IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Review of Computer Engineering Studies
Vol. 6, No. 2, June, 2019, pp. 44-49
Journal homepage: http://iieta.org/journals/rces

Accurate Positioning of License Plate in Video Stream Based on Concatenated Convolutional Neural Network

Pan Ding, Hua Sun*, Chuping Xiong, Yao Li

School of Software, Xinjiang University, Urumqi 830008, China

Corresponding Author Email: xj_sh@163.com

https://doi.org/10.18280/rces.060203

Received: 13 February 2019
Accepted: 29 May 2019

Keywords:
accurate positioning of license plate, concatenated convolutional neural network (CCNN), You Look Only Once, Version 3 (YOLO v3), real-time detection

ABSTRACT

One of the key functions of intelligent traffic management system is the accurate positioning of license plate in the video stream. However, the traditional license plate positioning algorithms are greatly affected by environmental factors, such as license plate covers, cloudy weather and varied colors. To overcome this defect, this paper designs a three-level concatenated convolutional neural network (CCNN) with multi-task learning ability. The first level detects the vehicles in the video, using the target detection algorithm You Look Only Once, Version 3 (YOLO v3). Based on the images detected on level 1, the second level performs rough detection of the license plate. On this basis, the third level accurately positions the key points on the license plate. The experimental results show that the CCNN achieved a mean accuracy of 95.8 % and a positioning speed of 63f/s in license plate detection, much better than the traditional license plate positioning algorithms. The proposed method can pinpoint the license plates in video in real time at a high accuracy.

1. INTRODUCTION

Intelligent video surveillance is a cutting-edge video technology that extracts and screens the abnormal behaviors within the video in real time, and issues early warnings without any delay. Compared with conventional surveillance technologies, intelligent video surveillance not only supports passive monitoring, but also achieves active control of abnormalities. To realize deep mining and quick search of video contents, the technical enterprises in China are competing to develop core technologies (e.g. digital signal processing and video analysis algorithms), especially the automatic identification of the attributes of specific items in video.

License plate recognition is an important application of video surveillance. The existing license plate positioning methods are generally based on the color recognition algorithm, the edge recognition algorithm, or mathematical morphology. Specifically, the color-based positioning method needs to go through grayscale operations before acquiring the features of the image on license plate, and is affected by natural light intensity and license plate covers. Thus, the features extracted by this method is rather unstable. Edge-based positioning mainly collects the order of gradient and computes the local gradient changes of the license plate image, failing to handle images with complex backgrounds. The morphology-based positioning method, with a certain morphological structure, supports erosion, dilation, opening and closing operations of binary images. This method usually needs to be combined with the other license plate recognition algorithms [1].

The license plate recognition is essentially to detect a target in the image. During the recognition, the first step is to judge whether the target exists in the video. If the target presence is confirmed, the target should be differentiated from non-region of interests, and be positioned accurately. In recent years, deep learning has become a hotspot in the field of image recognition, thanks to the in-depth research into human neural network. A typical deep learning algorithm is the convolutional neural network (CNN) [2]. With a deep network structure, the CNN contains a series of operations, ranging from convolution to pooling. The emergence of regional CNN (R-CNN) [3] and Over Feat [4] in 2013 marked the dawn of deep learning-based image target detection. The relevant algorithms include Fast R-CNN [5], Faster R-CNN [6], Single Shot Multi Box Detector (SSD) [7], You Look Only Once (YOLO) series [8~10], and the latest method Pelee. In less than five years, deep learning-based image target detection has evolved from two stage to one stage, from bottom-up only to top-down, from single scale network to feature pyramid network, and from the PC-end to the mobile-end. All these algorithms boast excellent detection performance on open target detection datasets.

Based on K. Zhang's concatenated neural network [11, 12], this paper designs a three-level concatenated CNN (CCNN) with multi-task learning ability. The innovation lies in the three-level structure of the CCNN. The first level selects the candidate window of the vehicle in the video, using the network model You Look Only Once, Version 3 (YOLO v3) [13]. The second level extracts the license plate with three CNN models. The third level accurately positions the four key points of the license plate with six CNN models, and then outputs the license plate.

2. MODEL CONSTRUCTION

The network architecture is a three-level CCNN, with several CNN models on each level. On the first level, the YOLO v3 target detection algorithm is introduced to detect and classify the vehicle features in each frame of the video.

