

## Research Article

# License Plate Detection with Shallow and Deep CNNs in Complex Environments

Li Zou,<sup>1</sup> Meng Zhao ,<sup>1</sup> Zhengzhong Gao,<sup>1</sup> Maoyong Cao,<sup>1</sup>  
Huarong Jia,<sup>2</sup> and Mingtao Pei <sup>2</sup>

<sup>1</sup>College of Electrical Engineering and Automation, Shandong University of Science and Technology, Qingdao 266590, China

<sup>2</sup>Beijing Laboratory of Intelligent Information Technology, Beijing Institute of Technology, Beijing 100081, China

Correspondence should be addressed to Meng Zhao; zhaomeng@sdust.edu.cn

Received 14 August 2018; Revised 31 October 2018; Accepted 11 November 2018; Published 2 December 2018

Guest Editor: Junpei Zhong

Copyright © 2018 Li Zou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

License plate detection is a challenging problem due to the large visual variations in complex environments, such as motion blur, occlusion, and lighting changes. An advanced discriminative model is needed to accurately segment license plates from the backgrounds. However, effective models for the problem tend to be computationally prohibitive. To address these two conflicting challenges, we propose to detect license plate based on two CNNs, a shallow CNN and a deep CNN. The shallow CNN is used to quickly remove most of the background regions to reduce the computation cost, and the deep CNN is used to detect license plate in the remaining regions. These two CNNs are trained end to end and are complementary to each other to guarantee the detection precision with low computation cost. Experimental results show that the proposed method is promising for license plate detection.

## 1. Introduction

License plate recognition is an important and popular research issue in image processing and computer vision. Applications [1–5] based on license plate detection are playing an increasingly important role in our daily life, such as unattended parking lots, security control of restricted areas, congestion pricing, and automatic toll collection. License plate detection is the basis of license plate recognition. Although the research of license plate detection started relatively early, the detection systems today are still not perfect; a number of factors such as motion blur, occlusion, and lighting changes can lead to large visual variations in plate appearance, which can severely degrade the performance of the license plate detector.

The license plate feature is usually designed manually based on the license plate standard defined by the industry, such as text, edge, and color. Category Specific Extremal Region proposed by Matos [1] is a representative of the method based on the characteristics of the text region. This algorithm does not need to set a global threshold for the characters and background of the license region and is robust

to motion blur, lighting changes, and occlusion. However, this algorithm requires that the characters and background of individual license can be binarized and thus cannot handle the situation of license staining, motion blur, and intense lighting. Besides, if the license includes certain characters of which the pixels are not simply connected, the algorithm cannot be used directly. Liang [2] used Sobel [6] operator to detect edges in the input image, applied mathematical morphology processing to the image to obtain connected candidate regions of the license plate, and analyzed the candidate regions to locate the license plate region. License plate detection based on edge features has relatively high detection efficiency, especially in the situation in which the environment is constrained such as the front gate. Nevertheless, this kind of methods is sensitive to edge noise and illumination; it is difficult to apply this kind of methods to complex scene.

Viola [7] proposed a cascade classifier based on Haar-like feature. It is successfully applied to face detection, text detection, and license plate detection. However, due to the simple nature of the Haar feature, it is relatively weak in the uncontrolled environments. CNN is very popular in

