International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

License Plate Recognition Using Opencv and Easyocr

Likitha B D

Department of MCA, The National Institute of Engineering, Visvesvaraya Technological University, Belagavi, India

ABSTACT

With the rapid growth of vehicles, manual monitoring of license plates has become inefficient and error-prone. Automated License Plate Recognition (LPR) systems provide an effective solution by identifying vehicle numbers without human intervention. This paper presents a real-time license plate recognition system that captures live video through a webcam, detects vehicle license plates, extracts alphanumeric characters using Optical Character Recognition (OCR), and displays the results on a web-based dashboard. The proposed system uses OpenCV for image processing, EasyOCR for text recognition, Flask for backend integration, and MySQL for data storage. Detected license plate numbers are stored along with timestamps and corresponding images, enabling efficient tracking and future analysis. The system is designed to be accurate, user-friendly, and suitable for real-time surveillance applications such as parking management and traffic monitoring.

Keywords: License Plate Recognition, OpenCV, EasyOCR, Flask, Computer Vision, OCR

I. INTRODUCTION

The increase in the number of vehicles has created challenges in traffic management, parking control, toll collection, and security monitoring. Traditional manual methods of recording vehicle numbers are time-consuming and prone to human error. Automated License Plate Recognition (LPR) systems aim to overcome these limitations by automatically detecting and recognizing vehicle license plates from images or video streams.

Recent advancements in computer vision and OCR technologies have enabled the development of efficient real-time recognition systems. By combining image processing techniques with web-based applications, LPR systems can provide instant results and centralized data storage. This paper proposes a real-time LPR system that integrates OpenCV, EasyOCR, Flask, and MySQL to achieve accurate and efficient license plate recognition.

II. RELATED WORK

Several studies have explored license plate recognition using traditional image processing and deep learning techniques. Early approaches relied on edge detection, morphological operations, and template matching for plate detection and character recognition. More recent methods utilize convolutional neural networks (CNNs) to improve detection accuracy.

However, deep learning-based solutions often require large datasets and high computational resources. In contrast, OCR-based solutions using libraries such as EasyOCR offer a lightweight and efficient

alternative for real-time applications. The proposed system adopts this approach to balance accuracy and computational efficiency while maintaining real-time performance.

III. PROPOSED SYSTEM

The proposed system follows a layered architecture consisting of input, preprocessing, recognition, data management, and output layers.

A. Input Layer

A webcam captures live video streams of vehicles. The user initiates the camera operation through a web-based interface.

B. Preprocessing Module

The captured video is divided into frames, and image preprocessing techniques such as grayscale conversion, noise reduction, and resizing are applied to enhance image quality. License plate regions are localized using OpenCV-based methods.

C. License Plate Recognition Module

The localized license plate images are processed using EasyOCR to extract alphanumeric characters. Data validation techniques are applied to filter incorrect or duplicate plate readings.

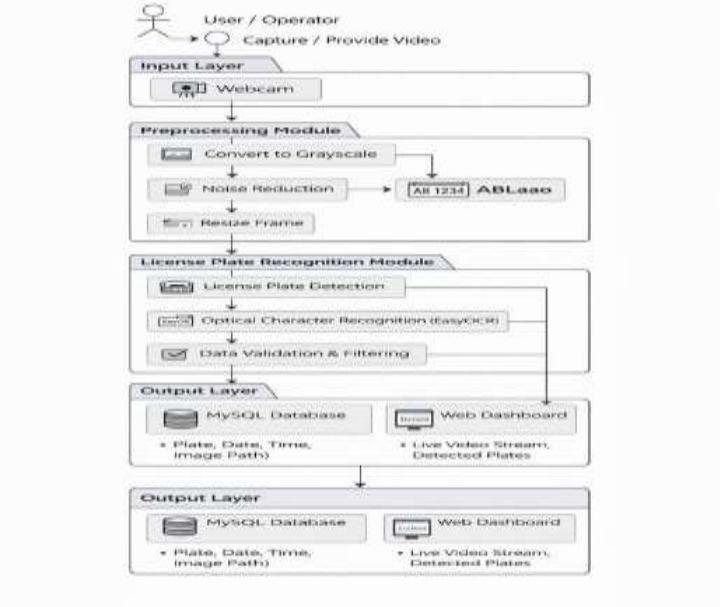
D. Data Management Module

Recognized license plate numbers, along with timestamps and image paths, are stored in a MySQL database. Cropped license plate images are saved in a structured directory for future reference.

E. Output Layer

The results are displayed on a web dashboard that shows the live video feed and detected license plate numbers in real time.

IV. SYSTEM ARCHITECTURE



The system architecture is designed using a modular approach to ensure scalability and maintainability. Each module performs a specific task and communicates with other modules through the Flask backend. This layered design simplifies debugging and future enhancements.

V. METHODOLOGY

The system follows a structured processing pipeline:

- Video Acquisition: Live video frames are captured using OpenCV.
- Preprocessing: Frames are converted to grayscale, denoised, and enhanced.
- License Plate Detection: Plate regions are identified using contour analysis.
- OCR Processing: EasyOCR extracts alphanumeric characters with confidence scores.
- Data Validation: Duplicate detections and low-confidence results are filtered.
- Database Storage: Valid records are stored with timestamps.
- Web Visualization: Results are displayed in real time on a web dashboard.

