ANALYSIS OF VEHICLE & PARKING MANAGEMENT SYSTEM

Shanigdh Raj, **Utkarsh Goel, Ritu Agarwal, Satyam Mishra**, Department Of Electronics And Communication Engineering, Abes Engineering College Ghaziabad, India

Abstract

Efficient management of parking spaces is essential in modern facilities. This research paperpresents a vehicle authorization and parking management system based on license plate recognition. Implemented with an Arduino Uno microcontroller,the system integrates IR sensors, server motors, Python, and the OpenCV library for image processing and computer vision tasks. IR sensors detect vehicle presence, while an LCD screendisplays real-time information on available parkingspaces. The developed system demonstrates accurate detection and recognition of vehicle number plates in real-world scenarios. It provides real-time updates on parking space availability and effectivelymanages available spaces. In summary, the proposed vehicle authorization andparking management system offers a reliable and efficient solution for optimizing parking space usagein modern facilities. By reducing parking-related stress and enhancing the overall parking experience, it contributes to streamlined parking management in urban environments.

Keywords— OpenCV, Python, Arduino UNO, Cascade, JSON-DB, Image Processing, HAAR, ANPR,Image Processing.

Introduction

Automated parking systems are rapidly gaining popularity in today's world as they offer numerous benefits such as improved parking efficiency and reduced driver stress. These systems play a crucial role in the development of smart city surveillance systems. License plate recognition is a vital component of such systems as it enables the identification of moving vehicles by capturing license plate information. Vehicle authorization, on the other hand, involves the process of verifying the captured image against a database of registered vehicles to determine whether the licensenumber belongs to a known or unknown vehicle.

In our research, we utilized Tesseract, an Optical Character Recognition (OCR) engine, to capture and recognize license plate information [1]. By employingthis technology, we were able to accurately extract thenumber plate number from the captured images, enabling us to authenticate registered vehicles.

The front-end application of our system is tightly integrated with the database, allowing for seamless verification of information related to registered vehicles. Python, a powerful programming language widely used in computer vision applications, waschosen to develop the computer vision and image processing components of our system. Python's versatility and extensive library support make it an ideal choice for handling complex image processing tasks. Our proposed system aims to detect and track vehiclesin real-time, accurately recognize license plate information, and efficiently authenticate registeredvehicles to effectively manage parking spaces. By harnessing these advanced technologies, we can provide drivers with real-time updates on the availability of parking spaces and ensure that only authorized vehicles can access the parking facility. Through the integration of Tesseract, Python, and a robust database system, our research endeavors to enhance the capabilities of vehicle authorization and parking management systems. The utilization of thesecutting-edge technologies holds great potential in revolutionizing the efficiency and security of parking facilities, ultimately contributing to a smoother and more streamlined parking experience for drivers.

Literature review

Automated vehicle recognition and authorizationsystems have become increasingly important in modern security and surveillance applications. Such systems can be used to track and detect vehicles

in real-time, helping to prevent theft and unauthorizedentry into restricted areas. Additionally, they can provide convenience for drivers by allowing forautomatic parking management and allocation of available parking spaces. With advancements in computer vision and image processing technologies, these systems are becoming more accurate and cost- effective. Several studies have been conducted on license plate recognition and vehicle detection using machine learning
algorithms and cascade classifiers [2].However, there is still a need for further research on improving the accuracy and speed of these systems, especially in challenging environments such as low- light conditions and high-speed vehicle movement. In this paper, we propose a novel approach to vehicle recognition and authorization using Python and OpenCV, with a focus on addressing these challenges and improving the overall performance of the system.

There has been a growing demand for automated systems in a variety of applications, including vehicle authorization and parking management, in recent years. Automated parking systems are becoming more common, and their efficient management is heavilyreliant on computerized systems. License plate recognition, which involves reading license plate data from moving vehicles and using this information for classification or other purposes, is a critical componentof these systems.
Numerous parking lot burglaries, vehicle breaches, and other intrusions have gone unreported, emphasizing theneed for a security system that recognizes vehicle license plate numbers. As a result, several researchers have proposed systems that use license plate recognitionto increase security while alleviating problems associated with manual data entry.

The elimination of manual data entry issues is a significant advantage of automated license plate recognition, and the data is secure thanks to database support. Adding or deleting saved data is simple with database management system crud operations. Recognition systems are increasingly being used in the modern world to identify vehicles by capturing and extracting data from images using image processing methods.

