ANPR Based Toll Tax Collection System

Darshankumar C. Dalwadi
Associate Professor
Department of Electronics & Communication Engineering
Birla Vishvakarma Mahavidyalaya Engineering College
Vallabh Vidyanagar, Gujarat, India
darshan.dalwadi@bvmengineering.ac.in

Sanket Amin
Department of Electronics & Communication Engineering
Birla Vishvakarma Mahavidyalaya Engineering College
Vallabh Vidyanagar, Gujarat, India
amin.sanket.sa@gmail.com

***Abstract*— Automatic Number Plate Recognition (ANPR) is a crucial technology for toll tax collection systems as it enables efficient and automated license plate recognition. In this paper, we propose a machine learning-based ANPR system developed using TensorFlow, OpenCV, and EasyOCR. We trained our model on a dataset of 345 car images along with their corresponding XML files containing license plate information. The trained model was integrated into a real-time image-capturing system to extract license plate numbers for toll tax collection. Our system achieved high accuracy and efficiency in recognizing license plates, with promising results for real-world applications.**

Keywords— ANPR, license plate recognition, toll tax collection, machine learning, TensorFlow, OpenCV, EasyOCR

I. Introduction

Automatic Number Plate Recognition (ANPR) has become an integral part of toll tax collection systems worldwide due to its ability to automate the license plate recognition process, improve efficiency, and reduce human error [1]. ANPR is a challenging task that involves the detection and extraction of license plate numbers from images of moving vehicles [2]. In this paper, we present our ANPR system that utilizes machine learning techniques, specifically TensorFlow, OpenCV, and EasyOCR, to achieve accurate and real-time license plate recognition.

Gradual increase in vehicles on road, and complications to manage the ongoing traffic, there is a demand to enforce Intelligent Transportation System (ITS) [3]. It is where ANPR, which is an important subsystem, comes into light. ANPR was first introduced by Police Scientific Development Branch in Britain in the year 1976 [4]. Since then, there have been momentous development in the ANPR system, but there is a condition to realize modern-day traffic.

ITS has been acknowledged and established by many developed countries to oversee and control vehicle traffic [5]. In addition, ANPR systems are also used in Electronic Toll Collection (ETC), law enforcement, and access control in a restricted area [6]. Access Control Systems are a highly helpful application of ANPR since it may help discover potential security issues. With ANPR, the effort put forth by service personnel can be greatly reduced, along with the influence of human factors and errors [7].

The primary objective of the ANPR system is to read license plates, without any human involvement [8]. An ANPR system usually receives an image of a vehicle as input and, outputs the content of the license plate, as text. The license plate can be found and read using methods including image processing and object recognition [9].

In recent times, there have been numerous variations in the policies of ANPR system across the world. Specifically in India, The Ministry of Road Transport and Highways (MoRTH) has issued several guidelines and regulations on the use of ANPR systems in India. The MoRTH guidelines also specify the types of ANPR systems that can be used in India. These systems must meet few requirements, such as, ANPR system should be reliable and accurate, it must be able to read vehicle registration plates in a variety of conditions, including low-light conditions and in the presence of dirt and debris, and also it must be secure and protect the privacy of data collected. In addition to recent variations in policies, there have been many emerging trends related to ANPR. The use of ANPR systems to collect and analyze traffic data, combat terrorism and other criminal activity, and use of ANPR systems to enforce parking regulations are a few of the trends.

Although there is much research conducted on the ANPR system, there are still a few research gaps that need to be filled. Improving the accuracy of ANPR systems in low-light conditions is one of the research gaps that most needed to be conducted. The efficiency of the model drops significantly in low light conditions, compared to normal lighting condition.

The three stages of the recognition process- number plate extraction, segmentation, and character recognition- are all included in the performance rate.

