

Licence Plate Detection Using Machine Learning

Navjeevan Chaudhary¹, Sunil Kumar S. Manvi²

^{1,2}School of Computing and IT REVA University Bangalore, India.

Email: jeevanchy@gmail.com¹, ssmanvi@reva.edu.in²

*Corresponding author's E-mail: jeevanchy@gmail.com

Article History	Abstract
Received: 06 June 2023 Revised: 05 Sept 2023 Accepted: 30 Nov 2023 CC License CC-BY-NC-SA 4.0	<i>License Plate Recognition (LPR) is one of the tough tasks in the field of computer vision. Although it has been around for quite a while, there still lies the challenges when we have to deal with; the harsh environmental conditions like snowy, rainfall, windy, low light conditions etc. as well as the condition of the plates which includes the bent, rotated, broken plates. The performance of the recognition and detection frameworks take a significant hit when it is concerned with these conditional effects on the license plate. In this paper, we introduced a model to improve our accuracy based on the Chinese Car Parking Dataset (CCPD) using 2 separate convolutional neural networks. The first CNN will be able to detect the bounding boxes for the license plate detection using Non-Maximus Suppression (NMS) to find the most probable bounding area whereas the second one will take these bounding boxes and use the spatial attenuation network and character recognition model to successfully recognize the license plate. First, we train the CNN to detect the license plates, then use the second CNN to recognize the characters. The overall recognition accuracy was found to be 89% in the CCPD dataset.</i> Keywords: Non-Maximus-Suppression(NMS), Spatial Network, Detection, Recognition

1. Introduction

Licence Plate Detection and Recognition (LPR) is a crucial element implied in the fields of vehicular transportation systems and is a difficult and necessary. It is implemented in traffic management systems, digital security systems, police work, vehicle recognition, parking management, etc. Most of the methods in the LPR use the following methods of implementation; analyse the image or videos captured by the capturing device, process the images using digital image processing, recognise the pattern using pattern recognition or alternative technologies to get the vehicle plate registration number and colour information of the plate. In the real world, the images are taken over days and nights in extremely complicated atmosphere conditions with variable illuminations and completely different lighting conditions. The key process includes localization, segmentation and finally the recognition of the plate which becomes a difficult task to perform. In this paper, we introduce a method that works using the convolutional neural network for the detection of plate region and finally the recognition of characters from the licence plate. This paper presents the following contributions to the pre-existing works:

- 1) We introduce a real-time convolutional network-based method to detect the licence plate by training it with a huge number of images of different weather conditions and other various defects.
- 2) We use the spatial networks to make the recognition performance higher.

The rest sections of the paper is described as follows. Section II, gives a brief about the literature survey conducted. Section III explains the methodology and the approach used. Section IV is about the proposed system, dataset description, data analysis, data pre-processing, data exploration. Section V is about result analysis followed by conclusion in Section VI and References.

Related Works

Chris Henry and Sang-Woong lee [1] used deep networks in multinational license plate classification and detection. Unified character recognition was the main method that was applied in recognizing the

characters using the YOLOv3-SPP. The features are combined to create a spatial pyramid of features. These features will be used in the recognizing the character features from the image. The plate regions are also searched using the YOLO framework itself. Xie and Jen [2] used a multi directional approach on top of the YOLO based framework. This combined framework formed a system to detect the plate. CNN was used as this provided a high accuracy in the detection and the detection time being small allowed it to be used in the real time scenarios. They used rotational angle prediction and Intersection Over Union (IoU) for the detection. This significantly reduced the problems that were faced with the rotated plates. L. Xie [3] introduced a new method to validate the intersection over union by the use of penalty factors. To get the precision higher in detection they used the intersection over union using 2 rotational angles and this was used in the penalty factor. They also used a CNN to find the region for the attention.

Yongjie Zou, Youngjun Zhang, Jun Yan, Xiaoxu Jiang [4] used the Bi-LSTM for a robust license plate recognition model. It had significant advantages like activating the regional features of characters and extracting the character of license plates, then using the Bi-LSTM to locate each license plate character and also their location information. 1D-Attention mechanism was the key to extract only the useful character. This reduced useless features to increase the effective recognition precision of character features from license plates.

V. Pustokhina [5] used the 3-step process to detect the license plate. First the license plate is localized and then Bersen Algorithm and CCA models are used for the licence plate detection. The characters are then segmented using clustering and Krill Herd Algorithm along with optical K-Means. The CNN Model is then used for the recognition of characters in the image.