The classification results are inputted to the next level. On the second level, the license plate is roughly detected by three CNN models, outputting the key points of the license plate. On the third level, six CNN models make accurate detection of the license plate based on the key points. The structure of the CCNN model is illustrated in Figure 1.

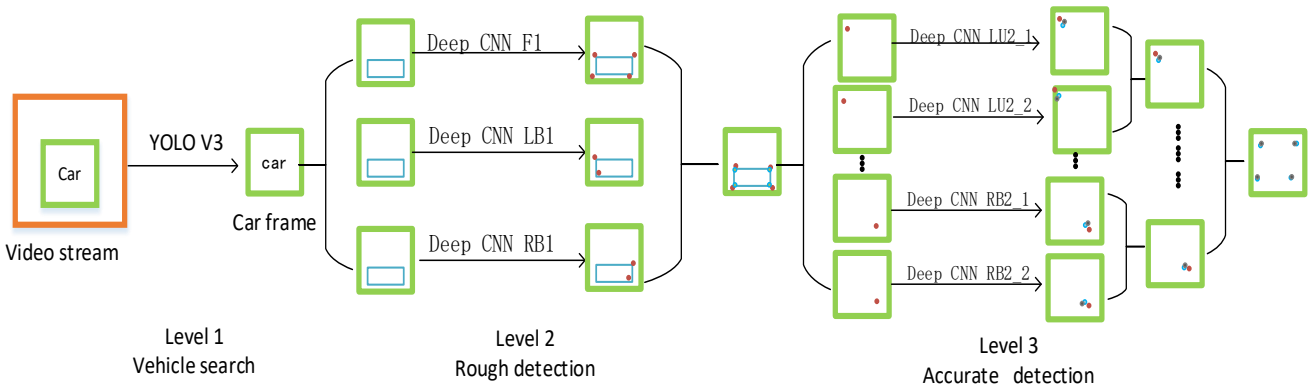

Figure 1. The structure of the CCNN model

2.1 Principle of the YOLO v3

The YOLO algorithm has a simple structure and makes prediction based on the global information of the image. The core idea is to take the entire image as the network output, and directly regress the position and class of the bounding box on the output layer.

2.1.1 Feature extraction network

In the YOLO v3, the Darknet-53 feature extraction network uses a 53-layer CNN, which is superposed by multiple residual blocks [14]. This network outperforms ResNet-101, ResNet-152and Darknet-19.

