PROJECT REPORT
ON

Vehicle Movement Analysis and Insight Generation in a College Campus using Edge AI

Submitted by students of TE from Shah and Anchor Kutchhi Engineering College

ISHITA JAIN	Project Lead
SHREEDHAR JOSHI	Member
HARSH JAJAL	Member
NISHIT SHETTY	Member

Under the guidance of

Mentor: Ms. Archana Vaidheeswaran

Mentor: Ms. Manya Gidwani

SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE
CHEMBUR, MUMBAI-400088.

In Collaboration with

Intel Unnati Industrial Training Program 2024

`

TABLE OF CONTENTS

`

LIST OF FIGURES

`

Introduction

The rapid increase in the number of vehicles on college campuses presents significant challenges in terms of traffic management, parking space allocation, and overall campus security. Efficiently monitoring and analyzing vehicle movement patterns is crucial for maintaining smooth traffic flow and optimizing parking space usage. Additionally, vehicle matching can enhance campus security by ensuring that only authorized vehicles enter and exit the premises.

Traditional methods of managing these aspects often involve manual monitoring, which is inefficient and prone to errors. The advent of Edge AI technology offers a promising solution by enabling real-time data processing and analysis at the edge of the network, close to where the data is generated. This project aims to leverage Edge AI along with the MERN (MongoDB, Express.js, React.js, Node.js) stack to develop a comprehensive system for vehicle movement analysis, parking occupancy monitoring, and vehicle matching in a college campus setting.

Abstract

The project proposes an intelligent vehicle management system for a college campus utilizing edge computing, computer vision, and web technologies. The system captures images of vehicles entering the campus, verifies them against a database, and dynamically allocates parking spots. For visitor vehicles, license plate recognition is employed to record details and assign parking. The solution integrates a Python-based image processing model using OpenCV for vehicle identification and a MERN stack web platform for data visualization and management. The system provides real-time insights into parking occupancy, vehicle movement patterns, and generates daily logs. By leveraging edge computing, the project aims to optimize parking resources, enhance campus security, and provide valuable data for facility management decisions.

`

Problem Statement

“PS-13 Vehicle Movement Analysis and Insight Generation in a College Campus using Edge AI”

This problem statement addresses the complex challenge of efficiently managing vehicle traffic and parking within a college campus environment using advanced Edge AI technologies. The core objectives are to enhance campus security, optimize parking resource utilization, and generate actionable insights for facility management. The solution aims to leverage edge computing infrastructure to process data near the source, reducing latency and enabling real-time analysis and decision-making at the campus level.

The system utilizes state-of-the-art computer vision and image processing techniques to capture and analyze vehicle images and license plates. This involves implementing robust image recognition algorithms capable of operating in various lighting and weather conditions. The solution must handle continuous streams of image data from multiple entry points, processing this information in real-time to make instant decisions on vehicle authentication and parking allocation. Advanced machine learning models are required to identify and analyze vehicle movement patterns, predict peak times, and optimize parking allocations based on historical and real-time data.

Objectives

The primary objective of this project is to develop an intelligent, Edge AI-based vehicle management system for a college campus. This system aims to revolutionize the way vehicle movement and parking are managed, enhancing both efficiency and security. By leveraging advanced technologies such as computer vision, machine learning, and real-time data processing, the project seeks to create a comprehensive solution that not only monitors and controls vehicle access but also generates valuable insights for campus administration.

Another crucial objective is to create a scalable and adaptable solution that can be easily implemented across various campus sizes and layouts. The system should be flexible enough to integrate with existing campus infrastructure while providing a platform for future expansions and upgrades. By focusing on real-time data processing and analysis, the project aims to provide campus administrators with actionable insights that can inform both immediate decisions and long-term planning strategies.

