Research Article

An Intelligent License Plate Detection and Recognition Model Using Deep Neural Networks

J.Andrew Onesimu[1], Robin D.Sebastian[1], Yuichi Sei[2,3,*] and Lenny Christopher[4]

[1]Karunya Institute of Technology and Sciences, India
andrewj@karunya.edu; robin.d.sebastian@gmail.com
[2]The University of Electro-Communications, Japan
seiuny@uec.ac.jp
[3]JST, PRESTO, Japan
seiuny@uec.ac.jp
[4]Karunya Institute of Technology and Sciences, India
lennyc@karunya.edu.in
*Correspondence: seiuny@uec.ac.jp

Received: 3th May 2021; Accepted: 26th September 2021; Published: 1st October 2021

Abstract: One of the largest automotive sectors in the world is India. The number of vehicles traveling by road has increased in recent times. In malls or other crowded places, many vehicles enter and exit the parking area. Due to the increase in vehicles, it is difficult to manually note down the license plate number of all the vehicles passing in and out of the parking area. Hence, it is necessary to develop an Automatic License Plate Detection and Recognition (ALPDR) model that recognize the license plate number of vehicles automatically. To automate this process, we propose a three-step process that will detect the license plate, segment the characters and recognize the characters present in it. Detection is done by converting the input image to a bi-level image. Using region props the characters are segmented from the detected license plate. A two-layer CNN model is developed to recognize the segmented characters. The proposed model automatically updates the details of the car entering and exiting the parking area to the database. The proposed ALPDR model has been tested in several conditions such as blurred images, different distances from the cameras, day and night conditions on the stationary vehicles. Experimental result shows that the proposed system achieves 91.1%, 96.7%, and 98.8% accuracy on license plate detection, segmentation, and recognition respectively which is superior to state-of-the-art literature models.

Keywords: ***car license plate recognition; convolutional neural network; deep learning; deep neural networks; license plate detection***

1. Introduction

Over the past years, Automatic License Plate Detection and Recognition (ALPDR) [1] has played a critical role in the development of smart cities as a monitoring device for car tracking, traffic control, enforcement of strict traffic rules and policies [2]. ALPDR has been commonly used in traffic toll systems, smart parking systems, and many more across several countries. In the parking area, toll gates, or any other crowded places, several vehicles pass by and there is a need for automation where a machine would identify the License plate number of all the vehicles. Hence ALPDR is used in this case where it automates the whole process thus saving human power and this automation would have been impossible without the development of Deep Learning (DL). Recent advances in Deep Learning, particularly neural networks have contributed to the advancement over a range of Computer Vision [3], such as Object Detection [4], Optical Character Recognition (OCR) [5], and identification of objects have led to the development of ALPDR.

J.Andrew Onesimu, Robin D. Sebastian and Yuichi Sei and Lenny Christopher, "An Intelligent License Plate Detection and Recognition Model Using Deep Neural Networks", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, pp. 23-36, Vol. 5, No. 4, 1st October 2021, Published by International Association of Educators and Researchers (IAER), DOI: 10.33166/AETiC.2021.04.003, Available: http://aetic.theiaer.org/archive/v5/v5n4/p3.html.

Deep learning is a branch of machine learning where Artificial Neural Networks (ANN) are algorithms admired by the human brain that learns from large volumes of data [6]. Deep learning helps computers or machines to solve complicated problems even while using a dataset that is very diverse from the one by which the model was trained [7]. Deep Learning predicts and classifies the information by using filters which are passed as inputs through layers. It uses several layers to gradually remove higher-level features from the raw input. Computer Vision [8] is a field of Artificial intelligence that uses digital images in form of videos and images captured from cameras to correctly recognize objects. Computer vision concerns the automated retrieval, interpretation, and comprehension of useful knowledge from a single or series of images. It involves the creation of a theoretical or algorithmic basis for automatic visual comprehension. Computer vision concerns the philosophy behind artificial systems that can extract knowledge from images. Image data can take several types, such as video sequences, multidimensional data, viewing from multiple cameras, etc. Such deep learning models have been effectively used in the field of vehicle automation [9]–[14].

Recognizing a license plate involves detection of the license plate as the first step and then recognize the characters in the license plate [15]. There are several methods to recognize the license plate. Free software called Google Tesseract [16] is available online that can recognize the characters in the license plate. A two-step approach called adaptive recognition is used which finds templates in pixels, words, sentences, and letters. It uses one data stage for recognizing the characters and a second stage to fill out the letters by matching the missed word or sentence. Google Tesseract is pretrained and can be used directly with any model. The proposed approach uses a trained Convolutional Neural Network (CNN) [17], [18] to identify characters in different situations which will be explained at the later stage of this paper

ALPDR is an open problem due to its vast diversity of image acquisition status and number plate identity which differs from country to country. ALPDR has been implemented in various places such as smart parking systems [19], tolls, and many more. Still, it faces problems with blurred images, images captured in the night, low-quality images, angle in which the images are captured. Many methods have been proposed, in some cases, three steps as detection, segmentation, and recognition are used. Detection of the license plate is one of the major tasks. Lin *et al.* in [20] have used You Only Look Once (YOLO) and Support Vector Machine (SVM) together to detect the license plate which provides higher accuracy. In some cases, only YOLO has been used for detection. The proposed method is based on edge detection where the RGB image is converted to a binary scale to check the edges of the license plate, once the edges are detected we make a bounding box over it. Then segmentation of all the characters is done using region props. Once done the characters boxed are recognized using trained Convolution Neural Network (CNN). All the processes are explained in the later stage of this paper.

The remainder of this paper is as follows. Section 2 presents the literature survey of various license plate recognition systems in the past. Section 3 explains the methodology of the proposed system and model architecture. Section 4 highlights the experiments and results performed to evaluate the model generalization in various cases. Section 5 presents the conclusion and future scope of the proposed ALPDR model.

2. Related Works

The section discusses the state-of-the-art literature in the field of license plate detection, recognition and segmentation. This section is divided into two parts they are license plate detection, and license plate recognition & segmentation.

2.1. License Plate Detection

Character recognition, segmentation, image processing, license plate location, and correction are needed for a license plate recognition system. By using the adaptive output algorithm image processing of the license plate is done. To detect the location of the license plate Lin *et al.* in [21] used a method called a Canny edge operator. To avoid the tilting problem in the license plate perspective transformation algorithm is used. For segmenting good image characters vertical projection method is used and for matching license plates with given characters in the library, the Huasdroff distance algorithm is used by the authors. Number plate detection in vehicles is important in case of any emergency where the police couldn’t reach the vehicle or if any vehicle has been stolen. Hsieh *et al.* in [22] proposed a Real-Time Vehicle Number Plate Recognition System (RVNPR). This system works in real-time where the characters are extracted from the number plate. High definition cameras are used to take pictures of the moving vehicle and then the RVNPR software is used to extract the information from the image. YOLO (You Look Only Once), Segmentation, and last character Recognition are different types of algorithms used to recognize the license plate in a vehicle. Using YOLO the license plate is detected from a vehicle and recognized. The proposed system is tested and proved to be highly accurate in detecting and recognizing the license plate. Many applications are available in License Plate detection and recognition work on a single image that can identify a single license plate in an image. Lin *et al.* in [23] proposed a method that can detect multiple license plates from a single image. They have proposed a two-stage method based on the deep learning algorithms that can detect all the licenses available in a single image and recognizes the character using character recognition in Convolutional Neural Networks. After evaluating the proposed method's results show that the method achieves 98.23% of license plate detection and 97.38% of character recognition rate. Color based license plate detection is another way to detect the license plate as it is different from the car body color. In [24] Zhang *et al.* proposed a Sobel-Color Algorithm combined with MSER Algorithm that can locate the color features and Sobel edge in an image. Support Vector Machine (SVM) is used to detect the location of the license plate with much higher accuracy. To segment the characters available in a license plate, the license plate segmentation algorithm is designed. After segmentation of the license plate, the characters need to be identified for that purpose leNet-5 depth model is used to improve the accuracy of character recognition. To avoid the problem of multilingual license plate (LP) detection problem, Kessentini *et al.* [25] used YOLOv2 for detecting license plates. The second method is used to capture license plate recognition with cropped images. The two datasets used in the system are actual road surveillance and parking access control environments. To minimize the cost and time of processing, a new semi-automatic annotation method for LP images with labelled bounding box components is proposed. When contrasted with other approaches using the public AOLP dataset, the proposed approach outperformed. It gave a 97.67% LP recognition rate in the GAP-LP dataset, and 91.46% in the Radar dataset. For further work, the correlation of data stored on homeland and police stations requires correlation and also enhances the identification of vehicle identity. There are many systems of surveillance available all over the world.

