

Vehicle License Plate Recognition and Detection Using Neural Networks

Alan Tian¹, Arnav Dani², Lillian Yu¹

1. Lynbrook High School, 1280 Johnson Ave, San Jose, CA 95129, USA; alantian@ucsb.edu

2. The Harker School, 500 Saratoga Ave, San Jose, CA 95129 USA; arnavdani@ucsb.edu

ABSTRACT: Automatic license plate recognition (ALPR) has many practical applications in law enforcement, highway and traffic control such as in electronic toll collection systems, as well as in rapidly growing industries where self-driving technology is used. In this paper, the accuracy of current ALPR algorithms is evaluated. Existing models were used as well as datasets they were trained on, and other datasets to measure generalization errors. A unique two-stage ALPR algorithm was tested, First, to detect the license plate, a modified Multi-task Cascading Convolutional Network (MTCNN) and employed spatial transformation layers were used to fit tilted and rotated images to a bounding box, making it easier for the neural network to segment and classify. This was combined with a convolutional Neural Network (CNN)-based model which can accurately localize and efficiently assess and identify license plates in real-time. A bounding box is a rectangle that surrounds the license plate in the image. In order to train and test the model, a dataset containing over 300,000 images of license plates at various angles, lighting, and resolutions was used. It was found that while the model is accurate in good conditions, it starts to deteriorate as conditions become more extreme. By analyzing the usability of the network under different circumstances, the advantages and limitations of the technology becomes clear.

KEYWORDS: Algorithms; Machine Learning; license plate recognition; CNNs; ALPR; image detection.

■ Introduction



Figure 1: Examples of the model successfully detecting multiple license plates.

As countries' transportation infrastructure services have advanced in the recent decades, the demand for automatic license plate recognition (ALPR) for security and law enforcement, especially in urban areas, and highway surveillance has also grown. In addition to performing surveillance, license plates are useful as identification for vehicle access in gated communities, private property, and electronic toll systems. Thus, ALPR will reduce the need for workers and will also be much more efficient for traffic intake. Installing ALPR into road surveillance systems, especially in road junctions and traffic zones, will decrease the need for physical surveillance by police for unauthorized or stolen vehicles and assist in crime prevention as well. The movement of vehicles can be more easily tracked by an ALPR system. ALPR can also be used for documenting any traffic accidents such as a hit-and-run.

License plate recognition has many difficulties and obstacles in practical situations and environments. A vehicle may be pho-

tographed or video-taped while experiencing conditions such as glare, low resolution, low lighting, blur, having low contrast to surroundings, or being tilted. In addition, weather interference may be present such as rain, snow, fog or hail, which increases the difficulty of the recognition process. The variety of license plate shapes, characters, and sizes further complicates successful recognition. Lastly, the different backgrounds of the images or videos may lead the detection system to misinterpret where the license plate is actually located.

Previously, many attempts to detect license plates were based on certain hand-picked features, such as the color of the plates.^{1,2} However, these models often perform significantly worse under low visibility conditions. The downscaling technique through a line density filter (LDF)² was proposed, followed by a cascaded license plate classifier (CLPC), where the localization of the license plate was based on color saliency. These methods are useful only when there are dominant colors, and images processed through a grayscale filter had the best performance.² Such models are limited by their inability to detect variations in license plates, as colors on the license plate must match the color the model was trained with. Many of these networks take over a second to process images, hampering the real-time applications of the software. It was then decided to analyze the performance of a pre-existing high-performance lightweight model.

The two main processes involved in automatic license plate recognition include license plate detection (LPD), where the localization of the license plates occurs, and license plate recognition (LPR), where the license plate's characters are inter

interpreted and read. To avoid needing to use high-resolution images, many of these models³⁻⁵ rely on convolutional neural networks (CNNs), networks that excel in image processing and feature extraction. These CNN-based models additionally use sliding windows, which result in over-processing of the data and slow efficiency. While using deep CNNs⁶ can ensure high accuracy, it involves the classification of too many unnecessary sliding windows and is not practical due to high computational cost. To combat this problem, a shallow CNN was combined with the deep CNN to remove non-license plate regions,⁷ creating a hybrid CNN model. Despite reducing the number of candidate license plate regions and using the non-maximum suppression (NMS) technique to remove highly overlapped sliding windows, the resulting regions that are processed by the deep CNN can still be reduced further as the reported speed was only 6 fps(frames per second).⁷ Many similar hybrid CNN models have recently been created,^{7,8} but inefficient and fail to recognize plates under more realistic conditions.

