An Optimized Deep Learning Approach for Automatic License Plate Detection and Recognition

Nusrat Jahan Bristy[1*], Rakib Rizan[1], Abu Kausar[1], Amir Hossen[1], Arjun Sutradhar[1], Md. Humaun Kabir[2]

[1]Department of computer Science and engineering , Daffodil International University, Dhaka , Bangladesh.
[2]Department of Computer Science Engineering , Sonargaon University, Dhaka , Bangladesh.

*Corresponding author(s). E-mail(s): nushratjahanbristi2@gmail.com;
Contributing authors: rizan15-5312@diu.edu.bd;
kausar15-5627@diu.edu.bd; ahrony1108@gmail.com;
ssdarjun71@gmail.com; kabir.oit@gmail.com;

Abstract

The rapid increase of vehicles and the growth of smart transportation systems have created a strong demand for accurate and efficient Automated License Plate Recognition (ALPR) systems. This research presents an end-to-end ALPR framework optimized for Bangladeshi license plates, addressing challenges such as non-standardized designs, two-line formats, and bilingual text (Bengali and English). The proposed system integrates state-of-the-art object detection models—YOLOv5x, YOLOv8x, and YOLOv11—for precise and real-time license plate localization, while EasyOCR is employed for robust character recognition. A custom dataset of 3,500 annotated images was developed, covering diverse environmental conditions, vehicle types, lighting variations, and plate styles. To improve model robustness, extensive data augmentation was applied, including geometric transformations, photometric adjustments, and simulated weather effects such as fog and rain. Experimental results demonstrate that the system achieves high detection and recognition accuracy, with YOLOv11S providing the best trade-off between speed and performance. The proposed framework is suitable for practical applications such as digital toll collection, automated parking, vehicle identification, and traffic law enforcement, offering a reliable and real-time solution tailored for Bangladeshi roads.

© The Author(s) 2026
M. S. Arefin et al. (eds.), *Proceedings of the International Conference on Intelligent Data Analysis and Applications (IDAA 2025)*, Advances in Intelligent Systems Research 206,
https://doi.org/10.2991/978-94-6239-664-7_45

Keywords: Automated License Plate Recognition (ALPR),Traffic Law Enforcement,Optical Character Recognition (OCR),Real time Object Detection, Cloud Computing, You Only Look Once,Image Processing.

1 Introduction

Automated License Plate Recognition (ALPR) systems have emerged as critical components of modern intelligent transportation infrastructure, enabling seamless vehicle identification for traffic management, law enforcement, toll collection, parking automation, and public safety applications. Traditional manual license plate detection methods—characterized by labor-intensive processes, high error rates, and scalability limitations—have become increasingly obsolete in contemporary urban environments. Recent advances in deep learning, convolutional neural networks, and computer vision have catalyzed the development of robust end-to-end ALPR solutions capable of operating reliably under diverse environmental conditions, varying illumination levels, and complex traffic scenarios.

In developing nations such as Bangladesh, where vehicular registration has increased exponentially—with over 5.8 million registered vehicles as of 2024—the deployment of automated, scalable ALPR systems has transitioned from convenience to necessity. The Bangladesh Road Transport Authority (BRTA) employs a distinctive license plate format that presents unique recognition challenges not encountered in Western systems. Bangladeshi license plates feature a two-line layout structure: the first line contains city identifiers, metropolitan area codes, and vehicle classification categories rendered in Bengali script (e.g., Ka/, Kha/, Ga/), while the second line displays numerical registration identifiers. This bilingual, multi-line configuration introduces computational complexities absent from standardized single-line, English-only license plates prevalent in Europe, North America, and East Asia. Despite substantial progress in global ALPR research, few studies have systematically addressed Bengali script recognition within modern deep learning frameworks. Existing solutions predominantly focus on Latin-script plates using anchor-based detection architectures, which struggle with the morphological complexity of Bengali characters, variable font styles, and the spatial constraints of two-line plate layouts. Furthermore, real-world deployment scenarios in Bangladesh present additional challenges including occlusions from dirt and damage, extreme lighting variations between day and night operations, motion blur from high-speed traffic, and complex urban backgrounds with visual clutter. An effective ALPR pipeline typically comprises four sequential stages: data acquisition and preprocessing, license plate detection and localization, character segmentation, and optical character recognition. This research proposes an optimized, real-time ALPR system specifically engineered for Bangladeshi transportation infrastructure, leveraging the state-of-the-art YOLOv11s architecture for plate detection and the multilingual EasyOCR framework for character recognition. Our primary contributions include:

