Journal of
Image Processing & Pattern Recognition Progress

https://journals.stmjournals.com/joipprp

ISSN: 2394-1995
Volume 13, Issue 1, 2026
January–April
DOI (Journal): 10.37591/JoIPPRP

Review JoIPPRP

Automated Car License Plate Detection and Recognition Using Deep Learning

Prachi Pravin Durge[1,*], Suraj V. Dhole[2]

Abstract

The use of automated license plate detection and recognition (ALPR) systems to automate processes such as number plate detection is gaining popularity in traffic control, security, and law enforcement. This research focuses on achieving more accurate and efficient detection and recognition of number plates by leveraging deep learning techniques. The systems outlined in this study aim to improve the effectiveness of ALPR systems using advanced convolutional neural networks (CNNs) and other object detection algorithms in challenging and low-visibility scenarios. The methodology is based on a dataset containing images of different number plates, which undergo various processing stages across multiple models, with further adjustments implemented as needed. Moreover, the real-time detection and recognition framework demonstrates a balanced trade-off between speed and accuracy, maintaining a consistent recognition rate under complex and varying lighting conditions and across different plate designs. The system is shown to outperform previous works in detection accuracy and processing speed through rigorous evaluation. This work contributes to the advancement of reliable and efficient ALPR systems, with applications in automated traffic control and enforcement systems.

Keywords: Convolutional neural networks (CNN), plate detection, image processing, deep learning, optical character recognition (OCR)

INTRODUCTION

Automated license plate detection and recognition (ALPR) has emerged as a strong and impactful technology, with adoption growing across several industries, including law enforcement, traffic control management, and access control systems [1]. Automated number plate detection and recognition (ANPR) systems offer considerable advantages, such as improved security, automated toll collection, and streamlined traffic monitoring by interposing vehicle number plate identification and reading. Traditionally, ALPR systems depend on simple machine learning algorithms along with methodical image processing techniques [2]. However, these systems have shortcomings in handling complex scenarios, such as lighting changes, varied plate designs, and occluded or distorted plates [3].

The deep learning paradigm shift has marked a point of transformation in terms of challenges related to computer vision and image processing. Recognition of images through machines is a primary task in computer vision that identifies real-world objects [4]. With the help of a deep learning model, and especially with the use of convolutional neural networks (CNN), they have been proven to perform image recognition and tasks because of their feature learning of hierarchical structures. This

***Author for Correspondence**
Prachi Pravin Durge
E-mail: prachidurge349@gmail.com

[1]Assistant Professor, Department of Computer Science and Engineering, G H Raisoni University, Amravati, Maharashtra, India
[2]Assistant Professor, Department of Computer Science and Engineering, G H Raisoni University, Amravati, Maharashtra, India

Received Date: May 12, 2025
Accepted Date: August 23, 2025
Published Date: February 25, 2026

Citation: Prachi Pravin Durge, Suraj V. Dhole. Automated Car License Plate Detection and Recognition Using Deep Learning. Journal of Image Processing & Pattern Recognition Progress. 2026; 13(1): 14–29p.

has significantly improved the reliability and accuracy standards of ALPR systems [5]. Moreover, further developments in object detection frameworks and OCR techniques have significantly advanced the functionality of ANPR systems, enabling real-time detection, recognition, and tracking [6].

Objectives of the Project

The focus objectives of this project include the design and development of the ALPR, which recognizes, reads, and captures vehicle registration plates for different uses, such as automated traffic control and law enforcement [7]. Through the development of this system, we aim to achieve accurate counting and recognition of license plates by leveraging automated image processing techniques to facilitate toll collection, monitor traffic flow in and out of the city, and for security purposes. Furthermore, this project intends to maintain a high recognition rate with minimal processing time in real-world situations [8].

The Key to this Project

- We apply image processing and object detection algorithms to find a license plate patch and set the boundaries.
- Character recognition using deep learning methods for precise identification.
- Fast processing time and high-accuracy level solutions under various weather and lighting conditions.
- Ease of integration with CCTV systems, toll machines, and smart parking structures.

LITERATURE REVIEW AND RELATED WORK

The application of ALPR is crucial in contemporary traffic control, security, and even law enforcement. The cornerstone methods of ALPR resistance in image processing methods, such as edge detection, template matching, and morphological operations, are usually ineffective in real-life situations [9]. Real-world scenarios pose challenges, such as lighting changes, orientation shifts, and obstructions that profoundly affect accuracy. Fortunately, recent developments in deep learning technologies have offered more reliable and accurate solutions for detection and recognition, significantly enhancing the performance of both the detection of the number of plates and the recognition of characters [10].

