Automatic License Plate Recognition System for Bangla License Plates using Convolutional Neural Network

Conference Paper · December 2019

DOI: 10.1109/TENCON.2019.8929280

7 authors, including:

Nazmus Saif
United International University

Sayem Pasha
United International University

Md. Saif Khan Shahrin
United International University

Mahmudul Hasan
United International University

Automatic License Plate Recognition System for Bangla License Plates using Convolutional Neural Network

Nazmus Saif, Nazir Ahmmed, Sayem Pasha, Md. Saif Khan Shahrin,
Md. Mahmudul Hasan, Salekul Islam and Abu Shafin Mohammad Mahdee Jameel
Department of Computer Science and Engineering
United International University
United City, Madani Avenue, Badda, Dhaka 1212, Bangladesh
Email: nazmussaif@gmail.com, nahmmed151014@bscse.uiu.ac.bd, zun.sayem@gmail.com, saifkhan29797@gmail.com,
mhasan152185@bscse.uiu.ac.bd, salekul@cse.uiu.ac.bd, mahdee@cse.uiu.ac.bd

***Abstract*—We present an automatic license plate recognition system that can detect and recognize Bangla license plates from an image of vehicles. It is mandatory for vehicles in Bangladesh to have Bangladesh Road Transport Authority (BRTA) standard license plates attached in front and back of the vehicle. We build a database containing vehicles of these type. From this dataset, detecting the license plates in an input image is the first step of our system. We use a convolutional neural network (CNN) model for this. From that, we take the detected license plate image as a new input for the second and similar CNN model to segment and recognize license plate numbers. We tested our model on 200 images and achieved an accuracy of 99.5%. For speed, we tested our model on Google Colaboratory's Nvidia Tesla K80 GPU and attained a speed of 9 frames per second while detecting and recognizing the license plate numbers in a video.**

***Index Terms*—Automatic License Plate Recognition (ALPR), Convolutional Neural Networks (CNN), Optical Character Recognition (OCR), Character Segmentation, Bangla Licence Plate, Object Detection**

I. Introduction

Automatic License Plate Recognition System for Bangla License Plates using Convolutional Neural Network automatically (without manual interaction) detects a Bangla license plate in an image and outputs the license plate numbers in the same sequential order it has appeared in the original plate. Although automatic license plate recognition (ALPR) systems have been used for many useful purposes for a long time throughout the world, to best of our knowledge, it is not implemented successfully in Bangladesh yet. The use of the automated parking management system is increasingly becoming popular in Bangladesh. As an effect to this, a significant number of clients are now interested in the ALPR system. It is an integral part of the automated parking system. By market demand, it will be an expected feature for most of the commercial parking management projects throughout the metropolitan areas of the country. Apart from parking management systems, ALPR solution can be used in toll collection systems, for security purposes, for traffic control, even in gas stations and there are many more opportunities to explore. There are several international companies who provide ALPR solutions, but as these systems are language and license plate design specific, no support is available for Bangla license plates from these companies.

Since Bangla language itself has a complex lexical structure, detection, based on some handcrafted image features is not enough. Also, in [1] Gonalves et al. showed that even with various combination of different methods and thresholds, image processing techniques do not scale up very well for license plate segmentation on a large dataset. Therefore, instead of traditional image processing, we incorporate You Only Look Once (YOLO) [2], a convolutional neural network (CNN) based object detection algorithm. It has surpassed the accuracy measure of any ALPR system on Bangla language we have tried. The exciting part of our proposed method is that it doesn't require any controlled condition or environment setup. A decent clear image from a human-readable distance is enough to detect the number plate with our model.

The advantages of our model are accuracy, speed, and generalization. YOLO is one of the 'state of the art' algorithms for real time object detection and classification. Using YOLO as our CNN model, we achieved accuracy up to 99.5% in our dataset. Our model can also detect and recognize multiple license plates in a single image or video without any complication, although speed may decrease for multiple plates recognition. One of the best thing about YOLO model is that it generalizes very well, and thus our model scales well with the type of images it has never been trained on.

