Deep learning for object detection in video

Shengyu Lu*

School of Xiamen University, Xiamen, China

*Corresponding author e-mail: lu.s.y@foxmail.com

Abstract. With the development of deep learning technology in recent years, more and more fields are introducing deep learning to accelerate the development of the industry. Deep learning technology was originally used for image recognition. Nowadays, with the continuous improvement of video monitoring, the application of deep learning to surveillance video has profound implications for areas such as transportation and unmanned vehicle. The effects of traditional video recognition are not satisfactory, and deep learning is excellent in many image classification scenes. In this paper, we proposed an object detection algorithm of video based on YOLO network to realize vehicle identification based on surveillance video.

1. Introduction

At present, the pressure on traffic is increasing, in order to ease it, a smart transportation system emerged, the basis of smart transportation is vehicle identification. Only by identifying the vehicle information in the video and classifying the vehicle statistics, can we better control the road conditions, ease traffic pressure, reduce traffic congestion and traffic accidents. What's more, it can provide powerful information support for informational command and intelligent dispatch. Nowadays traffic monitoring is also developed, how to use the video of traffic monitoring is an important issue in smart transportation. Therefore, the proposed method has important significance for the development of smart transportation. At the same time in the area of unmanned driving, judging the road conditions also needs to be carried out on the basis of video recognition.

The current research on vehicle detectors is mainly based on virtual point technology. The main problem of traditional vehicle detection technology is instability. When the weather is bad or the camera shakes, it is prone to false detection results. Therefore, with the rise of deep learning, more and more laboratories are trying to apply deep learning to the work of vehicle inspection. Deep learning was excellent in ImageNet's competition, which fully demonstrated the talent of deep learning in video image processing, however, the main problem of deep learning now is the long training time, and it is difficult to detect real-time video for vehicles.

This paper uses the deep learning to complete the video vehicle detection work, so as to ensure the accuracy and stability of the detection, at the same time, it combines traditional detection algorithms, such as color-domain-based, frame difference algorithms, to improve the detection efficiency and lay the foundation for real-time detection. The main difficulty of this algorithm lies in the preprocessing of the video and building a suitable model to detect the vehicle, taking into account the speed and accuracy. Our main tasks include building deep learning models for training, researching video processing algorithms, and building network models.
2. Background and Related work

The goal of the object detection is to find the position of the target object in the image and make a category judgment on the target object. The traditional method of using machine learning to detect target features, such as SIFT, HOG, etc., essentially uses artificial design features and the application of the sliding window method, which has a series of drawbacks such as high time complexity and poor generalization ability.

With the rise of deep learning technology, object detection algorithms begin to integrate more with their powerful expressions. In 2014, the RCNN model [1] was proposed by the Ross Girshick team. The basic approach is to selectively acquire multiple candidate regions on the input image, use CNN to extract regional features, and then obtain the best region by non-maximal suppression. Considering that the RCNN has complex steps and high time cost in the training and testing process, Shaoming He and others proposed the SPP-net structure [2], the structure uses the strategy of spatial pyramid sampling, reduces the time cost by one operation of convolving full maps, and then maps the location information of candidate regions to the feature map. In 2015, the Ross Girshick team improved the RCNN and proposed the Fast RCNN model[3], which was mainly used to solve the problem of candidate window duplication and integrated all the models, however, it still adopts a strategy of selective search for candidate areas, and the time cost has not been reduced. In order to solve this problem, Girshick team designed the Faster RCNN algorithm[4], and innovatively proposed RPN to generate candidate regions, making the regional candidate, regression and other methods share the convolution feature, so that the efficiency and performance of the target detection have improved significantly.

The end-to-end approach is based on non-regional candidate, the two most representative algorithms at present are the YOLO[5] and SSD [6]algorithms. These algorithms obtain candidate regions in the image by means of uniform segmentation. The method of comparing region candidates is faster and can maintain high accuracy, object detection in a typical road environment has a good effect.

3. Method

YOLO is an improved CNN algorithm. The whole network is mainly composed of convolution layer, pooling layer and full connection layer. YOLO is an end-to-end object detection algorithm, which integrates the three processes of generating candidate regions, extracting regional features and classifying in image detection. Therefore, compared with the traditional methods, YOLO can implement the object detection of images quickly.