2.2. License Plate Recognition & Segmentation

The number of vehicle growth in India is more. Due to an increase in the number of automobiles Automatic Number Plate Recognition (ANPR) is needed. ANPR systems are used in tracing stolen vehicles, tracking vehicles, and traffic examining, whereas implementing that system is a challenging task. Automatic License Plate Detection and Recognition (ALPR) is the most efficient detection and recognition mechanism for intelligent transport systems [9]. Singh *et al.* in [26] developed a deep neural network model for license plate detection and LSTM based model is used for license plate character recognition. The drawback of this method is it requires high-resolution cameras to capture high-quality images. Omar *et al.* [27] used a Mask-CNN framework license plate recognition. In the analysis, four databases are used where the images are taken into account with low image quality, blurred image, complex environment, and orientation. The accuracy rates reached on the AOLP and Caltech datasets are 99.3% and 98.9%, respectively. The technique's weakness is that it failed to detect LP and even the framework speed is not impressive. Hendry *et al.* [28] used a deep learning framework called You Only Look Once (YOLO) - darknet. A YOLO seven convolutional layer sliding window method is used to detect a single class. A sliding window detects the license plate, and a YOLO framework is used to detect each window. The number plate characterization can vary from country to country, as well as in orientation sizes and different languages. Laroca *et al.* [29] presented an automatic licenses plate recognition system based on the YOLO-CNN model. The proposed system is layout independent and improves the recognition accuracy through post-processing. The system provides a balanced trade-off between speed and accuracy of license plate recognition. The methods proposed by Selmi *et al.* [30] include identification, segmentation, and recognition using two CNN models. The first CNN is used for classification between plates and non-plates. The second CNN is used for digit segmentation. The images are also taken as poor, good, bright with different viewpoints. Preprocessing is used for character recognition. Several works have been proposed on Automatic License Plate Recognition but the proposed model is limited to their country and does not perform well in other countries. To overcome this problem Henry *et al.* in [31] proposed a method that consists of three main approaches are License Plate (LP) detection, Unified Character Recognition, and Multinational LP layout detection. For character recognition, YOLOv3 is used that consists of the Spatial Pyramid Pooling (SPP) block. A layer detection algorithm is proposed that can extract the correct sequence of License Plate. The license plate from South Korea, Taiwan, the United States of America, Greece, and Croatia was used to validate the proposed model. The seven numeric plate characters are identified and recognized in this system from the available Brazilian License Plate dataset. Silva *et al.* in [32] proposed two YOLO-based neural networks, such as (FV / LPD-NET) and (LPS / CR-NET) respectively, which are developed to detect vehicle frontal view and license plate and also to recognize characters. When test 63.18% were correctly detected. It gave a precision of 97.39% for detecting at least five characters. By this form, 93% and 99% of characters are correctly identified and segmented. Car manufacturer/model identification can be performed in future works in the ALPR pipeline. Raju *et al.* [33] have proposed a method of developing a handheld device that can scan the vehicle's number and inform whether the vehicle belongs to the organization or not by checking in the database. To identify the license plate number Optical Character Recognition (OCR) is implemented in Raspberry pi. To test the handheld device approximately 70 vehicles have been tested including two-wheelers and four-wheelers. The proposed model gave a promising result when the quality of the input image was high and was not much accurate in the case of low-resolution images.

Image processing plays a significant role in numerous real-time applications. In these approaches, the still camera senses images of the number plate. A self-organized CNN function is used in [34] to identify the status of vehicles from number plate pictures. It is achieved with a low level of preparation and 90% accuracy. The actual obstacle that lies here is the real-time implementation of CNN. Dong *et al.* [35] proposed a fast area CNN method called RCNN. RCNN is used for regressing corner images of the detected plate. Networks and shared-weight recognizers are in the identification stage of parallel spatial transformation. The recognizer eliminates 57.5% of the errors and works well. Rectification is done for seeing distinct viewpoints and STN is done for character segmentation. This approach outperformed other approaches. For high-security surveillance purposes, a system called Vehicle Number Plate Recognition System (VPRS) was tested in tollgates by Rajathilagam *et al.* [36]. For detecting and recognizing vehicle number plate details, a neural network-based genetic algorithm is used in this approach. The optical characters of vehicles are identified through this algorithm.

Table 1 compares the state-of-the-art literature in the field of license plate detection and recognition. The table compares various techniques such as SVM, CNN, YOLO, and so on. It is perceived that the state-of-the-art literature used different combination models for license plate detection and recognition. It is also noticed that few approaches have used image segmentation techniques to extract the license plate characters. Hence, in our approach, we used a three-step process to detect, segment, and recognize the characters.

Table 1. State-of-the-art Techniques for License Plate detection and Recognition

Reference	Model to detect License Plate	Technique to Segment the Image	Model for Character Recognition
[20]	Support Vector Machine (SVM)	Horizontal Projection	LPRCNN model
[21]	Texture Feature-Based Algorithm	Template Matching Method	Hausdorff Distance algorithm
[37]	YOLO (You Only Look Once)	-	Real-time vehicle number plate recognition (RVNR) system
[27]	CNN object detection	Deep Segmentation network	CNN
[23]	Support Vector Machine (SVM)	-	Object Detection Architecture YOLO v2
[24]	Support Vector Machine (SVM)	License plate character segmentation algorithm	LeNet-5 network
[30]	CNN	Image Edge Detection	CNN model with four convolutional layers
[31]	YOLOv3-SPP	-	Multinational LP layout detection
[25]	YOLOv2	-	Convolutional Neural Networks (CNN)
[32]	FV/LPD-NET	YOLO	LPS/CR-NET
[26]	Faster RCNN	Tesseract	LSTM Neural Network
[38]	Sobel Edge Detector	Morphological operation	The normalized cross-correlation method
[39]	Template Matching	-	Template matching and bounding box methods

3. Proposed System and Architecture

The proposed model is divided into three stages they are License plate detection, Plate Segmentation, and plate recognition. The flow of the three stages is shown in Fig 1.

