**REPORT**

#### Problem Statement

The movement of vehicles within a college campus can significantly impact safety, traffic flow, and overall campus operations. The objective of this project is to analyze vehicle movement and generate insights using Edge AI to help manage vehicle traffic more efficiently, enhance security, and improve the overall campus experience.

#### Introduction

Efficient management of vehicle traffic within a college campus is crucial for ensuring safety, optimizing traffic flow, and enhancing overall campus operations. With the rapid advancement of technology, Edge AI provides a powerful tool to monitor, analyze, and manage vehicle movements in real-time. By leveraging computer vision, OCR, and data analytics, this project aims to provide a comprehensive solution for vehicle movement analysis and insight generation.

We need to collect image of data and make the dataset. Monitor the image processor make note of letter which are extracting by the processor. Check whether complete number is extracting or any number are letters are missing in the number plate.

In the image processor code, the tesseract OCR place main role to identify the letter and numbers to match with predefined alphabets and numbers. The image converted into multiple colours and then extracted the number plate.

After number plate extracted then it is compared with database and if it matches then it allowed if not, they won’t allowed. If there is no space to park the cars then no car is allowed inside.

The excel file is created the data is store after excel number plate, entry date, exit date, time, no of hours spent inside, which slot the vehicle is parked inside. Every time when we run the code the excel will be rested.

Pattern is identify based on the no of slots car parked. Based on the excel we can get the entry and exit time pattern. That are peak times of the entry and exit.

**Objectives**:

1. **Enhance Campus Safety:**
   * + Use OCR to detect and recognize vehicle number plates.
     + Cross-check detected number plates against a pre-registered database.
     + Provide immediate alerts for unauthorized vehicles attempting to enter.
2. **Optimize Traffic Flow:**
   * + Dynamically assign available parking slots to entering vehicles.
     + Track and update the status of each parking slot in real-time.
     + Prevent entry when all parking slots are occupied, avoiding congestion.
3. **Efficient Parking Management:**
   * + Log entry and exit times of vehicles to keep track of parking duration.
     + Maintain a real-time status of each parking slot, indicating whether it is occupied or available.
4. **Real-Time Monitoring and Alerts:**
   * + Display real-time status of vehicles on entry, including visual overlays and audio alerts.
     + Generate real-time notifications for unauthorized entries and full parking conditions.
     + Ensure real-time updates of the parking status on a centralized platform.
5. **Comprehensive Logging and Reporting:**
   * + Record detailed logs of each vehicle's entry and exit, including timestamps and duration.
     + Provide visual representations, such as charts and graphs, to highlight parking slot status and trends.
6. **Data-Driven Decision Making:**
   * + Analyze logs and reports to identify peak hours, frequently used slots, and potential improvements.
     + Use insights to optimize parking slot allocation and campus traffic regulations.
7. **Improved Campus Experience:**
   * + Reduce wait times for parking by ensuring availability information is accurate and up-to-date.
     + Provide a seamless entry and exit experience for authorized vehicles.

**Dataset**

**1. Image Dataset of Vehicles**

* **Description:** A collection of images capturing vehicles number plate.
* **Content:**
  + High-resolution images of vehicle number plates.
  + Images captured under various lighting and weather conditions.

**2. Registered Number Plate Database**

* **Description:** A text file containing the list of authorized vehicle number plates.
* **Content:**
  + A simple text file where each line represents an authorized vehicle number plate.

3. **Timestamps**

* **Description:** Timestamps are recorded for each image, representing the time the image was added to the dataset.
* **Content:**
  + Each image file's modification time serves as the timestamp.
  + Timestamps are used to log the entry and exit times of vehicles.

The dataset for this project consists of vehicle images collected from various locations, each with a timestamp indicating when the image was added. The images are preprocessed to enhance number plate detection, which is then achieved using OCR. The detected number plates are verified against a registered database, and the data is used to log and manage vehicle movements within the campus. This comprehensive dataset and its processing enable efficient vehicle movement analysis and insight generation, improving campus safety, traffic flow, and parking management.