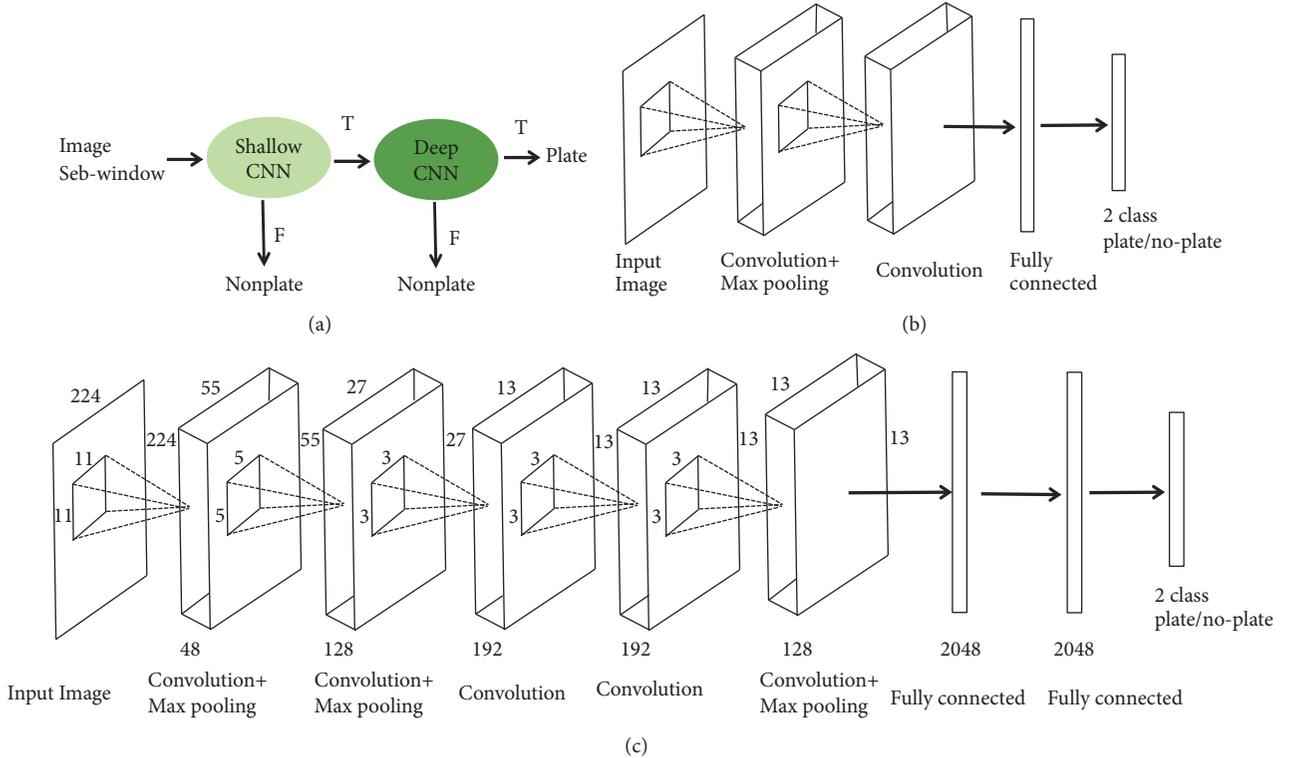


FIGURE 1: The framework of the proposed license plate detector. (a) The overall framework of our license plate detector. (b) The structure of the shallow CNN. (c) The structure of the deep CNN.

object detection now. Garcia [8] successfully applied CNN to face detection. Chen [9] modified the CNN's structure to detect the license plate, and the results turned out to be remarkable. Compared to previous hand-crafted features, CNN [10] automatically learns the features of license plate through tremendous training data. It is stable to geometric transformation, shape changes, and illumination changes. Xie [11] proposed a CNN-based framework for multidirectional license plate detection. Kurpie [12] modelled a CNN which produced a score for each image subregion to estimate the locations of license plates by combining the results obtained from sparse overlapping regions. Masood [13] proposed a deep CNN which could work across a variety of license plate templates. License plate detection based on CNN has achieved promising results and is more robust than traditional methods. However, exhaustively scanning the full image with a deep CNN is not a practical solution, since there is mass amount of scanning windows that need to be classified, which has a very high computational cost.

To reduce the computational cost, we propose to detect license plate based on two CNNs, a shallow CNN and a deep CNN. The shallow CNN is used to quickly remove most of the background regions to reduce the computation cost, and the deep CNN is used to detect license plate in the remaining regions. The two CNNs are trained end to end and are complementary to each other to guarantee the detection precision with low computation cost.

## 2. Methods

**2.1. Overall Framework.** The overall framework of our license plate detector is shown in Figure 1(a), and the structures of the two CNNs are shown in Figures 1(b) and 1(c), respectively. Given a test image, the shallow CNN scans the whole image densely to quickly reject most of the candidate windows. The remaining candidate regions will be classified by the deep CNN. Compared with the shallow CNN, the deep CNN is more powerful but slower since it is more complicated. We adopt the nonmaximum suppression (NMS) to the regions that passed the deep CNN to eliminate highly overlapped detection windows and finally obtain the plate's location.

**2.2. The Shallow CNN.** The shallow CNN is used to quickly scan the testing image to reject most of the candidate windows. A  $140 \times 40$  window is used to scan the whole image densely to obtain a map of confidence scores; the size of the map is as follows:

$$\left( \left\lfloor \frac{(W - 140)}{5} \right\rfloor + 1 \right) \times \left( \left\lfloor \frac{(H - 40)}{2} \right\rfloor + 1 \right) \quad (1)$$

where  $W$  and  $H$  are the width and height of the input image and the stride is 5 pixels in  $x$  direction and 2 pixels in  $y$  direction.

The shallow CNN is used to quickly remove most of the candidate windows; therefore, it just has 4 layers for time efficiency. The first layer is a convolution layer followed by



FIGURE 2: Candidate regions obtained by the shallow CNN.

a max pooling layer. The next layer is also a convolution layer. The last layer is a fully connected layer, which has a soft-max of 2 outputs. When training the shallow CNN, we increase the weight of the positive sample in the loss, which exacerbates the punishment of making a wrong classification for the positive samples. At the same time, we decrease the weight of the negative samples in the loss. That is, the shallow CNN will have high false positive rate and low miss rate. The false positives which passed the shallow CNN will be rejected by the following deep CNN. Figure 2 shows an example of the candidate regions obtained by the shallow CNN.