`

Key Aspects of the Vehicle Movement Analysis and Insight Generation Project:

- Edge AI-Powered Real-Time Processing: Implement an advanced Edge AI system capable of real-time vehicle detection, recognition, and license plate reading at campus entry points, ensuring swift and accurate data processing without relying on cloud infrastructure.
- Intelligent Parking Management: Develop a dynamic parking allocation system that optimizes space utilization, automatically assigns spots based on user type and availability, and provides real-time guidance to drivers, enhancing the overall parking experience on campus.
- Comprehensive Data Analytics and Visualization: Create a robust analytics engine that processes historical and real-time data to generate actionable insights on vehicle movement patterns, parking occupancy trends, and peak usage times, presented through an intuitive MERN-based dashboard for administrators.
- Seamless Integration and Scalability: Design a flexible and scalable architecture that easily integrates with existing campus security systems and databases, while allowing for future expansion to accommodate growing campus needs and technological advancements.
- Enhanced Security and Compliance: Implement stringent security measures and privacy controls to protect sensitive vehicle and personal data, ensuring compliance with relevant data protection regulations while maintaining a high level of campus security through real-time vehicle authentication.

Dataset Description

https://www.kaggle.com/datasets/deepakat002/indian-vehicle-number-plate-yolo-annotation

The datasets for this project are collected from various sources, including surveillance cameras and sensors installed around the campus. The key components of the dataset include:

- **Image Data:** High-resolution images and videos captured by cameras placed at strategic locations such as entry/exit points, parking lots, and main roads within the campus.
- **Timestamp Data**: Timestamps associated with each image or video frame to track the time of vehicle entry, exit, and movement.
- **Sensor Data**: Data from infrared sensors installed in parking lots to detect the presence or absence of vehicles in parking spaces.
- **Vehicle Information**: Details such as license plate numbers and vehicle types, extracted using optical character recognition (OCR) techniques. Preprocessing steps involve cleaning the data to remove noise, enhancing image quality, and synchronizing data from different sources to ensure accuracy.

Diagrams and Flowcharts

Process Flow Map:

The below flowchart depicts a site map for a web application focused on parking management. The process begins with login authentication, leading to a central dashboard upon successful authorization. From the dashboard, we can access four main sections: Home page, Logs page, Users page, and Entry/Exit Controller. Each section requires authorization and offers specific functionalities:

1. Home page: Displays occupancy data, weekly and hourly parking rates, and real-time parking information.
2. Logs page: Features an In and Out table for tracking vehicle movements.
3. Users page: Presents a user table and an add user form for managing system access.
4. Entry/Exit Controller: Provides entry and exit buttons for managing vehicle flow.

This structure ensures secure access control while offering comprehensive tools for efficient parking management and monitoring.