A recent study proposed by Paawan Sharma, Mukul K Gupta, Amit Mondal, and Vivek Kaundal, a cascade classifier-based HAAR-like feature-based car key detection system. The classifier was built using OpenCV-Python, which involved both detection and training processes [3]. For training, the authors used a diverse set of object images.

Another study by Sajjad, K.M, Department of ComputerScience and Engineering, MES College of Engineeringproposed using Python and OpenCV to recognize license plates automatically. The proposed system communicated with the camera via an Arduino Uno, andit was tested using OpenCV. Loading an input image, converting it to grayscale, denoising it, and comparing it to database images were all part of the experiment.

While several studies on license plate recognition and automated parking systems have been conducted, thereis still a need for improved accuracy and efficiency in these systems. As a result, the current study proposes asystem that uses cascade classifiers, OpenCV, and Tesseract to improve vehicle authorization and parkingmanagement accuracy [4]. The proposed system's methodology entails using a camera to take a picture ofthe target's number plate data, reading the image into the system using OpenCV, processing the image using various image processing methodologies, and comparing the plate number to information in the database to authorize the vehicle [5]. The proposed system aims to improve parking management and vehicle authorization security, speed, and accuracy. In conclusion, while several studies have proposed systems for automated parking management and vehicle authorization using license plate recognition, there is still a need for improved accuracy and efficiency [6]. The current study aims to close this gap by proposing a system that uses cascade classifiers, OpenCV, and Tesseract to improve accuracy and speedin vehicle authorization and parking management.

Methodology

The proposed vehicle authorization and parking management system follows a systematic approach to efficiently manage parking spaces. The system's methodology is divided into the following steps:

1. License Plate Recognition: The first step in the process is capturing the image of the vehicle's license plate using a camera module. The captured image is then read into the system using OpenCV [7]. The image is processed using various image processing methodologies to extract useful information.
2. Image Processing: To speed up the processing time, we convert the RGB image to grayscale, which eliminates the need for color information. We thenapply the edge detection filter to reduce unwanted dataand speed up the algorithm.
3. Character Detection: The system uses a rectangle detection technique to detect characters on the license plate. The algorithm looks through a rectangle in the image that may contain characters using all four of its edges.
4. Vehicle Authorization: The system compares the number plate information with the registered vehicles' information stored in a database to authorize the vehicle's entry [8]. We used SQLite3 or NoSQL as our database system to store and manage vehicle information.
5. Parking Space Management: The system uses IR sensors to detect the presence of vehicles in parking spaces.
The system keeps track of the number of available parking spaces and displays real-time information on an LCD screen for drivers.
6. Hardware Used: The proposed system uses a cameramodule, IR sensors, an LCD screen, and an Arduino Uno board to manage the parking spaces.
7. Software Used: The proposed system's image processing and computer vision components were implemented using Python and OpenCV. Python was chosen for its ease of use and robust libraries, making it ideal for image processing tasks.

The proposed system employs advanced image processing and computer vision techniques in a step-by-step approach to efficiently manage parking spaces.The system's methodology ensures that only registered vehicles can access the parking facility while also providing drivers with real-time information on the availability of parking spaces.

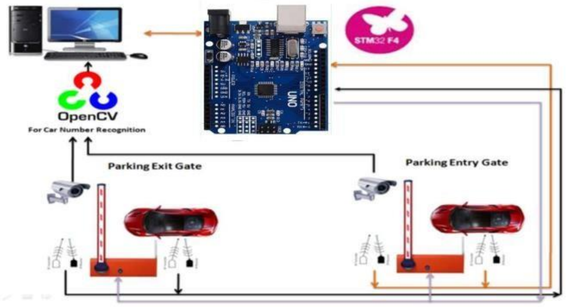


Fig 1: Block diagram of Vehicle Authorization Process

At the core of the block diagram is the Arduino Uno microcontroller, which serves as the central control unit. It is connected to various hardware components, including IR sensors for vehicle detection, an LCD screen for real-time information display, and motor control mechanisms for gate or barrier operations.

The system incorporates the OpenCV library for imageprocessing, vehicle detection, and license plate recognition. OpenCV algorithms receive input from the IR sensors and process the captured images to identify vehicles and extract license plate information [9]. The recognized license plate data is then sent to thevehicle authorization module for verification.

The vehicle authorization module interacts with the database, which stores information about registered vehicles and authorization details. Upon successfulverification, the authorization module signals the motor control mechanism to open the gate or barrier, allowing the authorized vehicle to enter the parking area.