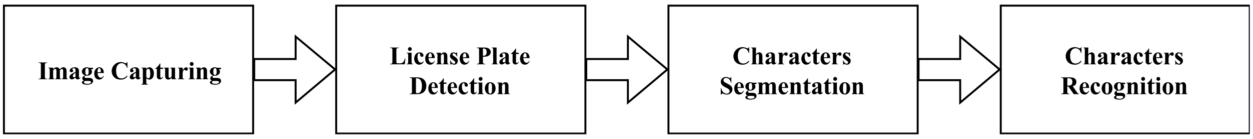


Figure 1. Flow Diagram of License Plate Recognition

The image or video is captured and passed to the detection stage to identify the license plate. Then the license plate is passed through the segmentation process where all the characters are segmented. The segmented images are stored in a list to be passed to the CNN model in the recognition stage to identify the letters. The architecture of the proposed ALPDR model is shown in Fig 2. The architecture consists of a camera to capture the image or video of the vehicle to be given as the input to the model. At first, we have the license plate detection module. It identifies the license plate from the input image by comparing it with the bounding box dimensions. Then the detected license plate is given as the input to the segmentation module which segments the characters from the license plate and saves the order of the characters. Finally, the CNN module is used for character recognition that recognizes the characters from the license plate and gives the output. The detailed operation of the proposed ALPDR model is presented as algorithm 1.

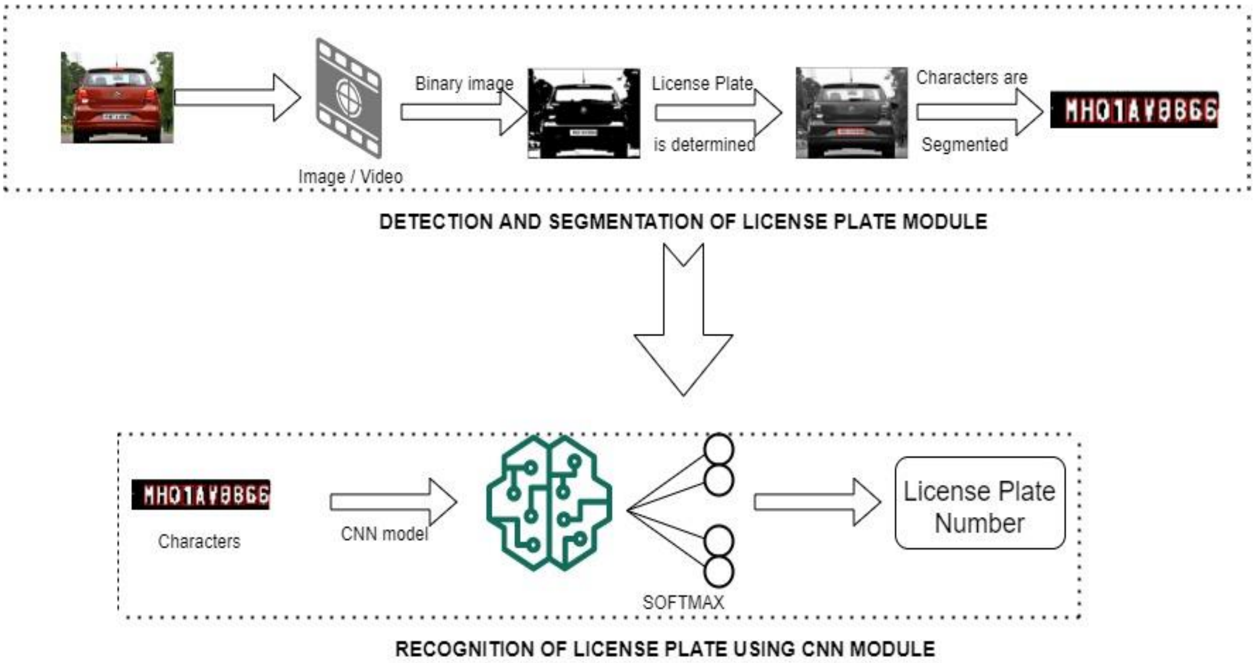


Image / Video
Binary image
License Plate is determined
Characters are Segmented
MH01AV8866
DETECTION AND SEGMENTATION OF LICENSE PLATE MODULE
MH01AV8866
Characters
CNN model
SOFTMAX
License Plate Number
RECOGNITION OF LICENSE PLATE USING CNN MODULE

Figure 2. Architecture Diagram

3.1. Image capturing

The input to this model can be in the form of videos or images. If video is provided as input then the whole video is separated into several frames and then one frame is passed to the model for further steps. The image captured is then passed to the license plate detection module to identify the license plate of the vehicle.

3.2. License Plate Detection

Image pre-processing is a crucial step while dealing with images. The captured images are preprocessed according to the dimensions required for the proposed ALPDR model. Then the license plate detection module identifies the license plate from the input images by performing RGB to grayscale conversion and binarization of the image.

Algorithm 1: Automatic License Plate Detection and Recognition

```
Input: License plate image, $b_{dim}$ - bounding box dimension
Output: Character recognition - Recognized characters from the License plate
1: Convert the captured license plate image to gray scale image
2: Perform binarization of the image using Otsu thresholding algorithm
3: Calculate $b'_{dim}$ of the input image
4:    if ($b'_{dim} < b_{dim}$)    then
5:        discard $b'_{dim}$
6:        goto step (3)
7:    else
8:        segment the characters
9:        convert the black pixel to white pixel and vice versa
10:       check the width and add it to the list
11:       save characters order
12:       Recognize the segmented characters using the trained CNN model
13:       Prints the recognized characters as output
```

3.2.1. RGB to Grayscale Conversion

In a computer vision system, edge detection is considered as an information reduction process that offers region boundary information by filtering out unnecessary information for the next steps of processes. Grayscale image use just one channel of color where only 8 bits is necessary to represent this image whereas in case of coloured images three channels are used where 24 bits are necessary to represent the image. Not in all cases of edge detection grayscale images can be used. In our case, we need to detect the edges of the license plate and the rest of the information is considered as noise and needs to be ignored. The extra information available in colored images can be considered as noise and be removed from the images when using grayscale images. Hence, we use the grayscale image over the colored image in our usecase.

3.2.2. Binarization of the Image

To convert the image containing pixels to a binary image is called binarization, i.e. containing pixels of black and white. The aim of binarization is to detect the edges available in the image. These edges detected in the binary image will be useful to detect the license plate. To create a binary image from a grayscale image Otsu's thresholding algorithm [40] available in scikit-image is used which is the simplest way to segment objects from the background.

Otsu's approach [40] works by iterating over all conceivable threshold values and calculating a measure of spread for the pixel levels on each side of the threshold, i.e. the pixels that either fall in foreground or background. The goal is to obtain the least threshold value for the total of foreground and background spreads. Once the edges in the license plate are detected a red color bounding box is formed over the license plate.

Before the bounding box is formed we check for a condition whether the bounding box predicted is very small (by comparing to the sample license plate dimension that is passed to the model). If the predicted bounding box is very small then we discard it and assume that it could not be a license plate. Once the license plate is detected by satisfying the condition a red color rectangular box is drawn over it as shown in Fig 3.