Additionally, some unorthodox methods have been tried in the past, such as using support vector machines (SVMs),^{2,9} but the consensus is that the exclusive use of a CNN is necessary for high-quality results. Car segmentation is used to divide images further, where each part is processed through a CNN-based feature extraction model and the SVM classifier,⁹ resulting in the classification of unrelated regions.

In order for the network to detect the license plate, the model has a two-stage process.¹⁰ The first stage, detection, utilizes a multi-task cascaded convolutional neural network (MTCNN) in order to efficiently generate a bounding box around license plates. In Figure 1, it can be seen that the network is capable of detecting multiple license plates at a time. The network consists of 2 sub-networks, Pnet and Onet, consisting of 3 and 4 convolutions, respectively, and employs the use of a parametric RELU activation function. The second stage is the reading of the license plates, which consists of 2D convolutional layers, 2D max-pooling layers, etc.

■ Methods

In order for the model to detect and identify various license plates, the researchers had trained and built a model with multiple fully connected convolutional layers.¹⁰ Although there are many well-established models of image processing to choose from, such as visual geometry group (VGG), the researchers decided to implement an algorithm called LPRNet,¹¹ as it had the benefit of being able to process data in real-time, as well as driving down the computational demands compared to well-established models.^{3,7} As the quality of character segmentation is dependent on the resolution of the images, it was found that it was important to test the network on a database that contained both high- and low-resolution images to compare their performances and potential improvements. Furthermore, the researchers were able to encompass a wide variety of images using the CCPD dataset.¹²

The Network:

The model that was used had two stages. The first stage was the detection stage. The model used a MTCNN¹³ to

detect the location of the license plate using bounding box regression. MTCNN originally consisted of 3 sub-networks: proposal network (Pnet), refine network (Rnet), and output network (Onet), where the outputs of Pnet and Rnet are used as proposals for Onet. In this specific implementation, however, the researchers discarded Rnet to minimize the processing time, as it did not contribute significantly to the accuracy.¹⁰ After the data was split into train, validate and test sets, it was passed into the Pnet where images were scaled and candidate bounding boxes with the dimensions of 12x47 were made and was processed through 4 convolutional layers as seen in Figure 2. The candidate bounding box was then searched if it contained a part of the license plate and each pixel was classified as containing a part of a license plate or not. Each layer has a parametric RELU activation function and Cross Entropy loss function, and the final output is a bounding box around the plate as seen in Figure 3.

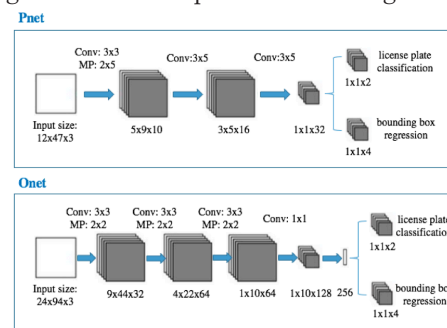


Figure 2: Model of Modified MTCNN Network.¹⁰



Figure 3: The final output, a bounding box around the plate.

Realistically, cars will not be in the direct view of a surveillance camera and many plates seen in images or videos are only visible at various horizontal angles, heights, and vertical degrees of rotation. This poses a challenge for some license plates to fit even the best candidate bounding boxes. To address this problem, a spatial transformation layer¹⁴ is implemented to resize and align the edges of the license plate with the bounding box as seen in Figure 4. This stage consists of two convolutional layers, each with RELU activation functions. The result of the process is a straighter image which is easier for the final stage, identification, to recognize the numbers and characters on the license plate.