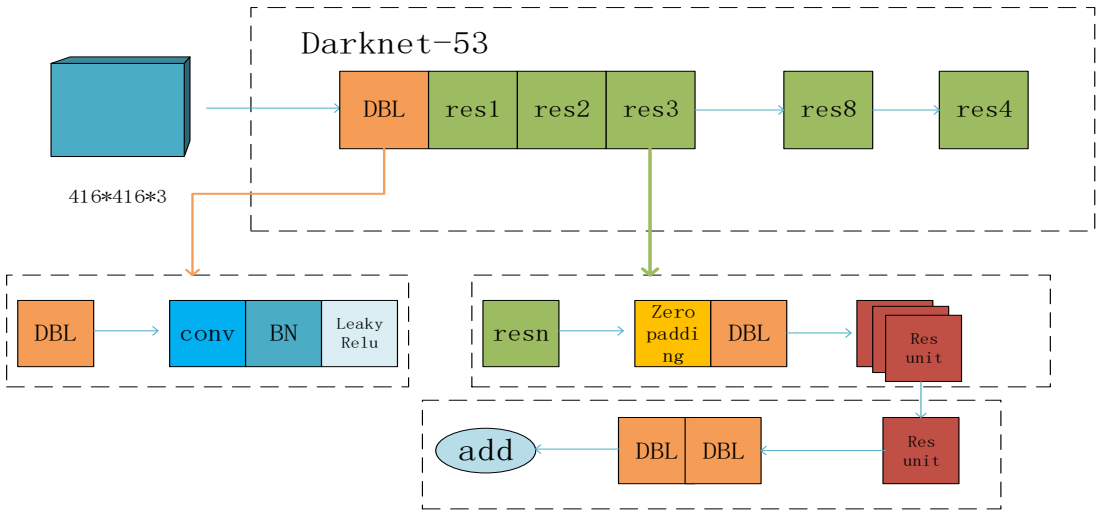

Figure 2. The structure of Darknet-53

The structure of Darknet-53 is shown in Figure 2, where DBL (conv+BN+Leaky ReLu) is the basic unit of YOLO v3. For YOLO v3, Bayesian network (BN) and Leaky Rectified Linear Unit (ReLU) are indispensable from the convolutional layer (conv), except that in the last layer. The three elements form the smallest component of the YOLO v3. The "resn" indicates the number ("n") of residual units (res_units) in the residual block (res_block), a large component of YOLO v3. Drawing on the residual structure of ResNet, YOLO v3 enjoys a deeper network structure than YOLO v2 (darknet-53 vs. darknet-19), which does not have the residual structure.

2.1.2 Input-output mapping

Each input image is mapped by YOLO v3 to output tensors of three different scales. The existence of the target differs with the positions in the image. As shown in Figure 3, for the 416*416 input image, three a priori boxes are set in each grid of the feature map on each scale. Thus, there are a total of 10,647 forecasts (13*13*3 + 26*26*3 + 52*52*3 = 10,647). Each forecast is an 85-dimensional vector ((4+1+80)=85), which covers the border coordinates (4 values), the border confidence (1 value) and the probability of target class.

Mimicking the structure of residual network, the YOLO v3 is a deep network capable of multi-scale detection. The network boasts an exceptionally high mean average precision (mAP), especially on small objects. Judging by COCO mAP-50, the YOLO v3 can complete the detection 3 or 4 times faster than the other models (Table 1). Figure 4 is the vehicle detection results of the YOLO v3 model on our dataset.