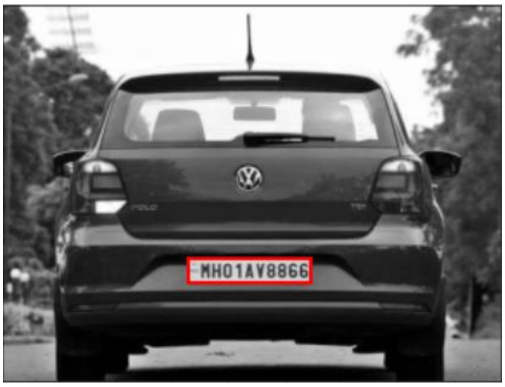


Figure 3. License Plate Detection

3.3. Characters Segmentation

In the detection step, the license plate is detected through a bounding box over the license plate. In this segmentation stage, the bounding boxes are used over all the characters available in the license plate. This is the important step, only if the characters are segmented properly the model will be able to recognize the characters.

Character segmentation is performed using the existing library functions available in matplotlib. At first, we invert the detected license plate from black pixel to white pixel and vice versa. In this step, we again specify the width of the license plate and ensure that it does not exceeds the specified width. We use region props available in Skimage and patches from matplotlib to make the bounding boxes over the characters as shown in Fig 4. The rectangle class from matplotlib patches is used to segment the specific characters based on the width, height, and rotation angle. We resize the boxed characters and append them to a list so that we don't change the order in which we are predicting the letters.

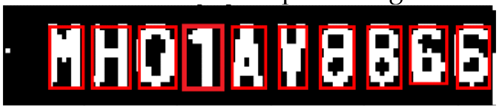


Figure 4. License Plate Characters Segmentation

3.4. Character Recognition

The final step is to recognize the segmented characters from the license plate. We train a Convolutional Neural Network (CNN) with two convolutional layers for recognizing the segmented characters. Since the focus is to develop a lightweight model, we proposed the CNN model with a minimum number of layers and parameters. The CNN model can efficiently learn the low level features from the input and can classify it. Hence, it is used in the recognition module of this work.

3.4.1. Convolutional Neural Network Model

To map image data to an output variable, CNNs were designed. They have been so popular that they have been given priority for any sort of prediction issue involving image information as input. The justification for using the Convolutional Neural network is that the important features of an image can be identified automatically without any human supervising the model. CNN's are very effective in the case of classification problems. Hence, the CNN model is used to correctly recognize the segmented images from the license plate.

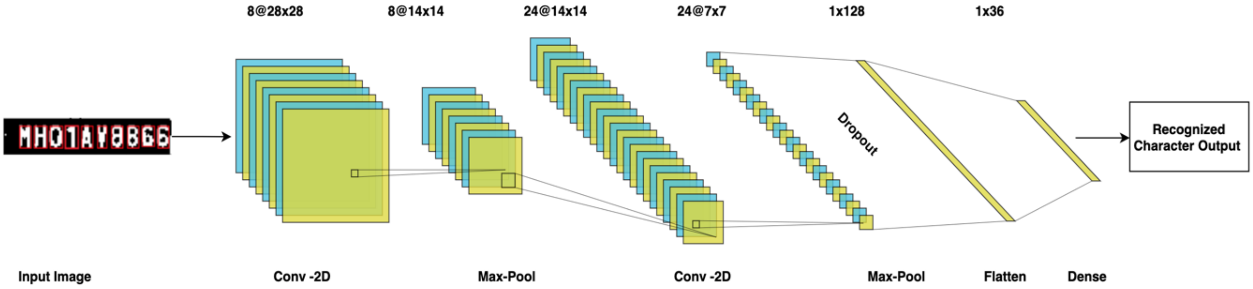


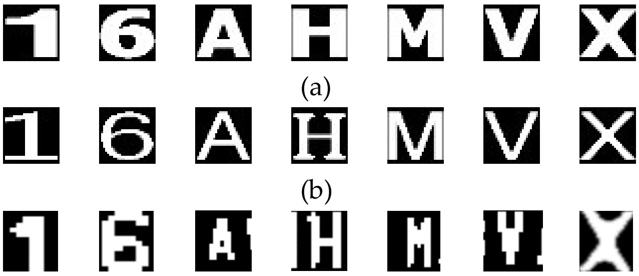
Figure 4. Proposed CNN Architecture

The proposed CNN architecture consists of combinations of 2D convolution and pooling layers at the end we have flatten and dense layers. The proposed CNN model takes input from the segmentation process output. The model accepts the input of size 28x28. Since we feed in individual segmented characters as input 28x28 image dimension is sufficient to recognize the character. The model contains two 2D-convolution layers. We used max pooling layers of size 2x2 between the convolution layers to extract the low level features from the images. At the end of the model, we used a flatten layer to convert the input dimensions into a single dimension and dense layers at the last for multi-nominal classification. To measure the loss, categorical cross-entropy whereas the optimizer used was Adam. The architecture of the proposed CNN model is shown in figure 4. The original input and the recognized output by the CNN model are shown in Table 2 with different datasets.

4. Experiments & Results

4.1. Dataset Details

To train and test the model we used 3 different types of datasets extracted from the Chars74k [41] dataset which is commonly used for license plate character recognition. The variations in all the three datasets are as follows, Dataset A contains 834 images with 36 classes (A-Z and 0-9) that were extracted from the Chars74K dataset. These 834 images are clear without any noise being added to them. The width of the characters is almost the same. But in a real-life scenario, the images may be thicker and sometimes very thin. The sample dataset is shown in fig 5 (a). In dataset B we added some more images manually and it contained 998 images with the same 36 classes. Totally 164 more images with the existing images were added. These newly added images were manually cropped from using the snipping tool from the Chars74K dataset and even reduced the pixels for some images in all 36 classes. Some sample images are shown in fig 5 (b). This is the third type of dataset that we used to train the model. Dataset C had a total of 1229 images. A total of 231 images were added. All the 231 images added in this C dataset were collected manually from the internet. Some images included dark and light backgrounds. The thickness of the characters was different and some images were cropped manually. Some samples of images are shown in fig 5 (c).


(a)

(b)

(c)

Figure 5. Sample Images from Chars74k dataset (a) Dataset A (b) Dataset B (c) Dataset C

4.2. Experimental Setup

The experiment was carried out in Google Colab, a cloud server. In this cloud server, we used Tesla P100 as a GPU with 16GB RAM for enhancing the running time. The training tools used in this case were Python, Keras, Tensorflow etc.

4.3. Experimental Analysis

Several experiments with different batch sizes, and with different datasets as shown in Table 2 were carried out to find a better model that would be more generalized in all cases. Table 2 shows the variations of the predicted output with its accuracy when using a different dataset for the same number of iterations. To validate the accuracy of the predicted output we have used the kappa score. Kappa score is one of the evaluation matrices for measuring accuracy. We trained the model by changing several parameters to find out the best point where our model would perform well. We have mentioned some of the hyperparameter changes and their effect on model performance is shown in Table 2.

Table 2. Experimentation of the model using different datasets

Dataset Used	Epoch / Batch Size	Training Accuracy	Validation Accuracy	Predicted Output	Prediction Accuracy (Kappa Score)	Letters Predicted Incorrectly
A	100 / 1	98.15	95.12	TSD7FI353Z	67.03%	3
B	100 / 1	97.68	95.57	TS07FI353Z	77.7%	2
C	100 / 1	98.97	96.88	TS07FX3534	100%	0

*The original output is TS07FX3534

When dataset A was used to train the model we got and maximum training accuracy of 98.15, still the model didn't perform well in predicting the letter and got a very low accuracy by predicting 3 characters incorrectly. The main reason for the incorrect classification is the variation in the dataset. Dataset A didn't have much variation and hence the model was unable to classify characters with noise or the one which was blurred. In the case of the C dataset, there were so many variations of the image like blurring and cropped images so the model could able to classify them with higher accuracy.