**2.3. The Deep CNN.** We use the Alex Net [14] as the deep CNN for license plate detection. The structure of the deep CNN is shown in Figure 1(c). The deep CNN is a powerful binary classifier. The deep CNN extracts the Hierarchical Feature through multilayer convolutional layers and gets invariance by pooling layer through down-sampling and finally classifies the images by a fully connected layer on the top.

Alex Net uses a method called Local Response Normalization. Meantime, Alex Net uses dropout when selecting the nodes. Both promote the performances of Alex Net. However, the simplest method to enhance the model's performance and avoid over-fitting is to increase the size of training data. The data we used comes from cameras on road which capture vehicles' image under natural scene. We label the license plate manually to collect 10774 license plates positive samples and randomly select 10029 patches from nonlicense plate regions to establish negative samples. These positive and negative samples are resized to  $227 * 227$  to fine-tune a pretrained Alex Net model.

The shallow CNN rejects most of the detection windows while keeping almost all of the license plate regions. The remaining candidate windows are classified by the deep CNN. If a candidate window is classified as a license plate, we add it into a ranked list  $L$ , ranked by the score given by the deep CNN. And we apply nonmaximum suppression to get the final license plate location. That is, for each candidate window  $W$  in  $L$ , starting with the highest score, we remove all windows that overlap with  $W$ , since two license plates will not overlap with each other in real scenes.

### 3. Baseline: Remove Candidate Regions by Edge Density

To evaluate the proposed method, we also implement a baseline; that is, we use the edge density as feature to remove candidate regions as in [15]. There are many edges in the license plates formed by the characters and the background. Therefore, edge density can be used to remove candidate regions.

We use Sobel operator for edge detection [16]. Technically, it is a discrete function's gradient approximation. Using this operator to process any point in the image can produce the corresponding gradient vector. This operator includes two matrices of  $3 * 3$  which represent transverse and portrait. Letting it convolute with the image will get brightness difference approximation of transverse and portrait. Let  $A$  represent the original image and  $G_x$  and  $G_y$  represent edge detection image of transverse and portrait; the function is presented as follows:

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A, \quad (2)$$

$$G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * A$$

Owing to plate's edge feature, here we only compute the  $|G_x|$ . The most intense response production is which  $X$  is perpendicular with, that is, vertical edge.

We use sliding window method to scan the whole image and compute the average edge density of every corresponding window. Similar to [15], we divide each window into 3 small windows and compute the difference between the 3 small windows; if the difference is too large, the window will be rejected. The window size and the strides in  $x$  and  $y$  directions are the same as in Section 2.2.



FIGURE 3: Some sample images of BIT-Vehicle Dataset.

TABLE 1: Comparison results.

Method	The number of candidate windows	Precision
Baseline	200	96.25%
Shallow + deep CNNs	20	98.1%

## 4. Experiments

**4.1. Datasets.** We collected a complex and challenging vehicle dataset called BIT-Vehicle Dataset to test our method. BIT-Vehicle Dataset is a complex and challenging dataset, which includes 10400 labelled vehicle images. The dataset has 1600\*1200 and 1920\*1080 resolution's images, but the top or bottom parts of some vehicles are not included in the images because of the capturing delay and the size of the vehicle. They are captured by two cameras at different time and place. The reason why the dataset is complex and challenging is that the images include differences in illumination, surface color, and angle of view. Figure 3 shows some sample images of the dataset. We split the dataset as training set, validation set, and testing set. The training set contains 8400 images, the validation set contains 1200 images, and the testing set contains 800 images.

**4.2. Results.** We implemented the proposed plate detector using the Cafe library and tested our program on BIT-Vehicle Dataset. When training the shallow and deep CNN, we use the labelled license plates in the training set as positive samples and randomly select nonlicense plate regions as negative samples. The number of positive samples is 10774, which is more than the number of images in the training set; this is because some images contain more than one license plate as shown in Figure 3. The number of negative samples is 10029. When training the shallow CNN, we increase the

weights of the positive samples and decrease the weights of the negative samples to guarantee a low miss rate. During the training of the deep CNN, the weights of the positive and negative samples are the same.

For each image, after passing the shallow CNN, there are about 20 candidate windows left for the deep CNN to classify. After training, the deep CNN achieves 99.2% accuracy on the validation set and achieves 98.1% accuracy on the testing set.

