Vehicle License Plate Recognition

Damian Filo
Computer Science and Technology
Xi'an Jiaotong University
Xi'an, China
splithor1zon@protonmail.ch

***Abstract*—License plate recognition systems have been an interest in recent years for their role in intelligent traffic management. This paper illustrates a design and development of an efficient Automatic number plate recognition system (ANPR), using image processing and deep learning techniques. The paper focuses on the deep learning SSD MobileNet model, which will be used to detect the number plate of the vehicle. We utilized two different sequential convolutional neural networks that are smart enough to recognize plates characters. Moreover, it is shown that with variations of pretrained architectures, one can obtain the desired trade-off between complexity and accuracy, and SSD MobileNet v2 and a simple CNN model can obtain the desired results with satisfying cost and accuracy.**

Keywords—Object detection, Object recognition, License plate, Convolutional neural network

I. Introduction

The license plate numbers provide access to valuable information, including the vehicle's ownership, condition, and record. The ability to detect license plates is an important aspect of developing intelligent transportation. Providing critical services including parking management, traffic monitoring, and policing [1]. Increasing the efficiency of law enforcement and public safety. There are several sub-tasks in Number Plate Recognition. Starting by gathering input images from different scenes. This can be in the form of static images or video. The region of interest is then extracted through a method known as Number Plate Extraction, in which input images are processed using the proposed detection algorithm. Finally, characters on the plate are segmented to be later converted into text using Optical Character Recognition (OCR) techniques. Segmentation and traditional machine learning techniques are the classic approaches to license plate recognition. The advantage of such systems is the fast run time. However, because of the wide variety of license plate styles, colors, fonts, and sizes, they fail as conditions get more complex. With the popularity of deep learning, new technologies replaced the traditional methodology. The rising techniques appeared to be fast and highly accurate for both detection and recognition tasks. This work presents a fast approach for license plate detection using the pre-trained model SSD MobileNet v2. The detected plate region is used to produce the plate number by simply utilizing two simple CNNs for recognition for plate recognition. Therefore, this work tackles the task of license plate recognition. Comparing the performance of two models character recognition of license to produce the final results.

II. Related Work

The process of developing a system that detects and recognizes license plates automatically involves two key phases. Character recognition and license plate extraction. In some cases, the input data may necessitate further image preprocessing techniques. Enhancing images with varying illumination or weather conditions.

A. Traditional Algorithms

Traditional and deep learning algorithms are the two main approaches to license plate detection and recognition systems. The traditional approach takes advantage of features such as edges. Searching for all rectangles in the image that make up the car's body, the license plate, or regions with distinct color transitions [2]. in the work of [3], the vertical and horizontal histogram information have been used for plate extraction, where the accuracy of extraction was tested on 50 input images ranging in font and luminance to achieve an accuracy rate of 90%. Instead of utilizing grayscale, another method groups the pixels of the image using the (HLS) hue, lightness, and saturation color space. The accuracy of [4] study succeeds at the recognition of plates with the prior technique achieving the accuracy of 90% under varying illuminations. Contour detection, on the other hand, uses a binary picture to locate a connected object [5]. Geometrically similar features to license plates are extracted to be processed. Such a method necessitates the use of high-quality images. For the task of recognition. The plate's bounding box must be defined before it can be fed through an (OCR) system or a convolutional neural network (CNN). The Template Matching technique is one of the most basic methods for character recognition. the similarities between template characters and recognized ones are measured. The predicted character is determined by the highest match. This method has a success rate of up to 98.1 percent [6].

B. Deep Learning

Deep learning approaches are popular for many intelligent takes since they do not require manual feature extractions techniques like traditional approaches. Faster-RCNN is one of the most common Convolutional Neural Network object detection models. Which can detect objects fast and accurately. The Faster R-CNN with InceptionV2 feature extractor was utilized by the authors in [7]. Testing with several types of license plates on traffic images. obtaining a 93.21 percent recognition accuracy. Single Shot Detection (SSD) is another architecture that may encapsulate computation in a single network. End-to-end detection accuracy for an SSD architecture using the MobileNet features extractor is 79.86 percent [8]. Another Convolutional Neural Network (CNN) for object detection and recognition is You Only Look Once (YOLO).