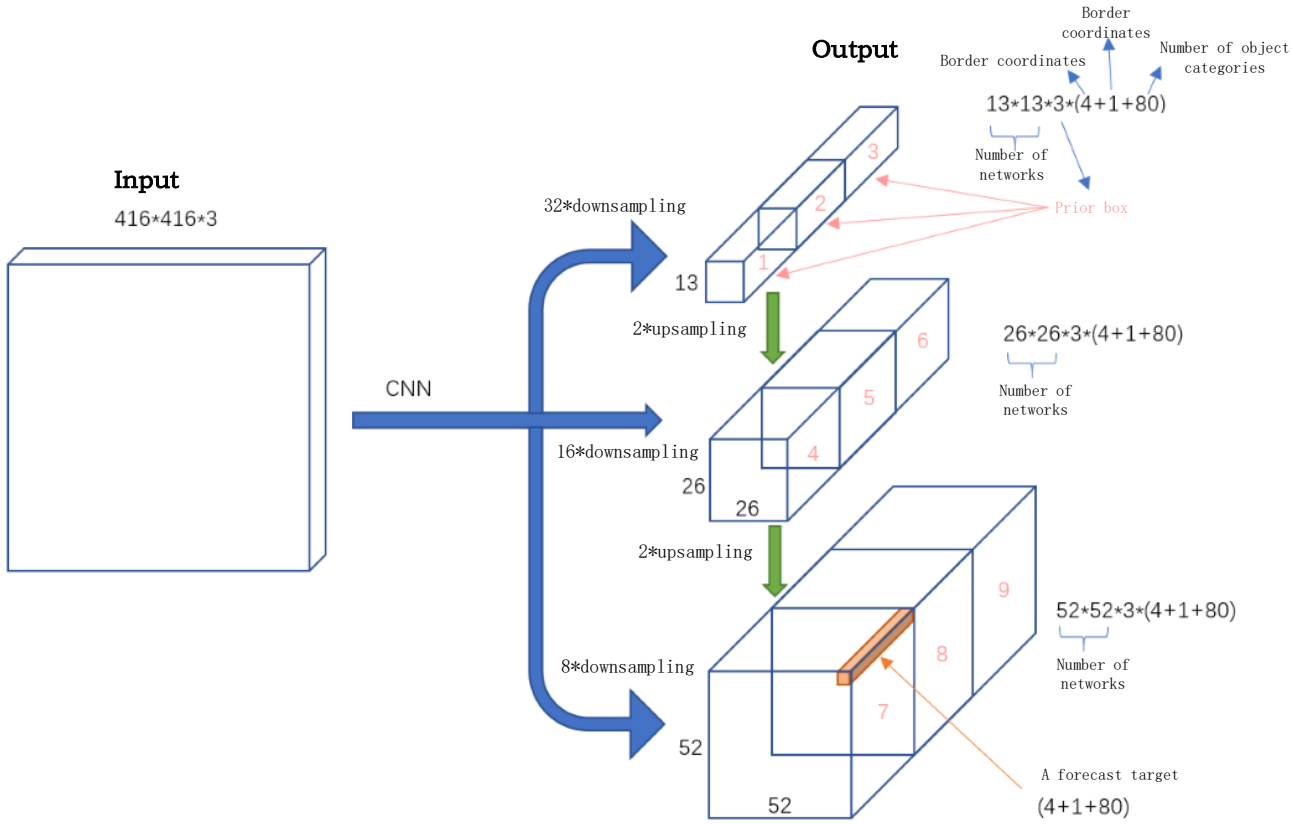

Figure 3. Input and outputs of YOLO v3

Table 1. Performance comparison between YOLO v3 and other networks

Method	mAP-50	time/ms
SSD321	45.4	61
DSSD321	46.1	85
R-FCN	51.9	85
SSD511	50.4	125
DSSD513	53.3	156
FPN FRCN	59.1	172
RetinaNet-50-500	50.9	73
RetinaNet-101-500	53.1	90
RetinaNet-50-800	57.5	198
YOLO v3-320	**51.5**	**22**
YOLO v3-416	55.3	29
YOLO v3-608	57.9	51

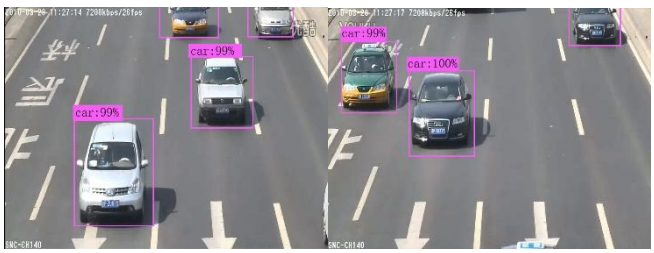

Figure 4. The vehicle detection results of the YOLO v3 model on our dataset

2.2 Rough detection of license plate on level 2

The detection on level 2 is realized through 3 CNN models: Deep CNN F1, Deep CNN LB1 and Deep CNN RB1 in Figure 1. Deep CNN F1 detects four key points; Deep CNN LB1 detects two key points in the upper left and the lower left; Deep CNN RB1 detects the two key points in the upper right and the lower right. The red points in Figure 1 are the key points roughly detected by these CNN models. Finally, the forecast values of each key point are averaged.

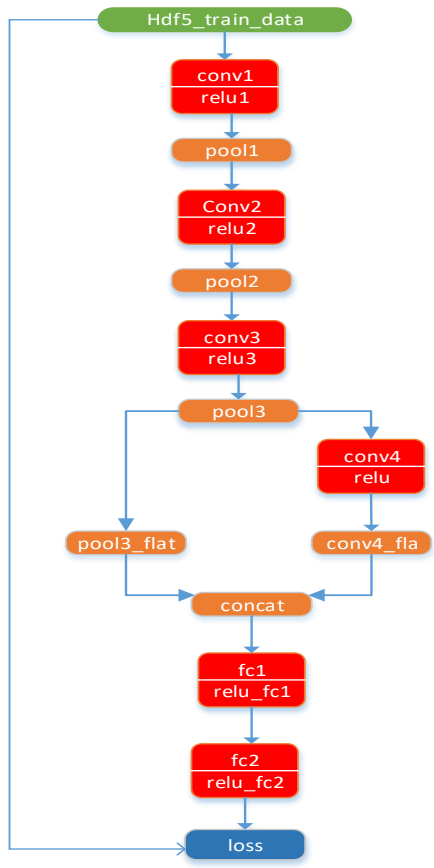

Figure 5. Network structure of Deep CNN F1

In Deep CNN F1, the key points are positioned by four convolutional layers (conv1, conv2, conv3, conv4) and two fully-connected layers (fc1, fc2). Each convolutional layer is followed by a ReLu operation and a max pooling layer (pool). The kernel sizes of the four convolutional layers are 4*4, 3*3, 3*3 and 2*2, respectively, and the kernel size is 2*2 for all max pooling layers.