The second stage¹¹ consists of a multi-layered process which includes another spatial transformer along with the convolution, normalization, RELU activation function, and max-pooling layers. The input of LPRNet is 24x94x3, equivalent to the output of the Onet and the spatial transformer

model. The first step performs a convolution on the input with 64 output channels and a stride of one. After normalizing the data and passing it through a RELU function, the data passes through four convolutions with a total input of the 64 channels and a final output of 128 channels. In the second iteration, the network is normalized with 128 features. After another RELU activation, the data is max-pooled to reduce the number of features to 64. Next, four convolutions are grouped with an initial input of 64 channels to an output of 256 channels. After being normalized, this process is then repeated with input and output of 256 channels. At this late stage in the model, dropout is introduced to increase the effectiveness of each neuron and to prevent overfitting to a specific set of training plates. Finally, a convolution layer takes in the 256 channels and has the same number of outputs as the number of characters possible, in China being 68 possible characters. This prediction is then evaluated against the ground truth to determine the accuracy of the prediction. The network only classifies an identification as correct if all characters in the ground truth match with the prediction, as this is the same criteria that the original researchers used. In total, this LPRNet network has around 30 layers.

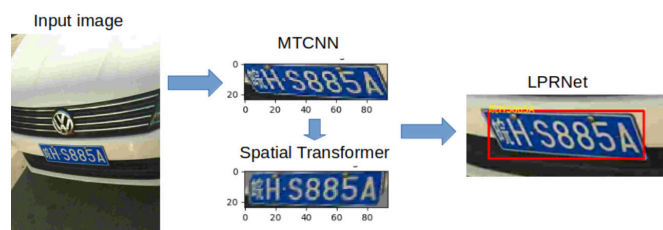


Figure 4: Process the neural network takes in order to determine the bounding box.

Dataset:

The researchers used a subset of a Chinese license plate dataset called CCPD¹² which contained around 300,000 images of license plates. A subset of this dataset was used, containing 25,000 images to test the network. The dataset is split into categories defining various images of plates taken in different scenarios and environments. There was a total of 7 categories that was used, those being: 1. blurred images, 2. challenging images, which were examples especially difficult to recognize, 3. darker images which also generally had lower resolution, 4. rotated images, 5. images with tilted license plates, 6. images for plates taken in varying weather conditions such as rain, snow, etc., and 7. the base images that were taken in good conditions and high visibility. Having so many different images exposed the network to a multitude of scenarios that would exist in the real world, allowing them to become more robust. In addition, a dataset with empty plates, having no characters on them, was used to train the detection model to be able to look for the shape of the plate. The training data consisted of randomly selected images so that the network would not learn order and recognize only a certain image. Of a chosen 5000 images, 3750 were used to train the networks while 1250 were used to validate it, establishing a 3:1 ratio. The researchers heavily pre-processed the data before it was used to train the model,¹⁰ where a series

of steps were taken to prepare for Pnet and Onet processing. The candidate bounding boxes made by Onet were sized to satisfy the input dimensions of LPRNet, whereas Pnet's candidate bounding boxes were half the size both in length and width compared to Onet to ensure strict moderation of the image so the target license plate would not be missed. Along with splitting the dataset into training and validation sets, the ground truth provided from each image was obtained to use as a comparison with the predicted characters made by LPRNet. After MTCNN detection, the network crops the image to be left with only the bounding box of the license plate, resulting in a 94x24 image in order to satisfy the input requirements for both the spatial transformation layer, as well as the final identification layer. In order to test the accuracy of the trained network, subsets of 1270 images from each dataset category were supplied into the main program and were able to calculate the final accuracy of the model using the ground truths. Through conducting such testing, the researchers were able to analyze the advantages and limitations in the current model, as well as common errors, mistakes and trends specific to each dataset category.

Results and Discussion

Although the model could not be trained, experiments were conducted by testing each category in the CCPD dataset with the pretrained network and thoroughly analyzed their accuracy and shortcomings, as well as stating some techniques that would improve their performance and efficiency. The goal was to find areas for improvement in LPR technology and pinpoint the direction the software should advance in the future.