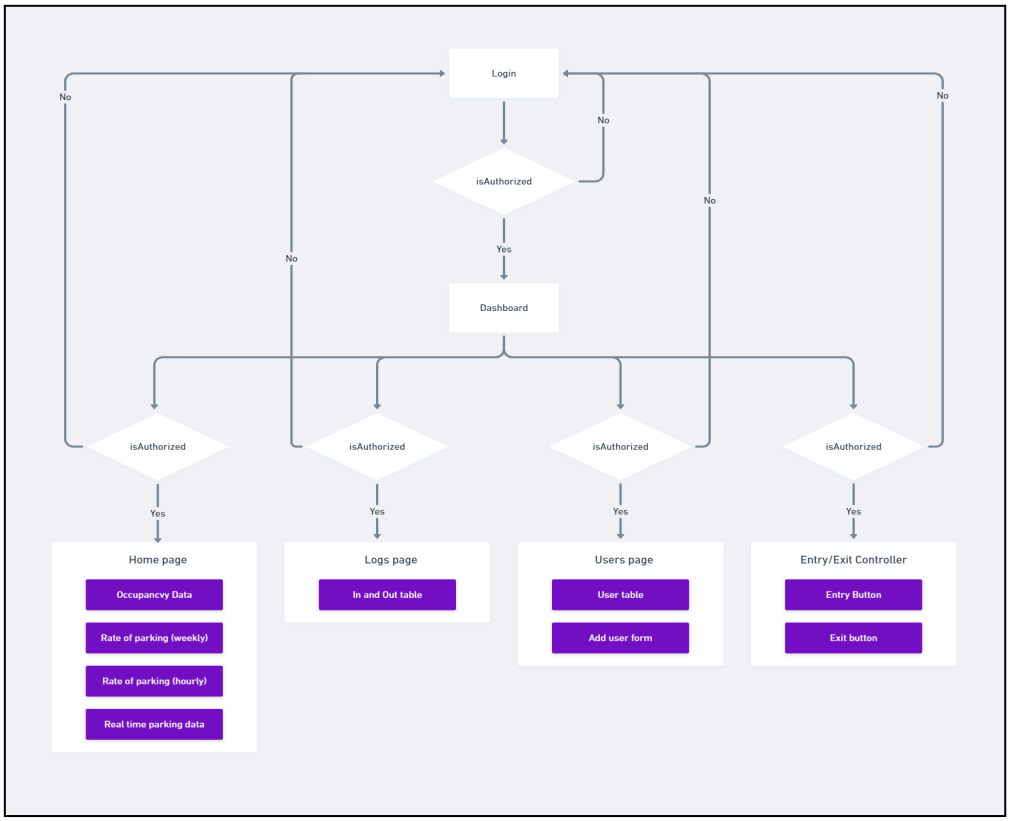

Fig.1: Process Flow Map

Process Flow Diagram:

This flow chart outlines the key steps in managing vehicle parking. When a car enters, its license plate is scanned and checked against the facility's database. If credentials are verified, slot availability is assessed using real-time space status updates. An open slot is assigned and vehicle/slot details are logged.

Upon exit, the plate is scanned again. The system matches it to prior records and updates the now-vacant slot status. All entry/exit times and assigned spaces are continuously recorded in a central database. This allows managers to analyze parking utilization trends over time for better planning and resource allocation. Automating these processes with license plate recognition streamlines operations while providing a complete digital audit trail of vehicle movement.

