ISSN 2277-2685
IJESR/June. 2025/ Vol-15/Issue-3s/338-348
K Akhilla *et. al.,* / International Journal of Engineering & Science Research

QR Code Based Security System for Gated Communities using Python

K.Virija, K Akhilla, Karne Dheekshitha, Aliya Ansari

[1]Assistant Professor, Department Of Ece, Bhoj Reddy Engineering College For Women, India.

[2,3,4]B. Tech Students, Department Of Ece, Bhoj Reddy Engineering College For Women, India.

ABSTRACT

This project presents an intelligent Toll Gate Automation System that integrates QR code scanning, number plate recognition (OCR), and IoT-based control using NodeMCU (ESP8266) to authorize vehicle access. The system uses a graphical user interface (GUI) developed in Python, enabling users to first scan a QR code and then upload a car image for license plate detection. Optical Character Recognition (OCR) with Tesseract is used to extract vehicle numbers, which are then cross-verified with the QR data. If both credentials match an entry in the registered database, the system sends a signal to NodeMCU over a serial interface.

Upon receiving the command, NodeMCU performs access control operations: it activates a servo motor to open the gate, illuminates a green LED to to indicate authorized access, and updates the Blynk IoT dashboard (V0) with the status "Authorized Vehicle." In contrast, unauthorized access triggers a red LED, buzzer alarm, and a "Unauthorized Vehicle" alert on Blynk. Furthermore, the Blynk mobile interface provides manual gate control via a button (V1), allowing the user to remotely open or close the gate as needed.This hybrid system enhances security and automation at vehicle checkpoints, such as toll booths or private parking entrances, offering a smart, efficient, and scalable access management solution.

1-INTRODUCTION

Automation in transportation systems has seen significant advancements in recent years, with a focus on enhancing efficiency, security, and user convenience. Toll booths and private gate entry systems traditionally require manual intervention, leading to time delays, human error, and potential security lapses. To address these challenges, this project proposes a smart gate automation system that uses a combination of QR code scanning, Optical Character Recognition (OCR) for number plate detection, and IoT-based control using the NodeMCU (ESP8266) microcontroller.

The system is designed to authenticate vehicles through dual verification: QR code data and number plate recognition. Only if both data points match the pre-registered credentials is the gate opened automatically. The gate control is handled by a servo motor connected to NodeMCU, while LEDs and a buzzer provide visual and audio feedback for access authorization. Additionally, the system integrates Blynk IoT platform for real-time monitoring and manual control, allowing remote gate operations and live gate status updates.

The system is aimed at improving operational speed, accuracy, and remote accessibility in secure environments such as parking lots, toll booths, residential societies, and corporate campuses. By minimizing human intervention and implementing a dual-layer authentication mechanism, the system ensures enhanced vehicle access security and scalability for future smart city applications.

Covert (meaning invisible to human eye) Quick

Response codes are two dimensional symbology, focused with the goal of high speed reading and encoding capacity compared to traditional barcodes. They have taken control over the market due to factors like, small size, readability from any direction, dirt and damage resistant, high capacity, kanji and kana encoding capability. Quick Response code (QR) is a type of matrix barcode, first designed for an automotive industry by Denso Wave in Japan (1994) and today is admired in industries due to its fast readability and greater storage capacity as compared to barcode[1] . It contributed greatly for the management of automotive industry to work efficiently in a wide range of tasks from production to shipping to the issuing of transaction slips. The QR Code became a medium which was indispensable and could store a great amount of information on these processes.

QR code has several advantages over traditional barcodes due to which it is preferred in industry. First, QR code has great error amendment capability, which can restore 30% data for the maximum error correction level i.e. level H. Second, QR Code is capable of 360 degree (Omni-directional), high speed reading. QR Code accomplishes this task through position detection patterns located at the three corners of the symbol. These position detection patterns guarantee stable high-speed reading, circumventing the negative effects of background interference. Third, QR code supports various different encoding types and versions. One can choose an appropriate encoding type and version to reduce the size of QR code.