The training accuracy and validation accuracy of the model using dataset C is plotted in Fig. 6 and the training and validation loss is shown in Fig. 7.

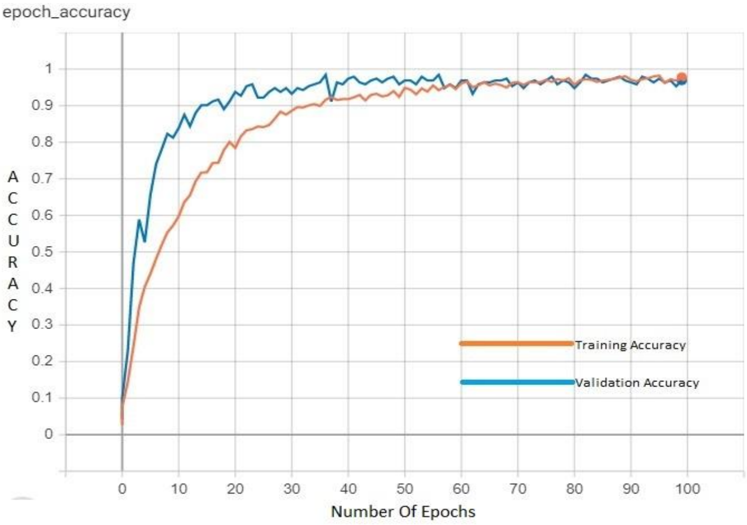


Figure 6. Training and Validation Accuracy

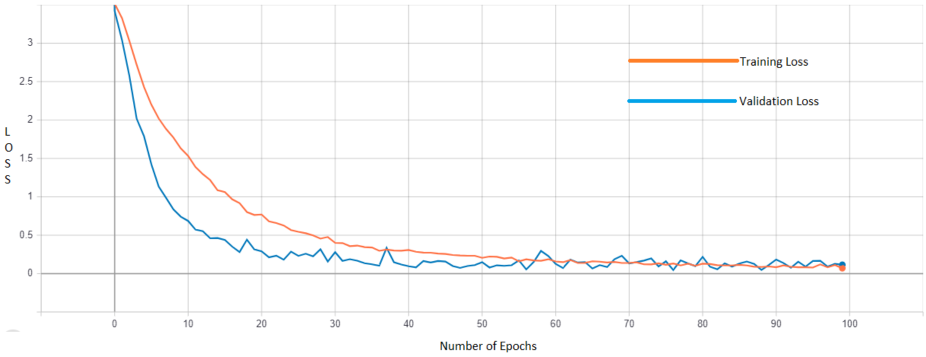


Figure 7. Training and Validation Loss

4.4. Model Performance in Each Stage

As discussed above the model includes three stages and the testing of all the three stages have been performed with images captured from real-time which had a total of 20 images and several videos. The performance of the model has been analyzed with eth real-time data in all stages to find out where the model lags. The result in all three stages are as follows:

4.4.1. Detection of License Plate

The model has been tested on several real-time vehicles by passing an image and video taken in real-time to check how well the model can detect the license plate and in which case it is failing to detect the license plate. Real-time images as shown in Fig. 8 is passed to the model in this stage to detect the license plate.

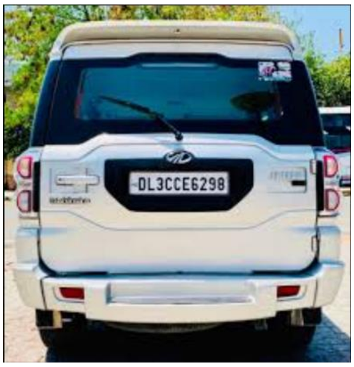


Figure 8. Example of input image to ALPDR model

When tried with more than 20 real-time images in different conditions, the model achieved 91.1% accuracy in detecting the license plate. The main reason for the reduction in accuracy was the additional edges available in most of the images.

4.4.2. Segmentation of Characters

From the License plate detected the character available in the license plate are segmented (or boxed). With all the 20 real-time images available, we test the segmentation accuracy and got a satisfactory accuracy of 96.67% which is efficient. The problem in the segmentation of the characters occurred due to the extra noise present in the license plate.

4.4.3. Recognition of Segmented Characters

All the characters segmented from the license plate were tested by the trained CNN model to identify the prediction accuracy. All the characters segmented from the license plate under different situations such as day and night images (from step 2 i.e. Segmentation of Characters) were tested to verify how well CNN can recognize these characters and it gave an accuracy of 98.8% accuracy. The model is unable to recognize the characters when blurred heavily. The performance of the proposed ALPDR model in each stage is visualized in Fig. 9.

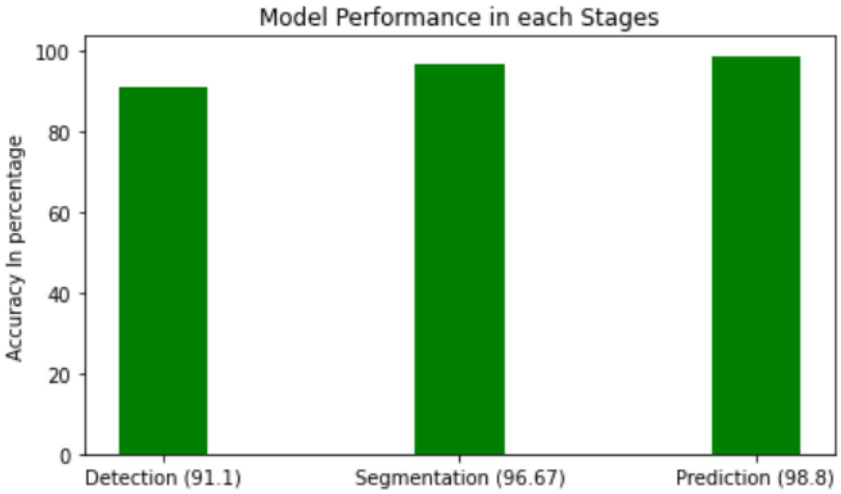


Figure 9. ALPDR model performance in detection, segmentation, and prediction

4.5. Performance Evaluation

This section evaluates the performance of the proposed ALPDR model with other state-of-the-art methods. Table 3 compares the results of the proposed model and other models. It also describes the methodology through which the proposed model can get better accuracy. From the evaluation, it is evident that the proposed ALPDR model is efficient than other state-of-the-art approaches.

Table 3. Performance evaluation with state-of-the-art approaches

Reference	Detection Accuracy (%)	Recognition Accuracy (%)	Methodology
Proposed Method	**91.1**	**98.8**	Bounding box for detection and CNN for classification
Henry and Chen [28]	78	98.2	YOLO deep learning framework
Xie *et al.* [42]	90	97.7	Image Processing and Neural Network
Astawa *et al.* [43]	73	96	Sliding Window, Hog and SVM
Sharma [38]	63.4	67.98	Template Matching Techniques
Kusumadewi *et al.* [39]	80	80	Template Matching and Bounding Box

5. Discussion

In this section, we discuss the highlights and limitations of the proposed ALPDR model. Based on the experimental analysis and the performance evaluation it is evident that the proposed ALPDR model is efficient in detecting, segmenting, and recognizing license plates of cars in different conditions such as day, night, varying distance, etc. The ALPDR model is trained with the public dataset Chars74k which contains images of different characters. We have generated three different types of the dataset from the Chars74k dataset to analyze the efficiency of the model. We observed that the dataset which consists of varying images such as blurred, and cropped images yield better accuracy in terms of detection and recognition. Then the proposed ALPDR model is tested in the real-time scenario, the model has recognized the characters in the license plate with 98.8 accuracy which is superior to many state-of-the-art approaches. The details of the performance evaluation are shown in Table 3.