Analysis of the Problem

The traffic management, security, and law enforcement uses of ALPR systems are widespread. Even with a lot of technological advances, there are still a lot of problems that limit the accuracy and efficiency of these systems [11]. The core issues of existing ALPR systems are analyzed in detail, and a number of major problem areas are revealed, essentially in dealing with the real-world complexities of environmental variability, plate diversity, and system scalability [12].

METHODOLOGY

Traditional image processing techniques or even deep learning models struggle to accurately detect and recognize number plates when the lighting conditions are poor (e.g., at night or in overcast weather), and there are strong shadows or glare from vehicle headlights. Recent systems are trained in controlled environments and suffer from poor performance in the presence of extreme variations in lighting [13].

Plate Orientation and Camera Angles

The number of plates may not be captured at the front-facing angle. Skewed, rotated, or angled plates may occur because of variations in the vehicle positioning or movement. The detection process is further complicated by these geometric distortions because many systems rely on the assumption that the plates will be almost parallel to the image plane [14].

Number plate detection from live video feeds or still images in real time is a very complex task with many technical challenges that must be resolved in real time with acceptable accuracy and speed. In high-traffic areas or with large-scale applications, the performance requirements may be difficult to meet by systems. To handle real-time constraints, NNs, such as other advanced machine learning models, must be optimized for speed and accuracy [15].

Real-Time Detection and Recognition

Real-time detection and recognition become increasingly complex as advanced machine learning models, such as neural networks (NNs), need to be optimized to achieve a balance between speed and accuracy. Model pruning, quantization, and the use of hardware accelerators (e.g., Graphics rocessing Units (GPUs) and Tensor Processing Units (TPUs)) can reduce the processing time by orders of magnitude without affecting the detection accuracy. In addition, the system can meet real-time constraint demands using real-time processing frameworks and parallel computing.

Plate Occlusion and Damaged License Plates

It is common for the number plate to be partially occluded by objects (e.g., dirt, mud, and other vehicles) or damaged through wear and tear, reducing the number plate recognition to an incomplete or inaccurate plate. Such obstructions may impede the system from identifying the full license plate or reading all characters correctly.

The system can be improved by adding inpainting techniques to restore the occluded parts of the number plate. In addition, training models on partially occluded and damaged plates in the dataset proved helpful in making the models more robust. Recognition accuracy in challenging scenarios can also be enhanced using advanced computer vision techniques such as generative models or super-resolution algorithms.

SYSTEM OVERVIEW

The proposed ALPR system consists of four main stages:

1. Image acquisition
2. License plate detection
3. Character segmentation
4. Character recognition

Each stage utilizes deep learning techniques to improve the accuracy and robustness under varying conditions, such as lighting, angle, and occlusions.

Data Collection and Preprocessing

- *Dataset:* We used public datasets (e.g., OpenALPR, CCPD, and private datasets) covering diverse conditions (day/night, multiple angles, and weather conditions).
- *Annotation:* Bounding boxes for license plates and labels for individual characters were annotated manually or using semi-automated tools.
- *Preprocessing:*
    - Resize images to a fixed size (e.g., 512×512 pixels).
    - Normalize pixel values.
    - *Data augmentation:* random brightness, rotation, blurring, and occlusion simulation.

License Plate Detection

- *Model:* YOLOv8/Faster R-CNN/SSD (depending on application needs—speed versus accuracy trade-off) (Figures 1 and 2).
- *Training:*
    - *Input:* Full vehicle images.
    - *Output:* Bounding box coordinates for the license plate.
    - *Loss:* Intersection over Union (IoU) loss + Classification loss.

