Natural scene text detection based on YOLO V2 network model

Dong Haifeng¹, Han Siqi¹

¹School of Computer Science, Xi'an Petroleum University, Xi'an 710065, China
Email: 854560388@qq.com

Abstract: As an important part in the field, the natural scene text detection has been widely applied in visual navigation system, content-based image and video retrieval, instant translation system and so on. In this paper, we introduce several object detection network based on deep learning, and apply the YOLO v2 into natural scene text detection, changing the multi objects detection problems into the two classification problems. The main works in the paper include the following: prepare the datasets; we train the YOLO v2 with the optimum parameters, carry out the regression analysis of the coordinate parameters and categories of bounding boxes, obtain the detection result; according to different detection models, the detection results of different datasets are compared and analyzed, YOLO V2 model detection speed 0.105s/image has certain advantages.

1. Introduction
Text detection in natural scenes has become a hot topic in current research. Due to the complexity and variability of natural scenes, text size and font, lighting and other reasons, it is difficult to accurately conduct text detection[1] Deep learning theory has been well researched and developed in pattern recognition image processing. This paper attempts to convert the multi-classification problem of target detection into binary classification (background and text), and apply it to text detection in natural scenes. Based on the analysis of the feasibility, advantages and disadvantages, Yolo V2 model was adopted[2]. Based on THE YOLO Model, THE YOLO V2 model introduced multiple technologies such as anchor mechanism in Fast-R-CNN, and improved the original YOLO target detection framework. With the advantage of maintaining the original speed, the accuracy was improved. At the same time, YOLO V2 also simplified the model and facilitated the optimization of the model[3]. The text detection system framework for natural scenes adopted in this paper is shown in Figure 1. First, the original data images are selected, the Chinese text information of the images is annotated, and VOC data sets are made. Then the samples are input into the YOLOV2 network model for training until convergence. Finally, the model is tested and verified.
Figure 1. Framework of text detection system for natural scenes

2. database

The YOLO V2 model itself can be trained on VOC2007\cite{4} data set. Therefore, this paper annotates the text information of the original data images and makes a database in the format of VOC2007 data set. The steps are as follows:

1) Rename the original data image to "00001.jpg" "00002.jpg" and so on;
2) Image text bounding boxes are annotated with LabelImg tool, and corresponding XML bounding information files are generated at the same time. They are similar to SVT data set, only corresponding to a single image with VOC format.
3) The image labeling XML file is parsed to generate the corresponding TXT format label file.
4) Generates train.txt and Val.txt for specifying training and test images.

Bounding Boxes The natural scene text detection database made in this paper includes Images: 850 /JPG, Labels: 850 /XML and 850 /TXT, Bounding Boxes: 1529 /Rectangles. The first 600 JPG images were selected for training, and the remaining images were verified. Some data images of the database are shown in Figure 2. The corresponding XML file is shown in Figure 3, and the label file in TXT format is shown in Figure 4.

![Database image](image1.jpg)

![The image "00013.jpg" corresponds to the XML file](image2.jpg)

Image names, paths and text position information are included in the XML file shown in Figure 3, among which <bndbox> <xmin>169</xmin> <ymin>99</ymin> <xmax>858</xmax> <ymax>455</ymax> </bndbox> refers to the specific position information of the image bounding box, which is classified as "text".
In Figure 4, the coordinate format of the label file is: Label X, Y, W, H. Label, namely "0", X and Y, are the central coordinates of bounding Box, namely (0.50,0.49), W and H are the ratio of width and height of Box and image, namely 0.67 and 0.20.

3. Training Process

3.1. Experimental Environment

Hardware environment: Intel(R)Core(TM) I7-4790 3.60GHz CPU

Software environment: Ubuntu14.04 LTS, cuda7.5, openCV2.4.20

YOLO model has its own unique open source framework Darknet, which can be run on BOTH CPU and GPU. In this paper, CUDA7.5 and CUDNN compatible with YOLO version are configured and run on GPU for experimental training.

3.2. Selection parameters

In this paper, the experimental training strategy by adopting the technology of fine - tuning, using ImageNet pre training weights files before 23 layer weight training, some parameters such as momentum (impulse), such as decay at parameter is set to the initial parameters of the original network ImageNet, only need to modify learning_rate (vector), max_batches (iterations), classes (number of categories in model checking), filters (the last layer convolution convolution kernel number) four parameters. In the experiment in this article, max_batches = 40000, classes = 1, Filters = num*(classes+coords+1) = 30. The main parameter is selected as learning_rate.

In this paper, SVT[5] data set and homemade database were used for training test, mainly for homemade database. Therefore, the selection of parameters mainly considers the training based on homemade database. During the model training, different learning_rates will have different influences on the training result loss function Loss. In this experiment, Matplotlib was used to visualize avg_loss. In Figure 5, Figure 6 and Figure 7, for example, in batches of AVG_loss of model training under different Learning_rates 40000 times, Table 1 shows the loss function loss value of loss under different Learning_rates at the end of training.

![Figure 5. Visualization curve of AVG_loss when LR =0.01](image1)

![Figure 6. Visualization curve of AVG_loss when LR =0.001](image2)
The learning rate (LEARning_rate) has a certain influence on the gradient descent trend of loss during training. When the value of LR is large, the gradient descent iteration process is fast and does not converge well. It is always hovering around the optimal solution and cannot calculate the optimal solution. When the value of LR is small, the iteration process of gradient descent is slow. Although more accurate optimal solution can be obtained, the time consumption is large.