Testing:

In order to have a better grasp of LPR technology's strengths and limitations: the advertised pre-trained models needed to be thoroughly tested. This was accomplished by modifying the main method to not only create an output of an image, but also to compare an image's output with the ground truth and calculate an overall accuracy. To test, a generator that would randomize the order of the images within the file was built to accurately test the capability of the networks and prevent any previous learning of image order to jeopardize the results. By comparing which images had a successful recognition to those that had failed, the true flaws and limitations of the technology became clear.

It was found that altering conditions for the worst case had a tremendous effect on the accuracy. In the base set, with simple, clear images, the network returned a 98.2 % accuracy. Although this set is strictly limited to images with good conditions, the high accuracy proves the network's potential to be used in the future. There was an impressive performance for images in the weather dataset, where cars were often covered in snow and rain, as the network returned an accuracy of 93.31 %. However, in every other category, the network failed to exceed 50 % accuracy, showing that many realistic scenarios are still unaccounted for and not trained enough for. The conditions that most heavily impacted performance was darkness and blur, where the overall accuracy for dark was only 10.6 % and blurred at 23.05 %. An example of an

incorrect prediction of a blurred license plate is shown in Figure 5, where the model identified more than one character as a Chinese character. With low contrast between the surroundings and the license plate, the localization of the license plate was difficult, and the recognition of the characters was even more so. After analyzing the results of the tilted and rotated dataset, it was clear that many candidate bounding boxes did not encompass the entire license plate, leading to incomplete predictions as seen in Figure 6. This is due to having the bounding boxes created not being parallel to the misaligned license plate, out of which many did not fit the bounding box correctly until the spatial transformer adjusted the angle and dimension of the license plate. Even then, the output of the spatial transformer did not always encompass the entire license plate. Many of the images were not only dark with little illumination, but also pixelated and blurry as seen in Figure 7, some to the extent that even one could not determine the license plate characters without using the ground truth. As a result, many outputs only accurately predicted the first half or last half of the license plate characters and failed to identify all the characters of the original license plate contained in the input image. A graph was used to visually interpret the data to see the stark differences shown in Figure 8.

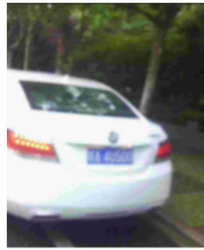


Figure 5: Blurred license plate predicted as 皖皖皖L皖 should have been: 皖A4U500.



Figure 6: Rotated license plate predicted as 闽AV2 should have been: 皖AV229P.



Figure 7: Dark license plate Predicted as: 皖AWX24Z Ground Truth: 皖AWX242.

Errors/Inconsistencies:

There were many instances where the model predicted the license plate characters incorrectly. The common trends as

to why errors were occurring were found in most of the poor performing datasets. First, characters that look alike such as “4” and “A” or “2” and “Z” were hard to differentiate in low light scenarios, blurry images, or tilted license plates where the letters would appear slightly narrower, and the lines could not be easily differentiated as shown in Figure 7. Next, the model guessed either too little or too many characters, just forgetting one or adding one with the rest of the characters being found correctly. Most commonly, the reason for this would be having the same character appear consecutively on the license plate and the network would not identify all of them, having narrow letters such as “1” or “T” being skipped. The model would also guess the same license plate sequence multiple times on different vehicles. This is most likely something the network learned while training and is an obvious indication of an overfit model. Many of the license plates that were hard to read were at extreme angles and in adverse conditions, some of which were not even identifiable by the human eye. On the other hand, there were cases when clear and obvious plates were misinterpreted, and the output was wildly different from the input. Inconsistencies can be overcome with more adjustment to the spatial transformation layer and training of the network. It was hypothesized that the results would be distinctly different from those reported because the dataset could be more exhaustive with more edge cases when it came to each category than the training dataset, but there would have to be more experimentation to isolate the cause.