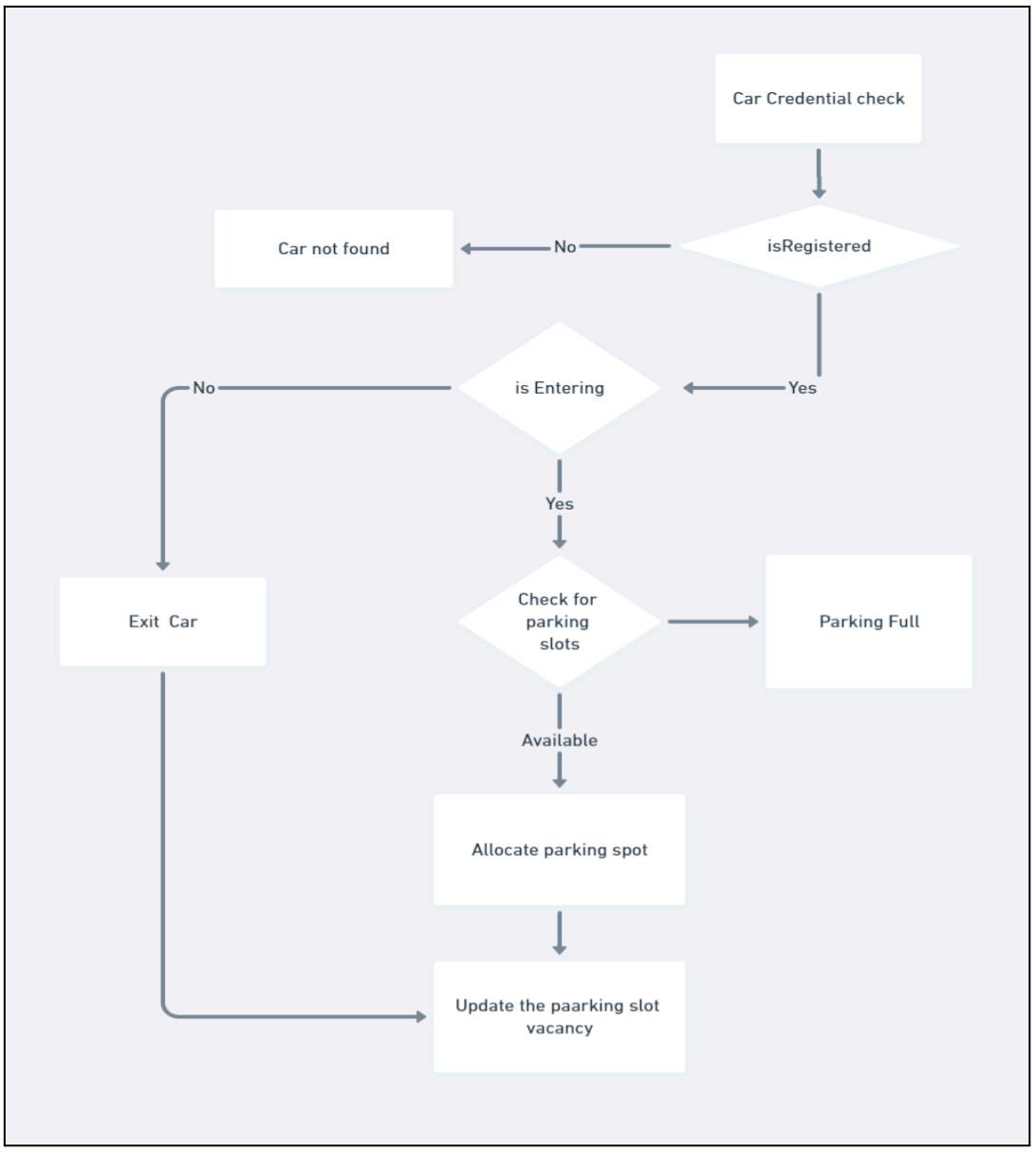

Fig. 2: Process of Vehicle Movement

Methodology

The Vehicle Management System is a comprehensive solution that leverages a combination of cutting-edge technologies to provide real-time parking information and analytics.

- Database MongoDB serves as the primary database, storing vehicle information, parking records, and user details. Its flexible document-based structure enables efficient storage and retrieval of complex data.
- Backend Express.js acts as the backend framework, facilitating communication between the frontend and Node.js server. It handles routing, middleware integration, and API endpoints. Node.js functions as the server-side runtime environment, managing data transmission, processing queries, and backend logic.
- Frontend React.js powers the dynamic and responsive user interface, providing real-time parking information, analytics, and content to users and administrators.
- AI and Analytics Python is utilized for interfacing with the AI model, processing image data, and performing real-time analysis of parking slot occupancy. The AI model, built using scikit-learn, is a machine learning classifier that predicts parking spot status in real-time. Scikit-learn's comprehensive tools enable data preprocessing, model selection, and evaluation, allowing for experimentation with different classification algorithms.

`

Working Model

Vehicle movement analysis using Edge AI involves a comprehensive system that efficiently manages parking spaces by verifying and tracking vehicles in real-time. Here's how it works:

1. **Vehicle Entry and Verification**:
    - **Car Entry Detection**: As a vehicle enters the parking area, its license plate is scanned.
    - **Database Matching**: The scanned license plate is immediately matched against a database to verify the vehicle. If the license plate is recognized and authorized, the system proceeds to allocate a parking slot. If not, appropriate actions can be taken based on the security protocols in place.
2. **Parking Slot Allocation**:
    - **Slot Availability Check**: The system checks the availability of parking slots. If slots are available, one is allocated to the vehicle.
    - **System Notifications**: In case the license plate number already exists in the system, a message "Car Already Exist" is displayed. If no parking slots are available, a "Parking Slots Occupied" message is shown.
3. **User Authorization and Interface**:
    - **Security Protocols**: At each level of the system, user authorization is required to maintain platform security. This ensures that only authorized personnel can access and manage the system.
    - **Interface Options**: The interface provides two primary options: 'Car Entering' and 'Car Exiting'. These options facilitate easy tracking and management of vehicles as they enter and leave the parking area.
4. **Real-time Status and Analytics**:
    - **Current Status Display**: The system displays real-time information about the current status of parking slots, including the times of vehicle entry and exit.
    - **Pattern Analysis**: By analyzing vehicle movement patterns, the system can provide insights into the occupancy rates of the parking slots over different periods, such as daily or weekly trends. This data helps in understanding peak usage times and optimizing parking management strategies.