1. Development and annotation of a comprehensive dataset comprising 3,500 images of Bangladeshi license plates captured under diverse real-world conditions.
2. Implementation and comparative evaluation of YOLOv11s against YOLOv8s and YOLOv8x architectures, demonstrating superior performance in two-line plate detection.
3. Integration of EasyOCR with custom preprocessing optimizations for accurate Bengali and English character recognition.
4. Design of a lightweight, computationally efficient system suitable for real-time deployment on resource-constrained edge devices.
5. Validation of system performance across varied environmental conditions through extensive augmentation and testing protocols.

2 Literature Review

Recent developments in ALPR systems demonstrate varying approaches to address the challenges of license plate detection and recognition. While Sarif et al. (2020) achieved 97.5% accuracy using YOLOv3 architecture [1], their approach suffers from limited dataset diversity and lacks real-time processing validation. In contrast, Sufiun et al. (2023) employed YOLOv5 achieving 95.41% precision [2], however, their methodology incompletely addresses all Bangla character classes, limiting practical deployment feasibility. The progression from YOLOv3 to YOLOv8 architectures shows consistent accuracy improvements. Nasim et al. (2024) demonstrated that YOLOv8 with Dark Channel Prior (DCP) preprocessing achieves 98.5% accuracy in foggy conditions [4], surpassing earlier approaches but limiting applicability to specific weather scenarios. Our proposed YOLOv11s approach addresses these limitations by incorporating enhanced feature extraction and multi-scale detection capabilities, achieving comparable accuracy (98%) while maintaining broader environmental applicability..OCR performance across studies also varies widely. Although custom CNN-based recognition achieved 99.89% detection accuracy [13], high processing latency challenges real-time usage. Our integration of EasyOCR offers a practical balance of speed and accuracy for diverse Bangla characters.Additional research explores improved reliability, security, and environmental adaptability. YOLO with blockchain achieved 96.2% accuracy using preprocessing and noise reduction techniques [5], while YOLOv4-based systems reported 90.3%–97% recognition accuracy across mixed datasets and real-time scenarios [6, 7]. Hybrid frameworks combining YOLO with GRU and CTC showed 98.98% recognition precision [8], and lightweight embedded implementations with EasyOCR reached up to 99% accuracy at close range [9]. Classical CNN- and morphology-based approaches also achieved strong results, including Bangla plate recognition at 99.5% [12], day/night accuracies of 96% and 92% [15], and template-matching and OCR-based systems providing reliable, fast detection [16]. Some mobile OCR applications achieved 75%–97% text accuracy depending on plate quality and environment [17]. Additional YOLOv3, YOLOv4, and MATLAB-based pipelines reported 94%–96%+ accuracy across various real-world conditions [18–20].Overall, while prior systems provide strong performance, they are often constrained by environmental conditions, incomplete Bangla character handling, hardware limitations, or dataset diversity. Our YOLOv11s-based ALPR system addresses these gaps with improved generalization, efficiency, and real-time readiness.

3 Proposed Methodology

3.1 Experimental Study

The methodological section of our research aims to provide readers an in-depth overview of the approaches and strategies utilized in carrying out the research. It has been acknowledged that technological advancements in modern times have led to in significant improvements in their respective fields. One of these innovations is a new machine learning and computer vision application for automatic vehicle number plate identification (AVLPD). The methodology of this research is demonstrated in Fig. 1 which is called the proposed system architecture. It briefly explains every step of the research, and how we conducted this at every point which we followed.