II. Methodology

Our ANPR system is developed using a combination of TensorFlow, OpenCV, and EasyOCR. TensorFlow, an open-source machine learning framework, is used to train our model on a dataset of 345 car images along with their corresponding XML files containing license plate information. The dataset is pre-processed using OpenCV, a popular computer vision library, to perform image resizing, normalization, and augmentation [10]. The pre-processed

2023 IEEE 11th Region 10 Humanitarian Technology Conference (R10-HTC) | 979-8-3503-2614-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/R10-HTC57504.2023.10461835

data is then used for training the neural network model in TensorFlow [11]. EasyOCR, a library for optical character recognition (OCR), is employed to extract license plate numbers from the captured car images in real time [12].

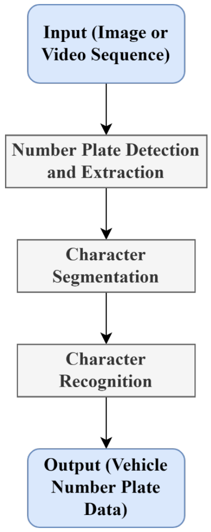

Fig. 1. Multi-stage ANPR System

A. Vehicle Image Capture

The first stage is the acquisition of an image, or capturing an image using the PC's built-in digital camera [13]. These captured photos are in RGB format, allowing for number plate extraction to proceed further. The database system includes the driver's personal information as well as a few license plate photos, abbreviations, and acronyms.

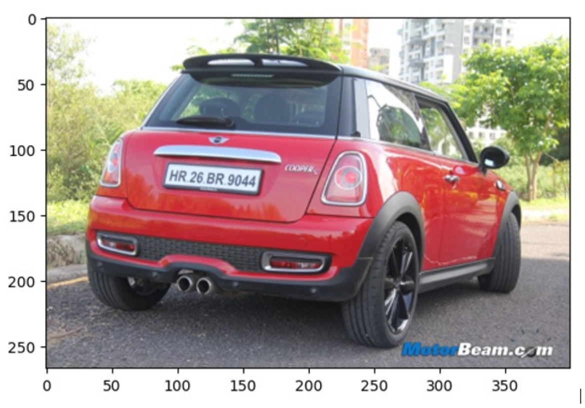

Fig. 2. Vehicle Image

B. Pre-Processing

In Fig. 2, the RGB image is shown that was captured. Numerous factors, such as optical system distortion, system commotion, a lack of presentation, the excessive relative motion of the camera or the vehicle, and others, have an impact on the captured image [14]. These factors degrade the captured vehicle image and have an unfavorable effect on subsequent image processing. Pre-processing of the collected image must therefore be performed before the primary image processing. This includes converting RGB to gray, noise elimination, and border enhancement for brightness, as shown in Fig. 3. In this figure, we have applied Bilateral filtering and Canny Edge Detection to the image.

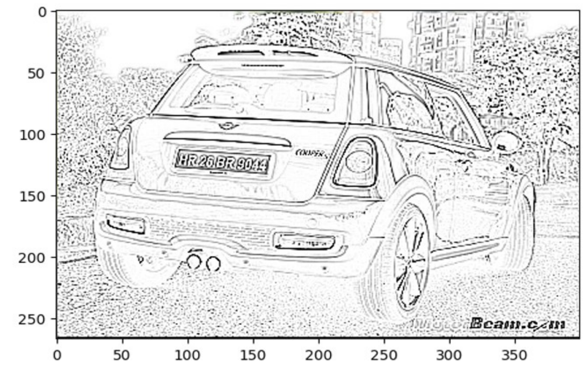

Fig. 3. Pre-Processed Image of Vehicle

C. Number Plate Extraction

Deep learning neural networks [15] have largely supplanted most statistical methods in computer vision systems in recent years due to their excellent object detection accuracy. Recognizing this, numerous studies on the detection of license plates have employed various forms of neural networks. The Basic step in the extraction of a number plate is to identify the plate size and a large number of plates are rectangular in shape. Since number plates are rectangular, we can use the shape as a key point to extract the number plate by eliminating the vehicle as shown in Fig. 6.