While working on the project, one of the main hurdles we face at the beginning is that there is no publicly available dataset for Bangla license plate. Also, unlike the most popular style of one line license plate format throughout the world, Bangla license plate is composed of two lines. For that reason, using a publicly available database of the license plate of different countries to extract features or to learn detecting license plate was not a feasible idea either. However, as we are using the CNN model for our system, and CNN is a data-dependent process, we prepare a dataset of our own counting multi-class vehicles for our system. To reduce the distribution of imbalanced data, we also fabricated synthetic data, which will be farther discussed in the V-A section.

The rest of the paper is organized as follows: section II explains the format of the Bangladesh Road Transport Authority (BRTA) standard number plates, section III summarizes the existing work, section IV presents the proposed method by explaining the number plate detection, segmentation and character recognition, section V presents the detailed results received through experimental analysis, section VI briefly discusses the limitations of our model and finally we conclude the paper.

II. Overview of the BRTA Standard License Plates

The Bangladesh Road Transport Authority (BRTA) is the only authoritative body in Bangladesh to issue various types of license plates for vehicles. The current version of the BRTA license plates started in 1973 but the digitization of this kind of license plates stared in 2012. Now it is a requirement for the vehicles to have digital license plates on both the front and rear sides of a vehicle, where the rear license plate is permanently attached. Figure 1 refers to the classes of license plates in Bangladesh.

Vehicle Category	Plate Color	Character Color	Example
Private	White	Black	
Commercial	Green	Black	

Fig. 1. Different classes of license plate

In the BRTA guideline, Bengali alphabet and Bengali numerals are used in the vehicle license plates. BRTA format of the digital license plates is divided into two lines. The first line presents the "city - vehicle class letter" while the second line shows the "class number - vehicle number". Figure 2 is an example of the BRTA standard license plate.

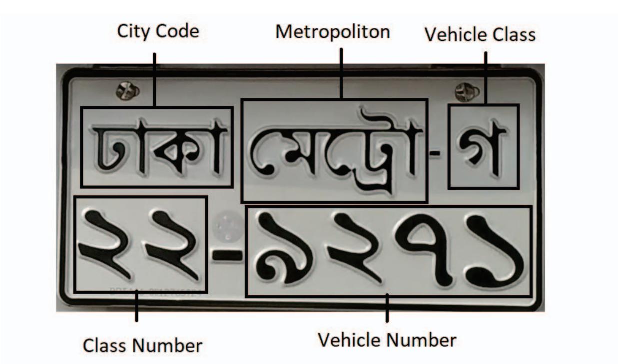

Fig. 2. BRTA standard license plate

III. Literature Review

As there exist many versatile applications for ALPR systems, many techniques have been developed around this field. We will present an overview of the popular techniques for ALPR systems used internationally and for our country as well.

A. Global Practice for Number Plate Detection

Anagnostopoulos et al. [3] proposed an algorithm that can detect license plate of various sizes and positions in the image. Also, their method is able to detect and segment more than one license plates from the same image. Their algorithm is based on 'novel adaptive image segmentation technique (sliding concentric windows)' and connected component analysis for the detection and two-layer probabilistic neural network (PNN) for recognition of the characters. They achieved 96.5% accuracy in detection and 89.1% accuracy in recognition.

Villegas et al. [4] used three layers of fuzzy neural network for detecting and recognizing the license plate. In this method, rectangular perimeter detection and pattern matching were used for detection of the license plate. Horizontal and vertical projections were used to segment out the characters. Finally, a fuzzy neural network was used to recognize the license plate. They attained up to 80% accuracy in detection and 87.22% accuracy in recognition of the license plate.