VI. IMPLEMENTATION

The system is implemented using Python as the primary programming language.

- OpenCV: Used for video capture, frame extraction, image preprocessing, and license plate localization.
- EasyOCR: Used for recognizing alphanumeric characters from license plate images.
- Flask: Acts as the backend framework for handling video streaming, API requests, and user interaction.
- MySQL: Used for storing detected license plate data with timestamps.

The frontend is developed using HTML, CSS, and JavaScript to provide an interactive dashboard for users.

VII. RESULTS AND DISCUSSION

The proposed system successfully detects and recognizes license plates in real time under normal lighting conditions. The web dashboard displays live video streams and recognized plate numbers with minimal delay. Experimental results show that the system provides reliable recognition accuracy for standard vehicle license plates. The use of EasyOCR simplifies the recognition process while maintaining acceptable performance.

VIII. EXPERIMENTAL SETUP AND PERFORMANCE EVALUATION

A. Experimental Setup

The proposed License Plate Recognition system was evaluated using live video streams captured through a standard USB webcam with a resolution of 720p. The experiments were conducted in both controlled and semi-controlled environments such as parking areas and entry gates. Vehicles with standard license plate formats were observed under varying lighting conditions, including daylight and moderate indoor illumination. The distance between the camera and vehicles ranged from 2 to 5 meters. The system was executed on a standard personal computer without GPU acceleration, demonstrating its suitability for low-cost hardware deployment.

B. Performance Metrics

To assess the effectiveness of the system, the following performance metrics were considered:

- Recognition Accuracy: Percentage of correctly recognized license plate numbers.
- Processing Time: Average time taken to process each video frame.
- Duplicate Handling Efficiency: Ability of the system to avoid repeated detection of the same license plate within a short interval.
- System Responsiveness: Delay between video capture and display of results on the web dashboard.

C. Performance Results

Experimental observations indicate that the proposed system achieves an average recognition accuracy of approximately 90–92% under normal lighting conditions. The average processing time per frame was around 35–45 milliseconds, enabling near real-time operation. The duplicate filtering mechanism effectively reduced redundant entries, improving database reliability. These results confirm that the system offers a good balance between accuracy and computational efficiency for real-time surveillance applications.

IX. COMPARATIVE ANALYSIS

A comparative analysis was performed between traditional license plate recognition approaches, deep learning-based methods, and the proposed system. Traditional image processing methods show low accuracy under real-world conditions, while deep learning models offer higher accuracy at the cost of increased computational requirements. The proposed system provides a middle-ground solution by offering reliable accuracy with minimal hardware demands.

Table I. Comparison of License Plate Recognition Approaches

Method	Accuracy	Hardware Cost	Real-Time Capability
Traditional Image Processing	Low	Low	No
Deep Learning-Based Methods	High	High	Yes
Proposed System	Moderate–High	Low	Yes

This comparison highlights the practicality of the proposed system for real-world deployment where cost and efficiency are critical factors.

X. APPLICATIONS

- Traffic monitoring and management
- Parking automation systems
- Toll booth automation
- Security surveillance and access control

XI. ADVANTAGES

- Real-time processing
- User-friendly web interface
- Efficient data storage and retrieval
- Low computational requirements

XII. SECURITY AND PRIVACY CONSIDERATIONS

The proposed system is designed with basic security and privacy considerations in mind. Access to the database is restricted through controlled backend authentication mechanisms. The system focuses exclusively on license plate recognition and does not perform face detection or identification, thereby reducing privacy concerns. All stored data is intended solely for monitoring and administrative purposes, ensuring ethical use of collected vehicle information.

XIII. LIMITATIONS

Despite achieving reliable performance, the proposed system has certain limitations. Recognition accuracy may decrease under extreme lighting variations, motion blur, or when license plates are partially occluded or damaged. OCR-based recognition may occasionally confuse visually similar characters such as 'O' and '0' or 'B' and '8'. Additionally, the system currently relies on a single camera input, which limits its coverage area.

XIV. FUTURE ENHANCEMENTS

Future work may include integrating deep learning-based plate detection, supporting multiple cameras, improving accuracy under challenging conditions, and deploying the system on cloud platforms for large-scale applications.

XV. CONCLUSION

This paper presented a real-time license plate recognition system using OpenCV, EasyOCR, Flask, and MySQL. The system effectively automates vehicle identification by detecting and recognizing license plates from live video streams. Its modular architecture, real-time performance, and web-based interface make it suitable for practical surveillance and monitoring applications.

REFERENCES

1. C. Anagnostopoulos et al., "A License Plate Recognition Algorithm for Intelligent Transportation Systems," IEEE Trans. Intelligent Transportation Systems, 2006.
2. S. Du et al., "Automatic License Plate Detection Using Color Edge Information," IEEE Trans. Intelligent Transportation Systems, 2013.
3. R. Laroca et al., "A Robust Real-Time Automatic License Plate Recognition System," IEEE Trans. Intelligent Transportation Systems, 2018.
4. S. Silva and C. Jung, "Real-Time License Plate Detection Using YOLO-Based Networks," IEEE Access, 2020.