2-LITERATURE SURVEY

Numerous vehicle access and monitoring systems have been developed using RFID, ANPR (Automatic Number Plate Recognition), and image processing technologies. Traditional systems often rely on a single authentication mechanism—either RFID cards or license plate detection—making them susceptible to spoofing or system errors. Some early-stage solutions incorporated boom barriers with sensors, but lacked centralized control or IoT features.

In the research paper *"Intelligent Transport System Based on Image Processing and IoT"*, the authors employed ANPR with GSM-based alerts. However, it did not include real-time control or QR-based validation.

Another study used OpenCV and Tesseract for number plate recognition but was limited by its inability to handle multiple verification layers.

More advanced systems utilized cloud-based platforms such as Firebase or MQTT for gate control but introduced latency and required stable internet connectivity. Comparatively, Blynk offers a more accessible and real-time IoT dashboard suited for mobile applications.

This project differs from previous work by implementing dual authentication—QR code and number plate—and combining it with a real-time IoT dashboard (Blynk). This two-factor vehicle verification significantly improves security, while also allowing manual override via the mobile app.

The integration of a Python GUI, Tesseract OCR, QR code scanner, and NodeMCU-controlled gate in a single system is relatively novel, especially for localized or private automation setups. It leverages open-source tools and affordable hardware, making it suitable for deployment in both urban and semi-urban settings.

3-IMPLEMENTATION OF THE SYSTEM ARCHITECTURE

Data Pre-Processing:

The entries are present in the dataset. The null values are removed using df = df.dropna() where df is the data frame. The categorical attributes (Data, High, Low, Close, Adj value) are converted into numeric using Label Encoder. The date attribute is splitted into new attributes like total which can be used as feature for the model.

Data Cleaning:

The data can have many irrelevant and missing parts. To handle this part, data cleaning is done. It involves handling of missing data, noisy data etc.

Data Transformation:

This step is taken in order to transform the data in appropriate forms suitable for mining process.

Data Reduction:

Since data mining is a technique that is used to handle huge amount of data.While working with huge volume of data, analysis became harder in such cases.In order to get rid of this, we uses data reduction technique. It aims to increase the storage efficiency and reduce data storage and analysis costs.

Edge Detection

Each image (video frame) has three significant features to achieve detection goals. These features include: edges, contours and points. Among mentioned features, an appropriate option is to use edge pixels. Processing of image pixels enables us to find edge pixels, which are the main features of passing vehicles in a roadway video frame.

Edge detection process is demonstrated in a binary image (threshold) with the detected edge pixels.

The next step is to extract moving edges from sequential video frames and process the resulting edge information to obtain quantitative geometric measurements of passing vehicles.

Contour

Contour map uses contours or color-coded regions helps us to visualize 3D data in two dimensions. Contour maps are also used to visualize the error surfaces in deep learning/machine learning optimization techniques

