Real Time Bangla Number Plate Recognition using Computer Vision and Convolutional Neural Network

Md. Naimul Islam Suvon
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh
naimul.suvon@northsouth.edu

Riasat Khan
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh
riasat.khan@northsouth.edu

Mehebuba Ferdous
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh
mehebuba.ferdous@northsouth.edu

***Abstract*— Automatic number plate identification in today's world plays a vital role in vehicle tracking and organization. Our proposed model of automation in the detection and recognizing vehicles through the use of number plate computerization is expected to create a new scope of evolution for large cities. The system can be used for the parking system of motor vehicles, as well as to collect tolls. The detection of the Bangla number plates from different cities and multi-class vehicles is the first step of the proposed system. The number plate detection has been performed with the computer vision approach, and the You Only Look Once v3 (YOLOv3) algorithm. Next, the Tesseract optical character recognition system, in conjunction with the Bangla character recognition model, has been used for vehicle indexing and convolutional neural network for the character recognition from the detected number plate. Numerical results demonstrate that the accuracies of license plate detection for the computer vision and YOLOv3 are 91% and 95%, respectively. For the character recognition, the accuracy for Tesseract and convolutional neural network are 90% if the license plate is detected and cropped successfully and 91.38%, respectively. Finally, our system has been tested using the convolutional neural network method in an environment of real-world where our system's Pi Camera captured video as input, which has a total of 18 different cars. From 18 cars, it has successfully detected 17 cars, which makes our overall system accuracy 88.89%.**

***Keywords*— convolutional neural network, computer vision, license plate detection, morphological analysis, optical character recognition, tesseract.**

I. INTRODUCTION

Bangladesh is a country with a population of 163 million. In 2018, there were a total of 497,374,000 units of the registered motor vehicles in Bangladesh, which is still growing with proportional to the economic development. We need the right amount of human resources to monitor these vehicles, which is extremely expensive as well as time-consuming work. In this work, we have developed a system that can do the same job without any hassle using only the images of the vehicles. It will be easy for everyone to park in the parking lot as insignificant time will be needed to detect the license plate during parking entry and exit. The users of the proposed system don't have to suffer and wait for the next car to enter the garage as it will take a small time to process the data. Also, it will make a great impact on the 5th industrial revolution, where machine will do the maximum of the human work and all the systems will be free of human interaction. The utilization of this system can also be used for automated toll collection, analysis of city traffic conditions, and could also be used for tracking and identifying stolen vehicles. It can also be used to enforce different traffic rules in busy megacities, e.g., if there is a person who is violating any traffic rules (not wearing a helmet or seat belt) in the motor vehicle, then it can fine the person by detecting the number plate of the specific vehicle.

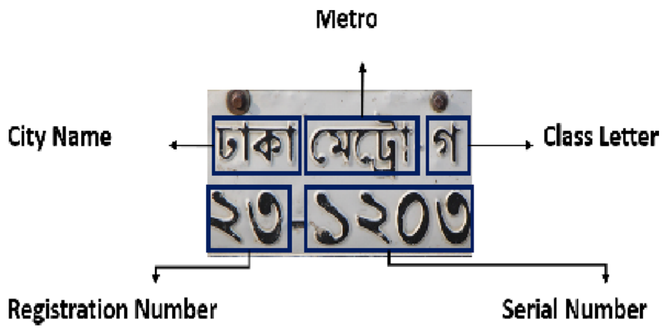

Fig. 1. License plate structure of Bangladesh.

All license plates in Bangladesh are written in a text format of two-line in the Bengali language. It also has different colours for various types of vehicles. In Fig. 1, name of the district (vehicle registration area) is in the first line, and the word "metro" denotes if the vehicle was registered in metropolitan territory. After the word "metro," there is a hyphen and a letter to show the category or variety of vehicles. There are six digits in the second-line or number line. The first two digits determine the vehicle's class, and the last four digits make up the unique vehicle registration number. The first two and the last four digits are separated by a hyphen.