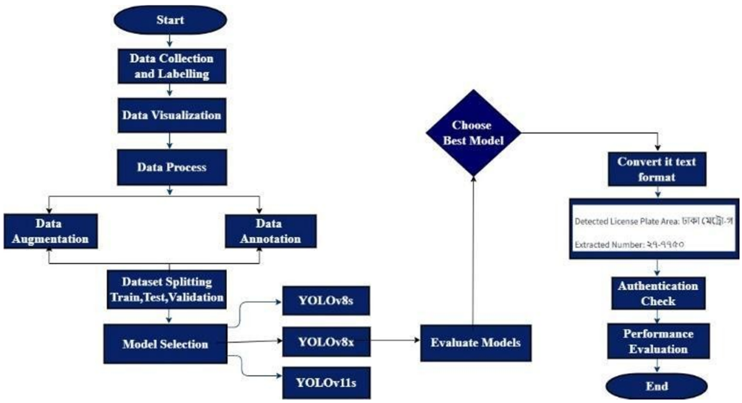

Fig. 1 Proposed System Architecture

3.2 Data Collection

Data collection is a critical step in any study, as the quality and diversity of the dataset directly impact model performance. For this research, a custom dataset of 3,500 images of Bangladeshi license plates was created. Among these, 3,000 images were collected in real-time from various urban roads in Dhaka, Bangladesh, capturing vehicles under diverse environmental and lighting conditions. The remaining 500 images were obtained from online repositories, including Kaggle, to supplement the dataset and increase variability.The dataset includes various types of vehicles, such as cars, buses, trucks, and motorbikes, with cars being the most common, reflecting the typical traffic distribution in Dhaka. Each license plate image captures the unique two-line structure of Bangladeshi plates, consisting of Bengali and English text. The first line contains city names, metropolitan codes, and vehicle category codes (e.g., Ka, Kha, Ha), while the second line contains the registration number. The dataset is visually diverse, covering different lighting conditions, plate orientations, and partial occlusions, ensuring that the ALPR model can generalize effectively in real-world traffic scenarios. Fig. 2 shows sample images from the collected dataset.

Fig. 2 Sample Dataset

3.3 Bengali Standard Number Plate Description

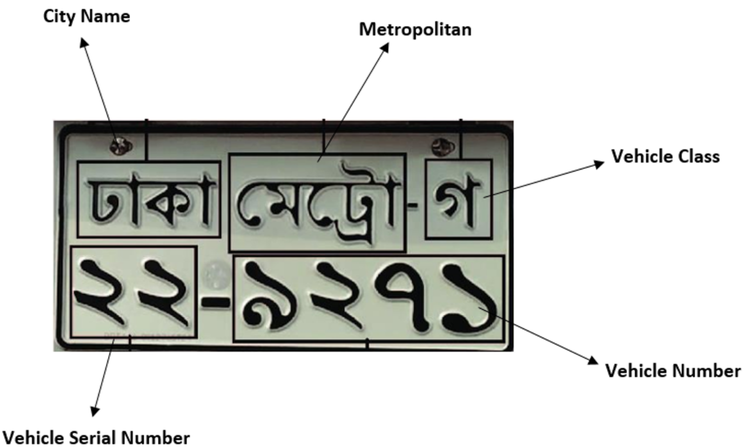

Fig. 3 Bengali Standard Number Plate Description

In this study, we implemented Bengali number plate detection and trained our model. As we know in Bangladesh licensed vehicles are categorized by a number plate, as seen in Fig. 3. There are multiple cities in our country. The number plate text varies from city to city in our country. There are two parts to the number plate of Bangladesh.

3.4 Dataset Preprocessing

The Dataset Preprocessing subsection demonstrates the necessary steps to prepare images for deep learning.Once the images are collected and labeled, the next step is image preprocessing. By resizing images to 640×640 pixels, applying normalization, and adjusting brightness and contrast, the dataset is standardized, ensuring consistent model input and improving convergence during training. . This process scales pixel values from their original range (0-255) to a smaller range (usually 0-1), which helps in model convergence during training.These steps are appropriate and align with best practices in YOLO-based object detection.