Since the focus of our proposed model is to build a lightweight ALDPR model that recognize the licence plate efficiently, there are few limitations included in our study. The proposed model uses an edge detection technique to detect the license plate rather than using object detection. Due to the limitation of the dataset we used for this study, we adopted an edge detection technique to detect the license plate which proves to be efficient for our model. Another challenge we faced with Indian vehicles is that even though there is a clear policy on the size and colors of the vehicle license plate, still there are no common standards followed for the font and size of the license plate. Hence, our model sometimes produced incorrect character recognitions when the characters and numbers are written in different fonts without complying with the policies. Further, the results we achieved are for standing vehicles, the model is not tested with moving vehicles. In the future, we would like to extend our work for moving vehicles by capturing the video frames. Also, we would like to make the deep learning model deeper with appropriate regularization to improve the detection and recognition accuracy.

6. Conclusion

Aiming at the practicability of License Plate Detection and Recognition in the parking area, tolls, and other places where a huge number of vehicles enter and exit, an efficient Automatic License Plate Detection and Recognition (ALPDR) model is designed. In this research paper, a three-step process model to detect the license plate and to recognize the characters from the license plate is proposed. The proposed model uses convolutional features to recognize the character and achieved an accuracy of 98.8% in recognizing the characters. In the detection of the license plate, the model achieved an accuracy of 91.1%. The proposed ALPDR model has experimented with real-time car images with different license plate formats and the performance evaluation proves that the proposed model is efficient compared to state-of-the-art methods. In the future work, the proposed model will be integrated with IoT-based systems to efficiently detect the license plate of the moving vehicles also would like to add some super-resolution techniques to improve the license plate detection in different weather conditions.

References

[1] Wengang Zhou, Houqiang Li, Yijuan Lu and Qi Tian, "Principal Visual Word Discovery for Automatic License Plate Detection", *IEEE Transactions on Image Processing*, Print ISSN: 1057-7149, Online ISSN: 1941-0042, pp. 4269–4279, Vol. 21, No. 9, 15th May 2012, Published by IEEE, DOI: 10.1109/TIP.2012.2199506, Available: https://ieeexplore.ieee.org/document/6200342.

[2] Shaoqing Ren, Kaiming He, Ross Girshick and Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Print ISSN: 0162-8828, Online ISSN: 1939-3539, pp. 1137–1149, Vol. 39, No. 6, 6th June 2017, Published by IEEE, DOI: 10.1109/TPAMI.2016.2577031, Available: https://ieeexplore.ieee.org/document/7485869.

[3] David A. Forsyth and Jean Ponce, *Computer Vision: A Modern Approach*, 1st ed. Pearson Publication, Prentice Hall Professional Technical Reference, 2002.

[4] Constantine Papageorgiou and Tomaso Poggio, "Trainable System for Object Detection", *International Journal of Computer Vision*, Print ISSN: 0920-5691, Online ISSN: 1573-1405, pp. 15–33, Vol. 38, No. 1, June 2000, Published by Springer, DOI: 10.1023/A:1008162616689, Available: https://link.springer.com/article/10.1023/A:1008162616689.

[5] Mori Shunji, Hirobumi Nishida and Hiromitsu Yamada, *Optical Character Recognition*, 1st ed. John Wiley & Sons, Inc., 1999.

[6] J. Andrew Onesimu and J. Karthikeyan, "An Efficient Privacy-Preserving Deep Learning Scheme for Medical Image Analysis", *Journal of Information Technology Management*, Print ISSN: 2008-5893, Online ISSN: 2423-5059, pp. 50–67, Vol. 12, No. Special Issue: The Importance of Human Computer Interaction: Challenges, Methods and Applications.,Published by Faculty of Management, University of Tehran, December 2021, DOI: 10.22059/jitm.2020.79191, Available: https://jitm.ut.ac.ir/article_79191.html.

[7] J. Andrew, M. Divyavarshini, P. Barjo and I. Tigga, "Spine Magnetic Resonance Image Segmentation using Deep

Learning Techniques", in *Proceedings of the IEEE 2020 6th International Conference on Advanced Computing and Communication Systems, ICACCS 2020*, 6-7 March 2020, Online ISBN: 978-1-7281-5197-7, E-ISBN: 978-1-7281-5196-0 DOI: 10.1109/ICACCS48705.2020.9074218, pp. 945–950, Available: https://ieeexplore.ieee.org/document/9074218.

[8] David A. Forsyth and Jean Ponce, *Computer Vision: A Modern Approach*, 2nd ed. Pearson Publication, Prentice Hall Professional Technical Reference, 2002.

[9] J. Andrew Onesimu, Abhishikt Kadam, K. Martin Sagayam and A. Ahmed Elngar, "Internet of Things Based Intelligent Accident Avoidance System for Adverse Weather and Road Conditions", *Journal of Reliable Intelligent Environments*, Print ISSN: 2199-4668, Online ISSN: 2199-4676, pp. 1–15, 25th January 2021, Published by Springer, DOI: 10.1007/s40860-021-00132-7, Available : https://link.springer.com/article/10.1007/s40860-021-00132-7.

[10] Shibin David, R. S. Anand, Sarah Sheikh, S. Jebapriya, J. Andrew and S. Basil Xavier, "A Comprehensive Overview on Intelligent Mechanical Systems and its Applications", *Materials Today: Proceedings*, Print ISSN: 2214-7853, pp. 733–736, Vol. 37, No. Part 2, 7th January 2021, Published by Elsevier, DOI: 10.1016/J.MATPR.2020.05.737, Available: https://www.sciencedirect.com/science/article/pii/S2214785320343492.

[11] Abhishikt Kadam, J. Andrew, K. Martin Sagayam and Dang T. Hien, "Vehicle Automation and Car-Following Models for Accident Avoidance", *Przegląd Elektrotechniczny*, Print ISSN: 0033-2097, Online ISSN: 2449-9544, pp. 118–123, Vol. 96, No. 1, 1st January 2020, Published by Sigma-Not, DOI: 10.15199/48.2020.01.26, Available: http://pe.org.pl/abstract_pl.php?nid=11975.

[12] Jithmi Shashirangana, Heshan Padmasiri, Dulani Meedeniya and Charith Perera, "Automated License Plate Recognition: A Survey on Methods and Techniques", *IEEE Access*, Online ISSN: 2169-3536, pp. 11203–11225, Vol. 9, 29th December 2020, DOI: 10.1109/ACCESS.2020.3047929, Publlished by IEEE, Available: https://ieeexplore.ieee.org/document/9310202.

[13] Irina Valeryevna Pustokhina, Denis Alexandrovich Pustokhin, Joel J. P. C. Rodrigues, Deepak Gupta and Ashish Khanna, *et al.*, "Automatic Vehicle License Plate Recognition using Optimal K-Means with Convolutional Neural Network for Intelligent Transportation Systems", *IEEE Access*, Online ISSN: 2169-3536, pp. 92907–92917, Vol. 8, 7th May 2020, DOI: 10.1109/ACCESS.2020.2993008, Available: https://ieeexplore.ieee.org/document/9088975.