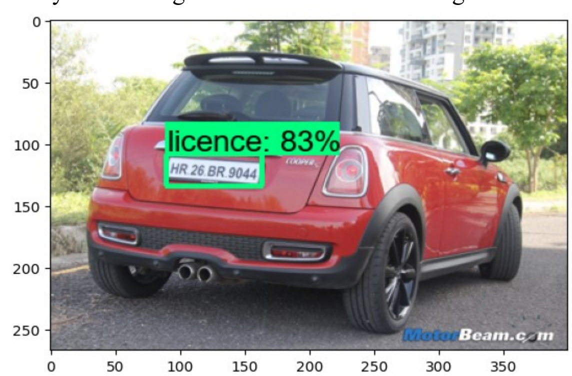

Fig. 4. Number Plate Acquisition by the Model

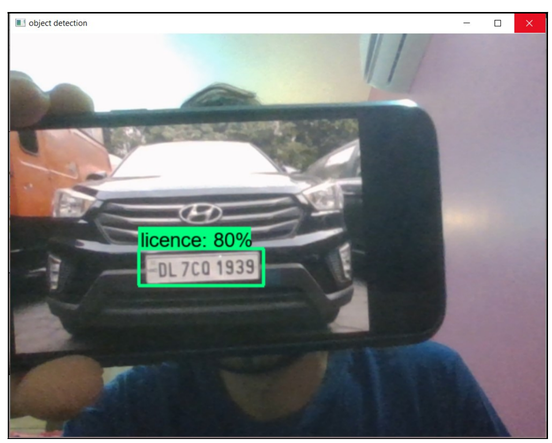

Fig. 5. Real-Time Number Plate Acquisition by the Model

D. Character Segmentation

The success of extracting number plates from the scene or image is absolutely necessary for the character segmentation stage [16]. The solitary number plate may have problems with contrast and different lighting circumstances, or it may be angled differently. Before segmenting the characters in such a case, pre-processing procedures, de-skewing, de-blurring, or any other approaches may need to be used, depending on the circumstances of the number plate. Based on the methods used, this step may be completed either during the extraction stage or after obtaining an isolated candidate area [17].

Neural networks are now a popular method for character segmentation, which makes use of CNN for computer vision-based tasks [18]. The CNN is given a localized number plate as input, and as an output, it generates the bounding boxes for each character. However, compared to more conventional computer vision-based algorithms, CNN execution can take longer and use more resources, depending on the dataset. Additionally, some deep learning-based number plate recognition pipelines have avoided explicit character segmentation in a further stage, which decreases the number of parameters and the processing expense.

E. Character Recognition

The act of identifying text from an image by comprehending and examining its patterns is known as text recognition. It is sometimes referred to as OCR or optical character recognition. OCR can be trained or deployed as a model that has already been trained. Recognition of the segmented characters is the last step in ANPR systems. Depending on the camera's position and focus level, the segmented characters may vary in size and thickness. Noise may break, distort, or affect the characters. This section discusses several character recognition techniques.

The advantages of implementing neural networks are that they can function independently as feature extractors and classifiers when given the raw pixel data. For character recognition, a variety of neural network architectures have been utilized, from straightforward multi-layer perceptron to probabilistic neural networks (PNN) and discrete-time cellular networks. Moreover, a lot of recent work has made use of CNN, which has demonstrated considerable promise in a variety of computer vision tasks.

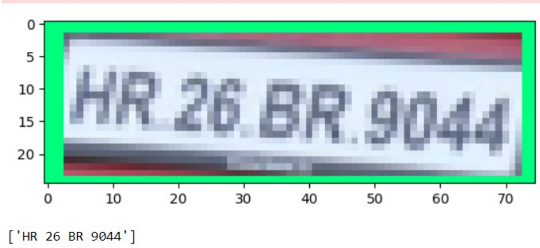

Fig. 6. Character Extraction of the Number Plate

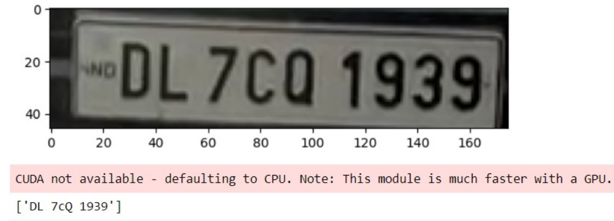

Fig. 7. Real-Time Character Extraction of the Number Plate

III. PROPOSED SYSTEM

In this research, we suggest an automated method for recognizing number plates and number plates that can extract number plate numbers from passing automobiles using image processing algorithms. The suggested method can be implemented without the installation of any additional hardware, such as a GPS or radio frequency identification system. Each passing car is photographed by unique cameras in the system, which then sends the photo to the computer to be processed by ANPR software. Different algorithms, including localization, orientation, segmentation, and optical character recognition (OCR), are used by plate recognition software.