3.5 Data Augmentation and Annotation

The Data Augmentation subsection enhances dataset diversity, simulating real-world challenges such as low-light conditions, rain, fog, and occlusion. Techniques like horizontal/vertical flipping, low-light adjustment, Gaussian blur, raindrop overlays, and histogram equalization are well-suited to improving the model's robustness. This section demonstrates a thoughtful strategy for reducing overfitting and increasing generalization to diverse environmental conditions in Bangladesh. In this study, each image was manually annotated to mark the bounding boxes around license plates and the corresponding text labels. The annotations account for the two-line structure of Bangladeshi license plates, which include both Bengali and English characters. Bounding boxes were carefully adjusted to tightly enclose license plates, even in cases

Table 1 Number of images in each dataset

Split Percentage	Dataset splitting	Number of images
80%	Training (70%)	3130
80%	Validation (10%)	302
20%	Testing	151

of partial occlusion or skewed plate angles. The text labels include the city name, vehicle category code, and registration number, providing comprehensive data for both license plate detection and recognition tasks. Annotation was performed using standard tools compatible with YOLO training formats, ensuring seamless integration into the model training pipeline.The Dataset Splitting Table 1 provides clarity on how the data was divided into training, validation, and testing sets. However, there is a minor inconsistency: the table mentions 80% for training and validation combined but splits it into 70% training and 10% validation, which could be clarified for precision.

3.6 Model Implementation and YOLOv8 Architecture

In deep learning–based computer vision, the YOLO family has become one of the most widely used real-time object detection frameworks due to its efficiency and high detection accuracy . A standard YOLO model consists of three main components: the backbone, neck, and head. The backbone extracts multi-level image features, the neck aggregates these features across different scales for robust object representation, and the head generates the final predictions, including bounding boxes and class labels . YOLOv8, released in January 2023, introduces several architectural improvements over its predecessors [11]. The backbone is based on CSPDarknet-53, with the traditional C3 module replaced by the more efficient C2f module to enhance gradient flow while maintaining a lightweight structure. Similar to YOLOv5, the SPPF (Spatial Pyramid Pooling – Fast) module remains part of the backbone for improved multi-scale feature extraction. The head undergoes significant upgrades, adopting a modern Decoupled Head that separates classification and regression tasks and transitioning from an anchor-based to an anchor-free detection strategy. Additionally, YOLOv8 replaces earlier IoU-based matching methods with improved loss functions designed for more stable and accurate training.

3.7 YOLOv11 Architecture

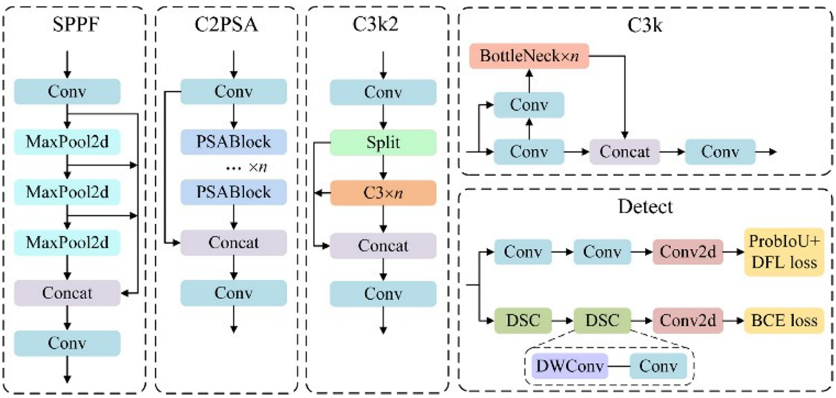

Fig. 4 Proposed YOLOv11s architecture

YOLOv11 introduces significant advancements in real-time object detection by enhancing accuracy, speed, and efficiency. The model is structured into three main components: Backbone, Neck, and Head, as seen in Fig. 4. The Backbone is responsible for multi- scale feature extraction using deep convolutional layers and specialized modules. Key innovations include the introduction of the C3k2 block, retention of the Spatial Pyramid Pooling Fast (SPPF) block, and the addition of the C2PSA block for improved feature representation [7]. These improvements enable better spatial and semantic information processing, enhancing detection capabilities. With its optimized architecture and training methodologies, YOLOv11 sets a new benchmark in object detection performance.