As shown in Figure 3, I(140,180) stands for the size of the input image (140*180); Conv(4, 20, 137, 177) means the kernel size (4*4) of the first convolutional layer, the number of feature images (20), and the size of the output image (137*177); P(2, 20, 69, 89) indicates that the max pooling, with the stride of 2, outputs twenty 69*89 compressed features. The final outputs of the fully-connected layers are the 4 feature points to be forecasted.

Deep CNN LB1 and Deep CNN RB1 have basically the same structure of Deep CNN F1. The only difference lies in the size of the input image. Deep CNN LB1 needs to predict the two feature points on the left of the license plate. Thus, the network is designed with 4 output layer neurons, and only the left half of the image (containing the two left feature points) is inputted. Similarly, Deep CNN RB1 needs to predict the two feature points on the right of the license plate. Thus, the network is designed with 4 output layer neurons, and only the right half of the image (containing the two left feature points) is inputted.

2.3 Accurate positioning of license plate on level 3

As shown in Figure 3, the forecast results after the training of level 2 (the red points in level 3) were not as accurate as the manually marked blue points. To solve the problem, the forecast position of each key point was inputted to the CNN models of level 3 to further reduce the detection scope of the key points.

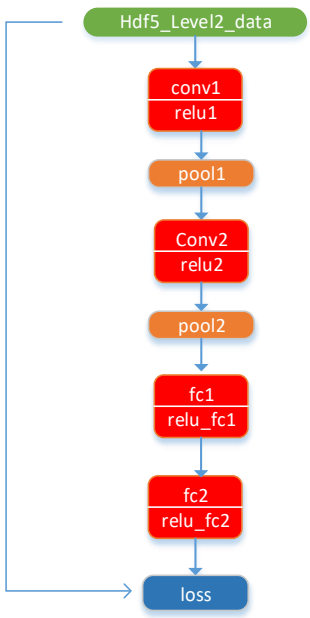

Figure 6. Network structure of Deep CNN LU2_1

Six simplified CNN models were designed for level 3, based on the rough detection of the key points in the license plate image. As shown in Figure 6, each model has two fewer convolutional layers and one fewer max pooling layer than that on level 2. In addition, the two datasets of each key point are trained separately by two CNN models. Thus, there are a total of 8 CNN models on level 3. Then, the forecast values of each key point are averaged, completing the accurate positioning of the four feature points in the image. The black points in Figure 1 are the key points estimated by level 3, which basically coincide with the manually marked blue points.

3. EXPERIMENTAL VERIFICATION

3.1 Data acquisition

The experimental data were collected from two sources: a library of the images extracted by OpenCV from surveillance video, and the open-source license plate database (3,000 images). The images with limited changes were eliminated, leaving 8,000 images. These images were divided into a training set (6,000 images) and a test set (2,000 images). Due to the resolution difference between video clips, the acquired images were of two sizes: 1,080c×720 and 1,920×1,080.

3.1.1 Data enhancement

The image data are often enhanced before image classification, because deep learning has a strict requirement on the size of dataset. If the original dataset is too small, the network model cannot be trained sufficiently, thus affecting the model performance. In this paper, the original dataset is expanded by rotating, scaling, cropping and translating the image files.

3.1.2 Data tagging

The LabelImg was employed to tag the data of all sample images of license plate (including the training set and the test set). The tagged text is shown in Figure 7, where the five red boxes are the coordinate tags of the plate frame, the upper left key point, the lower left key point, the upper right key point and the lower right key point, respectively.

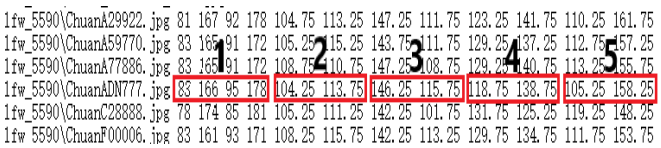

Figure 7. The tagged text

3.2 Results analysis

Considering the network structure of level 2 (rough detection of key points on license plate), 6,000 images were allocated into the training set and 2,000 into the test set. The network training was carried out in batches, each of which contains 32 images (140*180). In each batch of training, the weights were optimized, and the Euclidean distance between the forecast and actual key points, i.e. the loss, was computed. The initial learning rate was set to 0.01. After every 1,000 iterations, one test was conducted using the test dataset.