In our work, an automated number plate detection and recognition system without any human interaction has been proposed. The digital number plates have been extracted and detected first with the mathematical morphological based computer vision approach and the you only look once (YOLOv3) algorithm. The YOLOv3 algorithm performed better than the former approach concerning accuracy and detection frame rate. Tesseract optical character recognition (OCR) and a trained convolutional neural network (CNN) model have been used for character recognition. The creativity of the work belongs to the application and study of multiple approaches on recognizing the number plate, as well as, using a diverse dataset containing images of cars from different cities and vehicle class.

The paper is organized as follows: In Section II, we discuss the related works, and Section III describes the proposed methodology of our paper in detail. The performance evaluation of our system is shown in Section IV. Finally, Section V concludes the paper with some directions in future research of the following work.

978-1-7281-6946-0/20/$31.00 ©2020 IEEE

II. RELATED WORK

In this section, relevant papers that are related to our work have been discussed.The authors used different neural network approaches for detection, segmentation, and recognition of the license plates. YOLO, as a detection algorithm and CNN for recognition of the license plates, were used by some of the works. The authors in [1] describe an automatic license plate recognition system (ALPR) system that is consistent and effective to detect the number plate using the YOLO-based object detector. Using data augmentative tricks, they have presented two-stage approaches, particularly for the registration number, serial number, and metro name segmentation and recognition. Abdullah *et al.* have worked with the YOLOv3 algorithm in [4], for completing the localization of the number plates and for recognizing registration and serial numbers. They have trained a ResNet-20-based deep CNN for recognizing the character. A Bengali ALPR system is proposed in [5], where the dictation, segmentation, and recognition of the license plates are performed with the machine learning approach and chain code. The main contribution of [6] is the recognition and detection of the individual states of the vehicle by implementing feature learning capacity of the CNN algorithm. Liu *et al.* in [3] proposed an unorthodox method with CNN, which is slightly different from the traditional approaches that function straight away on image pixels working as an alternative to the integration of segmentation and recognition. The license plate images have characters that are changed into encoded texts. In [2], an artificial neural network constructed from the OCR algorithm for an effective framework and its ALPR application are presented.

Character classification has been done by using the forward background feed method in [7]. The back-propagation is used to develop the neural network. In the preprocessing scale, normalization and edge detection are included to achieve higher accuracy. To address the problem of character recognition, component survey, in addition to horizontal and vertical graph is used.

Prathamesh Kulkarni used another method where through connected component and binarization characters areas are selected [8]. The unwanted points are removed using the analysis of point method and splitting the points and also combining the split points. A combined 97.2% accuracy of character recognition is achieved from this unit, and it has 90.9% reliability of the character detection. In this paper, the analysis of the related component bad processes is overviewed, such as counting the pixel of images analysis and the analysis of the aspect ratio [9].

Jissy Joseph worked with a comparison of four algorithms where statistical properties are used sequentially. The results with the application of contours algorithm and transformation of rough, different morphological approach and medium transformation approach have been reported in [10].

Watershed algorithm is used to fragment the handwritten text in [11]. In the preprocessing, the noise removal, budgeting, normalization of the images, and the slope correction were eliminated. The extraction of the character from the image is then done by reverse integer so that it can be converted to the integer wavelet. And finally, the classification is sorted using the neuroscience method.

III. PROPOSED METHODOLOGY

Our approach for number plate recognition is divided into three sections. First of all, we have detected the number plate from the image. Next, we cropped the detected number plate, and lastly, from the cropped number plate, we have recognized the characters and digits.

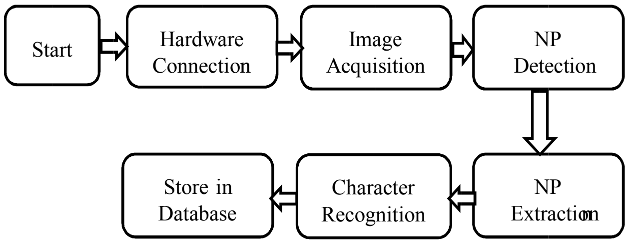

Fig. 2. Proposed license plate detection and recognition process.