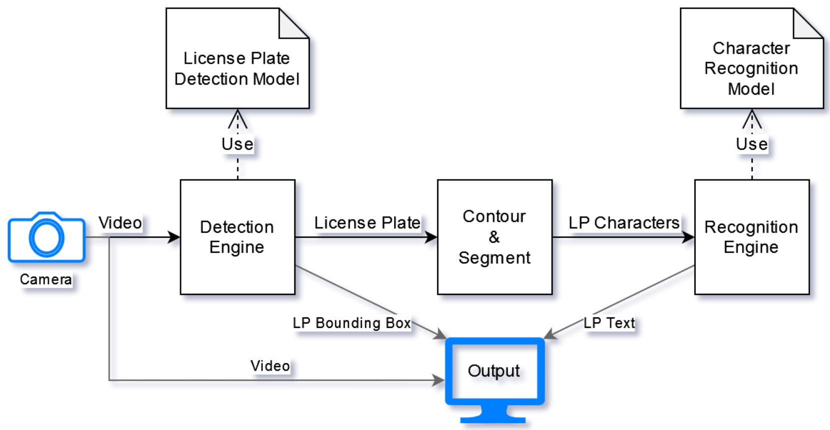

Figure 1. Proposed Methodology

The SSIG-SegPlate Database has yielded excellent results. With a recognition accuracy of 93.53 percent. For real-time Brazilian license plate detection and recognition, reference [9] employs the YOLO [10] network, whereas [11] uses a connected component-based approach to partition the characters before applying a deep CNN for character recognition. Deep CNN is also used in the commercial system "sighthound" in [12], which surpasses competitor approaches on various benchmarks. With the increasing usage of RNNs, approaches that do not need character segmentation have been developed. Reference [13] employs a VGG-Net with a 36-unit RNN for detection and identification, while [14] uses the same CNN with LSTM for sequence extraction and then CTC for final recognition. Meanwhile, for segmentation-free character recognition, [15] employs YOLO for plate localization and RDNet a hybrid of Dense-Net and Res-Net.

III. Proposed Method

Recent research has used Deep learning models in plate localization and OCR, thanks to developments in the field. CNN has recently gained popularity in detection and recognition tasks as a result of its recent breakthroughs. In addition, as compared to traditional models, LPR Systems that employ CNNs are more durable and accurate. In this section, we demonstrate the proposed method details, as well as architecture graphs, before discussing the findings and insights gained from the experiments. Refer to Fig 1 to see the proposed methodology.

A. *License Plate Detection*

For identifying the license plate in the input frame, we employ a fully convolutional network using MobileNet as the basic network for feature extraction and the SSD [16] as the detection framework, demonstrating the strength and potential of convolutional neural networks. Although deeper convolutional networks such as AlexNet obtain high accuracies on datasets like as ImageNet, this does not always imply that they are more efficient. In our situation, we place a premium on the system being light enough to run on embedded devices. The MobileNet model makes use of depth-wise separable convolutions, which convert a regular convolution into a depth-wise convolution and a 1 1 convolution, substantially lowering both computing cost and parameter count. MobileNet delivers great accuracies with a smaller number of parameters than bulkier networks, resulting in a cheaper computational cost.

As a result, for our work, the reasonable decision is to trade a slight loss of accuracy for a substantial difference in model parameters. The SSD network, which is utilized after the MobileNet for identifying the plate bounding boxes, provides accuracy comparable to models like Faster R-CNN and YOLO-V1, according to [16]. It is quicker than the other networks, and it is also less computationally expensive because it does not need pixel or feature resampling for the bounding box hypothesis.

B. *Pre-processing*

1) *Cropping*

Character width and the contrast between an image character area and its backdrop were used to identify the character region. The license plate was then retrieved by calculating the inter-character distance in the plate region.

Other ways, known as hybrid approaches, have been proposed, which reflect a mix of two or more methodologies. In [17], the authors merged color and texture characteristics by using fuzzy methods to extract certain color and texture features. [18] Have proposed combining the MSER approach with a sift-based unigram classifier.

The edge-based technique and the Back Propagation Neural Network were employed by [19]. The authors of [20] proposed an efficient and robust classifier for determining the precise location of the license plate in the picture. The authors of [21] proposed utilizing weak sparse network classifiers to locate the license plate area in the collected picture while extracting a list of candidate regions. They also used a CNN classifier to screen them.