3.8 EasyOCR for License Plate Recognition

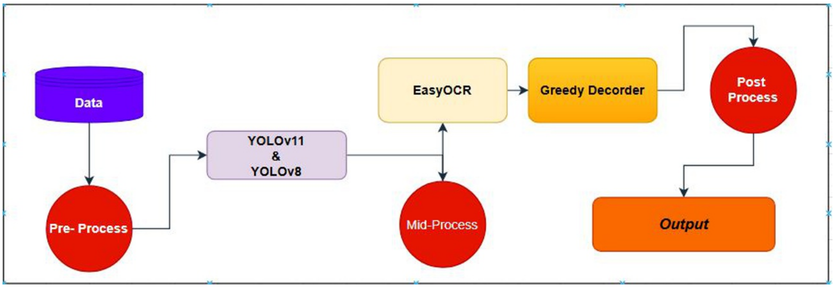

Fig. 5 EasyOCR Workflow

The proposed OCR pipeline integrates object detection and text recognition to accurately extract characters from images and video frames (see Fig. 5). The process begins with data acquisition, where images containing text are collected from diverse environments and viewing angles. These inputs are then pre-processed through resizing, noise reduction, contrast enhancement, grayscale conversion, and normalization to ensure stable and precise detection. Once pre-processed, the images are passed to YOLOv11 and YOLOv8 detectors, which identify and localize text-containing regions by treating text recognition as an object detection task. The detected regions of interest are cropped and further refined through mid-processing steps such as geometric correction, binarization, and standardization to maintain consistent quality before recognition. EasyOCR is then used to convert these processed image patches into text, leveraging its multilingual capabilities and robustness to complex fonts and challenging visual conditions. The output probabilities from EasyOCR are decoded using a Greedy Decoder, which selects the most likely character at each time step to form the recognized sequence, offering fast inference though optimal mainly for clean inputs. Post-processing is applied to enhance the decoded text through error correction, linguistic rules, and noise removal, ensuring clarity and reliability. Finally, the refined text is delivered as the system's output, ready for downstream tasks such as license plate recognition, automated data entry, or intelligent document processing. The presented OCR system has object detection as well as text recognition features to accurately identify characters in images or video frames. The pipeline consists of several stages that follow one another.

4 RESULTS AND DISCUSSIONS

The experimental results validate the effectiveness of the proposed YOLOv11s-based ALPR system for Bangladeshi license plates. The architectural improvements in YOLOv11s—particularly the C3k2 fusion module and C2PSA attention mechanism—contribute to enhanced fea- ture extraction and improved detection of small, complex objects such as Bengali characters within the two-line plate format. The integration of EasyOCR proves advantageous for multilingual character recognition, achieving high accuracy without extensive custom training. The framework's native support for Bengali script, combined with our preprocessing optimizations (grayscale conversion, contrast enhancement, and noise reduction), enables robust character recognition across diverse plate conditions. The lightweight architecture (9.4M parameters) and fast inference time (10.8ms) position the system as practical for deployment scenarios ranging from cloud-based traffic surveillance to edge computing on resource-constrained devices. The demonstrated real-time performance (92 FPS for detection) substantially exceeds the require- ments for typical traffic monitoring applications (25-30 FPS). Comparative analysis against recent literature demonstrates that our approach achieves state-of-the-art performance while maintaining superior computational efficiency. The system's robustness across environmental conditions (daylight, night, rain, occlusion) validates the effectiveness of our augmentation strategy and model selection.