Du et al. [5] reviewed different techniques of automatic license plate recognition. They have categorized different techniques according to the features used in each stage. They categorized the different algorithms according to advantages, disadvantages, recognition results and processing speed. They have also provided a guideline for how future work on ALPR should be done, what issues should be addressed, etc.

Patel et al. [6] also carried out a review of existing ALPR solutions and categorized them according to parameters including accuracy, performance, speed, image size, etc. They also provided a comprehensive study of the recent development of ALPR and future trends so that researchers who want to work on this field can have an idea.

B. Bangla License Plates Recognition

Neural Network Based Approach: Ghosh et al. [7] worked with a combination of image processing and neural network method. They segmented their work into three stages: license plate extraction, character segmentation, and recognition. They used Sobel filter [13] to extract the license plate. Morphological operations also had been used to aim to remove the unrelated objects. In the segmentation part, they used connected component analysis. In the classification and recognition part, they used a feed-forward neural network. Their result showed 84% accuracy for license plate extraction and 80% accuracy for character recognition.

Joarder et al. [8] proposed to use edge detection of the license plate using Sobel operator [13] and match filter [14] to remove most of the noisy part in the image. They used Gaussian kernel to smooth it. Horizontal and vertical projection analyses were used to detect the character in the plate.

TABLE I
SUMMARY OF THE REVIEWED LITERATURE ABOUT BANGLA LICENSE PLATES DETECTION AND RECOGNITION

Name	Detection	Segmentation	Recognition
Automatic License Plate Recognition (ALPR) for Bangladeshi Vehicles [7]	Sobel edge detection with additional morphological operations	Line segmentation, word segmentation based on area filtering	Feed forward neural network
Bangla Automatic Number Plate Recognition System using Artificial Neural Network [8]	Edge analysis	Horizontal and vertical projection analysis	Multilayer feed-forward network
Bangla License Plate Reader for Metropolitan Cities of Bangladesh Using Template Matching [9]	Connected component technique		Template matching
Line Segmentation and Orientation Algorithm for Automatic Bengali License Plate Localization and Recognition [10]	Sobel edge detector	Line segment orientation (LSO) algorithm	Template matching
Bangla License Plate Recognition Using Convolutional Neural Networks (CNN) [11]	CNN	CNN	CNN
A System Design for License Plate Recognition by Using Edge Detection and Convolution Neural Network [12]	Segmenting edges using Prewitt operators	Area filtering	Deep CNN

For the Bangla character recognition part, they implemented a multilayer feed-forward network (MLP Neural Network). The success rate of number plate detection for their method was 92.1% which took 1.3 seconds. Recognition was 84.16% accurate which took 1.3 seconds as well.

Baten et al. [9] proposed a method that uses a special feature in Bangla language called "matra" to segment out the characters from the license plate using connected component analysis. After that, they used template matching of words for recognizing the Bangla characters and numbers. They showed using "matra" to segment the characters decreases the complexity of the algorithm. They also claimed their algorithm takes 1.3 seconds to recognize the license plate.

Haque et al. [10] used image processing and genetic algorithms to detect and recognize the license plate in an image. In their approach, first, they resized their image by 800 x 1200 and cropped it. After that, they converted the picture into a gray-scale image and used Sobel filter [13] to detect the edges. Using image morphology and connected component analysis, they localized the license plate. After that, they used Otsu's method [15] to binarize the image of the license plate. They again used connected component analysis to segment out the words, characters, and numbers into 32 x 48 pixels images. Then, they used the line segment orientation (LSO) algorithm to recognize the Bangla characters in the image, and they achieved 95.8% accuracy in localization and 84.87% accuracy in recognition.

Image Processing Approach: Rahman et al. [11] proposed a Convolutional Neural Network (CNN) based approach to recognize the license plate number in Bangla license plate bearing vehicles. They used 1750 number of samples, where they sliced digits and characters of the license plate into 32 pixels by 32 pixels each. In their CNN, they used one input layer, two convolution layers, two subsampling layers and a final layer for classification. This way they achieved 88.67% testing accuracy.