**Methods and Tools Used:**

**1. Image Processing and Optical Character Recognition (OCR)**

* **OpenCV (cv2)**: Used for image processing tasks such as reading images, resizing, converting to grayscale, noise reduction, edge detection, and contour finding.
  + cv2.imread(), cv2.resize(), cv2.cvtColor(), cv2.bilateralFilter(), cv2.Canny(), cv2.findContours(), cv2.arcLength(), cv2.approxPolyDP(), cv2.boundingRect(), cv2.putText(), cv2.imshow(), cv2.waitKey(), cv2.destroyAllWindows()
* **Imutils**: Used for resizing the image.
  + imutils.resize()
* **Pytesseract**: Used for Optical Character Recognition (OCR) to extract text (license plate numbers) from images.
  + pytesseract.image\_to\_string()

**2. Data Storage and Manipulation**

* **Pandas**: Used for reading and writing data to Excel files and for manipulating data frames.
  + pd.read\_excel(), pd.concat(), pd.DataFrame()
* **Openpyxl**: Used for working with Excel files, including reading and writing data.
  + load\_workbook(), wb.save(), ws.append()
* **OS module**: Used for file and directory operations such as listing files, checking file existence, and getting file modification times.
  + os.listdir(), os.path.exists(), os.path.isfile(), os.path.getmtime(), os.path.join()
* **Datetime module**: Used for handling date and time operations, including formatting and calculating durations.
  + datetime.fromtimestamp(), datetime.strptime(), datetime.timedelta()

**3. User Alerts**

* **Winsound**: Used to play a beep sound as an alert when a registered vehicle is detected.
  + winsound.Beep()

**4. HTML Generation**

* **String manipulation and file writing**: The code dynamically generates HTML content and writes it to an HTML file to display vehicle status, including the image of the vehicle, its number plate, status, and slot number.

**5. Functions**

* **check\_if\_string\_in\_file(file\_name, string\_to\_search)**: Checks if a specific string (number plate) exists in a text file (database).
* **process\_image(image\_path)**: Processes an image to detect the number plate.
* **format\_duration(duration)**: Formats a duration into "days, hours, minutes".
* **get\_next\_available\_slot(df)**: Determines the next available parking slot.
* **update\_excel(entry\_data, exit\_data=None)**: Updates or adds data to the Excel file, managing vehicle entry and exit records.

**Results**

The results generated by the code can be summarized in terms of vehicle detection, status logging, and visualization. Here's an explanation of each component of the results:

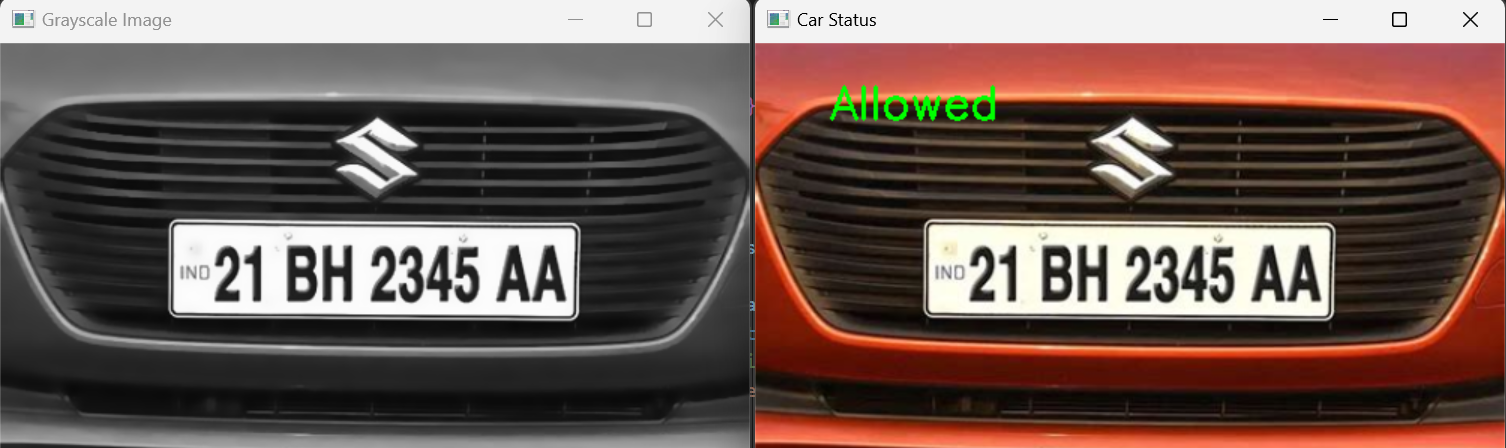
**1. Vehicle Detection**

**Image Processing:**

* The code processes images from the specified directory to detect vehicle number plates.
* **Contours** are used to identify the number plate regions in the images.
* **Optical Character Recognition (OCR)** with Tesseract extracts text from the identified number plate regions.