Just like what we expected, deep CNN can classify the candidate regions very well. The reason why the deep CNN can achieve high performance is that our images in our dataset are captured under different scenes with different viewing angles and under different illumination conditions and CNN can learn discriminative features from the training data automatically.

We enhance the accuracy rate, and our model can detect license plate faster. We implement this model on café and it can proceed 6 frames per second on a computer without GPU. With GPU's accelerating, our model's efficiency will be promoted dramatically and can detect license plate in real time.

**4.3. Comparison Results.** The experiment is divided into two parts. One is based on edge detection and CNN, and we regard its results as a baseline. The other is based on shallow and deep CNN. Finally, we compare the experimental results of the two methods as in Table 1.

The shallow and deep CNN perform better than the baseline; the reason is that the parameters of the baseline



FIGURE 4: The license plate detection results with our method.

are set manually, but the parameters of the shallow CNN are learned from the data. Figure 4 shows some detection results.

We also compare the running time of our method and just use deep CNN to detect license plate in sliding windows manner on a computer with an Intel Core i5-3230M CPU (2.6GHz). The average running time for one image with resolution 1600\*1200 is about 166 ms for our method and is about 500 ms for just using deep CNN to detect license plate in sliding windows manner.

## 5. Conclusions

In this article, we propose a license plate detection method based on cascade architecture. Firstly, a shallow CNN is used for time efficiency. A shallow CNN scans the whole image densely to quickly eliminate most nonlicense plate regions. Secondly, a deep CNN is used to classify the remaining regions to detect the license plate. Finally, we use nonmaximum suppression (NMS) to get the final license plate region. By using shallow and deep CNNs, we can detect license plate fast and accurately. We test out the method on BIT-Vehicle Dataset and the result shows that the proposed method promotes both the accuracy and the time efficiency.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

## Acknowledgments

This work was supported in part by the Natural Science Foundation of China (61471225).

## References

- [1] J. Matas and K. Zimmermann, "Unconstrained licence plate and text localization and recognition," in *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pp. 225–230, Vienna, Austria, September 2005.
- [2] L. Liang and X. Wang, "License plate positioning algorithm based on Sobel operator edge detection and mathematical morphology," *Modern Electronics Technique*, 2015.
- [3] C. Yang, H. Wu, Z. Li, W. He, N. Wang, and C. Su, "Mind control of a robotic arm with visual fusion technology," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 3822–3830, 2018.
- [4] C. Yang, C. Zeng, P. Liang, Z. Li, R. Li, and C.-Y. Su, "Interface design of a physical human-robot interaction system for human impedance adaptive skill transfer," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 1, pp. 329–340, 2018.
- [5] C. Wang, X. Liu, X. Yang, F. Hu, A. Jiang, and C. Yang, "Trajectory tracking of an omni-directional wheeled mobile robot using a model predictive control strategy," *Applied Sciences*, vol. 8, no. 2, 2018.
- [6] I. Sobel, "Camera models and machine perception," DTIC Document Technical Report, 1970.
- [7] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.
- [8] C. Garcia and M. Delakis, "A neural architecture for fast and robust face detection," in *Proceedings of the Sixteenth International Conference on Pattern Recognition*, vol. 2, pp. 44–47, 2002.
- [9] Y.-N. Chen, C.-C. Han, C.-T. Wang, B.-S. Jeng, and K.-C. Fan, "The application of a convolution neural network on face and license plate detection," in *Proceedings of the 18th International Conference on Pattern Recognition (ICPR '06)*, pp. 552–555, August 2006.
- [10] S. Ren, K. He, and R. Girshick, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2016.
- [11] L. Xie, T. Ahmad, L. Jin, Y. Liu, and S. Zhang, "A new CNN-based method for multi-directional car license plate detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, pp. 507–517, 2018.
- [12] F. D. Kurpiel, R. Minetto, and B. T. Nassu, "Convolutional neural networks for license plate detection in images," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '17)*, pp. 3395–3399, September 2017.
- [13] S. Masood Z, G. Shu, A. Dehghan et al., "License plate detection and recognition using deeply learned convolutional neural networks," *Computer Vision and Pattern Recognition*, 2017, <https://arxiv.org/abs/1703.07330>.
- [14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 26th Annual Conference on Neural Information*

*Processing Systems (NIPS '12)*, pp. 1097–1105, Curran Associates Inc., December 2012.

- [15] H. Zhang, W. Jia, X. He, and Q. Wu, “Learning-based license plate detection using global and local features,” in *Proceedings of the 18th International Conference on Pattern Recognition*, pp. 1102–1105, IEEE, Hong Kong, China, August 2006.
- [16] L. C. Chen, J. T. Barron, G. Papandreou, K. Murphy, and A. L. Yuille, “Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform,” *Computer Science*, vol. 9, no. 1, pp. e84558–e84558, 2016.

