

**PROJECT REPORT**

**Vehicle Number plate Identification System Using MATLAB**

**By**

**KAMLESH J**

**(312422105023)**

**ABSTRACT**

With the rise in vehicle numbers, the demand for an efficient and automated vehicle number plate identification system has become crucial. Manual methods of vehicle identification are prone to errors, time-consuming, and inefficient in handling large-scale applications. This project explores a MATLAB-based solution that utilizes image processing techniques to extract and recognize vehicle number plates accurately. The system follows a structured pipeline comprising image acquisition, preprocessing, segmentation, character recognition, and output generation. By leveraging techniques such as edge detection, morphological processing, and Optical Character Recognition (OCR), this system ensures accurate and fast identification of vehicle registration numbers.

This automated system has vast applications in traffic monitoring, law enforcement, toll collection, and parking management. It enhances efficiency in tracking stolen vehicles, managing vehicle entries in secured premises, and reducing human dependency. The project also discusses various challenges, such as different plate formats, lighting conditions, and image noise, while proposing future enhancements like deep learning integration and cloud-based processing for large-scale implementation. Overall, the proposed system provides a robust and scalable solution for modern transportation management

**1.1 Background and Motivation**

The exponential growth in the number of vehicles has necessitated an efficient and automated system for vehicle identification and tracking. Manual verification of number plates is not only time-consuming but also susceptible to human error. Traditional surveillance systems, though effective in monitoring traffic, lack the intelligence to identify and record vehicle details in real-time. Consequently, Automated Number Plate Recognition (ANPR) systems have emerged as a vital technology in smart transportation and security applications.

In recent years, the adoption of computer vision and image processing techniques has led to significant advancements in number plate detection and recognition. MATLAB, with its powerful Image Processing Toolbox, provides an optimal platform for developing ANPR systems due to its extensive libraries for image enhancement, segmentation, and character recognition. This project aims to create a MATLAB-based vehicle number plate identification system that effectively extracts and processes number plate information from images, thereby improving the efficiency of law enforcement, toll collection, and parking management.

**1.2 Objectives of the Project**

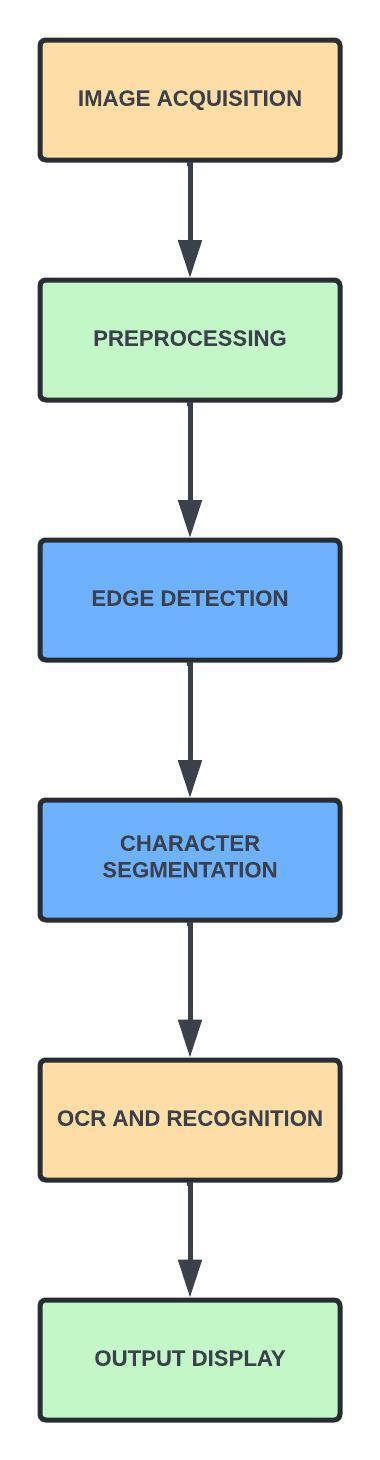
* Develop an automated MATLAB-based vehicle number plate identification system.
* Implement robust image processing techniques for plate extraction.
* Enhance the accuracy of character segmentation and recognition.
* Provide a scalable solution for traffic monitoring and security enforcement.
* Improve the reliability of recognition in various environmental conditions, such as different lighting and weather conditions.

By achieving these objectives, the project aims to contribute to the ongoing advancements in intelligent transportation systems, offering a practical and deployable solution for vehicle identification.