Figure 2a. Plate cropping

2) *Contour*

The procedure of binding the number plate from the larger scene is known as contour/localization. The number plate's position is determined, and the result is a sub-picture that just contains the number plate. The dataset is divided into training and testing sections and contains 1000 photos of number plates. For training, 800 photos are acquired and labeled. The testing is completed after the training is completed. After then, the detecting process is completed. For Character Recognition, all recognized number plates are saved in a separate folder.

Figure 2b. Plate contour

3) Binary Conversion

The Binary Conversion/character segmentation stage methodologies and techniques have been used to identify the area character in a plate license picture as the third phase in license plate detection and recognition systems. The authors in [17] employed an image segmentation approach termed the sliding window to determine the candidate region of a license plate. For character segmentation, an approach based on linked components was used [22]. The projection application [23] was the most often used character segmentation algorithm.

Figure 2c. Plate segmentation

C. License Plate Recognition

A Convolutional Neural Network (CNN) is a Deep Learning Network that can take in an image to assign weights and biases to different objects in the image, distinguishing one from the other. When compared to other traditional methods, it requires less pre-processing. While features in traditional approaches are hand-engineered, CNN can learn these features through training [24].

We proposed two different CNN architectures for character recognition. The models consist of an input layer, an output layer, and different numbers of hidden layers. The hidden layer is a network of a stack convolutional and pooling layers, which are used to extract features for the image. this eventually stops at one flatten layer and one or more fully connected layers. That is responsible for the classification of the image. The dropout layer is a common regularization method for reducing overfitting. It ignores a random set of neurons. CNNs' detailed design is illustrated in Fig. 2 and 3.

IV. Experimental Setting

A. Network Training

The input image of detection has a size of 300x300 pixels. Adam optimizer was used to train SSD MobileNet for 10,000 training steps. The batch size is set to 4, while the learning rate is set to 0.039. For recognition, the input image is 28x28 pixels, the learning rate is set to 0.0003 and the optimizer chosen is Adam.

B. Dataset

In this work, we utilized two publicly available datasets from Kaggle. For detection, the dataset includes 433 images with bounding box annotations of the automobile license plates contained inside the image. The second dataset has binary images of characters used for plate segmented character recognition.

C. Evaluation Metric

We calculated the mean average precision for detection and accuracy recognition tasks to evaluate the effectiveness of our models. Use the following equations:

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \quad (1)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