After the successful extraction of the characters from the license plate of the vehicle, the captured image is saved in a folder with a unique name assigned to each image. Along with that, the characters are too stored in a .csv Excel sheet. There are two columns in the Excel sheet, one is for the image name and the other is for the characters that are extracted.

Aside from toll tax collection, this technology can also be utilized for security. Roadside car theft can also be detected with this technology. In order to use this system, no new hardware needs to be installed on automobiles. These cameras capture images, which are then processed by a computer. The system database keeps track of all vehicle traffic data for a very long time. As a result, multiple parking gates can be used at various times to get detailed traffic information.

A digital image capture device (camera), a processing unit, and several video analytics algorithms are components of typical ANPR systems. Additionally, since infrared lighting is used, such systems can operate around the clock due to their ability to read license plates on moving vehicles at night.

1. To start, the ANPR camera records pictures (a video stream or a snapshot) that include a license plate. We can also detect the vehicle in real time through a high-speed image sensor.
2. Next, the plate is located using computer vision and machine learning techniques (Object Detection).
3. To return the number plate number in text format, OCR software is then applied to the detected plate area.

In order to integrate the transformed number with other IT systems, it is typically stored in a database.

A. Convolutional Neural Networks (CNNs)

Due to its capacity to extract both local and global characteristics from images, convolutional neural networks (CNNs) are a type of deep learning algorithm that is frequently utilized in image identification applications. Convolutional, pooling, and fully connected layers are some of the layers that make up CNNs, which are built to automatically identify significant features from the input data. The convolutional layer applies convolution operation to the input image, using a set of learnable filters, also known as kernels or feature maps. Each filter scans through the image and computes element-wise multiplications followed by summation to obtain a feature map. This operation helps in capturing local patterns and spatial hierarchies in the image. The mathematical equation for convolution operation can be represented as:

$$h[i,j]=activationfunction(\sum(x[m,n]*w[i-m,j-n]+b)) \quad .(1)$$

where ‘h[i,j]’ is the value of the feature map at position ‘(i,j)’, ‘x[m,n]’ is the value of the input image at position ‘(m,n)’, ‘w[i-m, j-n]’ is the value of the filter at position ‘(i-m, j-n)’, ‘b’ is the bias term, and the activation function applies an element-wise non-linear transformation to introduce non-linearity into the model.

B. Bilateral Filter

The bilateral filter can be formulated as follows:

$$GB[I]p=1/Wpq\in S\mathrm{G}\sigma s\ (||p-q||)G\sigma r\ (|Ip-Iq|)Iq \quad .....(2)$$

Here, new terms have been added to the original equation in the form of the normalization factor and the range weight. The terms “σs” and “σr” refer to the minimum amplitude of an edge and the spatial extent of the kernel, or the size of the neighborhood, respectively. It makes sure that only pixels with intensity levels comparable to the center pixel are taken into account for blurring while maintaining sharp intensity changes. The edge is sharper the lower the value of “σr”. The equation tends to a Gaussian blur as “σs” goes to infinity.

C. Canny Edge Detection

The edges of a picture are found using Canny Edge Detection. It employs a multistage method and accepts a grayscale image as input. With the use of the Canny edge detection technology, the amount of data that needs to be processed can be drastically reduced while still extracting meaningful structural information from various vision objects. It is frequently used in many computer vision systems. According to Canny, the prerequisites for applying edge detection to various vision systems are largely the same. Thus, an edge detection solution to address these requirements can be implemented in a wide range of situations.