The YOLO network model is shown below, including 24 layers. Where, the L(3) layer to the L(24) layer is similar to the L(1) layer and the L(2) layer. There are two convolution layers and max-pooling layers appearing alternately in each layer. The convolution kernel of all convolution layers are size=3, stride=1, pad=1, and the activation function is ReLU function. The all the max-pooling layers are size=2, stride=2. The output layer is a tensor of 7*7 (5*5+5) dimensions. The framework of YOLO network is as Figure1:

![Figure1](image-url) The framework of YOLO network
The training samples of YOLO do not need to extract all kinds of samples from the images, and use the whole images to train model directly. Of course, the labels of the samples includes both the information of the category and the location of the sample in the images. The YOLO divides the image into S*S grids and calculates the probability that the center of the object falling into the grids respectively. If the probability exceeds the threshold, there is an object in the grid. Then, B boundary boxes are built for the grids that exist objects, and the confidence of each box is calculated in the meantime. The confidence reflects the probability that an object exists in the boundary box. The formula is:

$$P_b = P_r(\text{Object}) \times IOU$$  \hspace{1cm} (1)

$$IOU = \frac{\text{area}(BB_{dt} \cap BB_{gt})}{\text{area}(BB_{dt} \cup BB_{gt})}$$  \hspace{1cm} (2)

Where, $P_r(\text{Object})$ represents the probability that the boundary box contains the target objects. $BB_{gt}$ represents the reference boundary box based on the training labels; $BB_{dt}$ represents the detection boundary box; $\text{area}(\cdot)$ represents the area.

Each boundary box includes five parameters: x, y, w, h, confidence. Where, (x, y) represents the center position of the boundary box relative to the original grid, w represents the height of the boundary box, and h represents the height of the boundary box. In the meantime. Each grid also predicts the probability of C categories which is $P_r(Class_i|Object)$, and it represents the probability that the center of class i fall into the the grid. Finally, the network outputs the tensor of S*S*(B*5+C) dimensions.

4. Experiment

4.1 Data sets

Our data sets were from urban intelligent traffic monitoring video. We preprocessed vehicle monitoring video with temporal difference method, and obtained a total of 3,000 pictures with the resolution of 1280*720. Among them, the training set has 2000 pictures, and the test set has 1000 pictures. In the training set, we marked 4,000 samples of five categories A, B, C, D and E, including cars, buses, large trucks, motorcycles and persons. As shown in Figure2.

![Figure2](image)

**Figure2** The simple images

We use sample pictures to train the model, and then input vehicle monitoring video into the trained model to get the object type and position information.

4.2 Evaluation metrics

We used the precision as evaluation metrics [7][9]. It defines as following:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Among them, TP and FP represent the number of true cases, false positive cases respectively.

4.3 Baseline methods

We used the following method for comparison:

1. Slide window: The image is divided into several grids, and the sliding window is used to extract the features of each grid and predict the category of objects [10][14].
2. CNN: Extract features of image by convolution operation [15].
3 Faster R-CNN: On the basis of RCNN, the extracted feature area is mapped to the feature map of the last convolutional layer of CNN [6], so only one feature extraction operation is needed.

4.4 Experimental results and Analysis

We used several data set to measure the performance of the methods. We obtained the experimental results of YOLO algorithm and other baseline algorithms respectively, and then calculated the average value of precision, as shown in the following Table1

| Algorithm      | Precision (%) | Frames every second(f/s) |
|----------------|---------------|--------------------------|
| Slide window   | 70.59         | 1/16                     |
| CNN            | 80.82         | 2/10                     |
| RCNN           | 84.19         | 1.5                      |
| YOLO           | 88.34         | 28                       |

We can see from Table1 that the precision of YOLO algorithm is better than baseline algorithms, especially in terms of accuracy rate. In the experiment, in order to ensure the real-time performance of the YOLO algorithm, we set the grid size to be relatively large. Therefore, the possibility of missing detection may reduce the recall rate to some extent.

| Algorithm | car  | trunk | bus  | motor | person |
|-----------|------|-------|------|-------|--------|
| Slide     | 65.35| 70.26 | 68.65| 68.34 | 72.16  |
| CNN       | 81.88| 75.38 | 79.98| 80.21 | 72.31  |
| RCNN      | 75.38| 80.21 | 85.36| 82.52 | 85.33  |
| YOLO      | **87.26**| **84.88**| **89.53**| **81.37**| **89.01**|

From Table 2, we can see the accuracy rate of object recognition with various algorithms. Compared with other algorithms, YOLO has higher accuracy rate for various objects. For categories recognition, the precision of the cars and buses are significantly higher than the trunk. There is something with our vehicle monitoring video that the appearance probability of cars and buses is higher than trunks. The larger number of samples to train, we obtain a more accurate model. Motors and persons are similar, and their samples appear more frequently in our data set.