As in Fig. 2, our system will run sequentially. It will take the captured image frame from the video using the Pi camera, and then will detect the area of the number plate and crop it. Subsequently, the Bangla Characters will be recognized and extracted from the cropped image and will be stored in a database.

A. Hardware System

The initial stage of our system proceeds with the design of the hardware components. First of all, we have used a Raspberry Pi 3 Model B single-board computer. Next, we have used a Raspberry Pi Camera Module v2 to capture the video. As we need 5Volts and 2Amperes current to power the Raspberry Pi 3B. There are two ways to power-up our system. The first way is a portable way by using Redmi mobile power bank, and the second way is using USB wire. As the Redmi power bank will power up the Raspberry Pi 3B maximum 2 hours, that's why we have also added a wired system.

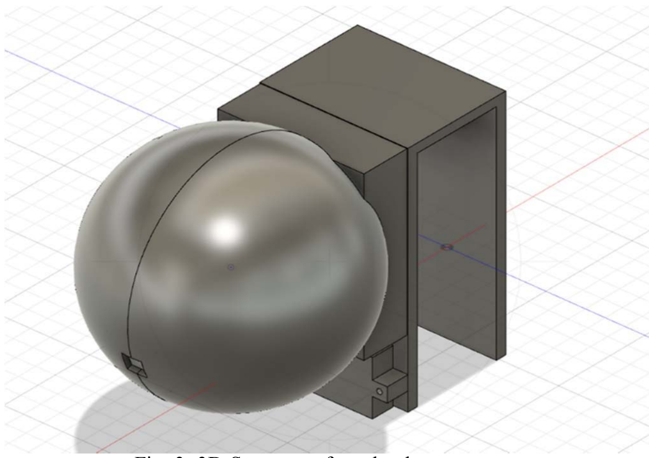

Fig. 3. 3D Structure of our hardware system.

In Fig. 3, we have shown the initial hardware structure of our project in 3D. It can be placed in the entrance and exit gate of the parking lot. From there, it will be capturing images and take the number plate information of every car. As we are in the primary stage of this work, we are interested in building the hardware in such a way so that it will co-up with the environment of the parking lot and fit in it properly.

B. License Plate Detection

1)Method 1: Computer Vision & Morphological Operation

To detect the character on cars' number plates first, we have to detect the number plates and then crop it for better character recognition.

First, the input from the user is an image of a motor vehicle that is shown in Fig. 4, which is an RGB image. RGB stands for red, green, and blue. RGB image is simply a composite of three independent grayscale images that correspond to the intensity of red, green, and blue lights.

Fig.4. Original image.

Fig.5. Grayscale image.

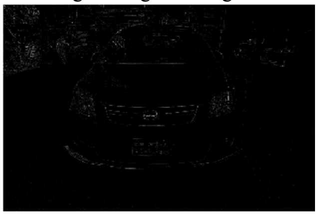

Fig.6. topHat image.

Fig.7. blackHat image.

Fig.8. Blur image.

Fig.9. Threshold image.

Fig.10. Contours image.

Fig.11. Possible character contours image.

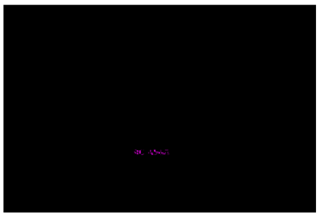

Fig.12. Final contours image.

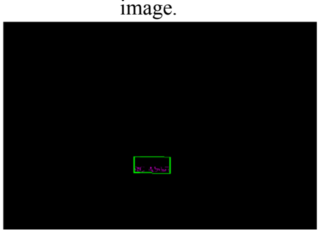

Fig.13. Detected image.

Fig.14. Detected original image.

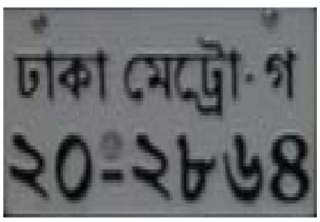

Fig.15. Cropped number plate image.