OUR RESULTS VS REPORTED RESULT

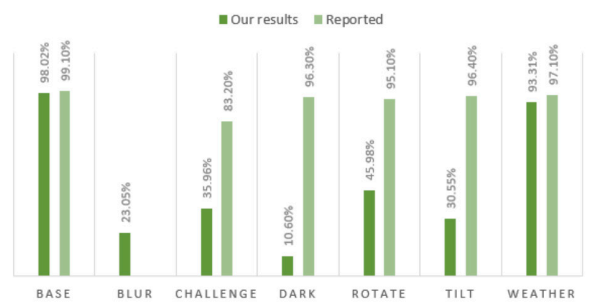


Figure 8: Comparison of the accuracies between reported results¹⁰ and the results: there were many discrepancies due to incorrect bounding boxes with tilted or rotated license plates as well as low contrast of license plate to surroundings for dark and blurred license plates.

The Future of License Plate Recognition:

An external method will probably be necessary since license plates are unique and the predictions must be perfectly accurate. A possible solution would be to substitute a temporary character for all the characters on the plate that the model is not over 80 % confident about. Then, the plate can be run against a government's national database of registered vehicles to find one that matches the known characters to identify the unknown characters. Another solution would be to spatially transform the image earlier, which might solve incomplete bounding box problems, increasing the accuracy

for the tilted and rotated images. To improve the contrast, especially when there is glare or poor lighting, one could increase the intensity of the image and saturate the blue and yellow colors of the RGB image,¹⁰ since they are the main colors found in Chinese license plates. With the implementation of color, brightness, and orientation feature maps as seen in Figure 9 in addition to increasing contrast, this will allow for more defined edges and lines, which results in easier localization of license plates as well as a way to differentiate characters that look similar. In addition, a possible implementation of a line density filter will ensure the candidate bounding box's accuracy.

As ALPR has many different applications in a vast range of fields, in the future, developing an accurate model that is able to perform reliably in all different types of conditions should be considered. Due to the fact that this particular model that was analyzed was trained using data from Chinese license plates, it would also be possible for the model to be trained on a dataset containing plates not limited to one particular country. This way, the technology could be used in a variety of different countries, which would increase the versatility of the network and become more applicable for more countries. Another limitation is storage, as in many situations, there is simply not enough storage to keep hours upon hours of high-resolution video. As a result, many surveillance cameras have low-quality videos, and if any frame of those videos containing a vehicle were to be put into an ALPR model, the license plate would most likely not be detected. This is due to the low performance for the blurred images category. Due to this limitation, there is hope that it would be possible to train the network to be able to localize license plates in low-resolution images in the future. More extensive tests and comparisons should be performed on larger sets of data in the future for better accuracy and better overall results. There is a lot of potential for this network and with the improvement of these areas, it can be used in the future for a wide range of applications

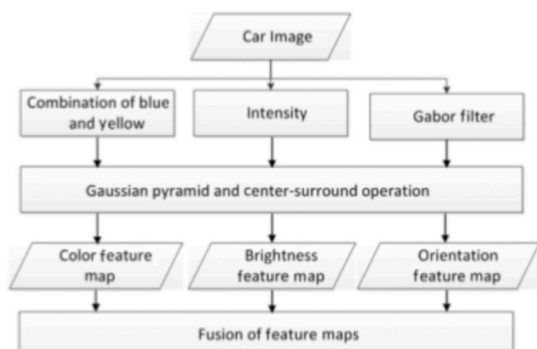


Figure 9: Amplify color intensity and certain layers to improve contrast of license plate compared to surroundings. SPIE. digital library May 4, 2015.⁹

Conclusion

It was concluded that the two-stage MTCNN and LPR-Net model that was studied was efficient, however, both the MTCNN and LPRNet were susceptible to many different inaccuracies and errors when non-optimal scenarios were presented. There were many limiting factors and drawbacks

that the networks had, such as the inability to locate license plates that are in a low contrast environment, tilted images, and rotated images, as well as identify all characters on the license plate given correctly. Even with the help of the spatial transformer layer, the model that was analyzed still struggled to handle such images effectively. On the other hand, images captured under bad weather conditions performed well, and such performance shows the potential that the network may become worthy of use in the real world in the future.