Table 1. Loss at the end of iteration under different learning_rates

| Learning_rate | 0.01       | 0.001      | 0.0001     |
|---------------|------------|------------|------------|
| Loss          | 4.593606   | 2.709078   | 0.901963   |

From the visualization graph of AVG_loss, it can be intuitively seen that as the number of iterations increases, the avG_loss trained in the model keeps decreasing, and the number of iterations reaches about 1000. The reduction range of AVG_loss obviously slows down, showing a convergence trend. However, when learning_Rate=0.01, the avG_loss curve diverged several times in the later stage of training, compared with the stable convergence when learning_rate=0.001 and 0.0001. By comparing the iteration ending loss in Table 1, the experiment was conducted with learning_rate=0.0001.

4. Experimental data

Under the condition of optimal parameters, the text detection method of natural scenes based on YOLO V2 is trained on SVT data set and self-made database respectively. During the training process, the network will save the training weight file after every 100 or 1000 iterations. The purpose is to continue the training with the intermediate training weight file instead of starting from the beginning when the training is interrupted. After 40,000 iterations of training, the network will generate the final weight file Yolo-voc_final.weights. We can test with the final weight file. A partial weight file is shown in Figure 8.
5. Experimental Results

The experiment uses the final weight file generated by training, YOLO - voc_final.Weights, to test and verify. First, the single picture is tested, and the image of the box text as well as the test time, category and confidence rate are directly returned. Figure 9 shows the training test results based on the self-built database.

In order to analyze the experimental results in a substantial and real way, this paper mainly evaluated the experimental results through several index parameters, including Avg_IOU, Precision and Recall. The specific calculation method of these indicators is described in detail below.

Positioning accuracy Avg_IOU refers to the degree of coincidence between the frame detected by the model and the image marker frame. Bounding Boxes Precise, that is, the ratio of correct bounding boxes among the detected images. Recall refers to the ratio of bounding boxes accurately detected in the test data set. Precision and Recall can be calculated using equations (1) and (2).

\[
\text{Precision} = \frac{tp}{tp + fp}
\]

\[
\text{Recall} = \frac{tp}{tp + fn}
\]

Tp represents the number of right boxes where the model can flourish. False FLOURISH represents the number of wrong boxes among the text that the model detects. Fn (false negatives) is denoted as the number of undetected text boxes.
In the actual calculation, we need to adjust the threshold to get the optimal result. In the experiment of this paper, the recall threshold \( \text{THRESH} = 0.25 \). The test data set is verified, and the final results are shown in Table 2.

| The data set            | Avg IOU (%) | Precision (%) | Recall (%) |
|-------------------------|-------------|---------------|------------|
| SVT                     | 74.38       | 61.23         | 73.33      |
| Self-made database      | 88.71       | 78.92         | 74.52      |

As can be seen from Table 2, compared with the self-built database, the accuracy of SVT data set is lower, which is due to the text annotation of SVT data set itself, many small texts are not annotated, which leads to the underestimation of detection results.

Table 3 shows the test results for different detection models and data sets. Respectively are the YOLOv2 model based on homemade database, test results on SVT, VOC2007 data set and YOLO V2 model based on CNN network, Alsharif proposed network test results on SVT data set respectively.

| Detection model | The data set | The mAP (%) | Time/image (s) |
|-----------------|--------------|-------------|----------------|
| YOLO v2         | Self-made database | 73.54     | 0.105          |
| YOLO v2         | SVT          | 67.25       | 0.122          |
| CNN [26]        | SVT          | 76.29       | 6.9            |
| Alsharif [25]   | SVT          | 74.3        | -              |
| YOLO v2 [20]    | VOC2007      | 76.8        | 0.014          |

As can be seen from Table 3, different networks are based on training tests on SVT data sets, and CNN-based networks and Alsharif propose that the test results of networks are better than those of YOLO V2. However, COMARED with other networks, YOLO V2 network model has a high advantage in detection speed, with only 0.122s for processing a picture. Training network generally requires a large amount of data, and the high-speed detection speed of YOLO V2 ensures good time performance for model training.

6. Conclusion
This paper mainly introduces the training test of target detection framework YOLO V2 applied to text detection in natural scenes. The training is based on self-made database. First, the data set is briefly introduced, including the production of self-made database. Secondly, the training process and loss function of text detection are introduced, experimental data and results are explained, and the detection results of different detection models and data sets are compared and analyzed. Finally, in view of the disadvantages of YOLOv2 applied to text detection in natural scenes, two assumptions were proposed, and MSER was combined to extract candidate region algorithm and K-means clustering extension candidate region algorithm to improve the model.
Acknowledgments
I am grateful to my school for giving me the opportunity to learn and become talented, and thank the teacher for giving me patient guidance and criticism and correction when I have no direction, so that I can improve the deficiencies of the thesis in time.

Reference
[1] Liu Jinyang. Research on context-based Target Detection Algorithm [D]. Nanjing University,2017.
[2] Liu Jie, Zhu Xun, SONG Mimi. Improved Chinese character detection for YOLOv2 in end-to-end natural scenes [J/OL]. Control and Decision :1-7(2020-07-04).
[3] Tan Boshuai. Text detection and Recognition of natural scenes based on Deep learning [D]. Shanghai Normal University,2020.
[4] wu ya-an, li jun-jun, zhang bin-bin. Multi-target detection network under contextual information constraints [J]. Intelligent computer and application,2019,9(06):6-12.
[5] Zhang Xiaofeng. Research on the Application of Convolutional depth confidence Network in Text Recognition [D]. Xi ’an University of Technology,2018.