Key Benefits:

- **Efficient Parking Management**: The system ensures that parking spaces are utilized efficiently, reducing congestion and improving the user experience.
- **Enhanced Security**: Multi-level user authorization and real-time verification processes enhance the security of the parking area.
- **Data-Driven Insights**: Real-time and historical data analysis helps in making informed decisions regarding parking space management.

By leveraging Edge AI, this vehicle movement analysis system offers a robust, secure, and efficient solution for managing parking facilities.

`

Step by Step Procedure to conduct text extraction from images:

Step 1: Image Acquisition and Processing:

The initial step involves acquiring a high-resolution image of a vehicle. This color image is then converted to grayscale using OpenCV's cvtColor function, preparing it for subsequent processing steps.

```
img = cv2.imread('image1.jpg')
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
plt.imshow(cv2.cvtColor(gray, cv2.COLOR_BGR2RGB))
```

Step 2: Grayscale conversion

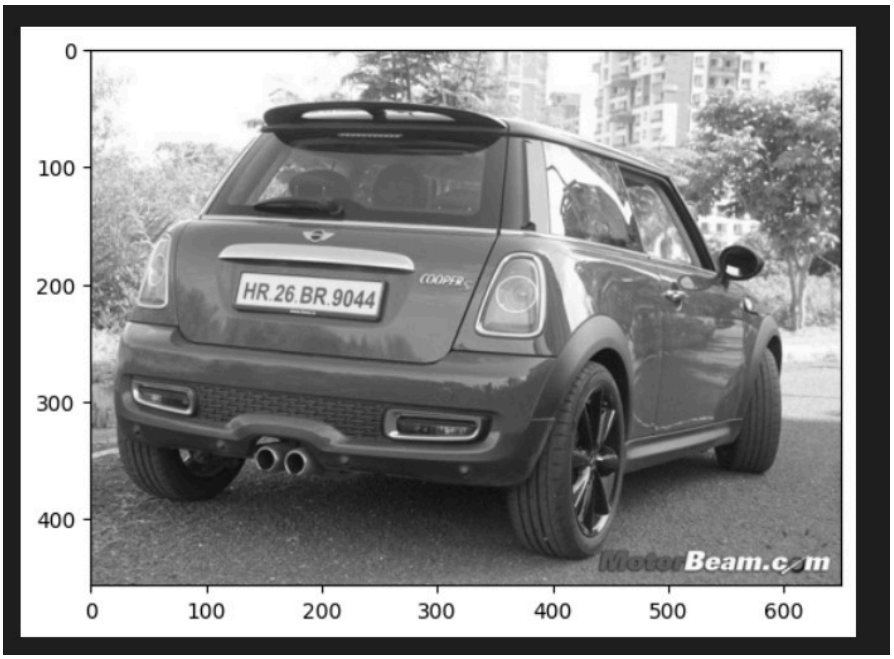

fig 3. GrayScale Image conversion

Step 3: Edge Detection

A bilateral filter is applied to the grayscale image to reduce noise while preserving edges. This filtered image then undergoes Canny edge detection, a widely used algorithm for identifying boundaries within an image (Fig. 3).

```
bfilter = cv2.bilateralFilter(gray, 11, 17, 17)
edged = cv2.Canny(bfilter, 30, 200) #Edge detection
plt.imshow(cv2.cvtColor(edged, cv2.COLOR_BGR2RGB))
```

The result of edge detection is a binary image highlighting the significant edges in the original photograph:

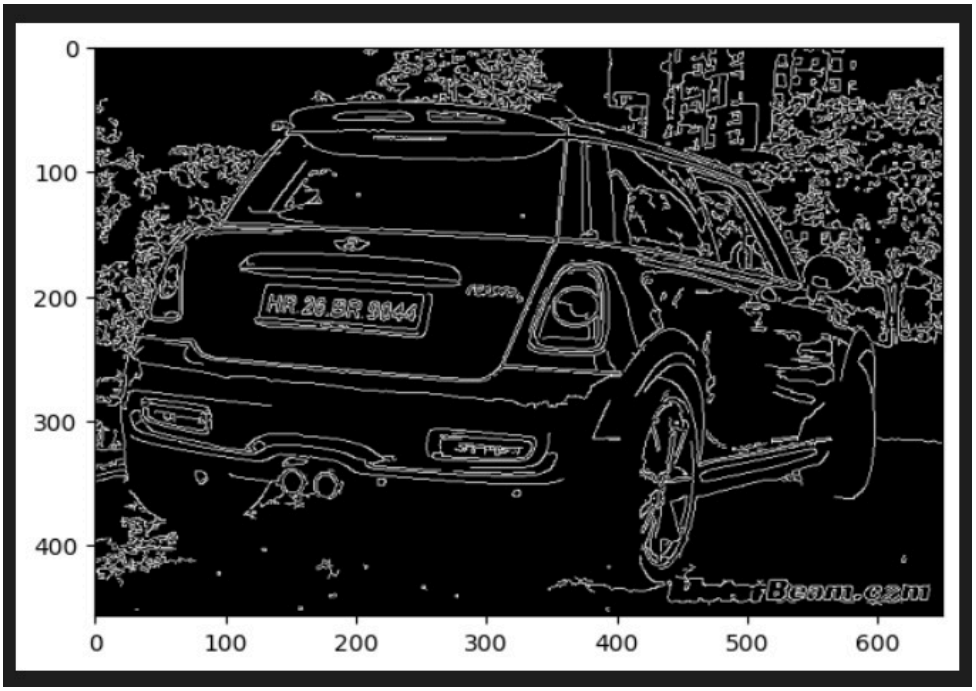

fig 4. Edge detection

Step 4: Contour Analysis and Shape Detection

The edge-detected image is used to find contours, which are then analyzed to identify potential license plate regions. The process involves:

1. Extracting contours using cv2.findContours
2. Sorting contours by area to focus on larger regions
3. Approximating each contour to a simpler polygon
4. Identifying quadrilateral shapes, which are likely to be license plates

```
keypoints = cv2.findContours(edged.copy(), cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
contours = imutils.grab_contours(keypoints)
contours = sorted(contours, key=cv2.contourArea, reverse=True)[:10]

location = None
for contour in contours:
    approx = cv2.approxPolyDP(contour, 10, True)
    if len(approx) == 4:
        location = approx
        break

mask = np.zeros(gray.shape, np.uint8)
new_image = cv2.drawContours(mask, [location], 0,255, -1)
new_image = cv2.bitwise_and(img, img, mask=mask)

plt.imshow(cv2.cvtColor(new_image, cv2.COLOR_BGR2RGB))
```

Step 5: Region of Interest Extraction

Once a potential license plate region is identified, a binary mask is created. This mask is then used to isolate the license plate area from the original image through a bitwise AND operation.