**Registration Check:**

* The extracted number plate text is checked against a database (text file) to verify if the vehicle is registered.



**2. Status Logging**

**Entry and Exit Logging:**

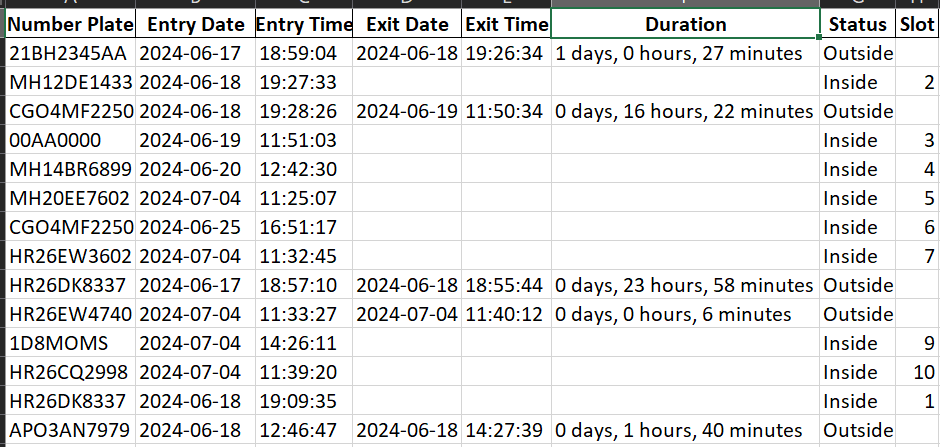
* For registered vehicles, the code logs entry and exit times into an Excel file.
* Each vehicle's status (inside or outside) and the duration of its stay are calculated and recorded.

**Slot Management:**

* The code assigns parking slots to vehicles and updates the slot information in the Excel file.

**Duration Calculation:**

* The duration of each vehicle's stay on the campus is calculated as the difference between entry and exit times.
* The calculated duration is formatted into "days, hours, minutes".



