients Features[C]// Intelligent Vehicles Symposium. IEEE, 2011.

[5] Girshick R, Donahue J, Darrell T, et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation[C]// IEEE Conference on Computer Vision & Pattern Recognition. 2014.

[6] Girshick R. Fast R-CNN[C]// IEEE International Conference on Computer Vision. 2015.

[7] Ren S, He K, Girshick R, et al. Faster R-CNN: Towards Real-Time Object Detection with region proposal Networks[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2015, 39(6):1137-1149.Dd

[8] Liu W, Anguelov D, Erhan D, et al. SSD: Single Shot MultiBox Detector[J]. 2015.

[9] G. Zhong, Y.-H. Tsai, Y.-T. Chen, X. Mei, D. Prokhorov, M. James, and M.-H. Yang, “Learning to tell brake lights with convolutional features,” in 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Nov 2016, pp. 1558–1563