In Fig. 5, the image is then converted into a grayscale image. First, we have converted the RGB (Red, Green, and Blue) image into HSV (Hue, Saturation, and Value) image, where Hue represents the color type, and the vibrancy is represented by Saturation, which ranges from 0 to 255. The brightness of the color is represented by Value, which ranges from 0 to 255, where 255 represents brightness, and 0 represents darkness. Next, we have split Hue, Saturation, and Value. Then by taking this value we have converted the HSV image to grayscale image. Next, we have done some morphological operation using kernel. In Fig. 6 and Fig. 7, we have done top-hat and black-hat filtering. In a dark background, the object of interest, whose brightness is enhanced by using the top-hat filter. The black-hat operation is done by doing the opposite task. In Fig. 8, Gaussian blur has been performed. It is widely used to reduce image noise and reduce unwanted detail. In Fig. 9, thresholding has been done to minimize the background noise by extracting the useful information. In Fig. 10, we have joined all the continuous points, which are known as contours, and respectively, we have to find the possible characters and final contours in Fig. 11-12, using geometric filtering equations (1), (2), and (3). In Fig. 13, we have box plotted around the possible characters. In Fig. 14, we have successfully detected the license plate, and finally, in Fig. 15, we crop the region with the license plate. The cropping of the license plate is not done manually. Equations used in this process are:

$$F(x) = h * R + s * G + v * B \tag{1}$$

$$Distance(\imath) = \sqrt{|X_1 - X_2|^2 - |Y_1 - Y_2|^2} \tag{2}$$

$$Angel(\theta) = tan^{-1} \frac{|X_1 - X_2|}{|Y_1 - Y_2|} \tag{3}$$

By using equation equation(1), we have converted the RGB image to HSV image. Using equation(2), we have found the possible character from the contours image of Fig. 10 by finding the distance between characters and filter it. Next, we have found the angel between those possible characters centers by using equation(3). After doing all the described operations, we have selected the possible characters which give more than 90% accuracy.

2)Method 2: CNN

a) Number Plate Dataset: The license plates of Bangladesh are different from the license plates of other countries. The plates of Bangladesh have a 2-line format compared to the 1-line format of other countries. On top of that, the Bengali characters and their complex lexical structure forced us not to use any public dataset of the license plates from any countries other than Bangladesh. Initially, approximately 3,000 pictures were captured by us. A camera of 12-megapixel was used to have good quality images. Different places of Dhaka city like, outdoor parking, indoor parking, and running vehicles from the road, were used to capture images. The images were captured in different environments to standard the images. Only clearly visible images of number plates were used as primary data to ensure accurate training of our dataset. Finally, we have taken only 2,000 images from our captured images. In. Fig. 16 (a) and (b) are some of our sample original images are showed.

(a) (b)

Fig. 16. (a, b) Running vehicle images captured from the road.

b) Number Plate Detection: In this work, YOLOv3 algorithm [12] has been used to identify the license plate. The newest version of YOLO (You Only Learn Once) algorithm, YOLOv3 has several design changes that make it a better approach. To do object detection in real time, YOLO is a smart convolutional neural network (CNN). Because YOLOv3 uses multi-scale prediction, it has better performance than its older versions, which is not that good for detecting small objects.

Initially, we started with 1,500 images, and some of the images that were not up to standard were removed after data cleaning. A tool named “labelimg” was used for marking bounding boxes. The YOLOv3 was customized before training to make it more suitable in detecting Bengali license plates. The batch size was set to 64, and for faster training, the subdivisions were set to 16. The filter size was equal to 18, as we had only one class in the license detection phase. For training, the initial weights of the pre-trained weights of the darknet-53 convolutional layers were used. The model was iterated 200 times.

C. *Character Recognation*

1)Method 1: Tesseract OCR

After detecting the license plate and cropped it, we have used a simple method to detect the Bangla character. It is a complex task to detect the Bangla letters and numbers. Most of the paper detects the license plate using CNN, but we have used both OCR and CNN method.