The LCD screen is synchronized with the system and displays real-time information about parking spaceavailability. This information is continuously updated based on the data received from the IR sensors and thedatabase.

Overall, the block diagram provides a visual representation of the system's architecture and highlights the interactions between its major components. It showcases how the different modules work together to achieve efficient vehicle authorization, parking management, and real-time information dissemination.

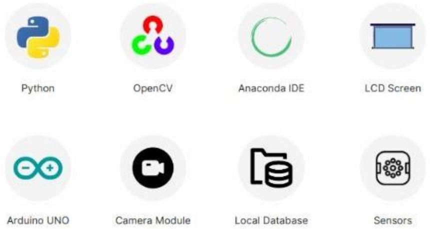


Fig 2: Materials and Tools Used

The software and hardware tools and technologiesutilized in the system are discussed as follows:

LCD Display

An LCD is a type of electronic display module that produces a visible image by using liquid crystal. The 16*2display shows the number of vacant spots. It is also updated on the display LCD when a vehicle parks or unparks.

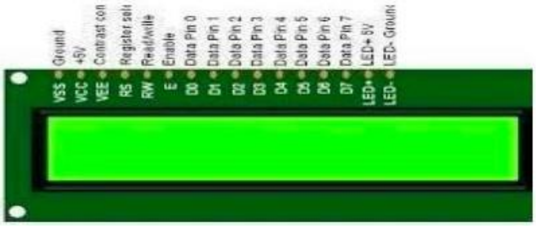


Fig 3: 16x2 Display

IR sensor

An infrared (IR) sensor is a piece of electronic equipmentthat measures and detects infrared radiation in its surroundings. In our project, we use three IR detect sensors: one to sense the vehicle near the parking sensor, and the other two to send data to the node MCU, which isthe brain of our system, to determine whether a vehicle isparked in that slot or not.

Fig 4: Infrared Sensor

Python

Python has built-in libraries for image processing, suchas OpenCV. Python, with only minor changes to the fundamentals of coding, is one of the most versatile andwell-known options for a wide range of platforms and technologies. Python is platform agnostic. Building reliable AI and ML models and systems is simple due to the language's simplicity. As a result, selecting this technology becomes one of the best options.

OpenCV

OpenCV is a Python open-source library for image processing, AI, and machine learning. It is free for academic and business purposes. It is compatible with Windows, Linux, Mac OS, iOS, and Android, and its interfaces are written in C++, C, Python, and Java. It was designed to be computationally efficient, with a strong emphasis on real-time applications. As a result, it's ideal for real-time detection with a camera. It can analyze images and videos to find objects, people, and so on.

Arduino Uno

The Arduino Uno is a small, low-cost microcontroller board with a flexible and user-friendly interface for connecting to various electronic components. It can be used to power the project's motors and sensors.

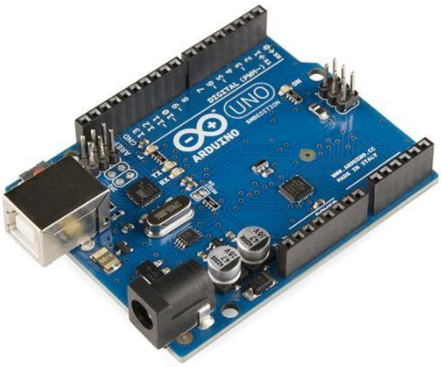


Fig 5: Arduino Uno

NoSQL/SQL Database

A database is a structured collection of data that can bestored and read electronically. Databases are classified into two types: NoSQL databases like MongoDB and SQL databases like MySQL. We also have an SQLite library, which does not require a separate server and canbe easily integrated into programs.

Computer Vision

The goal of computer vision is to develop automated systems that can interpret visual data (such as pictures or videos) in the same way that humans do. The goal ofcomputer vision is to teach computers how to interpret,process, analyze, and comprehend data on their own.

Image Processing

Image processing is the process of converting an imageto a digital format and then performing specific operations on it to extract useful information. The image processing system typically treats all images as 2D signals when using specific predetermined signal processing techniques.

The following steps are used in the image processing:

1. Capture the image from the camera into the system.
2. Convert the image to grayscale to simplify processing.
3. Apply a Gaussian blur to smooth the image and reduce noise.
4. Use an edge detection algorithm to identify the edges of objects in the image.
5. Apply morphological transformations [10], such as dilation and erosion, to improve the quality of the edges and remove unwanted noise.
6. Find contours in the image to identify individual objects.