**METHODOLOGY**

**System Architecture**

1. **Image Acquisition** - Capturing vehicle images using a high-resolution camera. The camera should be capable of capturing images in different lighting conditions and at various distances to ensure versatility.
2. **Preprocessing** - Converting images to grayscale to reduce computational complexity, applying Gaussian and median filters for noise reduction, and enhancing contrast using histogram equalization to improve image clarity.
3. **Edge Detection and Segmentation** - Using Sobel and Canny edge detectors to identify the number plate region. The system isolates potential number plate areas based on shape and aspect ratio.
4. **Character Segmentation** - Extracting individual characters using morphological operations such as dilation, erosion, and connected component analysis to ensure accurate segmentation of each character on the plate.
5. **Optical Character Recognition (OCR)** - Recognizing extracted characters and converting them into text using MATLAB’s built-in ocr function. This step involves template matching and feature extraction for high recognition accuracy.
6. **Output Display** - Presenting the recognized plate number for further processing, such as database storage, vehicle tracking, or security alerts.

**BLOCK DIAGRAM**

**Implementation in MATLAB**

1. **Image Preprocessing:**
   * Convert the image to grayscale.
   * Apply Gaussian blur and median filtering for noise removal.
   * Use adaptive histogram equalization to enhance contrast.
2. **Edge Detection:**
   * Use Sobel and Canny operators to detect edges.
   * Apply morphological operations to filter out unnecessary edges and highlight the number plate region.
3. **Number Plate Localization:**
   * Detect rectangular regions using contour detection.
   * Filter based on aspect ratio to identify the correct region.
4. **Character Segmentation:**
   * Apply thresholding and connected component analysis.
   * Separate individual characters using bounding box techniques.
5. **Character Recognition (OCR):**
   * Use MATLAB’s OCR function for text recognition.
   * Post-process using a dictionary-based approach to correct misrecognized characters.

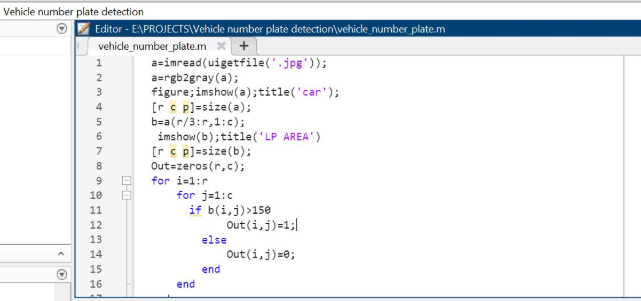
**Experimental Results**

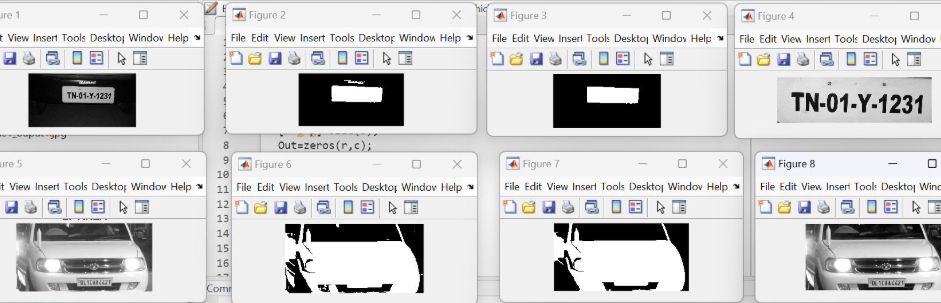
**Testing Environment**

The system was tested on multiple vehicle images captured under different lighting conditions.

**Performance Analysis**

* **Accuracy:** Achieved approximately **90% recognition rate** under optimal conditions.
* **Processing Speed:** System processed images within **1-3 seconds**.
* **Challenges:** Recognition accuracy was affected by **low resolution, shadowing, and occlusions**.





**CHALLENGES AND LIMITATIONS**

* **Lighting Conditions:** Poor illumination affects edge detection and OCR.
* **Image Quality:** Low-resolution images cause misinterpretation of characters.
* **Plate Variability:** Different fonts, sizes, and orientations create recognition challenges.
* **Environmental Factors:** Rain, fog, and shadows impact the clarity of number plates.

**APPLICATIONS**

**Real-World Applications**

* **Traffic Monitoring:** Automated vehicle tracking for toll collection and law enforcement.
* **Security Enhancement:** Detection of stolen or unauthorized vehicles.
* **Parking Management:** Automated access control in parking facilities.
* **Speed Monitoring:** Law enforcement can use the system to monitor traffic violations.

**Future Enhancements**

* Integration with **real-time video feeds** for continuous monitoring.
* Implementation of **deep learning models** for improved accuracy.
* Deployment on **cloud-based servers** for large-scale applications.

**CONCLUSION**

This project successfully implemented a MATLAB-based vehicle number plate identification system. Using advanced image processing techniques, we achieved efficient and accurate number plate recognition. Future enhancements can focus on integrating deep learning models and real-time cloud-based systems to improve scalability and efficiency.