[14] Muhammet Sebul, Beratoğlu and Behçet Uğur Töreyin, "Vehicle License Plate Detector in Compressed Domain", *IEEE Access*, Online ISSN: 2169-3536, pp. 95087–95096, Vol. 9, 28th June 2021, DOI: 10.1109/ACCESS.2021.3092938, Available: https://ieeexplore.ieee.org/document/9466149.

[15] K. Indira, K. V. Mohan and Theegalapally Nikhilashwary, "Automatic License Plate Recognition", in *Recent Trends in Signal and Image Processing, Advances in Intelligent Systems and Computing*, Singapore: Springer Nature, Vol. 727, No. 1, pp. 67–77, 2019, Print ISBN: 978-981-10-8862-9, Online ISBN: 978-981-10-8863-6, DOI: 10.1007/978-981-10-8863-6_8, Published by Springer, Available: https://link.springer.com/chapter/10.1007/978-981-10-8863-6_8.

[16] R. Smith, "An Overview Of The Tesseract OCR Engine", in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2007, 23-26 September 2007, Vol. 2, Online ISBN: 0-7695-2822-8, E-ISBN:978-0-7695-2822-9, DOI:10.1109/ICDAR.2007.4376991, pp. 629–633, Published by IEEE, Avaiable: https://ieeexplore.ieee.org/document/4376991.

[17] S. Ben Driss, M. Soua, R. Kachouri and M. Akil, "A Comparison Study Between MLP And Convolutional Neural Network Models for Character Recognition", in *Real-Time Image and Video Processing 2017*, pp. 1022306, 1st May 2017, Vol. 10223, Published by International Society for Optics and Photonics, DOI: 10.1117/12.2262589, Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/10223/1/A-comparison-study-between-MLP-and-convolutional-neural-network-models/10.1117/12.2262589.short.

[18] T. S. R. Mhathesh, J. Andrew, K. Martin Sagayam and Lawrence Henesey, "A 3d Convolutional Neural Network For Bacterial Image Classification", in *Intelligence in Big Data Technologies—Beyond the Hype, Advances in Intelligent Systems and Computing*, Singapore: Springer Nature, Vol. 1167, 2021, pp. 419–431, Print ISBN: 978-981-15-5284-7, Online ISBN:978-981-15-5285-4, DOI:10.1007/978-981-15-5285-4_42, Published by Springer, Available: https://link.springer.com/chapter/10.1007/978-981-15-5285-4_42.

[19] Abhirup Khanna and Rishi Anand, "IoT Based Smart Parking System", in *Proceedings of the 2016 International Conference on Internet of Things and Applications, IOTA 2016*, Pune, India, 22-24 January 2016, Print ISBN: 978-1-5090-0045-6, Online ISBN: 978-1-5090-0044-9, DOI: 10.1109/IOTA.2016.7562735, pp. 266–270, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/7562735.

[20] Cheng- Hung Lin, Yong- Sin Lin and Wei-Chen Liu, "An Efficient License Plate Recognition System Using Convolution Neural Networks", in *Proceedings of 4th IEEE International Conference on Applied System Innovation 2018, ICASI 2018*, 13-17 April 2018, Chiba, Japan, Print ISBN: 978-1-5386-4343-3, Online ISBN: 978-1-5386-4342-6, pp. 224–227, DOI: 10.1109/ICASI.2018.8394573, Available: https://ieeexplore.ieee.org/abstract/document/8394573.

[21] Guocong Lin, Binqiang Xue, Boyang Xu and Chuanguang Chen, "License Plate Recognition Based on Mathematical Morphology and Template Matching", in *Proceedings of the 2019 Chinese Automation Congress, CAC 2019*, 22-24 November 2018, Hangzhou, China, Print ISBN: 978-1-7281-4095-7, Online ISBN: 978-1-7281-4094-0, DOI: 10.1109/CAC48633.2019.8996973, pp. 405–410, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/8996973.

[22] Ching-Tang Hsieh, Liang-Chun Chang, Kuo-Ming Hung and Hsieh Chang Huang, "A Real-Time Mobile Vehicle License Plate Detection and Recognition for Vehicle Monitoring and Management", in *Proceedings of the 2009 Joint Conferences on Pervasive Computing, JCPC 2009*, 3-5 December 2009, Tamsui, Taiwan, Print ISBN: 978-1-4244-5227-9, CD: 978-1-4244-5228-6, DOI: 10.1109/JCPC.2009.5420189, pp. 197–202, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/5420189.

[23] Cheng-Hung Lin and Yi-Sin Sie, "Two-Stage License Plate Recognition System Using Deep learning", in *Proceedings of the 2019 8th International Conference on Innovation, Communication and Engineering, ICICE 2019*, Zhengzhou, China, 25-30 October 2019, Print ISBN: 978-1-7281-5840-2, Online ISBN: 978-1-7281-5839-6, DOI: 10.1109/ICICE49024.2019.9117277, pp.132–135, Available: https://ieeexplore.ieee.org/abstract/document/9117277.

[24] Mingjun Zhang, Wenjing Yu, Jinsheng Su and Weibin Li, "Design Of License Plate Recognition System Based on Machine Learning", in *Proceedings of the 2019 IEEE 4th International Conference on Image, Vision and Computing, ICIVC 2019*, 5-7 July 2019, Xiamen, China, Print ISBN: 978-1-7281-2326-4, Online ISBN: 978-1-7281-2325-7, DOI: 10.1109/ICIVC47709.2019.8981074, pp. 518–522,2019, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/8981074.

[25] Yousri Kessentini, Mohamed D. Besbes, Sourour Ammar and Achraf Chabbouh, "A Two-Stage Deep Neural Network For Multi-Norm License Plate Detection And Recognition", in *Expert Systems with Applications*, Print ISSN:0957-4174, pp. 159170,2019, Vol. 136, 18th June 2019, Published by Elsevier, DOI:10.1016/j.eswa.2019.06.036, Available: https://www.sciencedirect.com/science/article/abs/pii/S0957417419304361.

[26] Jaskirat Singh and Bharat Bhushan, "Real Time Indian License Plate Detection Using Deep Neural Networks And Optical Character Recognition Using LSTM Tesseract", in *Proceedings - 2019 International Conference on Computing, Communication and Intelligent Systems, ICCCIS 2019*, 18-19 October 2019, Greater Noida, India, Print ISBN: 978-1-7281-4827-4, Online ISBN: 978-1-7281-4826-7, DOI: 10.1109/ICCCIS48478.2019.8974469, pp. 347–352, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/8974469.

[27] Naaman Omar, Abdulkadir Sengur and Salim Ganim Saeed Al-Ali, "Cascaded Deep Learning-Based Efficient Approach for License Plate Detection and Recognition", in *Expert Systems with Applications*, 13th February 2020, Print ISSN:0957-4174, pp. 113-280, Vol. 149, DOI: 10.1016/j.eswa.2020.113280, Published by Elsevier, Available: https://www.sciencedirect.com/science/article/abs/pii/S0957417420301056.

[28] Hendry and Rung Ching Chen, "Automatic License Plate Recognition via Sliding-Window Darknet-YOLO Deep Learning", *Image and Vision Computing*, Vol. 87, July 2019, Print ISSN: 0262-8856, pp. 47–56, 2019, DOI: 10.1016/j.imavis.2019.04.007, Available: https://www.sciencedirect.com/science/article/abs/pii/S0262885619300575.