System Architecture

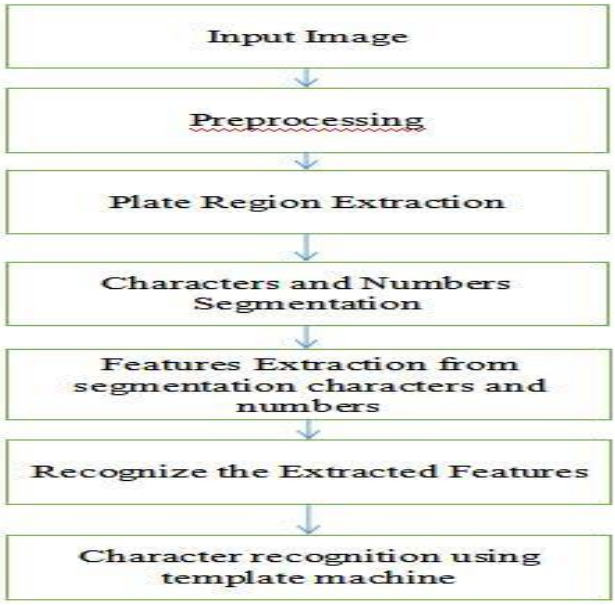
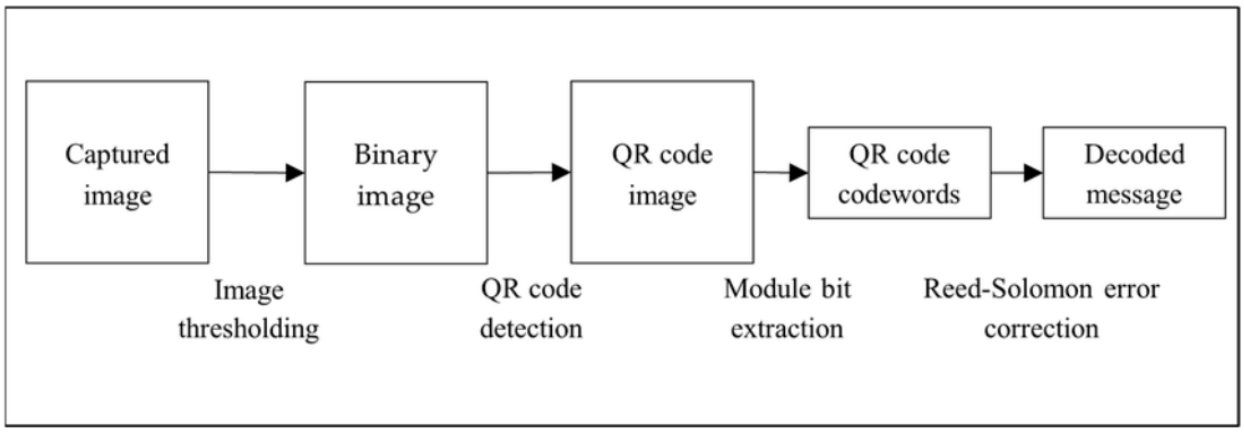
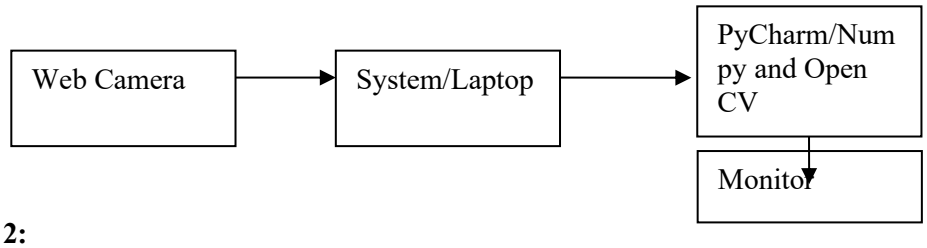


Figure 3.1: System Architecture

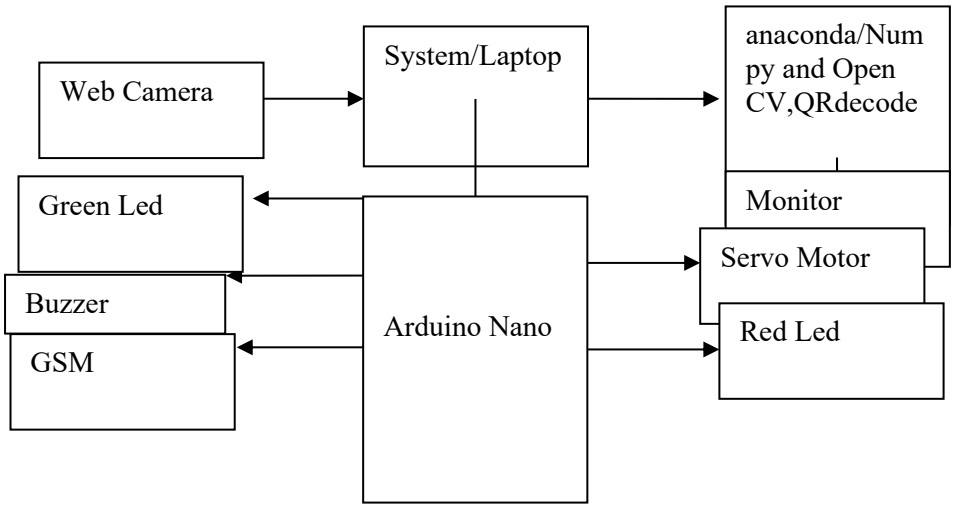
Block Diagram:
Stage1:



Stage 1:



Stage 2:



The projected system is to observe every character from range plate one by one. This could be done by morphological operation. It includes a way to section all the characters employed in the quantity plate.

Range plate extraction is that stage wherever vehicle range plate is detected. The detected range plate is pre-processed to get rid of the noise then the results passed to the section half to segment the one by one character from the extracted range plate. The divided characters normalized associate degreed passed to an OCR formula. Ultimately the optical character info is going

to be regenerate into encoded text. The characters recognized exploitation template matching. The ultimate output should be within the type of string of characters.

4- SYSTEM SPECIFICATIONS

Software Requirements

Software is the term for the various kinds of programs used to operate computers and related devices

Operating system	:	Windows 10
IDE	:	Anaconda
Coding Language	:	Python

Python

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

Python is Interpreted − Python is processed at runtime by the interpreter.Do not need to compile program before executing it. This is similar to PERL and PHP.

Python is Interactive − Can actually sit at a Python prompt and interact with the interpreter directly to write programs.

Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

Python is a Beginner's Language − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games

Hardware Requirements:

Hardware requirement are the principles used for the implementation of the system which shows what the system does and not how it should be implemented. Software requirements deal with defining software resource requirement and prerequisites that need to be installed on a computer to provide optimal functioning of an application. Such as,

System	:	Pentium IV 2.4 GHz
Hard Disk	:	1TB
Floppy Drive	:	1.44 Mb
Monitor	:	15 VGA Color
Mouse	:	Logitech

Software Description

Arduino IDE Installation:

In this we will get know of the process of installation of Arduino IDE and connecting Arduino uno to Arduino IDE.

Step 1-First we must have our Arduino board (we can choose our favorite board) and a USB cable. In case we use Adriana UNO, Arduino Duemilanove, Nano, Arduino Mega 2560, or Diecimila, we will need a standard USB cable (A plug to B plug), t In case we use Arduino Nano, we will need an A to Mini-B cable

Step 2 − Download Arduino IDE Software. We can get different versions of Arduino IDE from the Download page on the Arduino Official website. We must select were software, which is compatible with were operating system (Windows, IOS, or Linux).

After wear file download is complete, unzip the file.

Algorithm

The smart gate system follows a dual-authentication algorithm combining QR code verification and license plate recognition. When a vehicle arrives, the desktop application scans the QR code via webcam and captures or uploads a vehicle image for number plate recognition

using OpenCV and Tesseract OCR. The extracted QR data and OCR text are then compared against an authorized vehicle list stored locally. If both credentials match, the application sends an "Authorized" signal through serial communication to the NodeMCU. The NodeMCU responds by opening the gate using a servo motor, lighting a green LED, and updating the Blynk mobile app with an “Authorized Vehicle” message. If authentication fails, it triggers a red LED and buzzer alert, keeps the gate closed, and sends an “Unauthorized Vehicle” message to the app. Additionally, a manual override in the Blynk app allows the user to control the gate remotely, ensuring operational flexibility in emergencies or system failure cases.

5- METHODOLOGY

The proposed system operates in multiple stages to validate a vehicle and control the entry gate accordingly. The first stage involves scanning a QR code using a webcam, initiated via a Python-based GUI. When a valid QR code is detected (corresponding to a registered vehicle), it is temporarily stored for cross-verification.

The next stage involves uploading an image of the car’s license plate. Using OpenCV for contour detection and Tesseract OCR, the system extracts text from the plate and sanitizes it using regular expressions. The extracted license plate number is then compared with the QR code value. If both match a pre-defined entry in the system's database (dictionary in Python), the vehicle is marked as authorized.