Firstly, In Method 1, We have used Tesseract optical character recognition (OCR) for the Bangla character recognition part. Firstly, we have prepared a dataset of Bangla character text. Next, the text has been rendered to an image and then to a box file. After that, we have made a unicharset file for each unichar or symbol and made a starter trained data from the unicharset data. Then the Tesseract OCR engine is executed to make a complete training dataset from the processed image and box files. The Long-Short Term Memory (LSTM) training program, a multi-purpose tool for training neural networks, has been used for final training of the dataset. Lastly, we have fed the trained model into the Tesseract OCR. The unique feature about Tesseract OCR training is that the detection of the right character can be seen using Textbox Editor. If the character detection is not correct, it can be changed manually using Textbox Editor.

2)Method 2: CNN

a) Character Dataset: After detecting the number plate and cropped it from the original image, we sliced the words “DHAKA”, “CHOTRO”, “METRO”, characters “KHA”, “GA”, “GHA”, “CHA”, “VOH” and digits from 0 to 9 manually. This how we prepared our dataset for training our model.In Fig. 17 and Fig. 18 show the cropped characters for our model database. It is evident from the sample pictures that exceptionally challenging images contains in the database with various scales, interpretations, shearing, etc.

Fig. 17. Words and characters images from our character dataset.

Fig. 18. Digits 0-9 images from our digit dataset.

Consequently, this turns into a challenging job to accomplish better accuracy for Bangla LPB. To achieve better accuracy, we have used the CNN model for preparing and testing.

b) CNN network parameters and architecture: A model with six different layers, including two layers for convolution, two layers for subsampling or pooling, one fully connected layer, and one final classification layer were used in this experiment. In the pooling layer, the number of feature maps does not change. Sixteen filters are used, which produces 16 feature maps from the second convolutional layer, in the second convolutional layer. Then the second maxpooling layer is applied. A fully connected layer with 400 neurons is used, followed by a final classification layer with 18 neurons for 18 different classes. In TABLE I. , it is shown that the number of parameters of our model is calculated for 32 x 32 input image.

TABLE I. DIFFERENT CNN LAYERS AND THEIR PARAMETERS

Layer	**Layer of operation**	**Number of feature maps**	**Feature maps size**	**Window size**	**Parameters**
C1	Convolution	6	28X28	5X5	156
S1	MaxPooling	6	14X14	2X2	0
C2	Convolution	16	10X10	5X5	2,496
S2	MaxPooling	16	5X5	2X2	0
F1	Fully Connected	400	1X1	N/A	166,400
F2	Fully Connected	18	1X1	N/A	7,218

For the first convolutional layer, the output feature maps size is 28×28 for 6 feature maps. The filter mask size is 5×5 for both of the convolutional layers. The number of network parameters used to learn is $(5 \times 5 + 1) \times 6 = 156$, and the one is added with filter dimension because of bias. The number of trainable parameters of the subsampling layer is 0. However, the size of the feature maps reduced with factor 2 due to a 2×2 pooling mask; therefore the size of the outputs of the first subsampling layer is 14 × 14 with 6 feature maps. The second convolutional layer's output dimension is (14−5) + 1 = 10, 10×10 with 16 feature maps are used and the second sub-sampling filter mask size is 2×2. The learning parameters for second convolution layer are ((5×5+1) ×6) ×16 = 2,496. The zero number of network parameters exist in the second sub-sampling layer. The total number of features per sample from the second sub-sampling layer is 5×5×16 = 400. Since we have used 400 hidden neurons for this layer, therefore, the number of parameters for the first fully connected layer is 400×16× (5×5+1) = 166,400, whereas the amount of the final layer parameter is: 18 × (400+1) = 7,218. The total number of parameters is 1,76,270 for this model.

IV. Performance Evaluation

We have analyzed the performance of our system in two steps. Firstly, we have analyzed the performance of the number plate detection. Secondly, we have analyzed the performance of words, characters, and digits recognition. Finally, we have also analyzed the performance of the entire system.