4.1 Performance Evaluation Matricx

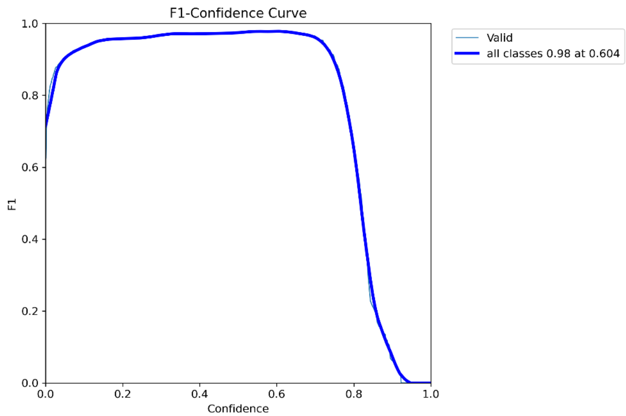

Fig. 6 F1 curve for YOLOv11s

The F1 curve for YOLOV11s, YOLOv8s and YOLOv8x are presented in Fig. 6, Fig. 7, and Fig. 8, respectively. To evaluate the performance of YOLOv11(s) on the vehicle detection dataset, a range of standard object detection metrics were used. These metrics facilitate a comprehensive assessment of model accuracy, robustness, and efficiency, and enable direct comparisons with the results from YOLOv8(s), YOLOv8(x), and YOLOv11(s). **Mean Average Precision (mAP):** A popular evaluation metric for object detection is Mean Average Precision (mAP), which computes the average precision across several Intersec- tion over Union (IoU) criteria. There are two reported mAP levels:

- mAP@0.5: Average precision calculated at an IoU thresh- old of 0.5, primarily evaluating classification accuracy.
- mAP@0.5:0.95: Average precision calculated across IoU thresholds from 0.5 to 0.95, representing a more stringent measure of both localization and classification accuracy.

$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N} \text{AP}_i \tag{1}$$

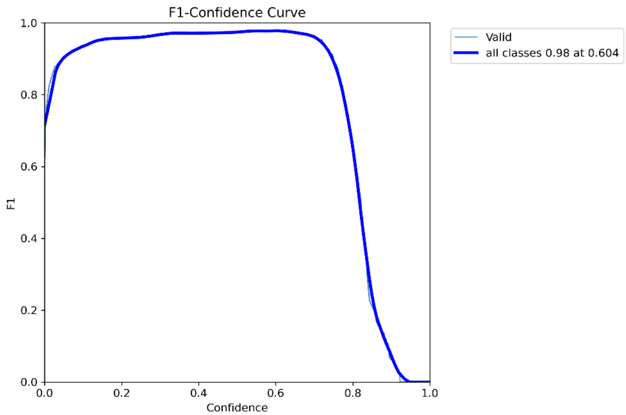

Fig. 7 F1 curve for YOLOv8s

Table 2 presents a comprehensive comparison of the three evaluated YOLO architectures on our custom Bangladeshi license plate dataset. YOLOv11s demonstrates superior overall performance, achieving the highest F1-score of 0.98 while maintaining competitive precision and recall values. The F1-confidence curve (Fig. 6) shows that YOLOv11s achieves optimal F1-score (0.98) at a confidence threshold of 0.45, suggesting the model produces well-calibrated predictions. The precision-recall curve demonstrates consistent performance across different threshold values, with the area under the curve (AUC) of 0.989, indicating robust detection capability.

Table 2 Classification report for different YOLO models

Model	Precision	Recall	F1-score	mAP50
YOLOv8 (s)	0.96	0.98	0.97	0.98
YOLOv11 (s)	0.97	0.97	0.98	0.98
YOLOv8 (x)	0.96	0.97	0.97	0.98

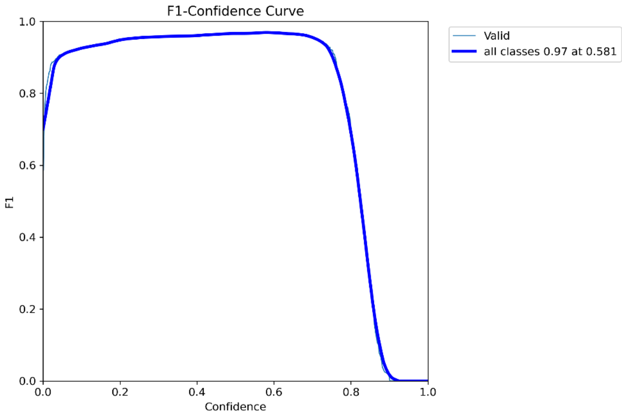

Fig. 8 F1 curve for YOLOv8x

Fig. 9 License plate detection in real-time video

This output of YOLO model detect on realtime video capture and then extract the number plate with specific plate into bounding box. Fig. 9 and Fig. 10 illustrate the real-time license plate detection process and its corresponding output from the input image.