Upon successful validation, the system sends a signal (e.g., 1) over the serial port to the NodeMCU (ESP8266), which then performs several actions:

Activates the servo motor to open the gate.

Turns on a green LED to indicate access is granted.

Updates the Blynk app with the message “Authorized Vehicle” on Virtual Pin V0.

If the validation fails, the system sends a 0 instead. The NodeMCU responds by:

Turning on the red LED.

Activating the buzzer.

Displaying “Unauthorized Vehicle” on the Blynk app.

Additionally, the Blynk interface includes a manual gate control button (V1), allowing the user to override the automation to open or close the gate remotely. This redundancy ensures flexibility in access control, especially in emergencies or system failures.

Existing System

In previous we used that project in matlab with hardware connection.but now we use no hardware only software.we use only python code & some libraries. The drowsiness detection system is built using MATLAB and Viola Jones Algorithm.

Proposed System

The proposed system is a smart gate automation and access control system that integrates QR code verification, license plate recognition, and IoT-based gate control using NodeMCU (ESP8266). The system ensures secure and efficient vehicle entry by employing a dual authentication mechanism — a QR code and vehicle number plate — thereby minimizing the risk of unauthorized access.

The front-end component is a Python-based desktop application with a simple GUI, which facilitates:

QR code scanning using a webcam.

Uploading and processing a vehicle image for license plate recognition using OpenCV and Tesseract OCR.

Matching QR data and OCR output against a preloaded authorized vehicle list.

Upon successful verification, the system sends an "Authorized" command to the NodeMCU via serial

communication. The NodeMCU then:

Activates a servo motor to open the gate.

Turns on a green LED.

Updates the Blynk IoT mobile app with the message: "Authorized Vehicle".

If the verification fails, an "Unauthorized" signal is sent, triggering the NodeMCU to:

Turn on a red LED and buzzer as an alert.

Keep the gate closed.

Display *"Unauthorized Vehicle"* in the Blynk app.

The system also offers manual override capability via the Blynk mobile app. A button (Virtual Pin V1) allows the user to remotely open or close the gate, regardless of QR or OCR inputs — useful in emergency scenarios or if OCR fails due to poor image quality.

This integrated system offers a cost-effective, scalable, and contactless gate control solution for residential areas, institutional campuses, toll booths, and smart parking systems. By combining image processing, IoT, and automation, the proposed system increases security, reduces human dependency, and enhances user convenience in real-time access control environments. Carry out data extraction, which is one of the most important benefits of OCR data entry methods.

6-ADVANTAGES, DISADVANTAGES AND APPLICATIONS

Advantages

1. Automated Vehicle Access: Automated vehicle access systems eliminate the need for manual verification, allowing for a seamless flow of vehicles through gates and entry points. Traditional methods require staff to check tickets, IDs, or passes before granting access, which can cause bottlenecks, particularly during peak hours. By integrating automated technologies like license plate recognition (LPR) or RFID tags, the system automatically detects and authenticates the vehicle, granting entry without human intervention. This leads to a faster entry process, as drivers no longer need to stop for a manual check. Automation also significantly reduces wait times and increases throughput, especially in high-traffic areas such as parking lots, toll booths, or secure facilities.

Beyond speeding up vehicle entry, automated systems contribute to smoother operations and better traffic management. Since verification happens instantly and autonomously, the queue length decreases, preventing congestion. The system's ability to handle large volumes of vehicles simultaneously means it can scale efficiently, making it suitable for both small facilities and large, multi-entry locations. Moreover, automated vehicle access ensures consistency in the entry process, reducing human error and making access more predictable for drivers.