A. Performance of number plate detection

We have used computer vision and morphological operation on images to localize the number plate, which is the first step of our proposed system. As we did not require any training data for the detection process, it reduces the computational time. We do not need any training for this method, we have tested this method using 100 images of car with license plate, and it has successfully detected 91 of them, so the accuracy for this step was 91%. But there are some limitations in this method, e.g., if the distance of the car is too far from the camera and there are a lot of irrelevant stuffs in the captured images, then the detection accuracy deteriorates.

To measure the efficacy of the proposed ALPR system for CNN, we performed some experiments on our collected number plate dataset. The YOLOv3 was trained with 80% of our dataset images for 2,00 iterations. The moving average loss was found to be 0.09 in the validation with IoU more than 85%. The license plate identified with IoU more than 50% is considered to be a correct detection instance according to many researchers. Our proposed method from that perspective outperforms other state-of-the-art methods with more than 95% total detection accuracy.

B. Performance of Character Recognation

For Character recognition, we have also applied two methods, firstly using Tesseract OCR and secondly using convolutional neural network classifier.

Tesseract is an optical character recognition engine for different working frameworks. The model that we created manually using LSTM training for character recognition has a testing accuracy of 90%. So, if the license plate has been localized from the input image perfectly, then its accuracy is 90%. Convolution Neural Network has an amazing learning capacity with fewer epochs. In Fig. 19, character recognition accuracy during the training and test phase are showed with each epoch using CNN classifier.

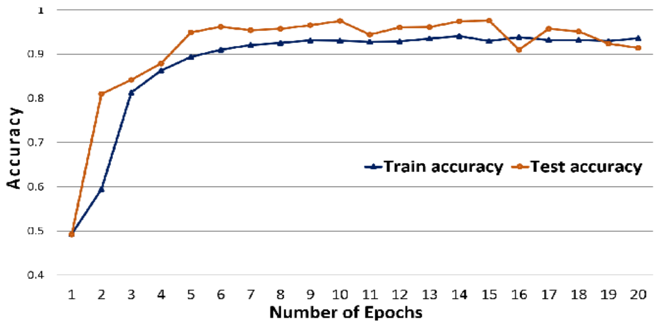

Fig. 19: Train vs testing accuracy.

In Fig. 19 shows that, we have accomplished 93.54% accuracy on the training set and 91.38% accuracy on the test set after 20 epochs.

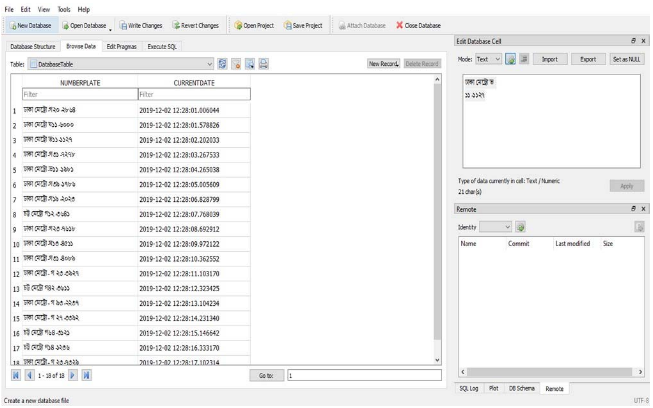

Fig. 20. Database of recognized number plates.

In Fig. 20, we have shown the database of the Recognized number plate by running our system.

TABLE II. Recall, precision and F-measue for character recognition

Class	Precision	Recall	F1-Score
Class 0 (DHAKA)	0.85	0.82	0.86
Class 1 (METRO)	0.88	0.90	0.86
Class 2 (CHOTRO)	0.96	0.89	0.93
Class 3 (KHA)	0.87	0.95	0.91
Class 4 (GA)	1.00	0.92	0.96
Class 5 (GHA)	0.30	0.85	0.44
Class 6 (CHA)	0.98	0.90	0.94
Class 7 (VOH)	0.55	0.89	0.78
Class 8 (0)	0.92	0.89	0.87
Class 9 (1)	0.89	0.86	0.91
Class 10 (2)	0.94	0.90	0.89
Class 11 (3)	0.95	0.89	0.94
Class 12 (4)	0.90	0.92	0.93
Class 13 (5)	0.96	0.87	0.92
Class 14 (6)	0.91	0.90	0.87
Class 15 (7)	0.88	0.83	0.90
Class 16 (8)	0.91	0.88	0.93
Class 17 (9)	0.92	0.86	0.90

Finally, In TABLE II. the recall, precision and F1-Score results are given for character recognition.