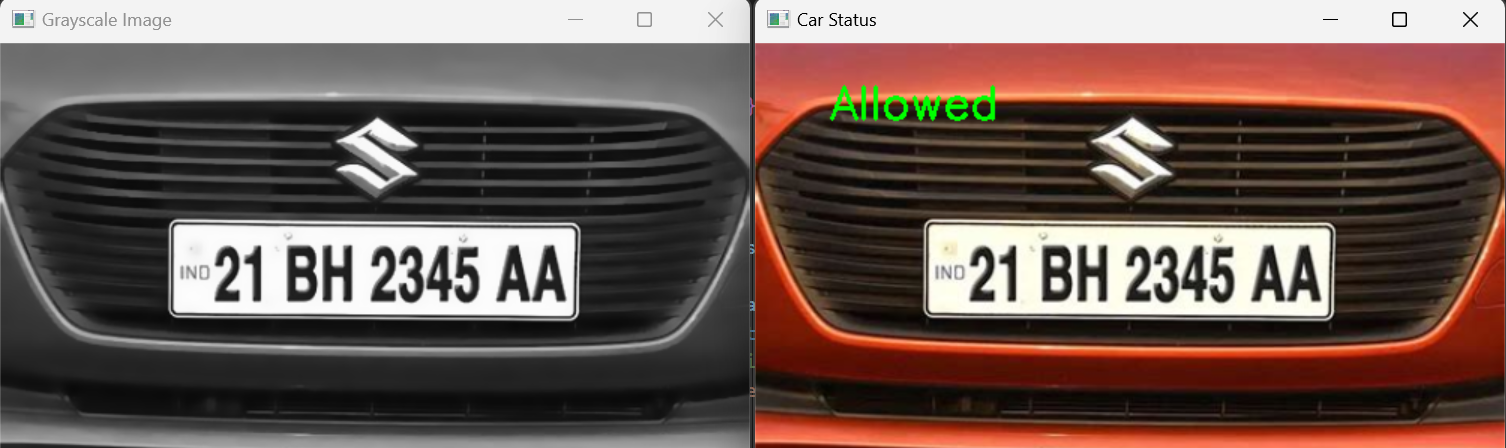
**3. Alerts and Notifications**

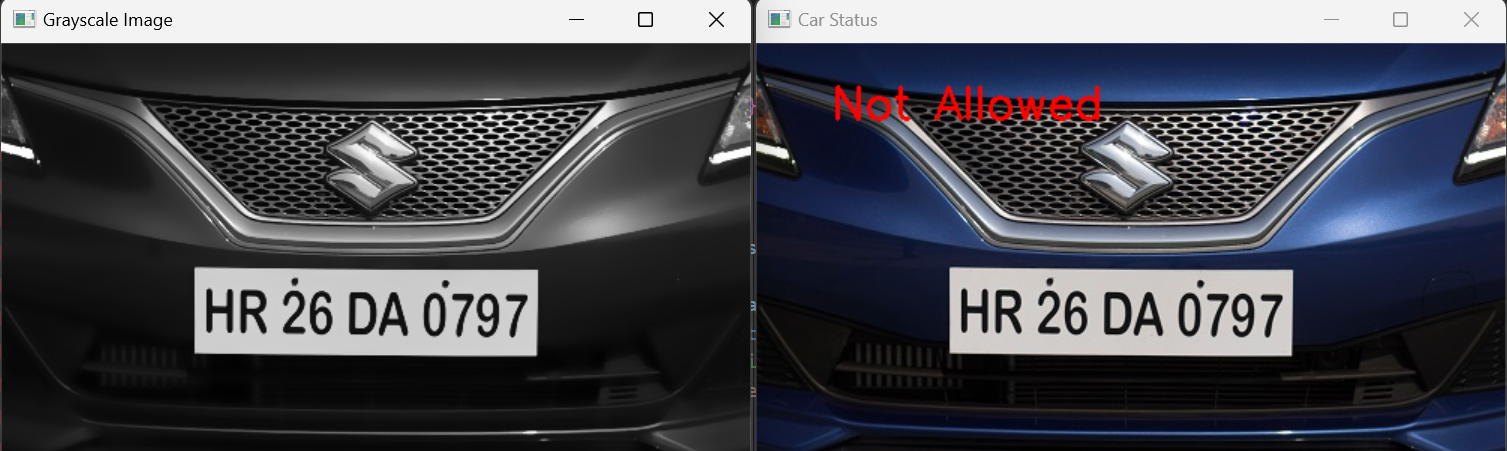
**Sound Alerts:**

* A beep sound is played for registered vehicles to alert their detection.

**Visual Alerts:**

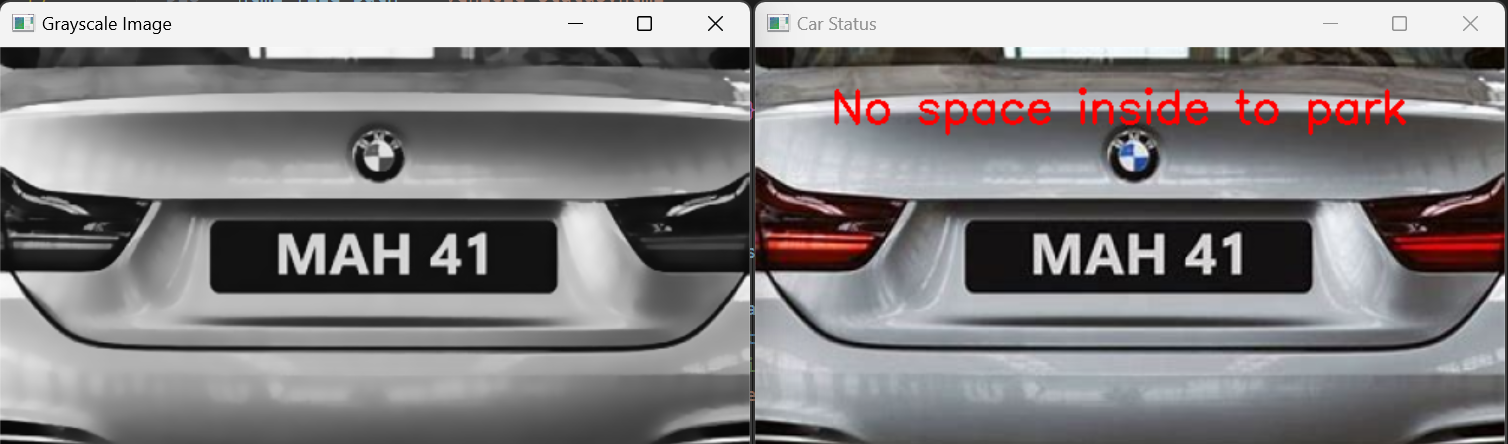
* Images of vehicles are displayed with either "Allowed" or "Not Allowed" text based on registration status.
* If the vehicle is not registered, it is marked as "Not Allowed".