C. Y. Suen [6] used separate YOLO frameworks for the detection and recognition process. The system used separate CNN for the detection of the licence plate and then separate one for the character recognition process. Various data augmentation techniques were applied to increase the dataset size for training the model to get a better accuracy.

Ching Yee Suen [4] combined the bi-directional LSTM and the 1D attention using heat maps and the position to detect the licence plates. It contains the region extraction and the useful and less useful region classification from the image. The features are taken from the useful region and it is used to detect the characters.

V. Pustokhina [5] used the 3-step process to detect the license plate. Bersen Algorithm is used for localization of the licence plate and CCA models are used for detection. The characters are then segmented using clustering and Krill Herd Algorithm along with optical K-Means. The CNN Model is then used for the recognition of characters in the image.

approach

In this section, architecture of the 2 CNN's that we have used for detection and the underlying recognition network for the license plate is shown.

Detection Structure

Neural Networks have proven to be efficient in finding the features. Deep Neural Networks even take this to a much further domain. Real world applications require a high precision as well as faster processing times. In today's world we have seen a great improvement in the precision as well execution times due to the advancement of the computer vision. Li et al. [16] used a cascaded CNN method which was able to maintain a high accuracy rate but with the lower time consumption by cascading the lower performance classifier to create a strong classifier. Our system is based on cascaded CNN method [17] and it is implemented using the MTCNN algorithm [18]. The MTCNN are generally used for face detection and recognition. We have modified the algorithm so that it can be used in license plate detection. The detection network consists of 2 CNN's. The first CNN will be used to find the bounding boxes for the license plates. These boxes are nothing but the coordinates for the license plate. The input images are scaled to various sizes and they will generate a feature pyramid. These give us candidates of the licence plate and the regression vector. All these candidates are then combined using the non-Maximum suppression. NMS [18] is a technique where use the regression method to find the most optimal features for our detection purpose. Each feature has a score which will give a measure of its precision. For the choosing of the windows the one with the highest score is considered and other windows which are overlapped in the image is discarded.

The modified architecture of P-Net and O-Net can be shown below

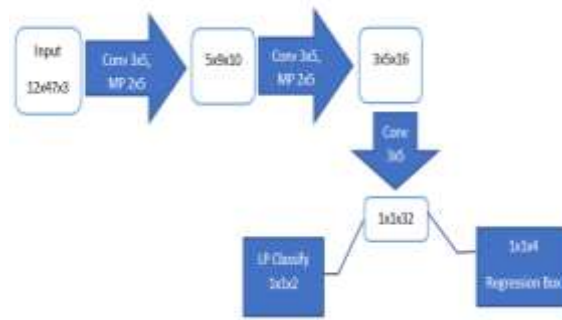


Fig 1: Modified P-Net Architecture

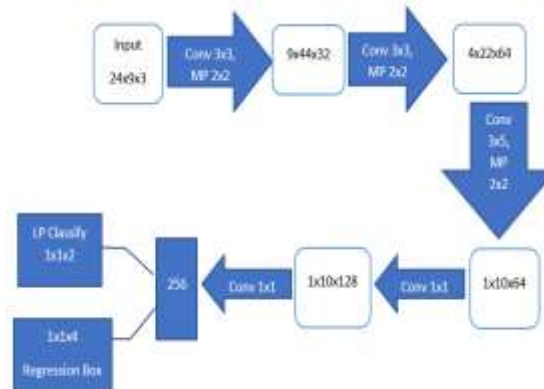


Fig 2: Modified O-Net Architecture

Here convolution layer is using stride of 1 and the pooling layer uses a stride of 2.

Training The Network

Here we employ 3 simultaneous tasks for training our CNN. The tasks include plate classification, bounding box regression and plate localization. The tasks will generate various bounding boxes from the smaller CNN and then refined again using the second CNN and merged using non-Maximus suppression. For the training purpose, we take the images from the dataset containing all the different types of challenges. The images consist of rotating plates, broken plates, low illumination plates and other various challenging images which are harder for the detection algorithms. This improves the performance of the model.