D. Convolution Operation

The convolutional operation is a fundamental operation in convolutional neural networks (CNNs) used for image recognition. It involves applying a filter or kernel to an input image to obtain feature maps. The convolutional operation can be mathematically represented as follows:

$$OutputFeatureMap(O) = InputImage(I)*Filter(W)+Bias(B) \quad ....(3)$$

where * denotes the convolutional operation, W represents the filter weights, B denotes the bias term, and O represents the output feature map.

E. Rectified Linear Unit (ReLU) Activation Function

ReLU is a popular activation function used in CNNs that introduces non-linearity to the model. It is mathematically defined as follows:

$$f(x) = max(0,x) \quad ....(4)$$

where x is the input to the ReLU function and f(x) is the output.

The rectified linear activation function, often known as ReLU, is a non-linear or piecewise linear function that, if the input is positive, outputs the input directly; if not, it outputs zero. In neural networks, particularly convolutional neural networks (CNNs) and multilayer perceptrons, it is the most widely employed activation function. Despite being straightforward, it outperforms earlier models like the sigmoid or tanh.

F. Cross-Entropy Loss

Cross-entropy loss is commonly used as a loss function in classification tasks. It measures the dissimilarity between predicted probabilities and ground truth labels. Entropy is the number of bits needed to send a probability distribution's randomly chosen event. A distribution with unequal probabilities of events has a higher entropy than one with a skewed distribution. The concept of cross-entropy, which is based on the concept of entropy from information theory, determines how many bits are needed to represent or convey an average event from one distribution in comparison to another. The cross-entropy loss for binary classification can be represented as:

$$L(y,y') = -[y*log(y')+(1-y)*log(1-y')] \quad ....(5)$$

where y represents the ground truth label (0 or 1), y' represents the predicted probability, and log denotes the natural logarithm.

G. Gradient Descent Optimization

Gradient descent is an iterative optimization algorithm used for updating the weights in a neural network during training. In order to train machine learning models, gradient descent is one of the most often utilized optimization techniques. It does this by reducing the discrepancy between the actual and expected outputs. Furthermore, neural networks are trained using gradient descent. The weight update rule for gradient descent can be expressed as:

$$W(t+1) = W(t)-learningrate*gradient \quad ....(6)$$

where W(t+1) represents the updated weight at time step t+1, W(t) represents the current weight at time step t, the learning rate denotes the hyperparameter that controls the step size, and the gradient represents the gradient of the loss function with respect to the weights.

IV. DATA COLLECTION AND PRE-PROCESSING

We collected a dataset of 345 car images along with their corresponding XML files containing license plate information. The car images were captured from various angles, lighting conditions, and weather conditions to ensure diversity in the dataset. The XML files provided the ground truth labels for the license plate regions in the images. The dataset was pre-processed using OpenCV, including resizing the images to a consistent size, normalizing pixel values, and augmenting the data with image flipping and rotation to enhance the training data.

V. TRAINING AND EVALUATION

We trained our ANPR model using the pre-processed dataset in TensorFlow. The dataset was acquired from the Kaggle website, consisting of a total of 433 images along with the XML file of each. The model architecture consists of a convolutional neural network (CNN) with multiple layers, including convolutional, pooling, and fully connected layers, along with activation functions such as ReLU. The model was trained using the backpropagation algorithm, which updates the weights in the neural network. It took around 6-7 hours to initially train the model with the 346 images (80% of the total dataset) of vehicles. The training process involves minimizing the loss function, typically the cross-entropy loss, through gradient descent optimization. The performance of the model was evaluated using cross-validation techniques, where the dataset was split into training and validation sets in an 80-20 ratio. The performance metrics used included accuracy, precision, recall, and F1-score, which were calculated based on the predicted license plate numbers compared to the ground truth labels in the XML files.

For the testing part, we acquired 86 images which was 20% of the total dataset, 433 images, we had. The cross-validation technique was conducted on the testing dataset in which we got 96% of detection accuracy and 90% of character accuracy. Detection accuracy is the accuracy of the model to successfully detect the position of the number plate. Character accuracy is the accuracy of the model to successfully extract the characters of the number plate.