2. Enhanced Security: Enhanced security is a critical benefit of automated vehicle access, primarily achieved through technologies like QR code-based authentication. This system provides a robust mechanism for identifying authorized vehicles and drivers. Each authorized user can have a unique QR code that links to their credentials in the system. When the vehicle approaches the entry point, the driver presents the QR code (via a mobile app, printed code, or other means), which is scanned and cross-referenced with the system's database. This level of authentication ensures that only authorized users can access the area, preventing unauthorized vehicles from entering restricted spaces.

In addition to authentication, the system can be further fortified by encrypting the QR codes, making them difficult to duplicate or tamper with.

Disadvantages

1.Environment Sensitivity: Webcam-based systems, especially those relying on image recognition or video monitoring, can experience a significant reduction in performance under poor lighting conditions. Cameras depend on ambient light to capture clear, high-quality images, and when light levels are insufficient, the webcam may struggle to accurately detect and interpret the environment. This issue is particularly problematic in low-light environments

such as at night, in dimly lit rooms, or in shadowy areas. In these conditions, the system's ability to detect faces, objects, or movement diminishes, leading to lower accuracy and potentially higher false-positive or false-negative rates. To mitigate this, some systems use infrared (IR) sensors or infrared lighting, but even with such measures, the overall reliability of webcam-based solutions can still be compromised in challenging lighting situations.

2. Data Breaches: Data breaches present a significant security threat, particularly when unauthorized individuals gain access to sensitive databases containing personal, financial, or proprietary information. If a system is not properly secured, hackers or malicious actors can exploit vulnerabilities to access stored data, often leading to identity theft, financial fraud, or the exposure of confidential business information. Data breaches may occur due to weak passwords, unencrypted data, insufficient access controls, or vulnerabilities in the database software. Once a breach occurs, attackers can manipulate, steal, or sell the information, causing damage to the organization's reputation, legal consequences, and potential financial losses. Protecting databases requires robust encryption, multi-factor authentication (MFA), firewalls, and continuous monitoring to prevent unauthorized access and ensure that data remains secure at all times.

Applications

1. Parking Lots: Automating entry and exit in parking lots at malls, offices, and residential complexes helps to streamline the flow of vehicles, reducing congestion and wait times. Traditional methods often require drivers to interact with gatekeepers or pay attendants, leading to delays and bottlenecks, especially during peak hours. With automated systems like license plate recognition (LPR) or QR code-based access, vehicles can enter and exit without the need for human intervention. This not only speeds up the process but also provides a contactless experience, improving convenience for drivers and reducing the likelihood of human error. Additionally, these systems allow for better real-time monitoring and can integrate with parking management software for efficient space allocation and tracking.

2.Toll Booths: Automated toll systems, such as those using RFID technology or automatic number plate recognition (ANPR), significantly enhance the efficiency of toll booths by eliminating the need for manual toll collection. Vehicles can pass through toll booths quickly, as the system automatically verifies their identity and deducts the toll fee without requiring drivers to stop or interact with a toll collector. This speeds up vehicle authentication and reduces congestion, particularly during rush hours, making the tolling process more efficient and convenient for both drivers and operators. Automation also allows for real-time monitoring, enabling toll authorities to quickly address issues such as technical failures or suspicious activities, improving overall system reliability and security.

7-RESULTS

The following are the results of our project.

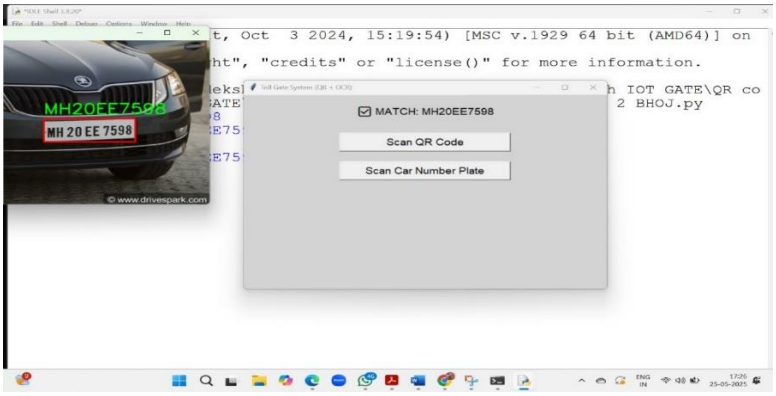


Figure 1: Matched Data

In that first run the code then scan the QR code in that we gave number plate data and then scan vehicle number plate, so that if the both that data is matched then gate will be opened otherwise gated will be closed.