Recognition Structure

The recognition network consists of a small basic block that is integrated into the CNN. The architectures of the small basic block and the underlying recognition network are shown below

Layer	Dimensions
Input	94x24 pixels RGB Image
Convolution	64 channel 3x3 (stride 1)
Max Pooling	64 channel 3x3 (stride 1)
Small Basic block	128 channel 3x3 (stride 1)
Max Pooling	64 channel 3x3 stride(2,1)
Small Basic block	256 channel 3x3 (stride 1)
Small Basic block	256 channel 3x3 (stride 1)
Max Pooling	64 channel 3x3 stride(2,1)
Dropout	0.5
Convolution	256 channel 4x1 (stride 1)

Dropout	0.5
Convolution	1x13 (stride 1)

Table 1: Recognition Network Backbone

Layer	Dimensions
Input	H× W feature map
Convolution	1x1 1 stride
Convolution	3x1 (stride, pad = 1)
Convolution	1x3 1 (stride, pad = 1)
Convolution	1x1 (stride 1)

Table 2: Small Basic Block

Experimental Result

A. DATASET

We mainly use and compare the performance of the Chinese Car Parking Dataset (CCPD) [19] which have been already applied to this dataset. The CCPD dataset contains nearly about 250K images with different conditions like different backgrounds, different rotation angles of the image, different shooting angles of the license plate and various other plates which were taken in different environmental conditions. It also contains the sub folders for various types of images. The folders include base, DB, FN, Rotate, Weather and Challenge. Each of these contains the various types of images for our purpose.

B. EVALUATION METRICS

The dataset CCPD which we have been using contains only the license plate location embedded in the label of the image. Hence, we can get compare our predicted location with the original location for the license plate which is in the annotation. The final detection model will use all the previously explained techniques to give a final output location. The only way to measure whether our output is correct or not is if it matches the annotated name, then it is considered as correct; Else it considered as incorrect. The annotated number of license plate can be taken as the ground truth or the actual value. We will consider how many our model has predicted correctly. Also, one thing to consider is that some of the images don't contain any plate number in them. Hence incorrectly annotated. We will not consider them in the ground truth values. The regression using the IoU values will give us a score in percentage determining their accuracy. If the accuracy is 70%, we will consider them to be correct. The Accuracy precision can be defined by the below-given formula

$$Accuracy = \frac{Char.Matched}{Total Chars} \times 100\%$$

The final recognition accuracy can be calculated from the above given formula. Since the model will predict something from the image, it will be matched with the characters after decoding from the label.

3. Results and Discussion

Here, we will compare the accuracy of our result set with the existing methods that have already been applied to the CCPD Dataset. The system was implemented in PyTorch; a tool to implement our neural network using visual studio code and google collab notebook. The training and validation purposes were carried out in the google collab notebook. For the detection framework, we have chosen 15 epochs. This can be shown as below

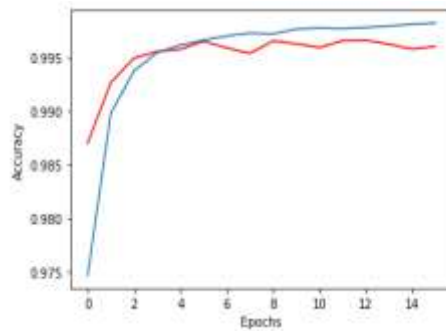


Fig 3. Training accuracy vs epochs

The execution time for the recognition framework for each image can be shown as below

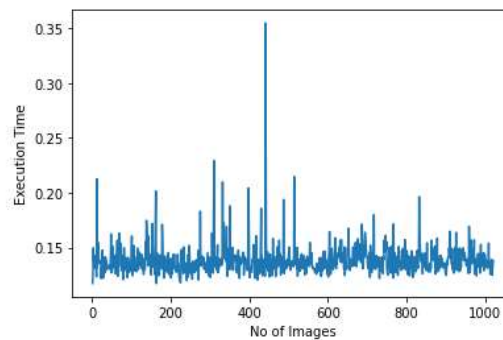


Fig 4. Per image execution time

As we can see that the execution time for each image is nearly 0.15s and max of 0.35s. the graph shows execution time taken by each image when the system was implemented over 1k images. The final model accuracy was tested over the various subsets of images containing all the various conditions of the license plate as well as the challenging ones. Each of the algorithms have been already tested and a base comparison has been provided in the CCPD dataset. We compare our results with the existing ones. They have been shown in the below table.