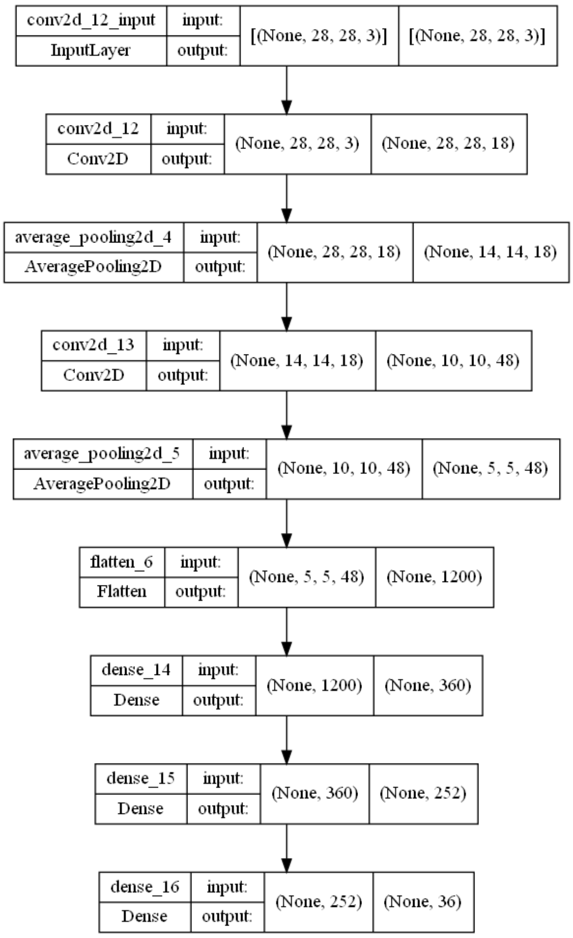

Figure 3. Architecture graph of CNN first Model

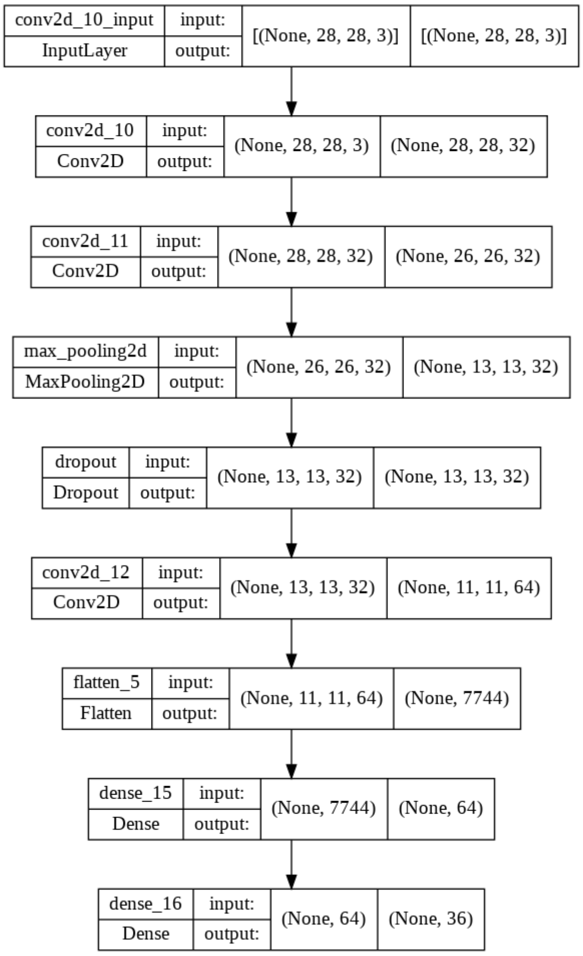

Figure 4. Architecture graph of CNN second Model

V. Results and Discussion

In this project, the process of License Plate Recognition was composed of a detection part done using SSD MobileNet v2, and the recognition part is done using two different CNN algorithms. The SSD MobileNet detection model detects the license plate with a certain probability as shown in Fig 5.

Figure 5. The output of the SSD MobileNet Detection model

Utilizing the output of the SSD model the license plate image is preprocessed i.e., cropping (already done), assigning the contours, and finally segmenting the individual digits or letters of the image, as shown in Fig 2.

This segmented output (Fig. 2c) is introduced to the CNN algorithm which predicts the license plate number utilizing one of the proposed models as selected by the user. The first CNN algorithm referred to as CNN first model in the project, was able to achieve a training accuracy of 94% and a validation accuracy of 94% as shown in figure Fig. 6. The second algorithm referred to as CNN second was able to achieve a training accuracy of 97.9% and a validation accuracy of 97.8% as shown in figure Fig. 7. The CNN second recognition algorithm was more accurate for recognizing the license plate number. Refer to Fig 9. The first model incorrectly predicted the “2” character as “z” as shown in Fig. 8. To see prediction results. Based on the accuracy metrics, the combination of the SSD MobileNet with the CNN second convolutional network is proposed as the final deliverables for this project. For detailed accuracy and loss metrics for both recognition algorithms refer to Table 1.