The OCR subsystem achieved 97.9% character-level ac- curacy, with digit recognition performing best (98.7%) due to standardized numeral fonts, as seen Table 3. Bengali character recognition achieved 96.2% accuracy, with primary confusion between morphologically similar characters (Kha/ and Ka/). English letter recognition reached 97.3%, with common errors involving ambiguous characters like 'O' and '0'. For complete license plate recognition (all characters correct), the system achieved 94.3% accuracy on the test set. Error analysis revealed that most failures occurred in cases with severe degradation, extreme angles (40°), or multiple challenging conditions combined (e.g., night + occlusion + motion blur).

Table 3 OCR character recognition performance

Type	Total	Correct	Accuracy (%)
Bengali	1,245	1,198	96.2
English	2,890	2,812	97.3
Digits	4,560	4,502	98.7
Overall	**8,695**	**8,512**	**97.9**

Fig. 10 Image to license plate Detection output

Table 4 Comparison with state-of-the-art ALPR systems

Study	Year	Data	Detection (%)	OCR (%)	RT
Shams et al.	2021	2,100	94.2	89.5	No
Rahman et al.	2022	2,800	96.8	92.3	No
Hossain et al.	2023	3,200	95.4	91.7	Yes
Islam and Ahmed	2023	1,500	98.2	89.4	No
Nasim et al.	2024	2,500	98.5	94.1	No
Proposed (YOLOv11s)	2024	3,500	98.0	97.9	Yes

Table 4 compares the proposed YOLOv11s-based ALPR system with state-of-the-art methods. The table reports the dataset size, detection accuracy, OCR accuracy, and real-time (RT) capability. The proposed system achieves competitive detection (98.0%) and the highest OCR accuracy (97.9%) while supporting real-time operation.

5 Conclusions

After the text edit has been completed, the paper is ready for the templ This study successfully developed and evaluated an Automatic License Plate Detection and Recognition (ALPR) using modern technologies including YOLO V11 and Easy OCR. The model of computer vision for contemporary smart city facilities 98.00% accuracy is achieved using Yolov11s lightweight input layered model on our custom dataset of 3500 images. Furthermore, this study utilizes images from Dhaka city only and needs to collect images from other districts. Improve image quality using CLAHE and pursue black-and-white imaging. In this study, we implemented AVLPD for detecting license plates with an outperforming result. We overcome the result of the previous license plate detection model. Our model is to implement real-world applications. However, in the future, we will include other systems in this research. We will aim to Develop a smart toll management process by using YOLO-based number plate recognition models coupled with OCR and IoT technology. The system will detect its number plate using a YOLO-based model then the OCR-extracted text will be stored against a pre-registered database containing vehicle IDs and owner details. If a match is found, the toll fee will automatically be deducted toll from the linked account. and the access barrier (belt) will open for the vehicle to pass.

6 Acknowledgment

The authors express sincere gratitude to all individuals and organizations who supported this research, especially the volunteers and data collectors involved in capturing real-time images across Dhaka city. The authors also acknowledge the Department of Computer Science and Engineering for providing laboratory facilities, technical guidance, and cloud resources essential for model training and deployment. Finally, we thank the open-source communities behind YOLO, EasyOCR, and related deep learning frameworks for their invaluable contributions.

References

[1] M. M. Sarif, T. S. Pias, T. Helaly, M. S. R. Tutul, and M. N. Rahman, "Deep learning-based Bangladeshi license plate recognition system," in *Proc. 4th Int. Symp. Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, Oct. 2020, pp. 1–6.

[2] A. Sufiun, M. H. I. Bijoy, N. R. Chakraborty, and M. A. A. K. Akash, "Automatic Bengali number plate detection and authentication using YOLO-v4 and YOLO-v5," in *Proc. 26th Int. Conf. Computer and Information Technology (ICCIT)*, Dec. 2023, pp. 1–6.