The training losses of level 2 and level 3 are displayed in Figures 8 and 9, respectively. It can be seen that both levels witnessed a decline in the loss with the increase in the number of iterations. For level 2, the mean loss of the four key points fell between 0.8 and 0.9, after 10,000 iterations of the training dataset. For level 3, the loss gradually approached the interval of 0.4~0.5 after 10,000 iterations. The mean loss of level 2 was greater than the loss of level 3. This means the CCNN can enhance the positioning accuracy of key points on license plate,

through the combination of rough detection and accurate positioning.

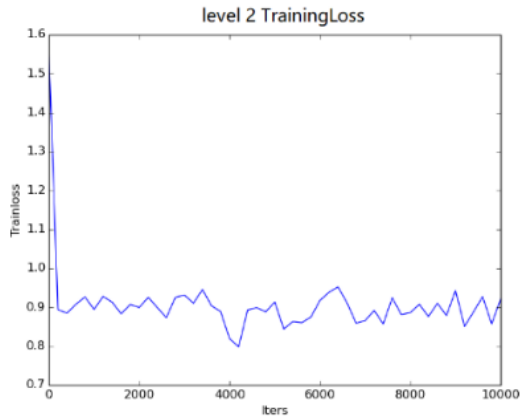

Figure 8. Loss trend on level 2

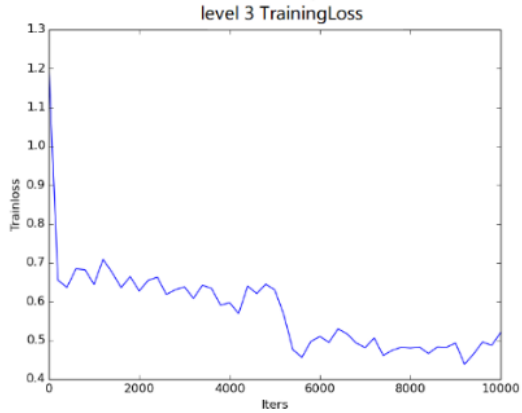

Figure 9. Loss trend on level 3

On level 2, the positions of the four key points were forecasted by the trained model, and the errors between the forecast positions and manually marked positions were calculated. Next, the errors on level 2 were compared with those on 3 (Figure 10). The comparison shows that the error of each key point on level 2 was greater than that on level 3. This further confirms the CCNN's ability to accurately position the key points.

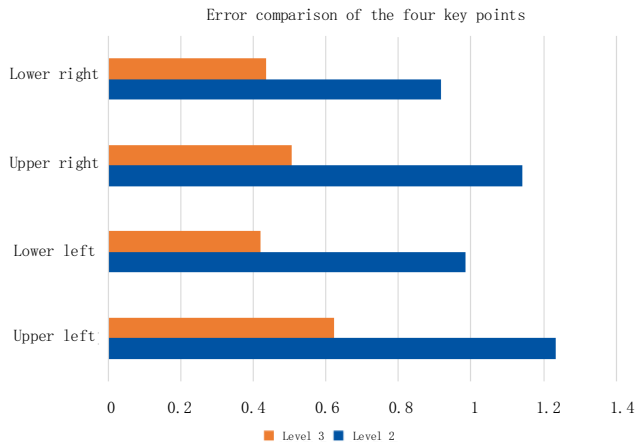

Figure 10. Comparison between the errors of key points on level 2 and level 3

Table 2. Key point positioning accuracies of different algorithms

Algorithm	Number of samples	Correct positioning	Incorrect positioning	Hit rate (%)
Pixel difference edge detection algorithm	2,000	1,652	348	82.6
Texture feature detection algorithm	2,000	1,786	214	89.3
Single CNN algorithm	2,000	1,831	207	89.6
The CCNN	2,000	1,916	84	95.8

Figure 11. Positioning effects on level 2 and level 3

Furthermore, the classic edge-based license plate positioning algorithm (pixel difference edge detection algorithm) and texture-based license plate positioning algorithm (texture feature detection algorithm) were tested on the 2,000 images in our test dataset. The results show that our CCNN algorithm outperformed the pixel difference edge

detection algorithm [15], the texture feature detection algorithm [16] and the single CNN algorithm [17] in positioning accuracy. The relatively poor effects of the traditional algorithms are attributed to their sensitivity to the environmental factors, as there is no limit on the scenes in the test images. By contrast, the CCNN model, trained by deep learning, can stay immune to the complex background and light intensity in the images.