**Visual Alerts:**

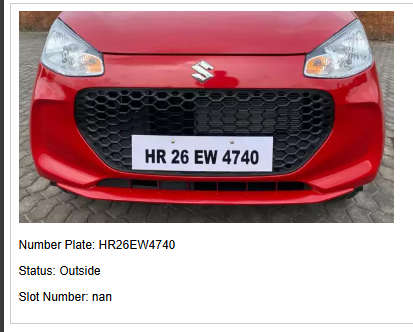
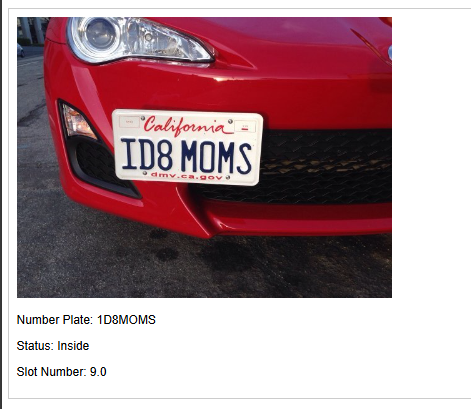
* If there is no space inside then advise to park outside.



**4. Visualization**

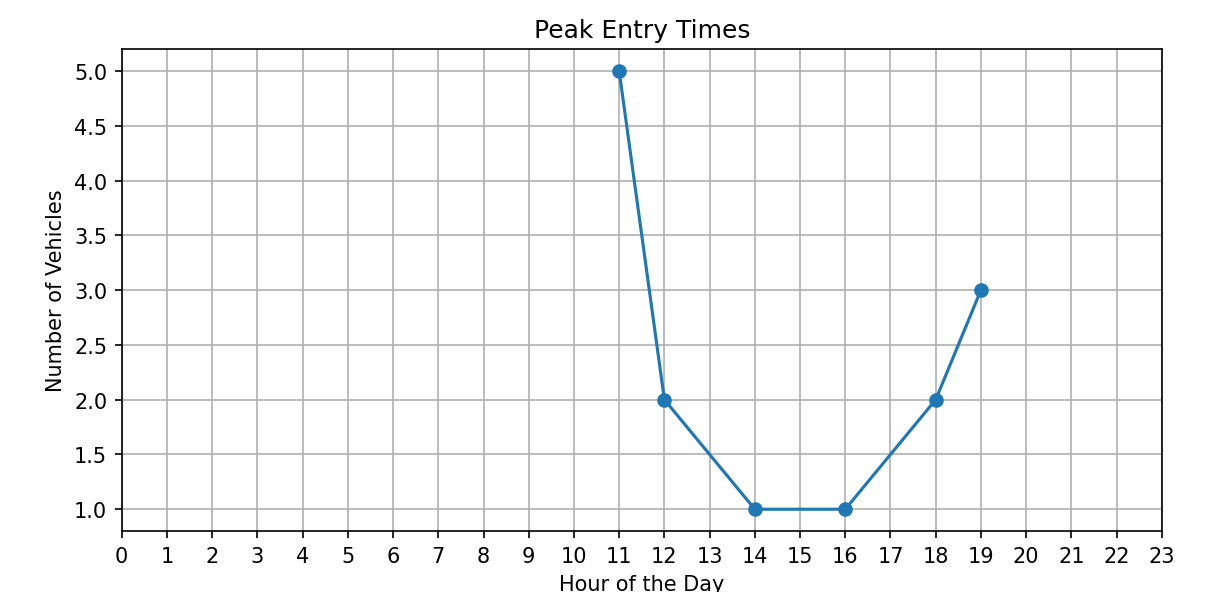
**HTML Report Generation:**

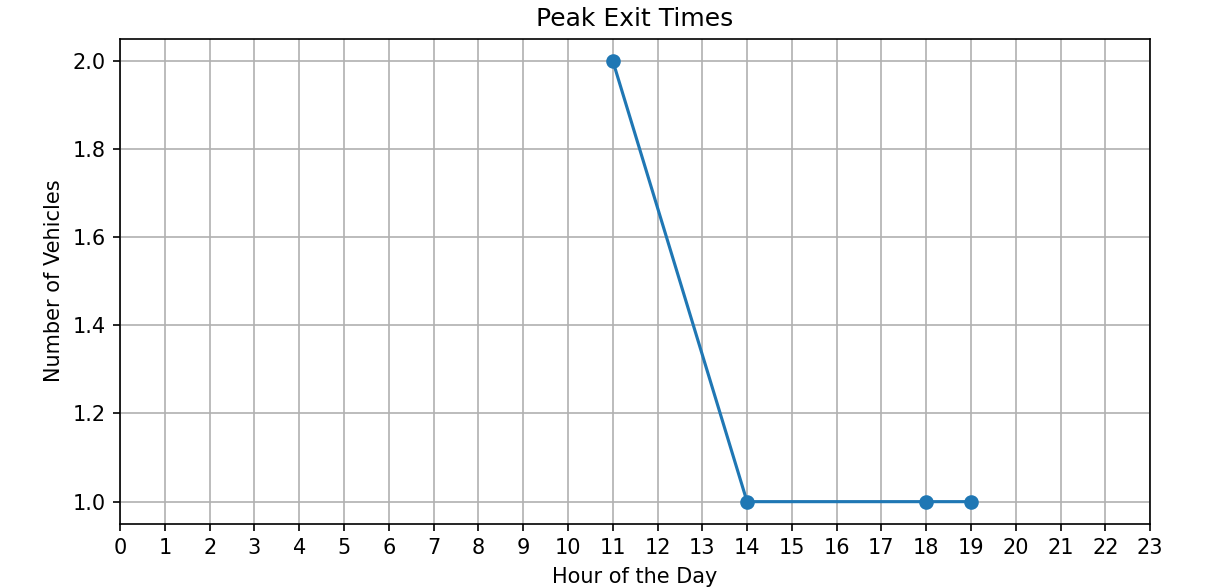
* An HTML page is dynamically generated to visualize the status of all detected vehicles.
* The HTML page includes:
  + Images of vehicles, Number plate information, Status (Inside/Outside), Slot number.



**5. Pattern:**

Peak times of the entry and exit times of the cars.





**Conclusion:**

 **Effective Vehicle Detection and OCR**:

* The system successfully detects vehicle number plates from images using OpenCV for image processing and Tesseract for Optical Character Recognition (OCR).
* The detection accuracy is high for clear and well-captured images.
* In the image processing code, Tesseract OCR plays a crucial role in identifying letters and numbers by matching them with predefined alphabets and numbers. The image is converted into multiple colours, and then the number plate is extracted.

 **Registration Verification**:

* Number plates are verified against a predefined database to check if the vehicle is registered.
* Registered vehicles are allowed entry, while unregistered vehicles are flagged as not allowed.
* Database is main part to store and to check with extracted number plate.

 **Efficient Logging**:

* Entry and exit times of vehicles are logged in an Excel file.
* The system calculates and records the duration of each vehicle’s stay on the campus.
* Slot numbers are assigned to vehicles, and the status (inside or outside) is updated accordingly.
* It shows the max output and is used to generate the patterns.

** Real-time Alerts**:

* The system provides immediate visual and audio alerts for registered vehicles.
* "Allowed" or "Not Allowed" labels are displayed on the images of the vehicles based on their registration status.

 **Comprehensive Visualization**:

* An HTML report is generated to visualize the status of all detected vehicles.
* The report includes images of vehicles, their number plate information, status, and assigned slot numbers.