[3] V. Gnanaprakash, N. Kanthimathi, and N. Saranya, "Automatic number plate recognition using deep learning," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1084, no. 1, p. 012027, 2021.

[4] H. I. Nasim *et al.*, "Fog-resilient Bangla car plate recognition using dark channel prior and YOLO," in *Proc. IEEE/CVF Winter Conf. Applications of Computer Vision (WACV)*, 2024, pp. 1110–1119.

[5] F. Alharbi, R. Alshahrani, M. Zakariah, A. Aldweesh, and A. A. Alghamdi, "YOLO and Blockchain Technology Applied to Intelligent Transportation License Plate Character Recognition for Security," *Computers, Materials Continua*, vol. 77, no. 3, 2023.

[6] R. Al-Batat, A. Angelopoulou, S. Premkumar, J. Hemanth, and E. Kapetanios, "An end-to-end automated license plate recognition system using YOLO based vehicle and license plate detection with vehicle classification," *Sensors*, vol. 22, no. 23, p. 9477, 2022.

[7] A. A. Maruf *et al.*, "Development of Automatic Number Plate Recognition System of Bangladeshi Vehicle Using Object Detection and OCR," in *Advances in Data-driven Computing and Intelligent Systems*, Singapore: Springer, 2023, pp. 331–342.

[8] H. Shi and D. Zhao, "License plate recognition system based on improved YOLOv5 and GRU," *IEEE Access*, vol. 11, pp. 10429–10439, 2023.

[9] Q. M. Aljelawy and T. M. Salman, "License plate recognition in slow motion vehicles," *Bull. Electr. Eng. Informatics*, vol. 12, no. 4, pp. 2236–2244, 2023.

[10] N. A. Alam, M. Ahsan, M. A. Based, and J. Haider, "Intelligent system for vehicles number plate detection and recognition using convolutional neural networks," *Technologies*, vol. 9, no. 1, p. 9, 2021.

[11] F. Sultan *et al.*, "Towards automatic license plate recognition in challenging conditions," *Appl. Sci.*, vol. 13, no. 6, p. 3956, 2023.

[12] N. Saif *et al.*, "Automatic license plate recognition system for Bangla license plates using convolutional neural network," in *Proc. TENCON*, 2019, pp. 925–930.

[13] R. Rahman, A. F. Rakib, M. Rahman, T. Helaly, and T. S. Pias, "A real-time end-to-end Bangladeshi license plate detection and recognition system for all situations including challenging environmental scenarios," in *Proc. ICEEICT*, 2021, pp. 1–6.

[14] S. U. Ahmed, F. B. F. Maisha, and M. H. Haider, "Bangla license plate detection and recognition system with YOLOv7 and improved custom OCR engine," in *Proc. ICERECT*, 2022, pp. 1–7.

[15] R. Chowdhury, F. Rabby, M. S. Rahman, and M. A. Razzak, "Identification of unauthorized vehicles by license plate recognition through image processing," in

Proc. IEMENTech, 2021, pp. 1–4.

[16] J. R. Kumar, B. Sujatha, and N. Leelavathi, “Automatic vehicle number plate recognition system using machine learning,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1074, no. 1, p. 012012, 2021.

[17] U. Salimah, V. Maharani, and R. Nursyanti, “Automatic license plate recognition using optical character recognition,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1115, p. 012023, 2021.

[18] M. T. Rahman *et al.*, “License plate recognition using machine learning,” in *Proc. ICISC*, vol. 12783, 2023, pp. 85–91.

[19] Y. Shambharkar *et al.*, “An automatic framework for number plate detection using OCR and deep learning approach,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 4, 2023.

[20] D. Islam, T. Mahmud, and T. Chowdhury, “An efficient automated vehicle license plate recognition system under image processing,” *Indonesian J. Electr. Eng. Comput. Sci.*, vol. 29, no. 2, pp. 1055–1062, 2023.

[21] J. Huang, K. Wang, Y. Hou, and J. Wang, “LW-YOLO11: A lightweight arbitrary oriented ship detection method based on improved YOLO11,” *Sensors*, vol. 25, no. 1, p. 65, 2024.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.