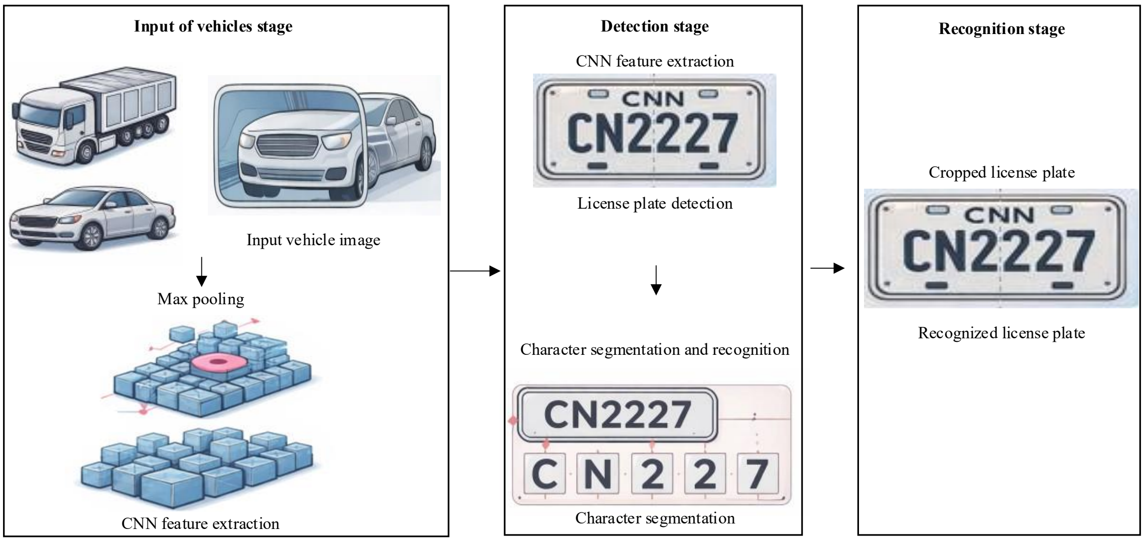

Figure 1. Flowchart diagram of CNN in the ANPR system.

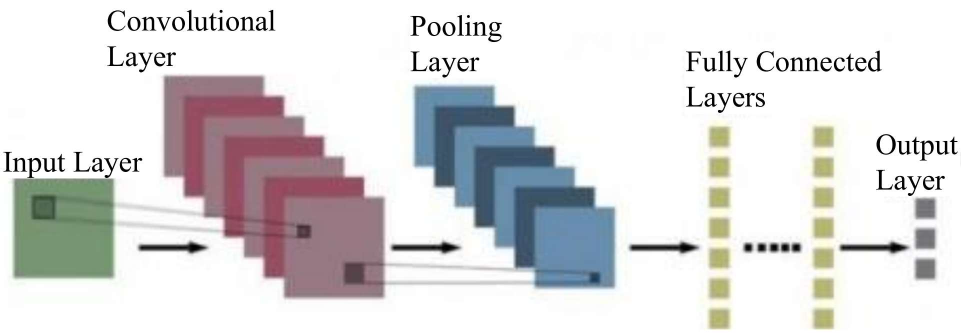

Figure 2. Typical architecture of CNN.

License Plate Cropping and Alignment

- After detection, plates are cropped from the image.
- *Optional:* Apply perspective transformation to correct skewed plates using homography estimation.

Character Segmentation

- *Methods:*
  - Traditional methods (morphological operations, connected components) or
  - *Deep learning methods* (like U-Net for segmentation masks).
- *Goal:* Isolate each character from the license plate image accurately.

Character Recognition

- *Model:* CNN-based classifier (e.g., ResNet, DenseNet) or sequence models convolutional recurrent neural network (CRNN) Transformers.
- *Training*
  - *Input:* Cropped characters (or full license plate for sequence models).
  - *Output:* Character class (A-Z, 0-9).
- *Loss:* Cross-entropy loss for classification.

Post-Processing

- Merge recognized characters into a full license plate string.
- Apply regex patterns based on country-specific plate formats to correct minor recognition errors.

Evaluation Metrics

- *Detection:* mAP (mean average precision) at different IoU thresholds.
- *Recognition:*
  - Character accuracy.
  - Plate accuracy (all characters correct).
- *System:* Overall plate recognition rate.

Deployment

- *Framework:* TensorFlow, PyTorch, or open neural network exchange (ONNX) for real-time inference.
- *Optimization:* Quantization or pruning for lightweight deployment.

SOFTWARE DEVELOPMENT PHASES

- *Hardware requirements*
  - Core i3/i5/i7 processor
  - At least 4 GB RAM
  - At least 20 GB of usable hard disk space
- *Software requirements*
  - Windows 10 above, VSCode IDE, SQLite Database, RestAPI, Chrome Browser, Django Backend Framework.
- *Frontend technology:* HTML, CSS, JavaScript, Bootstrap version 5.

Backend Processing

The backend was built using Python and Django, acting as a bridge between the user interface (UI) and artificial intelligence (AI) models. Text inputs were processed using natural language processing (NLP) techniques to ensure accurate interpretation across the three languages. Django handles HTTP requests, manages sessions, and routes data between the user interface and image generation model.

- *TensorFlow/PyTorch*: For building and training deep learning models.
- *OpenCV:* For image preprocessing, object detection, and general image manipulation.
- *YOLO (You Only Look Once):* Object detection in real time (can be integrated with OpenCV).
- *Scikit-learn:* For classical machine learning algorithms and preprocessing.
- *RESTful APIs:* For communication between the frontend and backend.
- *SQLITE :* SQLITE: Lightweight SQL database engine
- *Django:* Django is a feature-rich Python web framework for building scalable web applications.
- *YOLO (You Only Look Once):* For real-time object detection, locate a number of plates in images or videos.