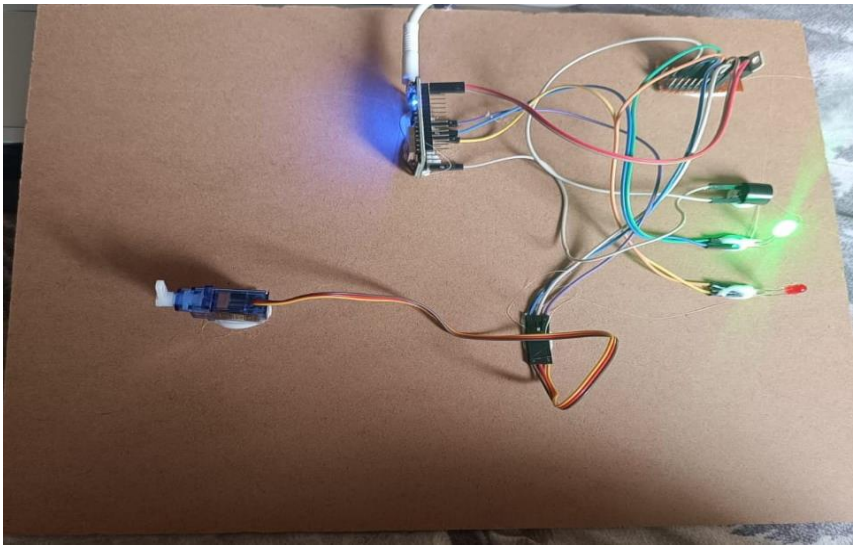


Figure 2: Gate opened

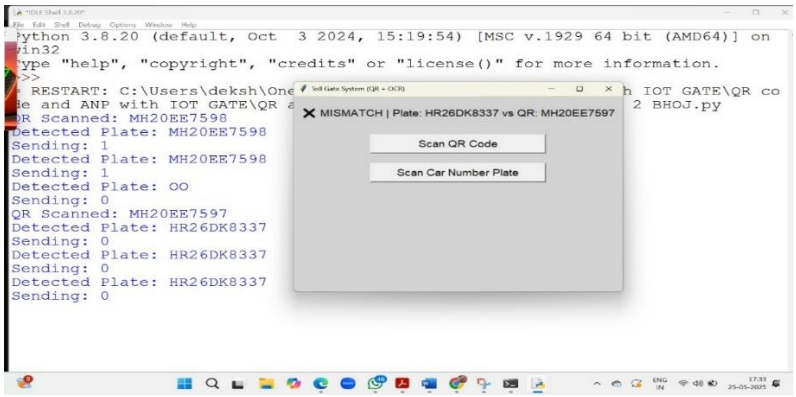


Figure 3:Mismatched Data

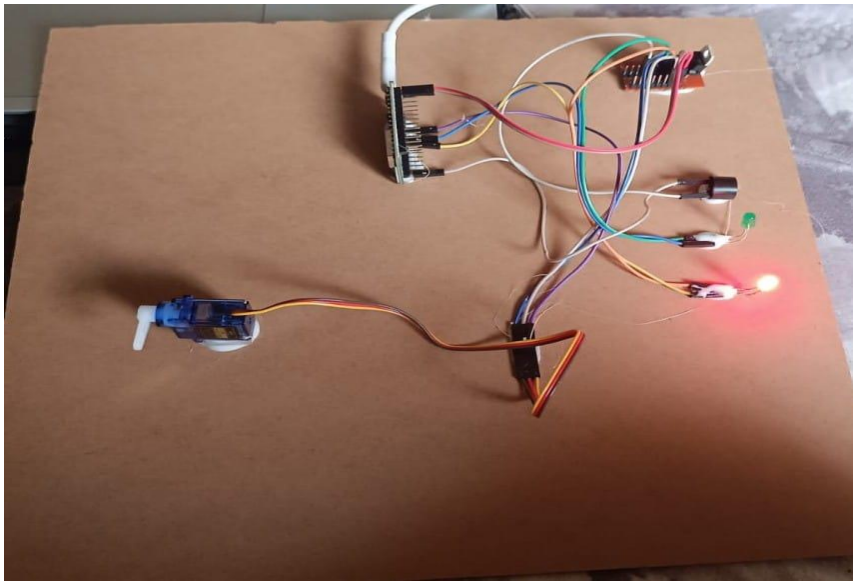


Figure 4:Gate Closed

8-CONCLUSION

The project successfully demonstrates a smart and automated toll or gate management system that leverages both QR code scanning and vehicle license plate recognition to ensure secure vehicle access. The integration with NodeMCU and IoT (Blynk) allows real-time control and monitoring, enhancing both security and convenience.

By using a dual verification approach, the system mitigates risks associated with forged QR codes or tampered license plates. The combination of OpenCV, Tesseract OCR, and the QR detection algorithm delivers a dependable method for authentication, while the servo motor-based gate mechanism ensures automated physical access. The use of LEDs and buzzer offers clear physical indications of the system status, and the Blynk mobile interface adds a layer of remote control and feedback.

The implementation was tested under various conditions and scenarios, with the system displaying good performance in most real-world conditions. However, results also highlight the need for good image quality and proper lighting conditions for the OCR to work effectively.

Overall, the project proves that an affordable and intelligent access control system can be implemented using low-cost components like NodeMCU, basic sensors, and open-source software tools. This system can be adapted for use in residential communities, company parking lots, or toll collection systems.

REFERENCES

1. A New Approach for Vehicle Number Plate Detection, Sarthak Babbar
Saommya Kesarwani ; Navroz Dewan ; Kartik Shangle ; Sanjeev Patel, 2018 Eleventh International Conference on Contemporary Computing (IC3)