Dhar et al. [12] used Prewitt operators [16] to segment edges, morphological dilation for enhancing the edges and distance to border vectors (DtBs) technique to filter noises. For character segmentation, they used region properties and morphological operation. For the recognition part, they collected still images of available cars, but they also created some pictures syntactically, by which they created 1400 images of 14 classes. In their Convolution Neural Network (CNN), they gave 30 x 30 pixel images of every character as input. Also, they used RELU [17] as activation function, and Max-Pooling was used to combat overfitting.

Table I shows a summary of the reviewed literature about Bangla license plates detection and recognition.

IV. PROPOSED METHODOLOGY

We chose to implement the convolutional neural network as our preferred model for the end-to-end pipeline for our system. For our case, CNN simply outperformed traditional image processing algorithms and CNN models comparatively generalized better in different scenarios. The main advantage we had for using CNN is that it learns to extract the features automatically which gave us independence from prior knowledge about different features and characteristics. Where in the case of traditional algorithms, features need to be hand-engineered. But in order to automate the process, CNN needs to train on a lot of data/images. For that, we prepared our own dataset, discussed in section V-A.

On completion of the literature review in section III, we find that the main problem with detecting or recognizing Bangla license plate is the lack of an enriched public dataset. It is very hard to compare these methods with each other as there is no proper dataset to make a good benchmark. Therefore, we had to experiment with these methods to our best ability to get close to the original work with our dataset. With that experience, we propose a new method for detection and recognition of Bangla license plate. Our pipeline consists of capturing the image and pre-process it to 416 x 416 pixels, then feed the image to our first CNN model to find the license plate in it. If the license plate is found, then the license plate is cropped from the input image. After that, word, character and number segmentation is completed with the second CNN model and from that, we finally recognize the text in the plate. Figure 3 shows the diagram of our proposed workflow.

A. Number Plate Detection

Detecting the license plate from the input image is the first challenge of our problem. For this, we used YOLOv3 [18]

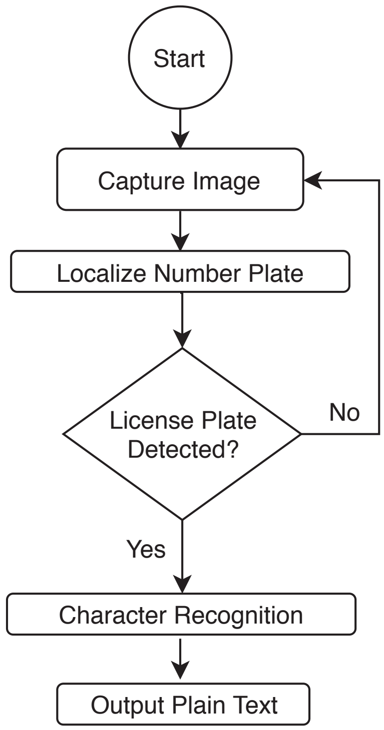

Fig. 3. Diagram of the proposed workflow

as our preferred convolutional neural network model. The reason for choosing YOLO is that it offers several benefits for our system over some other CNN models in the literature. YOLO is a real-time object detection model, which is accurate enough for detecting and localizing license plate images. We have used several image processing techniques for this same problem, while worked for some images, those techniques did not scale for the whole dataset, and was nowhere as accurate as YOLO for this task. Another benefit of using YOLO is that it generalizes very well, thus perform extraordinarily well on data that it has never been trained on. YOLO also can filter background noises from the actual data very well. For these reasons, we chose to implement YOLO in our work.