User Interface and Experience Design

- *Design principles:* Apply user-centered design principles to create an intuitive and engaging interface.
- *Prototype testing:* Conduct usability testing to refine UI elements and ensure a positive user experience.
- *Feedback integration:* Incorporating user feedback into the user experience (UX) design to enhance overall satisfaction and usability.
- *Accessibility features:* Ensure that the system is accessible to users with diverse needs, including those with disabilities.
- *Encryption:* Implementation of encryption protocols for data transmission and storage to protect user data from unauthorized access.
- *Access controls:* Establish access control mechanisms to ensure that only authorized personnel can access sensitive information.

CONCLUSION

This project aims to develop an efficient and reliable ANPR system using deep learning and computer vision techniques. By integrating state-of-the-art models for object detection, image processing, and optical character recognition (OCR), the system can accurately detect and recognize plates from vehicle images and video streams. The real-time performance, accuracy, and adaptability of the system demonstrated its potential for practical applications in traffic monitoring, law enforcement, toll management, and parking systems.

REFERENCES

1. Wang H, Li Y, Dang LM, Moon H. Robust Korean license plate recognition based on deep neural networks. Sensors. 2021;21(12):4140. doi:10.3390/s21124140.
2. Nguyen H. An efficient license plate detection approach using lightweight deep convolutional neural networks. Adv Multimed. 2022;2022:1–10. doi:10.1155/2022/8852142.
3. Gnanaprakash V, Kanthimathi N, Saranya N. Automatic number plate recognition using deep learning. IOP Conf Ser Mater Sci Eng. 2021;1084:012027. doi:10.1088/1757-899X/1084/1/012027.
4. Villena Toro J, Wiberg A, Tarkian M. Optical character recognition on engineering drawings to achieve automation in production quality control. Front Manuf Technol. 2023;3:1154132. doi:10.3389/fmtec.2023.1154132.
5. Jaerisch J, Matsuzaki K. Weighted cogrowth formula for free groups. Groups Geom Dyn. 2020;14(2):349–368. doi:10.4171/ggd/547.
6. Bai H, Liu C. A hybrid license plate extraction method based on edge statistics and morphology. Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. Cambridge, UK. 2004. Vol. 2. p. 831–834. doi:10.1109/ICPR.2004.1334387.
7. Silva SM, Jung CR. License plate detection and recognition in unconstrained scenarios. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y, editors. Computer Vision – ECCV 2018. Lecture Notes in Computer Science. Vol. 11216. Cham: Springer; 2018. p. 580–596. doi:10.1007/978-3-030-01258-8_36.
8. Selmi Z, Halima MB, Pal U, Alimi MA. DELP-DAR system for license plate detection and recognition. Pattern Recognit Lett. 2020;129:213–223. doi:10.1016/j.patrec.2019.11.007.
9. Omar N, Sengur A, Al-Ali SGS. Cascaded deep learning-based efficient approach for license plate detection and recognition. Expert Syst Appl. 2020;149:113280. doi:10.1016/j.eswa.2020.113280.
10. Almazán J, Gordo A, Fornés A, Valveny E. Efficient exemplar word spotting. In: Bowden R, Collomosse J, Mikolajczyk K, editors. Proceedings of the British Machine Vision Conference (BMVC 2012). Durham (UK): BMVA Press; 2012. p. 67.1–67.11. doi:10.5244/C.26.67.
11. Mirza M, Osindero S. Conditional generative adversarial nets. [Preprint]. 2014 Nov 6:arXiv:1411.1784. doi:10.48550/arXiv.1411.1784.
12. Hsiao CH, Lee H, Wang YT, Hsu MJ. Efficient license plate alignment and recognition using FPGA-based edge computing. Electronics. 2025;14:2475. doi:10.3390/electronics14122475.
13. Sung JY, Yu SB, Korea SP. Real-time automatic license plate recognition system using YOLOv4. 2020 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia), Seoul, Korea (South). 2020. p. 1–3. doi:10.1109/ICCE-Asia49877.2020.9277050.
14. Vaiyapuri T, Nandan Mohanty S, Sivaram M, Pustokhina IV, Pustokhin DA, Shankar K. Automatic vehicle license plate recognition using optimal deep learning model. Comput Mater Continua. 2021;67(2):1881–1897. doi:10.32604/cmc.2021.014924.
15. Chen GW, Yang CM, İk TU. Real-time license plate recognition and vehicle tracking system based on deep learning. 2021 22nd Asia-Pacific Network Operations and Management Symposium (APNOMS), Tainan, Taiwan. 2021.p. 378–381. doi:10.23919/APNOMS52696.2021.9562691.