7. Filter the contours to select only those that couldpotentially contain license plate characters.
8. Extract the individual characters from the selected contours.
9. Use optical character recognition (OCR) softwareto recognize the characters and convert them to text.
10. Compare the recognized text with the database ofregistered vehicles to determine if the vehicle is authorized to enter.
11. If the vehicle is authorized, send a signal to the Arduino to open the gate and display a message on the LCD screen.
12. If the vehicle is not authorized, display a messageon the LCD screen and do not open the gate.

Ehicle authorization process

Detection, authorization, and parking management are the three divisions of this process. The following processes break these three down into their individual components.

Data collection

Datasets are gathered from the camera module. These datasets, which contain information about vehicles, areused for number plate detection.

Grayscale conversion

The RGB images comprise of millions of shades, whereas grayscale images are synonymous. We changed the RGB photos to black and white so that ouralgorithm can operate more quickly [11]. With these, we don't need to worry about color information.

Number plate detection

The application of an edge detection filter speeds up ouralgorithm even more. Haar cascade [12] is then used to train the system to better detect the plates. The filters also assist in eliminating unnecessary data.

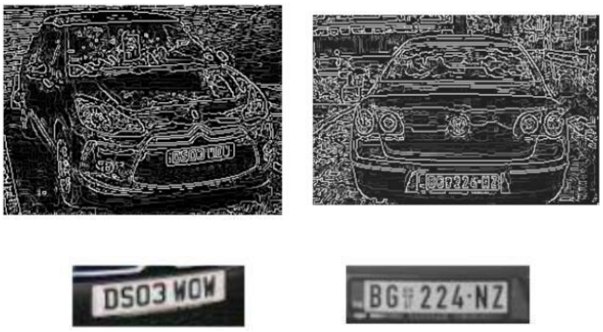


Fig 6: Number plate detected

Image processing & segmentation

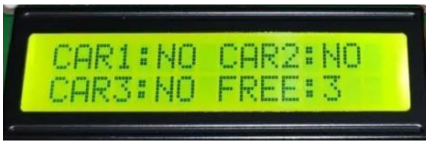
Operations are carried out after an image is used as input.Finding each contour in the input image is the process ofsegmentation. These pictures are now compared to shapes like rectangles and polygons. We write the command to find the rectangle shape, as our license platecan be in this manner.

Data authorization

After retrieving the data, we check our system to see if the input information pertains to a registered vehicle or not. If so, the user is allowed access to the system; otherwise, not. The administrator has the authority to use the crud operations to add or remove user data fromthe database.

Parking management

When a vehicle parks in a specific location, our system prompts it to fill the space. We use infrared sensors to determine whether a car is parked. An LCD screen will show the number of available parking spaces.



```
CAR1:NO CAR2:NO
CAR3:NO FREE:3
```

Fig 7: Parking Spaces notified on screens

1. At the parking spaces, we install infrared sensors.
2. It allows us to determine whether parking isavailable.
3. The availability of parking spaces is displayed onour LCD screen.

Algorithms

To locate license plates, the algorithms train the machineto recognize patterns [13]. The vehicle recognition and tracking method investigated in this paper employs the Haar classifier for recognition. The classifier is superimposed after numerous trainings to create a powerful classifier, and the recognition results areoptimized.

The image preprocessing algorithm is the first in the system. This algorithm uses various image processing techniques to prepare the input image captured by the camera for further analysis. The algorithm does things like converting the image to grayscale, applying noise reduction filters, and detecting edges [14]. These steps aidin the simplification of the image and the extraction of important features that can be used for further analysis.

The license plate detection algorithm is the next algorithmin the system. This algorithm determines the location of the license plate in the preprocessed image. It scans the image for areas that are likely to contain a license plate based on features like color, shape, and texture. When a license plate is detected, it is isolated and forwarded to thenext algorithm for further analysis.

The license plate recognition algorithm is the system's third algorithm. This algorithm extracts the alphanumericcharacters from the isolated license plate image. It worksby segmenting the characters and then matching the segmented characters to a database of known license platecharacters using pattern recognition techniques [15]. This algorithm is crucial in the vehicle authorization process because it determines whether the vehicle is permitted topark in the parking lot.

Finally, to keep track of available parking spaces, the system employs a parking space management algorithm. Based on check-ins and check-outs from the parking lot, this algorithm updates the number of available parking spaces in real-time. It monitors the number of parked vehicles and available parking spaces and displays this data to drivers in real time.