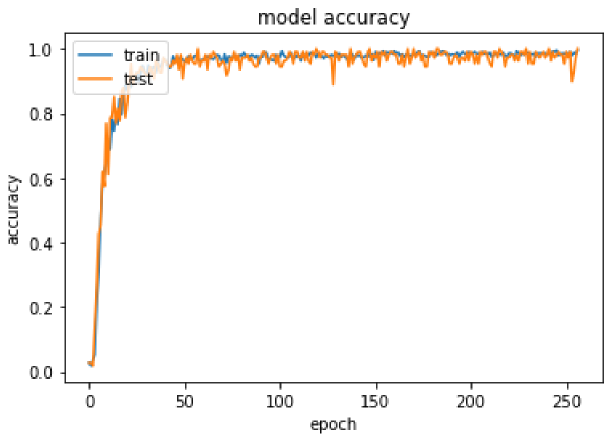

Figure 6. Accuracy graph for CNN first Model

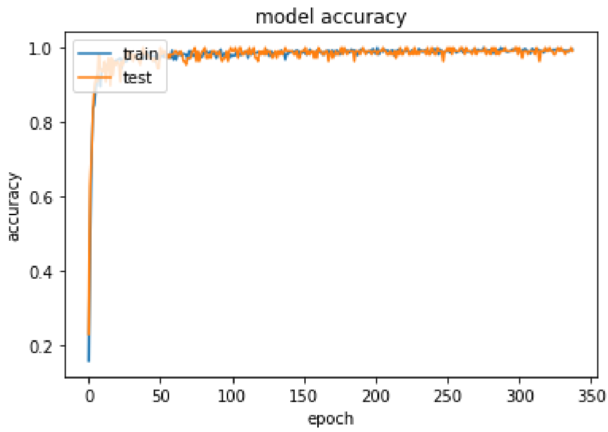

Figure 7. Accuracy graph for CNN second model

TABLE I. Training and Validation metrics of CNN1 and CNN2

Metrics	Models	
	CNN1	***CNN2***
Training Accuracy	0.941	0.979
Validation Accuracy	0.936	0.978
Training Loss	0.191	0.059
Validation Loss	0.208	0.060

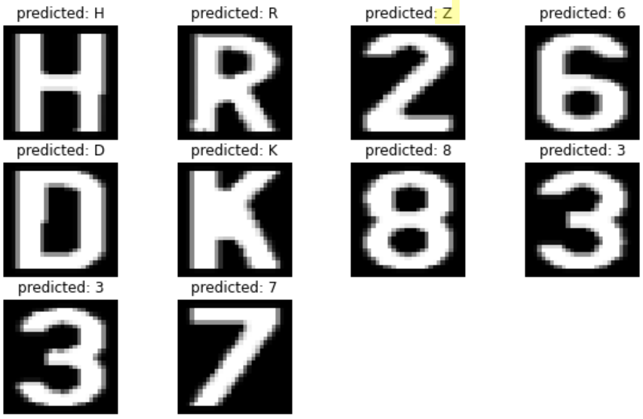

Figure 8. Prediction results of CNN first model

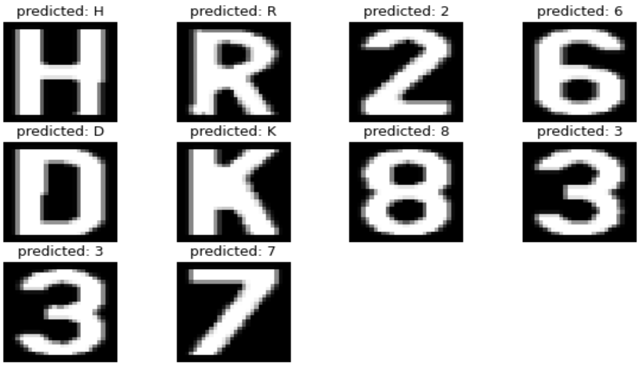

Figure 9. Prediction results of CNN second model

VI. Conclusion

This paper presented the car license plate recognition. It is trained two datasets to cover each subtask of the system. SSD MobileNet v2 is utilized for the detection phase, while for the recognition we experimented on two CNN models. The results of the tests are satisfactorily close. It is important to note that for different regions of the world a system must be provided. the developed system is not sufficient and needs to be designed according to the deploying region, keeping all the affecting factors into consideration. However, our system can be easily trained for different license plates based on a dataset of the characters used in the targeted license plates of the country. The recognition part of the system is trained and tested on the Indian license plate. It has an accuracy of 98%.

Future research in plate recognition systems still faces several challenges; For instance, there is a need to concentrate on more robust algorithms for non-standardized formats, irrespective of regions. Also, all proposed/designed algorithms need to be tested for real-time scenarios rather than pre-acquired images. For the class presentation, we presented a real-time demo using a webcam. However, high-resolution cameras need to be integrated, allowing robust algorithms to reduce processing times and increase recognition capabilities.

Acknowledgment

We would like to express our gratitude to Professor Na Lu, for exceptional guidance throughout this course.

References

1. Shi, X., W. Zhao, and Y. Shen, *Automatic license plate recognition system based on color image processing*, in *Proceedings of the 2005 international conference on Computational Science and Its Applications - Volume Part IV*. 2005, Springer-Verlag: Singapore. p. 1159–1168.