4. CONCLUSIONS

This paper designs the CCNN, an accurate positioning method for license plate. Firstly, the vehicles in the video stream were detected by the YOLO v3 network. Then, two CNN layers were designed to roughly detect and accurately position the license plate, respectively. The CCNN was proved to be highly robust and accurate, despite natural light intensity, license plate covers, or noises outside the license plate. The research findings lay a solid basis for character segmentation and recognition on the license plate.

REFERENCES

[1] Yang, S., Zhang, B., Zhang, Z.J. (2016). Vehicle license plate localization algorithm based on multi-feature fusion. Journal of Computer Applications, 36(6): 1730-1734. https://doi.org/10.11772/j.issn.1001-9081.2016.06.1730

[2] Zhang, S., Gong, Y.H., Wang, J.J. (2019). The development of Deep Convolution Neural Network and Its Applications on Computer Vision. Chinese Journal of Computers, 42(3): 453-482.

[3] Girshick, R., Donahue, J., Darrell, T. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. Computer Vision and Pattern Recognition, 580-587.

[4] Debojit, B., Su, H.B., Wang, C.Y., Blankenship, J., Aleksandar, S. (2017). An Automatic car counting system using OverFeat framework. Sensors (Basel, Switzerland), 17(7): E1535. https://doi.org/10.3390/s17071535

[5] Wu, W.Q., Yin, Y.J., Wang, X.G., Xu, D. (2019). Face detection with different scales based on faster R-CNN. IEEE Transactions on Cybernetics, 49(11): 4017-4028. https://doi.org/10.1109/TCYB.2018.2859482

[6] Ren, S.Q., He, K.M., Girshick, R., Sun, J. (2017). Faster R- CNN: Towards real- time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6): 1137-1149. https://doi.org/10.1109/TPAMI.2016.2577031

[7] Liu, W., Anguelov, D., Erhan, D. (2016). SSD: Single shot multibox detector. European Conference on Computer Vision, 9(17): 21-37.

[8] Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You only look once: Unified, real-time object detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society. https://doi.org/10.1109/CVPR.2016.91

[9] Zhang, X., Yang, W., Tang, X.L., Liu J. (2018). A fast learning method for accurate and robust lane detection using two-stage feature extraction with YOLO v3. Sensors (Basel, Switzerland), 18(12): 4308. https://doi.org/10.3390/s18124308

[10] Li, J.Y., Su, Z.F., Geng, J.H., Yin, Y.X. (2018). Real-time detection of steel strip surface defects based on improved YOLO detection network. IFAC PapersOnLine, 51(21). https://doi.org/10.1016/j.ifacol.2018.09.412

[11] Sun, Y., Wang, X., Tang, X. (2013). Deep convolutional network cascade for facial point detection. Computer vision and Pattern Recognition, 9(4): 3476-3483.

[12] Kimura, M.,Yamashita, T., Yamauchi, Y., Fujiyoshi, H. (2015). Facial point detection based on a convolutional neural network with optimal mini-batch procedure. IEEE International Conference on Image Processing, 9: 2860-2864.

[13] Girshick, R., Donahue, J., Darrelland, T., Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 580-587. https://doi.org/10.1109/CVPR.2014.81

[14] Cireşan, D., Meier, U., Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. Eprint Arxiv, 157(10): 3642-3649. https://doi.org/10.1109/CVPR.2012.6248110

[15] Wei, Y.K., Ou, Y.F. (2018). License plate location algorithm based on adjacent pixels difference. Computer Engineering and Design, 39(5): 1387-1392+1404

[16] Chen, H.Z., Xie, Z.G, Lu, H.L. (2018). License plate location method combining color and edge texture. Modern Electronics Technique, 41(21): 67-70+75.

[17] Lin, Z.C., Zhang, J.X. (2018). License plate recognition method based on GMP-LeNet network. Computer Science, 45(S1): 183-186.