Our network consists of 53 convolutional layers based on YOLOv3 model. It uses the leaky rectified linear unit as its activation function, which is defined as:

$$\phi(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.1x, & \text{otherwise} \end{cases} \tag{1}$$

YOLO uses the sum squared error as its loss function, where it has to minimize for 5 different parameters: x, y, w, h, and C_i. The x and y are coordinates of a point that represents the center of the object that is to be detected. The w and h are the width and height of the object, is predicted relative to the entire image. Finally, the confidence C_i is the intersection over union (IOU) between the predicted box and the ground truth box. Also, if any object is found, $p_i(c)$ is the class probability for that object.

Our network uses the same loss function from the YOLO [2] model, which is defined as:

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{\text{ij}}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right]$$

$$+ \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{\text{ij}}^{\text{obj}} \left[(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2 \right]$$

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{\text{ij}}^{\text{obj}} (C_i - \hat{C}_i)^2$$

$$+ \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{\text{ij}}^{\text{noobj}} (C_i - \hat{C}_i)^2$$

$$+ \sum_{i=0}^{S^2} 1_{\text{i}}^{\text{obj}} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2 \tag{2}$$

Here: $\lambda_{coord} = 5$ and $\lambda_{noobj} = 0.5$ are two constant that increases the loss from bounding box that contains objects and decreases the loss from the boxes that does not contain any objects to balance out the effect. Here, $\hat{x}_i$, $\hat{y}_i$, $\hat{w}_i$, $\hat{h}_i$, $\hat{C}_i$, $\hat{p}_i(c)$ are ground truth parameters. Details of the equation can be found in [2].

B. Segmentation and Character Recognition

One of the advantages of using YOLO is that it can detect and localize objects in an image simultaneously using only one convolutional network. Using this property, we can segment out the license plate's words and characters and classify them at the same time if a license plate's image is given to the network. It significantly reduces segmentation and recognition time as well. For that reason, we used the same type of YOLO model we used in the detection process and used same hyper-parameters at the time of training.

After detecting the license plate in the original image, it will crop out the license plate area and send it to the second YOLO model for segmenting and recognizing the license plate in the image.

Redmon Et al. stated in their YOLO (version one) paper [2] that, YOLO "struggles with small objects that appear in groups". For that reason, using only one model did not produce a good result for detecting and recognizing license plate numbers in our case. The resolution of the image was too low if the entire image with vehicles and other objects is given as the input. License plate numbers essentially behaved as small objects in a group. For that, we first detect the license plate in the image using one model and then use the second model with the cropped image of the license plate. This way, the second model gets a better resolution image for recognizing the license plate number in the image, thus reduce the problem of small objects that appear in a group.

V. Experimental Analysis

A. Dataset

For our dataset, we have used 12-megapixel cameras to capture the image. The images were in color mode with standard Joint Photographic Experts Group (JPEG) format. Our dataset contains 1050 images for training and 200 more images of private vehicles for testing. Every license plate has two word and a letter in the first line and 6 numbers in the second line, resulting in 9450 words and characters (alphanumeric symbols) that have been manually annotated with bounding boxes.

Our dataset has also been manually augmented to avoid overfitting. For that, we randomly translated and scaled up to 20% of the captured image sizes. We also changed the saturation and the exposure of the original images randomly by 30% in the HSV color space. We also added some random noise in the images to make it more robust.

B. Training

Our model has 53 convolutional layers. For detecting the license plate, we trained our model for 128 epochs. For segmenting and recognizing the license plate, we trained our model for 80 epochs.

In the time of training, the hyperparameters for our model ware similar to the original YOLO model, which is given below:

- Batch size = 16
- Momentum = 0.9
- Decay = 0.0005
- Learning rate = We started with 10^{-4}, slowly decreased when validation accuracy did not improve in a few previous epochs.

C. Accuracy Measurement

For detecting the license plate in the image, we measured the accuracy by measuring if it successfully localized the license plate in a way that all the numbers, word and characters are recognizable in the localization area. For the segmentation and recognition part, we measured accuracy in a binary fashion. If each and every characters and numbers in the license plate are correctly classified and appear in the same order as it is placed in the plate, only then we considered it as accurate. Otherwise, even if one single character is misclassified, we considered the whole license plate as misclassified.