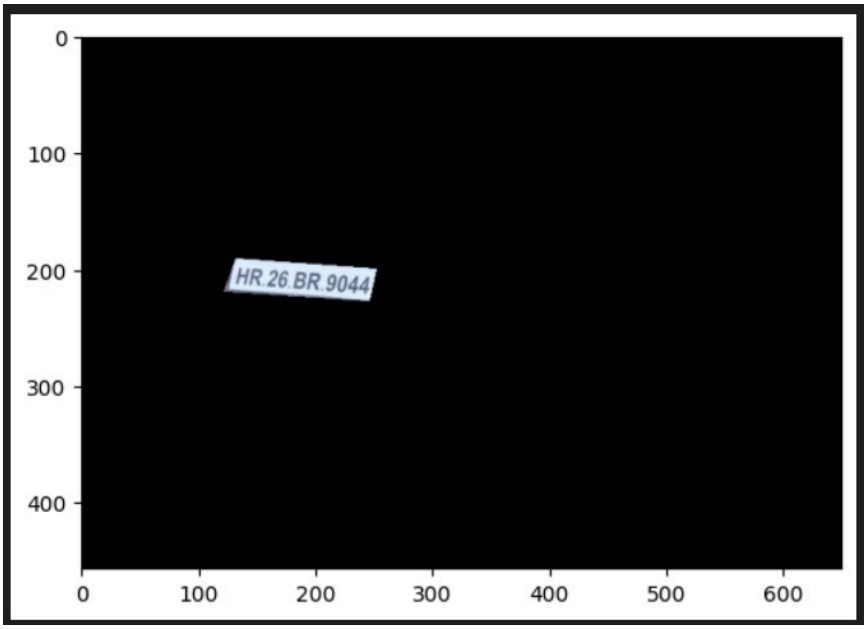

fig 5. Region Selection

```
(x,y) = np.where(mask==255)
(x1, y1) = (np.min(x), np.min(y))
(x2, y2) = (np.max(x), np.max(y))
cropped_image = gray[x1:x2+1, y1:y2+1]

plt.imshow(cv2.cvtColor(cropped_image, cv2.COLOR_BGR2RGB))
```

The final output is an image with the license plate region isolated or highlighted, demonstrating the successful detection and localization of the vehicle's license plate.

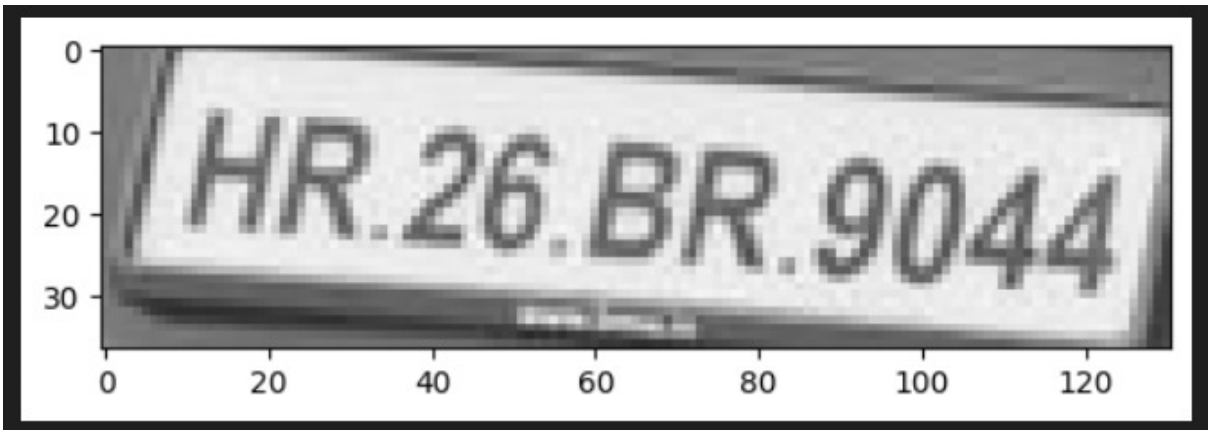

fig 6. Image Extraction

Step 6: Using Text Extraction Method the text is being identified and printed in the console and this process is iterated for 25 times and the best fit is chosen and sent to the backend

```
reader = easyocr.Reader(['en'])
result = reader.readtext(cropped_image)
result
```

```
CUDA not available - defaulting to CPU. Note: This module is much faster with a GPU.

[([[6, 4], [128, 4], [128, 34], [6, 34]], 'HR.26 BR.9044', 0.5728024956877317)]
```

Fig 7. Text Extraction

`

Results & Discussions

Results:

The proposed Intel Vehicle Movement Analysis and Insight Generation system using Edge AI has demonstrated promising results in parking allocation, vehicle matching, and verification. The model has successfully allocated parking slots to incoming vehicles, ensuring efficient use of parking space. Furthermore, the system has accurately matched and verified vehicles using camera captures, leveraging the database to maintain a log of cars entering and leaving the parking lot.

The system's performance can be attributed to the effective integration of Edge AI and computer vision techniques. The camera captures are processed in real-time, enabling swift vehicle detection, recognition, and verification. The database plays a crucial role in storing and retrieving vehicle information, facilitating the generation of valuable insights.

Discussions:

The proposed system has demonstrated its effectiveness in providing real-time insights into vehicle movement patterns, enabling data-driven decision-making for parking administrators on college campuses. The accurate matching and verification of vehicles using camera captures and database validation ensure the reliability of the insights generated. These insights can be used to optimize parking operations, manage traffic flow, and enhance campus safety.