Table 3: Accuracy evaluation

Mo del	Fast er- RC NN	SSD 300	SSD 512	YO LO v3- 320	Pro pose d Mo del
Bas e	84.9	86.9	87.8	87.2	92.1
DB	66.7	72.9	69.9	71.3	96.3
FN	76.4	74.8	80.6	82.4	97.3
Rota te	94.4	96.5	96.5	96.6	95.1
Tilt	88.1	91.8	91.2	89.1	96.4
Chal leng e	89.9	90.0	92.1	91.4	88.8

4. Conclusion

License plate detection and recognition face the challenge of getting higher accuracy with lower execution times to be able to be used in the real world. Smaller models can help the system to be sustainable in a long time. Challenges have always emerged to improve the accuracy within reasonable execution times and the main goal of this paper also extends to the creation of a system that could be used in real-time. On using the model for the detection and recognition, the overall

execution time was found to be to an average of 1.5s. Hence, we can successfully deduce that the proposed system can be implemented in real-world scenarios.

References:

- W. Wang, J. Yang, M. Chen and P. Wang, "A Light CNN for End-to-End Car License Plates Detection and Recognition," in *IEEE Access*, vol. 7, 2019.
- [2] C. Henry, S. Y. Ahn and S. -W. Lee, "Multinational License Plate Recognition Using Generalized Character Sequence Detection," in *IEEE Access*, vol. 8, 2020.
- [3] L. Xie, T. Ahmad, L. Jin, Y. Liu and S. Zhang, "A New CNN-Based Method for Multi-Directional Car License Plate Detection," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, Feb. 2018.
- [4] Y. Zou et al., "A Robust License Plate Recognition Model Based on Bi-LSTM," in *IEEE Access*, vol. 8, 2020.
- [5] I. V. Pustokhina et al., "Automatic Vehicle License Plate Recognition Using Optimal K-Means with Convolutional Neural Network for Intelligent Transportation Systems," in *IEEE Access*, vol. 8, 2020.
- [6] A. Tourani, A. Shahbahrami, S. Soroori, S. Khazaei and C. Y. Suen, "A Robust Deep Learning Approach for Automatic Iranian Vehicle License Plate Detection and Recognition for Surveillance Systems," in *IEEE Access*, vol. 8, 2020.
- [7] J. Zhang, Y. Li, T. Li, L. Xun and C. Shan, "License Plate Localization in Unconstrained Scenes Using a Two-Stage CNN-RNN," in *IEEE Sensors Journal*, vol. 19, 2019.
- [8] C. Liu and F. Chang, "Hybrid Cascade Structure for License Plate Detection in Large Visual Surveillance Scenes," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, June 2019.
- [9] M. Darji, J. Dave, N. Asif, C. Godawat, V. Chudasama and K. Upla, "Licence Plate Identification and Recognition for Non-Helmeted Motorcyclists using Light-weight Convolution Neural Network," 2020 International Conference for Emerging Technology (INCET), Belgaum, India, 2020.
- [10] M. S. Al-Shemarry and Y. Li, "Developing Learning-Based Preprocessing Methods for Detecting Complicated Vehicle Licence Plates," in *IEEE Access*, vol. 8, 2020.
- [11] X. Yan, C. Wang, D. Hao and M. Chen, "License Plate Detection Using Bayesian Method Based on Edge Features," 2021 IEEE 5th International Conference on Cryptography, Security and Privacy (CSP), Zhuhai, China, 2021.
- [12] K. D. Rusakov, "Automatic Modular License Plate Recognition System Using Fast Convolutional Neural Networks," 2020 13th International Conference "Management of large-scale system development" (MLSD), Moscow, Russia, 2020.
- [13] W. Weihong and T. Jiaoyang, "Research on License Plate Recognition Algorithms Based on Deep Learning in Complex Environment," in *IEEE Access*, vol. 8, 2020.
- [14] Joshua, J. Hendryli and D. E. Herwindiati, "Automatic License Plate Recognition for Parking System using Convolutional Neural Networks," 2020 International Conference on Information Management and Technology (ICIMTech), 2020, pp. 71-74, doi: 10.1109/ICIMTech50083.2020.9211173.
- [15] C. Zhang, Q. Wang and X. Li, "IQ-STAN: Image Quality Guided Spatio-Temporal Attention Network for License Plate Recognition," ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020, pp. 2268-2272, doi: 10.1109/ICASSP40776.2020.9053966.
- [16] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection" in *Proc. IEEE Conf. Computer Visual. Pattern Recognit.*, Jun. 2015
- [17] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, Oct. 2016.
- [18] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Proc. IEEE 18th Int. Conf. Pattern Recognition.*, Aug. 2006.
- [19] Z. Xu, W. Yang, A. Meng, N. Lu, H. Huang, C. Ying, and L. Huang, "Towards end-to-end license plate detection and recognition: A large dataset and baseline," in *Proc. 15th Eur. Conf. Computer. Vis.*, 2018.