D. Five Fold Cross Validation

For our system, recognition of the license plate is the crucial part. Because in our experiment, our model did produce successful results on detecting the license plate in the image. Even if it could not detect the whole license plate, it always gave a general localized area of it in the image. We then could use a threshold padding value to get the full license plate from the image. But that is not the case for segmentation and recognition part of our model. If a single character is misclassified, the whole output would be wrong. So, in order to check the stability of our model, we did five-fold cross-validation on our model's segmentation and recognition part. For that, we took 1000 images of license plates, and divided them into five sets, each containing 200 images. For every fold, we created the training dataset with four sets of 200 images totaling 800 images and the other set is considered as testing dataset. Table III shows the result.

From this result, we find the mean accuracy $\mu = 95.678$ and the standard deviation $\sigma = 0.801$.

E. Testing

1) Dataset for Testing: For testing, we used 200 images. 100 images were from the front of the license plate. 50 images were from the left angle, and the last 50 images were from the right angle compared to the license plate. The angle for left and right images was approximately 45 degrees compared to the front images.

2) Accuracy: From our 200 images, we found an error in only one license plate. so, the accuracy is 99.5%. It is noticeable that the test accuracy is better than the five-fold cross-validation accuracies. We think the major reason for this is because it was able to get 200 more images for training, which was unavailable to the five-fold training sets for testing.

3) Speed: We measured our model for video feed to get a speed measurement. The video size was 1280 x 720 and the frame rate was 30 with The Moving Picture Experts Group (MPEG) format. For Google Colaboratorys Nvidia Tesla K80, we got a speed of 9 frames per second.

F. Comparison With Other Models

Table II refers to a comparative analysis of our model with the other models in the section III-B. As the codes for these papers are not publicly available, this table only shows the reported measurements presented on their papers.

VI. Limitations of Our Model

Dataset: Our model is trained on a dataset which contains images mostly from the Dhaka city, the majority of the vehicles in the dataset are also privately owned vehicles. So, our model can not recognize license plates which are registered form other districts, cities or classes of vehicles that were not present in the dataset. With a diverse dataset, this problem can be easily solved.

Multiple Plates Detection and Recognition Speed: Our network can perform multiple license plate detection and recognition without any problem. But the speed decreases slightly if there are multiple license plates present in the image.

VII. Conclusion

In this paper, we present a methodology for detecting and recognizing the license plate number for Bangla license plates. As there is no public dataset to our knowledge about the

TABLE II
COMPARATIVE ANALYSIS BETWEEN OUR MODEL AND OTHERS

Name	Sample Image	Localization	Recognition	Time
Automatic License Plate Recognition (ALPR) for Bangladeshi Vehicles [7]	Testing: 300	84%	80%	
Bangla Automatic Number Plate Recognition System using Artificial Neural Network [8]		92.1%	84.16%	Detection: 1.3 seconds, Recognition: 1.2 seconds
Bangla License Plate Reader for Metropolitan Cities of Bangladesh Using Template Matching [9]				Recognition: 1.3 seconds
Line Segmentation and Orientation Algorithm for Automatic Bengali License Plate Localization and Recognition [10]	Testing: 119	95.8%	84.87%	
Bangla License Plate Recognition Using Convolutional Neural Networks (CNN) [11]	Training: 1650, Testing: 350	88.67%		
Our model	**Training: 1050, Testing: 200**	**100%**	**99.5%**	**111 milliseconds for the whole process**

TABLE III
FIVE FOLD CROSS VALIDATION RESULT

No. of test	Train data	Test data	Accuracy
One	800	200	94.50%
Two	800	200	95.68%
Three	800	200	95.47%
Four	800	200	95.73%
Five	800	200	97.01%

Bangla license plate, we created a dataset consisting of license plates for multi-class vehicles. For detecting the license plate, we used CNN based method and extract the license plate for recognition. The extracted image from the first network is the input for the second CNN network which is responsible for segmenting and recognizing the license plate number. Our model was tested with 200 images and correctly produced the license plate number for 199 images (99.5%). Also, for video feed, our model performed a speed of 9 frames per second for a single vehicle in the video. Although the frame rate may slightly decrease if there were more than one vehicles in the video feed.