2. Alborzi, Y., et al. *Robust real time lightweight automatic license plate recognition system for Iranian license plates*. in *2019 7th International Conference on Robotics and Mechatronics (ICRoM)*. 2019. IEEE.

3. Farhat, A.A., et al. *OCR-based hardware implementation for qatari number plate on the Zynq SoC*. in *2017 9th IEEE-GCC Conference and Exhibition (GCCCE)*. 2017. IEEE.

4. Lin, H.-Y., et al., *A vision-based driver assistance system with forward collision and overtaking detection.* Sensors, 2020. **20**(18): p. 5139.

5. Slimani, I., et al. *Vehicle License Plate Localization and Recognition System for Intelligent Transportation Applications*. in *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*. 2019. IEEE.

6. Chacon, M.I. and A. Zimmerman. *License plate location based on a dynamic PCNN scheme*. in *Proc. Int. Joint Conf. Neural Netw*. 2003.

7. Haider, S.A. and K. Khurshid. *An implementable system for detection and recognition of license plates in Pakistan*. in *2017 International Conference on Innovations in Electrical Engineering and Computational Technologies (ICIEECT)*. 2017. IEEE.

8. Amon, M.C.E., et al. *Philippine License Plate Character Recognition using Faster R-CNN with InceptionV2*. in *2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*. 2019. IEEE.

9. Montazzolli, S. and C. Jung. *Real-time brazilian license plate detection and recognition using deep convolutional neural networks*. in *2017 30th SIBGRAPI conference on graphics, patterns and images (SIBGRAPI)*. 2017. IEEE.

10. Iandola, F.N., et al., *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size.* arXiv preprint arXiv:1602.07360, 2016.

11. Radzi, S.A. and M. Khalil-Hani. *Character recognition of license plate number using*

convolutional neural network. in *International Visual Informatics Conference*. 2011. Springer.

12. Masood, S.Z., et al., *License plate detection and recognition using deeply learned convolutional neural networks.* arXiv preprint arXiv:1703.07330, 2017.

13. Cheang, T.K., Y.S. Chong, and Y.H. Tay, *Segmentation-free vehicle license plate recognition using ConvNet-RNN.* arXiv preprint arXiv:1701.06439, 2017.

14. Shivakumara, P., et al., *CNN-RNN based method for license plate recognition.* CAAI Transactions on Intelligence Technology, 2018. **3**(3): p. 169-175.

15. Gao, P., Z. Zeng, and S. Sun. *Segmentation-free vehicle license plate recognition using CNN*. in *International Conference on Signal and Information Processing, Networking and Computers*. 2018. Springer.

16. Liu, W., et al. *Ssd: Single shot multibox detector*. in *European conference on computer vision*. 2016. Springer.

17. Nijhuis, J., et al. *Car license plate recognition with neural networks and fuzzy logic*. in *Proceedings of ICNN'95-International Conference on Neural Networks*. 1995. IEEE.

18. Lim, H.W. and Y.H. Tay. *Detection of license plate characters in natural scene with MSER and SIFT unigram classifier*. in *2010 IEEE Conference on Sustainable Utilization and Development in Engineering and Technology*. 2010. IEEE.

19. Wang, N., X. Zhu, and J. Zhang. *License plate segmentation and recognition of Chinese vehicle based on BPNN*. in *2016 12th International Conference on Computational Intelligence and Security (CIS)*. 2016. IEEE.

20. Yuan, Y., et al., *A robust and efficient approach to license plate detection.* IEEE Transactions on Image Processing, 2016. **26**(3): p. 1102-1114.

21. Bulan, O., et al., *Segmentation-and annotation-free license plate recognition with deep localization and failure identification.* IEEE Transactions on Intelligent Transportation Systems, 2017. **18**(9): p. 2351-2363.

22. Wang, S.-Z. and H.-J. Lee. *Detection and recognition of license plate characters with different appearances*. in *Proceedings of the 2003 IEEE International Conference on Intelligent Transportation Systems*. 2003. IEEE.

23. Duan, T.D., et al. *Building an automatic vehicle license plate recognition system*. in *Proc. Int. Conf. Comput. Sci. RIVF*. 2005.

24. Alzubaidi, L., et al., *Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions.* Journal of big Data, 2021. **8**(1): p. 1-74.