Overall, the algorithms in the vehicle authorization and parking system are critical for automating the parking process and ensuring that only authorized vehicles can park in the lot. These algorithms can be further optimizedand improved to improve the system's accuracy and efficiency.

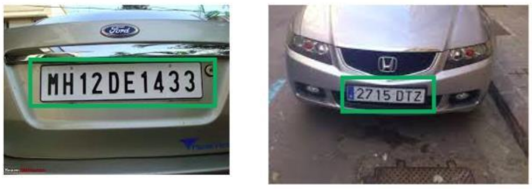


Fig 8: Number plate detected using OpenCV

Conclusion

Finally, the study presented a strong car authorization andparking management system based on license plate recognition. The technology successfully detected andrecognized car license plates, ensuring secure access for registered vehicles. The system achieved reliable license plate detection and verification by incorporating advancedalgorithms and technologies such as Cascade classifiersand

Tesseract. This improved security and efficiency in parking space management.

The system's implementation with an Arduino Uno microcontroller, IR sensors, and the OpenCV library demonstrated successful hardware and software integration. An LCD panel gave real-time updates on parking space availability, allowing drivers to make informed selections. The system's performance confirmed its usefulness and promise for real-world use. The research article also emphasized the necessity of Python and OpenCV for image processing and computer vision tasks in the system's development.

Overall, the proposed vehicle authorization and parkingmanagement system provides a dependable and effective solution for enhancing parking space utilization and overall parking experience. Further studycan investigate further improvements, such ascombining machine learning techniques and cloud-based data storage, to improve the system's capabilitiesand usability.

Finally, the created system has the potential to transformparking management by providing a secure, efficient, and user-friendly solution for modern parking facilities.

References

[1] Milan Samantaray, Anil Kumar Biswal, Debabrata Singh, Niju P Joseph, “Optical Character Recognition (OCR) based Vehicle's License Plate Recognition System Using Python and OpenCV”, Published in January 2022.

[2] MZ, Liu, “Vehicle recognition based on Haar features and Adaboost cascade classifier”, Journal of Physics Conference Series, published in July 2022, License: CC BY 3.0.

[3] Muhammad Abdurrahman, Haryadi Amran, “Face Recognition for Prevention of Car Theft with Haar Cascade and Local Binary Pattern Histogram using Raspberry Pi” , Akuwan Saleh's Lab, Published in 2020.

[4] Andrew Selasi, Jepthah Yankey, Ernest O, “An Automatic Number Plate Recognition System usingOpenCV and Tesseract OCR Engine”, International Journal of Computer Applications, Published in 2018.

[5] Martins Ugwu, Onate Egerton Taylo, Nuka Nwiabu, “An Improved Visual Attention Model for Automated Vehicle License Plate Number RecognitionUsing Computer Vision”, License: CC BY-NC 4.0,Published in May 2022.

[6] V Keerthi Kiran, Priyadarsan Parida, “Vehicle Detection and Classification: A Review”, Machine Intelligence Lab Published in 2022.

[7] M.N Tondra, Ebrahim Karami, “Vehicle detection and tracking”, Published in 2021.

[8] Phyu Phyu Khaing, Myint San, “Study for License Plate Detection”, published in International Journal of Image, Graphics and Signal Processing, 2019.

[9] Syed Hasan Adil, Syed Saad Azhar Ali, Mohsin Shahzad, “Towards Automatic License Plate Detection”,License: CC BY 4.0, Published in February 2022.

[10] S. Obulakshmi, M. Ramya, J. Mythili, “License plate detection using AI”, International Journal of HealthSciences, 2022, License: CC BY-NC-ND 4.0.

[11] Hadi, RA, Sulong, George, “Vehicle detection andtracking techniques: a concise review”, 2014.

[12] H. Rabiu, “Vehicle detection tracking and color- based classification in video”, International Journal of advanced and innovative Research (IJAIR), vol. 2, 2013.

[13] Kyung-Soo Lim, Seoung-Hyeon Lee, Jong Wook Han, Geon-Woo Kim, “Design considerations for an intelligent video surveillance system using cloud computing”, 2019.

[14] N Vishwanath, S Somasundaram, MR Rupesh Ravi, N Krishnan Nallaperumal, “Connected component analysis for Indian license plate infra-red and color imagecharacter segmentation”, IEEE, 2012.

[15] Rohit Tiwari, Dushyant Singh, “Vehicle Control Using Raspberry pi and Image Processing”, Innovative Systems Design and Engineering, Published in 2017.