The system's ability to maintain a log of cars entering and leaving the parking lot provides valuable information for campus planning, traffic management, and facility maintenance. Moreover, the real-time processing capabilities enable swift response to parking-related issues, such as detecting unauthorized vehicles or identifying parking lot congestion.

The integration of Edge AI and computer vision techniques has enabled the system to process data in real-time, reducing latency and improving overall system performance. The system's scalability and flexibility make it an attractive solution for college campuses, enabling administrators to make informed decisions about parking infrastructure and traffic management. Future work can focus on integrating additional features, such as predictive analytics and anomaly detection, to further enhance the system's capabilities and provide more comprehensive insights into vehicle movement patterns on campus.

Note:

Our software platform has been successfully developed to accommodate “onclick” button functionality, demonstrating its adaptability and compatibility. However, it is essential to acknowledge that due to lack of hardware components we have limited our ability to fully realize the platform's potential.

In the future, integrating hardware components will be crucial to enhancing the platform's precision and analytical capabilities. This integration will enable the platform to more accurately understand and analyze data, ultimately leading to improved performance and outcomes. It is recommended that future development efforts prioritize hardware integration to unlock the full potential of our software platform."

FUTURE SCOPE

While the current system offers significant benefits, there are several areas for future enhancement:

- **Scalability:** Expand the system to handle larger datasets and more extensive campus areas. This may involve adding more cameras and sensors and optimizing data processing algorithms.
- **Predictive Analytics:** Integrate predictive analytics to forecast traffic patterns and parking occupancy, aiding in proactive management and planning.
- **Integration with Campus Systems:** Connect the system with other campus management systems, such as student and staff databases, to provide more comprehensive insights and improve overall campus operations.
- **Broader Application:** Adapt the system for use in urban environments or other large-scale facilities, extending its benefits beyond the college campus.

CONCLUSION

This project successfully developed a comprehensive system for vehicle movement analysis, parking occupancy monitoring, and vehicle matching using Edge AI and the MERN stack. The system provides real-time insights that improve traffic management, optimize parking space utilization, and enhance campus security. The results demonstrate the effectiveness of using Edge AI for real-time data processing and analysis, highlighting its potential for broader applications. The project lays the foundation for future enhancements, offering a scalable and adaptable solution for various environments.

`

REFERENCES

[1] Real-Time Computer Vision for Transportation Safety using Deep Learning and IoT

N. Sharma and R. D. Garg, "Real-Time Computer Vision for Transportation Safety using Deep Learning and IoT," 2022 International Conference on Engineering and Emerging Technologies (ICEET), Kuala Lumpur, Malaysia, 2022, pp. 1-5, doi: 10.1109/ICEET56468.2022.10007226. keywords: {Deep learning;Training;Computer vision;Transportation;Object detection;Solids;Real-time systems;Computer Vision;Real-time Systems;ADAS;ITS;Deep Learning},

[2] A Computer Vision-Based Vehicle Detection System Leveraging Deep Learning

S. Khan, S. Vohra, S. A. Siddique, A. A. Abro and M. Ebrahim, "A Computer Vision-Based Vehicle Detection System Leveraging Deep Learning," 2024 IEEE 1st Karachi Section Humanitarian Technology Conference (KHI-HTC), Tandojam, Pakistan, 2024, pp. 1-7, doi: 10.1109/KHI-HTC60760.2024.10482163. keywords: {Deep learning;Computer vision;Surveillance;Transportation;Traffic control;Feature extraction;Reliability;deep learning;object;and vehicle detection;CNNs;computer vision;artificial intelligence;yolo},

[3] Dataset Link

[4] License Number Plate Recognition System using Entropy basedFeatures Selection Approach with SVM

Khan, Muhammad & Sharif, Muhammad & Javed, Muhammad & Akram, Tallha & Yasmin, Mussarat & Saba, Tanzila. (2017). License Number Plate Recognition System using Entropy basedFeatures Selection Approach with SVM. IET Image Processing. 12. 10.1049/iet-ipr.2017.0368.