[29] Rayson Laroca, Luiz A. Zanlorensi, Gabriel R. Gonçalves, Eduarto Todt, William Robson Schwartz and David Menotti, "An Efficient and Layout-Independent Automatic License Plate Recognition System Based on the YOLO Detector", *IET Intelligent Transport Systems*, Print ISSN 1751-956X, Online ISSN 1751-9578, pp. 483–503, Vol. 15, No. 4, September 2021, DOI: 10.1049/itr2.12030, Published by IET, Available: https://arxiv.org/abs/1909.01754.

[30] Zied Selmi, Mohamed Ben Halima and Adel M. Alimi, "Deep Learning System for Automatic License Plate Detection and Recognition", in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 9-15th November 2017, Print ISBN: 978-1-5386-3587-2, Online ISBN: 978-1-5386-3586-5, Vol. 1, pp. 1132–1138, DOI: 10.1109/ICDAR.2017.187, Available: https://ieeexplore.ieee.org/abstract/document/8270118.

[31] Chris Henry, Sung Yoon Ahn and Sang- Wung Lee, "Multinational License Plate Recognition using Generalized Character Sequence Detection", in *IEEE Access*, 19th February 2020, Online ISSN : 2169-3536, Vol. 8, pp. 35185–35199, 2020, DOI: 10.1109/ACCESS.2020.2974973, Available: https://ieeexplore.ieee.org/abstract/document/9003211.

[32] Sérgio Montazzolli and Claudio Jung, "Real-Time Brazilian License Plate Detection and Recognition using Deep Convolutional Neural Networks", in *Proceedings - 30th Conference on Graphics, Patterns and Images, SIBGRAPI 2017*, 17-20 October 2017, Niteroi, Brazil, Print ISBN: 978-1-5386-2220-9, Online ISBN: 978-1-5386-2219-3, DOI: 10.1109/SIBGRAPI.2017.14, pp. 55–62, Available: https://ieeexplore.ieee.org/abstract/document/8097294.

[33] Jampu Raju, C. V. Raghu, Sudhish N. George and T. S. Bindiya, "Development of a Hand Held Device for Automatic License Plate Recognition", in *Proceedings of the 2020 IEEE International Conference on Industry 4.0, Artificial Intelligence and Communications Technology, IAICT 2020*, 7-8 July 2020, Bali, Indonesia, Print ISBN: 978-1-7281-9337-3, Online ISBN: 978-1-7281-9336-6, DOI: 10.1109/IAICT50021.2020.9172026, pp. 102–106, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/9172026.

[34] Madhushee Mondal, Parmita Mondal, Nilendu Saha and Paramita Chattopadhyay, "Automatic Number Plate Recognition using CNN based Self Synthesized Feature Learning" ,in *Proceedings of the 2017 IEEE Calcutta Conference, CALCON 2017*, 2-3 Decemeber 2017, Kolkata, India, Print ISBN: 978-1-5386-3746-3, Online ISBN: 978-1-5386-3744-9, DOI: 10.1109/CALCON.2017.8280759, pp. 378–381, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/8280759.

[35] Meng Dong, Dongliang He, Chong Luo, Dong Liu and Wenjun Zeng, "A CNN-Based Approach for Automatic License Plate Recognition in the Wild", in *Proceedings of the British Machine Vision Conference 2017, BMVC 2017*, ISBN: 1-901725-60-X, September 2017, DOI: 10.5244/c.31.175, pp. 1–12, Published by BMVC, Available: http://www.bmva.org/bmvc/2017/papers/paper175/paper175.pdf.

[36] R. Rajathilagam, K. Sivamani, R. Seetharaman and D. Nedumaran, "Neural Network Based Vehicle Number Plate Recognition System",in *Proceedings of the 2019 2nd International Conference on Power and Embedded Drive Control, ICPEDC 2019*, 21-23 August 2019, Chennai, India, Print ISBN: 978-1-7281-2415-5, Online ISBN: 978-1-7281-2414-8, DOI: 10.1109/ICPEDC47771.2019.9036497, pp. 102–104, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/9036497.

[37] Divya Rastogi, Mohammad Shahbaz Khan, Kanav Jindal and Karan Singh, "A Real-Time Vehicle Number Plate Detection And Recognition System", J*ournal of Xi'an University of Architecture & Technology*, Print ISSN: 1006-7930, Vol. 12, No. 4, pp. 5005–5009, Published by Science Press, Available: http://www.xajzkjdx.cn/gallery/484-april2020.pdf.

[38] Gajendra Sharma, "Performance Analysis of Vehicle Number Plate Recognition System using Template Matching Techniques", *Journal of Information Technology & Software Engineering*, Print ISSN: 2165-7866, Vol. 08, No. 02, 2018, DOI: 10.4172/2165-7866.1000232.

[39] Ira Kusumadewi, Christy Atika Sari, De Rosal Ignatius Moses Setiadi and Eko Hari Rachmawanto, "License Number Plate Recognition using Template Matching and Bounding Box Method", *Journal of Physics: Conference Series*, Print ISSN: 1742-6588, Online ISSN: 1742-6596, May 2019, Vol. 1201, No. 1, DOI: 10.1088/1742-6596/1201/1/012067, Published by IOP Science, Available: https://iopscience.iop.org/article/10.1088/1742-6596/1201/1/012067.

[40] Ta Yang Goh, Shafriza Nisha Basah, Hanizha Yazid, Muhammad Juhairi Aziz Safar and Fathinul Syahir Ahmad Saad, "Performance Analysis Of Image Thresholding: Otsu technique", *Measurement: Journal of the International Measurement Confederation*, Print ISSN: 0263-2241, Online ISSN: 1873-412X, Vol. 114, pp. 298–307, 30th September 2017, DOI: 10.1016/j.measurement.2017.09.052, Published by Elsevier, Available: https://www.sciencedirect.com/science/article/abs/pii/S0263224117306243.

[41] O. Akbani, A. Gokrani, M. Quresh, F. M. Khan, S. I. Behlim and T. Q. Syed, "Character Recognition In Natural Scene Images", in *Proceedings of the 2015 International Conference on Information and Communication Technologies, ICICT 2015*, 12-13 December 2015, Karachi, Pakistan, Online ISBN: 978-1-4673-8907-5, DVD ISBN:978-1-4673-8906-8, Vol. 7, DOI: 10.1109/ICICT.2015.7469575, pp. 1-6, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/7469575.

[42] Fei Xie, Ming Zhang, Jing Zhao, Jiquan Yang, Yijian Liu and Xinyue Yuan, "A Robust License Plate Detection and Character Recognition Algorithm based on a Combined Feature Extraction Model and BPNN", *Journal of Advanced Transportation*, Print ISSN: 0197-6729, Online ISSN: 2042-3195, Vol. 2018, Article ID 6737314, 14 pages, 26th September 2018, DOI: 10.1155/2018/6737314, Available: https://doi.org/10.1155/2018/6737314.

[43] Inga Astawa, I. Gusti Ngurah Bagus Caturbawa, I Made Sajayasa and I Made Ari Dwi Suta Atmaja, "Detection of License Plate Using Sliding Window, Histogram of Oriented Gradient and Support Vector Machines Method", *Journal of Physics: Conference Series,* Print ISSN: 1742-6588, Online ISSN: 1742-6596, Vol. 953, No. 1, pp. 0–6, 2018, DOI: 10.1088/1742-6596/953/1/012062, Available: https://iopscience.iop.org/article/10.1088/1742-6596/953/1/012062.