As we can see, the CNN method gives a better result. Our system has been tested using the CNN method in an environment of real-world where our systems Pi Camera captured video as input with a total of 18 different cars. From 18 cars, it has successfully detected 17 cars, which makes our overall system accuracy by 88.89%.

V. Conclusion

Our prototype of the designed system, according to our model, was fully capable of vehicle number plate detection and processed the images, without the need for any human interactions. This could solve not only the issue of car parking crisis in Bangladesh but also open a new scope of the market for business-oriented car parking rental system. This could be a solution for megacities around the world, which are facing the tremendous burden of modernization and overpopulation. Not only could it solve the parking crisis, but also it could be a solution to track and locate any stolen vehicles, which is a concerning issue in modern cities. Both the computer vision and CNN based YOLOv3 algorithms have been used for number plate detection. We have achieved 91% accuracy for computer vision, where it has limitations on distance and irrelevant colors. On the other hand, with the YOLOv3 algorithm, we have got 95% accuracy. The YOLOv3 outperforms the computer vision method in terms of accuracy. On the other hand, for the character recognition, CNN outperforms tesseract OCR with 91.38% accuracy. Tesseract OCR has 90% accuracy, which is satisfactory. Though this method is simple, there is a limitation; i.e., the number plate should be detected and cropped perfectly. The authors are working on implementing the proposed model to design an app-based intelligent parking management system with a 3D view of the parking garage.

References

[1] R. Laroca *et al*., "A robust real-time automatic license plate recognition based on the YOLO detector," *2018 International Joint Conference on Neural Networks (IJCNN)*, Rio de Janeiro, 2018, pp. 1-10, doi: 10.1109/IJCNN.2018.8489629.

[2] X. Zhai, F. Bensaali and R. Sotudeh, "Real-time optical character recognition on field programmable gate array for automatic number plate recognition system," in *IET Circuits, Devices & Systems*, vol. 7, no. 6, pp. 337-344, November 2013, doi: 10.1049/iet-cds.2012.0339.

[3] J. Liu, X. Li, H. Zhang, C. Liu, L. Dou and L. Ju, "An implementation of number plate recognition without segmentation using convolutional neural network," *2017 IEEE 19th International Conference on High Performance Computing and Communications; IEEE 15th International Conference on Smart City; IEEE 3rd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, Bangkok, 2017, pp. 246-253, doi: 10.1109/HPCC-SmartCity-DSS.2017.32.

[4] S. Abdullah, M. Mahedi Hasan and S. Muhammad Saiful Islam, "YOLO-based three-stage network for Bangla license plate recognition in Dhaka metropolitan city," *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*, Sylhet, 2018, pp. 1-6, doi: 10.1109/ICBSLP.2018.8554668.

[5] A. K. Ghosh, S. K. Sharma, M. N. Islam, S. Biswas, and S. Akter, "Automatic license plate recognition (ALPR) for Bangladeshi vehicles," *Global Journal of Computer Science and Technology*, vol. 11, pp. 69-73,2011.

[6] M. Mondal, P. Mondal, N. Saha and P. Chattopadhyay, "Automatic number plate recognition using CNN based self synthesized feature learning," *2017 IEEE Calcutta Conference (CALCON)*, Kolkata, 2017, pp. 378-381, doi: 10.1109/CALCON.2017.8280759.

[7] N. Abirami and, J. S. L. Jasmine, "Accurate vehicle number plate recognition and real time identification using raspberry pi," *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, pp. 2563–2569, 2018.