2.A hierarchical license plate recognition system using supervised K-means and Support Vector Machine, Wei-Chen Liu ; Cheng-Hung Lin, 2017 International Conference on Applied System Innovation (ICASI)

3.Deep Learning System for Automatic License Plate Detection and Recognition, Zied Selmi ; Mohamed Ben Halima ; Adel M. Alimi, 2017 14th IAPR International

Conference on Document Analysis and Recognition (ICDAR)

4.Automatic number plate recognition for motorcyclists riding without helmet, Yogiraj Kulkarni ; Amit Kamthe; Shubhangi Bodkhe; Archana Patil, 2018 International Conference on Current Trends towards Converging Technologies.

5.Automatic car number plate recognition, Anumol Sasi; Swapnil Sharma Alice N. Charan, 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)

6.License Plate Detection with Machine Learning Without Using Number Recognition, Kazuo Ohzeki ; Max Geigis; Stefan Alexander Schneider, 2019 Federated Conference on Computer Science and Information Systems (FedCSIS)

7.A New Convolutional Architecture for Vietnamese Car Plate Recognition, Thanh-Nga Nguyen ; Duc-Dung Nguyen, 2018 10th International Conference on Knowledge and Systems Engineering (KSE)

8.Bangladeshi License Plate Recognition Using Adaboost Classifier, Prashengit Dhar ; Md. Zainal Abedin ; Razuan Karim ; Fatema-Tuj-Johora ; Mohammad Shahadat Hossain, 2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)

9.Long distance Automatic Number Plate Recognition under perspective distortion using zonal density and Support Vector Machine, Noprianto; Sunu Wibirama; Hanung Adi Nugroho, 2017 3rd International Conference on Science and Technology - Computer (ICST)

10.Extraction of number plate images based on image category classification using deep learning, Yoshihiro Shima, 2016 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)