5. Conclusion

In this paper, we proposed an object detection algorithm in vehicle video based on YOLO, which can apply for object detection of surveillance video. We constructed the deep learning network model based on YOLO to object detection and obtain the vehicle categories and location information. Our method combined the advantages of traditional methods and deep learning techniques. Compared with relevant algorithms, our method implemented high accuracy based on deep learning network, and greatly reduces the time of image detection. Our method can implement the object detection of video in real-time, and obtain vehicle information quickly and accurately.

References

[1] D. M. Ramík, C. Sabourin, R. Moreno, and K. Madani, ‘A machine learning based intelligent vision system for autonomous object detection and recognition’, Applied Intelligence, vol. 40, no. 2, pp. 358–375, Mar. 2014.
[2] W.-L. Zhao and C.-W. Ngo, ‘Flip-Invariant SIFT for Copy and Object Detection’, IEEE Transactions on Image Processing, vol. 22, no. 3, pp. 980–991, Mar. 2013.

[3] K. Mizuno, Y. Terachi, K. Takagi, S. Izumi, H. Kawaguchi, and M. Yoshimoto, ‘Architectural Study of HOG Feature Extraction Processor for Real-Time Object Detection’, 2012, pp. 197–202.

[4] K. He, X. Zhang, S. Ren, and J. Sun, ‘Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition’, p. 14.

[5] G. Li, J. Liu, C. Jiang, L. Zhang, M. Lin, and K. Tang, ‘Relief R-CNN: Utilizing Convolutional Features for Fast Object Detection’, in Advances in Neural Networks - ISNN 2017, vol. 10261, F. Cong, A. Leung, and Q. Wei, Eds. Cham: Springer International Publishing, 2017, pp. 386–394.

[6] S. Ren, K. He, R. Girshick, and J. Sun, ‘Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks’, p. 14.

[7] S. Lu, B. Wang, H. Wang, and Q. Hong, “A Hybrid Collaborative Filtering Algorithm Based on KNN and Gradient Boosting,” in 2018 13th International Conference on Computer Science & Education (ICCSE), Colombo, 2018, pp. 1–5.

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, ‘You Only Look Once: Unified, Real-Time Object Detection’, 2016, pp. 779–788.

[9] S. Lu, H. Chen, X. Zhou, B. Wang, H. Wang, and Q. Hong, “Graph-Based Collaborative Filtering with MLP,” Mathematical Problems in Engineering, vol. 2018, pp. 1–10, Dec. 2018.

[10] Sudowe and B. Leibe, ‘Efficient Use of Geometric Constraints for Sliding-Window Object Detection in Video’, in Computer Vision Systems, vol. 6962, J. L. Crowley, B. A. Draper, and M. Thonnat, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 11–20.

[11] W. Shuigen, C. Zhen, and D. Hua, ‘Motion Detection Based on Temporal Difference Method and Optical Flow field’, 2009, pp. 85–88.

[12] K. F. Wallis, ‘The Two-Piece Normal, Binormal, or Double Gaussian Distribution: Its Origin and Rediscoveries’, Statistical Science, vol. 29, no. 1, pp. 106–112, Feb. 2014.

[13] S. S. Seferbekov, V. I. Iglovikov, A. V. Buslaev, and A. A. Shvets, ‘Feature Pyramid Network for Multi-Class Land Segmentation’, arXiv:1806.03510 [cs], Jun. 2018.

[14] S. Lu, “An Image Retrieval Learning Platform with Authentication System,” in 2018 13th International Conference on Computer Science & Education (ICCSE), Colombo, 2018, pp. 1–5.

[15] S. Zha, F. Luisier, W. Andrews, N. Srivastava, and R. Salakhutdinov, “Exploiting Image-trained CNN Architectures for Unconstrained Video Classification,” in Proceedings of the British Machine Vision Conference 2015, Swansea, 2015, pp. 60.1-60.13.