Despite the limitations of the current technology, ALPR is an incredibly useful tool that has a vast range of applications. In the future, there is hope to develop a model capable of overcoming such challenges through better training techniques, more training data, implement new features such as a color feature map to increase the contrast between the license plate and the surrounding vehicle, and exposing the network to various countries' license plates to increase the versatility of this technology. In addition, human judgement can be added in the loop to improve the accuracy of the model. Hyperspecific networks, targeted to countries, character sets, or conditions, can also be developed on top of the base network. With these modifications, the model will become more robust and can ensure better results in the future for many applications.

Acknowledgements

It is worth mentioning our mentor Jędrzej (Jacob) Kozerawski and teaching assistant Aiwen Xu for helping us so much throughout the whole research process and revising our research paper, as well as the Director of Pre-College Programs at UCSB, Dr. Lina Kim for organizing the amazing SRA program and giving us this amazing learning and research opportunity. In addition, we would like to thank our Track 6 peers who have been very supportive and aided our learning experience throughout the program's duration.

References

1. A. H. Ashtari, M. J. Nordin, and M. Fathy, "An Iranian license plate recognition system based on color features," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 4, pp. 1690–1705, Aug. 2014.
2. Y. Yuan, W. Zou, Y. Zhao, X. Wang, X. Hu and N. Komodakis, "A Robust and Efficient Approach to License Plate Detection," in *IEEE Transactions on Image Processing*, vol. 26, no. 3, pp. 1102–1114, March 2017, doi: 10.1109/TIP.2016.2631901.
3. L. Zhang, P. Wang, H. Li, Z. Li, C. Shen, and Y. Zhang, "A Robust Attentional Framework for License Plate Recognition in the Wild," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, Jun. 2020.
4. 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, pp. 507–517, Feb. 2018, doi: 10.1109/TITS.2017.2784093.
5. P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *Proceedings of the International Conference on Document Analysis and Recognition, 2003 Proceedings (ICDAR)*, vol. 1. IEEE Comput. Soc, 2003, pp. 958–963.
6. 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, pp. 507–517.

7. L. Zou, M. Zhao, Z. Gao, M. Cao, H. Jia, and M. Pei, "License Plate Detection with Shallow and Deep CNNs in Complex Environments.," *Complexity*, 2018.
8. X. Yang, X. Wang, "Recognizing License Plates in Real-Time," Apr. 2020.
9. D. Zang, Z. Chai, J. Zhang, D. Zhang, and J. Cheng, "Vehicle license plate recognition using visual attention model and deep learning." *Journal of Electronic Imaging*, 24(3), 033001.
10. Adapted model from https://github.com/xuexingyu24/License_Plate_Detection_Pytorch to automatically recognize license plates.
11. S. Zherzdev, A. Gruzdev, "LPRNet: License Plate Recognition via Deep Neural Networks," Jun. 2018.
12. 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," *Computer Vision – ECCV 2018 Lecture Notes in Computer Science*, pp. 261–277, Apr. 2018.
13. K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks," *IEEE Signal Processing Letters (SPL)*, vol. 23, no. 10, pp. 1499–1503, 2016.
14. . M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial Transformer Networks," Feb 2016.

■ Author

Arnav Dani is a junior at The Harker School. He is interested in applications of computer science and he likes composing music. He would like to pursue computer science as a major in college.

Alan Tian is currently a sophomore at Lynbrook High School. In his free time, he enjoys writing computer programs and solving math problems. In the future, he would like to pursue a career in finance.

Currently a sophomore at Lynbrook High School, Lillian Yu has been organizing and hosting online coding classes and reading clubs for middle school students. She enjoys volunteering in her community in her free time.

Author Contribution Statement:

A.T. conceived the experiment(s), L.Y. conducted the experiment(s), A.D. collected the dataset(s), A.D. and L.Y. analyzed the results. All authors reviewed the manuscript.