VI. CONCLUSION

The technology we'll be applying, which comprises of character segmentation and recognition, can successfully be utilized to detect the number plate region from the image, according to our proposed method. The number plate region is comprised of the vehicle number and other characters. This technology, which we will be using on numerous photos, can be used to accurately identify the numbers on a vehicle's license plate. This project was created with the automation of the number plate recognition system for toll tax collection in mind.

ACKNOWLEDGMENT

The author acknowledges the support and informative feedback from Dr. Darshankumar Dalwadi, Professor at Birla Vishvakarma Mahavidyalaya (BVM).

REFERENCES

[1] Ganga Krishnan. G, "A Comparative Study of Four Deep Neural Networks for Automatic License Number Plate Recognition System," *International Journal of Engineering Research & Technology (IJERT,* vol. 11, no. 7, 2022.

[2] Jithmi Shashirangana, "Automated License Plate Recognition: A Survey on Methods and Techniques," *IEEE Access,* vol. 9, pp. 11203 - 11225, 2021.

[3] Z. Sanyuan, "Car plate character extraction under complicated environment," in *IEEE International Conference on Systems, Man and Cybernetics*, The Hague, Netherlands, 2004.

[4] L. Zou, "License Plate Detection with Shallow and Deep CNNs in Complex Environments," in *Complexity*, 2018.

[5] S. Du, M. Ibrahim, M. Shehata and W. Badawy, "Automatic License Plate Recognition (ALPR): A State-of-the-Art Review," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 23, no. 2, pp. 311 - 325, 2013.

[6] H. Ahmad Mushthofa, "The Improvement of Character Recognition on ANPR Algorithm using CNN Method with Efficient Grid Size Reduction," in *IEEE, 6th International Conference on Science and Technology (ICST)*, Yogyakarta, Indonesia, 2020.

[7] Pinaki Pratim Acharjya, "Study and Comparison of Different Edge Detectors for Image," *Global Journal of Computer Science and Technology,* vol. 12, no. 13, 2012.

[8] L. Luo, "An efficient method of license plate location," in *First International Conference on Information Science and Engineering*, Nanjing, China, 2009.

[9] Saqib Rasheed, "Automated Number Plate Recognition Using Hough Lines and Template Matching," in *World Congress on Engineering and Computer Science*, San Francisco, USA, 2012.

[10] X. Shi, W. Zhao and Y. Shen, "Automatic license plate recognition system based on color image processing," in *International Conference on Computational Science and Its Applications*, Springer: Singapore, 2005.

[11] R. Sferle and E. Moisi, "Automatic Number Plate Recognition for a Smart Service Auto.," in *15th International Conference on Engineering of Modern Electric Systems (EMES)*, Pusan, South Korea, 2019.

[12] K. Zimmermann, "Unconstrained licence plate and text localization and recognition," in *IEEE Intell. Transp. Syst.*, Vienna, Austria, 2005.

[13] F. Alegria and P. Girao, "Vehicle plate recognition for wireless traffic control and law enforcement system," in *IEEE International Conference on Industrial Technology*, Mumbai, India, 2006.

[14] H. Lim and Y. Tay, "Two-stage license plate detection using gentle Adaboost and SIFT-SVM," in *First Asian Conference on Intelligent Information and Database Systems*, Dong hoi, Vietnam, 2009.

[15] S. P. Kumar Parasuraman, "SVM Based License Plate Recognition System," in *IEEE International Conference on Computational Intelligence and Computing Research*, 2010.

[16] D. Shah, "Automatic Number Plate Recognition System (ANPR): A Survey," *International Journal of Computer Applications ,* vol. 69, p. 0975 – 8887, 2013.

[17] C. K. Sahu, S. B. Pattnayak, S. Behera and M. R. Mohanty, "A Comparative Analysis of Deep Learning Approach for Automatic Number Plate Recognition," in *Fourth International Conference on I-SMAC*, Palladam, India, 2020.

[18] F. Wu, "'Extracting characters from real vehicle licence plates out-of-doors," in *IET Comput. Vis*, 2007.