As deep neural network models are data depended, our belief is, a more diverse dataset for training our model will produce a better result for our test dataset. Also, with a diverse dataset, we will be able to classify more classes of vehicles in the future without changing any part in our model.

REFERENCES

[1] G. R. Gonçalves, S. P. G. da Silva, D. Menotti, and W. R. Schwartz, "Benchmark for license plate character segmentation," *Journal of Electronic Imaging*, vol. 25, no. 5, p. 053034, 2016.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[3] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, V. Loumos, and E. Kayafas, "A license plate-recognition algorithm for intelligent transportation system applications," *IEEE Transactions on Intelligent transportation systems*, vol. 7, no. 3, pp. 377–392, 2006.

[4] O. O. V. Villegas, D. G. Balderrama, H. Domínguez, and V. Sánchez, "License plate recognition using a novel fuzzy multilayer neural network," *International journal of computers*, vol. 3, no. 1, pp. 31–40, 2009.

[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] C. Patel, D. Shah, and A. Patel, "Automatic number plate recognition system (anpr): A survey," *International Journal of Computer Applications*, vol. 69, no. 9, 2013.

[7] M. N. I. S. B. S. A. Ashim Kumar Ghosh, Swapan K.D. Sharma, "Automatic license plate recognition (alpr) for bangladeshi vehicles," *Global Journal of Computer Science and Technology*, 2011. [Online]. Available: https://computerresearch.org/index.php/computer/article/view/853

[8] M. M. A. Joarder, K. Mahmud, T. Ahmed, M. Kawser, and B. Ahamed, "Bangla automatic number plate recognition system using artificial neural network," *Asian Transactions on Science & Technology (ATST)*, vol. 2, no. 1, pp. 1–10, 2012.

[9] R. A. Baten, Z. Omair, and U. Sikder, "Bangla license plate reader for metropolitan cities of bangladesh using template matching," in *Electrical and Computer Engineering (ICECE), 2014 International Conference on*. IEEE, 2014, pp. 776–779.

[10] M. R. Haque, S. Hossain, S. Roy, N. Alam, and M. J. Islam, "Line segmentation and orientation algorithm for automatic bengali license plate localization and recognition," *International Journal of Computer Applications*, vol. 154, no. 9, 2016.

[11] M. Rahman, M. S. Nasrin, M. Mostakim, and M. Z. Alom, "Bangla license plate recognition using convolutional neural networks (cnn)," *arXiv preprint arXiv:1809.00905*, 2018.

[12] P. Dhar, S. Guha, T. Biswas, and M. Z. Abedin, "A system design for license plate recognition by using edge detection and convolution neural network," in *2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2)*. IEEE, 2018, pp. 1–4.

[13] I. Sobel and G. Feldman, "A 3x3 isotropic gradient operator for image processing," *a talk at the Stanford Artificial Project in*, pp. 271–272, 1968.

[14] V. Abolghasemi and A. Ahmadyfard, "An edge-based color-aided method for license plate detection," *Image and Vision Computing*, vol. 27, no. 8, pp. 1134–1142, 2009.

[15] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[16] J. M. Prewitt, "Object enhancement and extraction," *Picture processing and Psychopictorics*, vol. 10, no. 1, pp. 15–19, 1970.

[17] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 2011, pp. 315–323.

[18] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *CoRR*, vol. abs/1804.02767, 2018. [Online]. Available: http://arxiv.org/abs/1804.02767