[8] P. Kulkarni, A. Khatri, P. Banga and K. Shah, "Automatic number plate recognition (ANPR) system for Indian conditions," *2009 19th International Conference Radioelektronika*, Bratislava, 2009, pp. 111-114, doi: 10.1109/RADIOELEK.2009.5158763.

[9] N. Vishwanath, S. Somasundaram, M. R. R. Ravi and N. K. Nallaperumal, "Connected component analysis for Indian license plate infra-red and color image character segmentation," *2012 IEEE International Conference on Computational Intelligence and Computing Research*, Coimbatore, 2012, pp. 1-4, doi: 10.1109/ICCIC.2012.6510323.

[10] J. Joseph, A. Prasad, T. S. Jithina and L. Mary, "A Study on Localization Techniques for Automatic License Plate Recognition System," *2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, Kannur,Kerala, India, 2019, pp. 1090-1094, doi: 10.1109/ICICICT46008.2019.8993319.

[11] P. Mathivanan, B. Ganesamoorthy and P. Maran, "Watershed algorithm based segmentation for handwritten text identification", *ICTACT Journal on Image And Video Processing*, vol. 4, no. 3, pp. 767–772, 2014, doi: 10.21917/ijivp.2014.0111.

[12] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: unified, real-time object detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[13] R. A. Baten, Z. Omair and U. Sikder, "Bangla license plate reader for metropolitan cities of Bangladesh using template matching," *8th International Conference on Electrical and Computer Engineering*, Dhaka, 2014, pp. 776-779, doi: 10.1109/ICECE.2014.7026925.

[14] J. Pyo, J. Bang and Y. Jeong, "Front collision warning based on vehicle detection using CNN," *2016 International SoC Design Conference (ISOCC)*, Jeju, 2016, pp. 163-164, doi: 10.1109/ISOCC.2016.7799842.

[15] Ching-Tang Hsieh, Yu-Shan Juan and Kuo-Ming Hung, "Multiple license plate detection for complex background," *19th International Conference on Advanced Information Networking and Applications (AINA'05) Volume 1 (AINA papers)*, Taipei, Taiwan, 2005, pp. 389-392 vol.2, doi: 10.1109/AINA.2005.257.

[16] A. Menon and B. Omman, "Detection and recognition of multiple license plate from still images," *2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET)*, Kottayam, India, 2018, pp. 1-5, doi: 10.1109/ICCSDET.2018.8821138.

[17] S. Huang, H. Xu, X. Xia and Y. Zhang, "End-to-End vessel plate number detection and recognition using deep convolutional neural network and LSTMs," *2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, Hangzhou, China, 2018, pp. 195-199, doi: 10.1109/ISCID.2018.00051.

[18] S. Lakshmi S., B. Sathiyabhama, T. K. Revathi, N. Basker and R. B. Vinothkumar, "Tracing of vehicle region and number plate detection using deep learning," *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, Vellore, India, 2020, pp. 1-4, doi: 10.1109/ic-ETITE47903.2020.357.

[19] S. Babbar, S. Kesarwani, N. Dewan, K. Shangle and S. Patel, "A new approach for vehicle number plate detection," *2018 Eleventh International Conference on Contemporary Computing (IC3)*, Noida, 2018, pp. 1-6, doi: 10.1109/IC3.2018.8530600.

[20] M. M. Shaifur Rahman, M. Mostakim, M. S. Nasrin and M. Z. Alom, "Bangla license plate recognition using convolutional neural networks (CNN)," *2019 22nd International Conference on Computer and Information Technology (ICCIT)*, Dhaka, Bangladesh, 2019, pp. 1-6, doi: 10.1109/ICCIT48885.2019.9038597.

[21] M. Atikuzzaman, M. Asaduzzaman and M. Z. Islam, "Vehicle number plate detection and categorization using CNNs," *2019 International Conference on Sustainable Technologies for Industry 4.0 (STI)*, Dhaka, Bangladesh, 2019, pp. 1-5, doi: 10.1109/STI47673.2019.9068049.