VEHICLE MOVEMENT ANALYSIS AND INSIGHT GENERATION IN A COLLEGE CAMPUS USING EDGE AI

- *BY TECH TURTLES*

TABLE OF CONTENTS

PROBLEM STATEMENT

To develop an intelligent system capable of analyzing vehicle movement patterns within a college campus using edge AI techniques, providing real-time insights for efficient traffic management and resource allocation.

SOLUTION

1. The system employs multiple cameras to capture real-time video feeds across the campus.
2. These feeds are processed on an edge device using OpenCV for image preprocessing and feature extraction.
3. Deep learning models (TensorFlow) detect and track vehicles, enabling analysis of traffic patterns, congestion points, and parking occupancy.
4. Insights are generated through data visualization using Matplotlib. To optimize performance, the model is deployed on the edge device using frameworks like TensorFlow Lite.

FEATURES OFFERED BY THE MODEL

1. Real-time Vehicle Detection: Accurate and efficient detection of vehicles within video frames.
2. Vehicle Tracking: Robust tracking of individual vehicles throughout the campus.
3. Traffic Flow Analysis: Generation of insights into traffic patterns, peak hours, and congestion points.
4. Parking Occupancy Monitoring: Real-time tracking of parking lot occupancy and availability.
5. Edge AI Implementation: Optimized model deployment for efficient processing on edge devices.

PROCESS FLOW

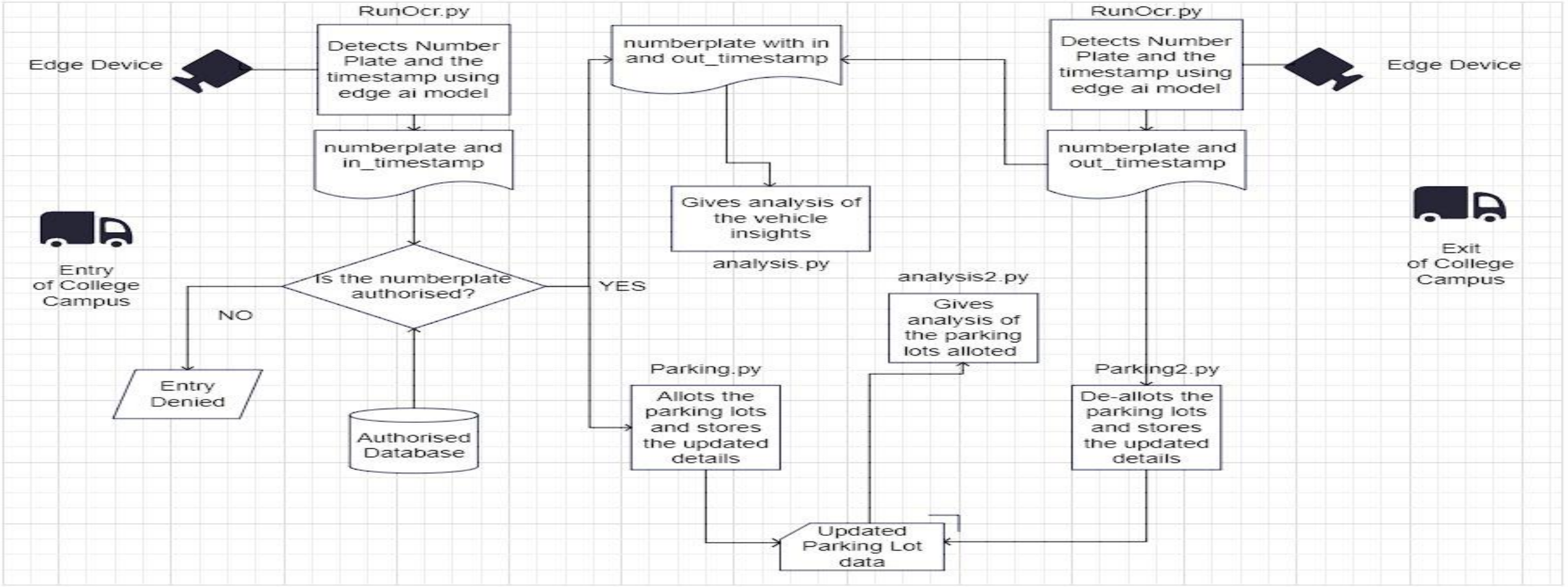
1. Image Capture: Cameras capture real-time video feeds.
2. Edge AI Processing:
   - Object detection identifies vehicles in each frame.
   - License plate recognition extracts plate numbers.
   - Vehicle tracking follows vehicles across frames.
3. Data Transmission: Processed data (vehicle type, license plate, timestamp, location) is sent to the cloud.
4. Data Storage: Data is securely stored in the cloud database.

PROCESS FLOW

1. Data Analysis:
   - Vehicle movement patterns are analyzed (entry/exit times, peak hours).
   - Parking occupancy is monitored (available/occupied spaces).
   - Anomalies (unauthorized vehicles, suspicious behavior) are detected.

2. Insight Generation:
   - Traffic flow optimization suggestions are provided.
   - Parking management strategies are recommended.
   - Security alerts are generated for anomalies.

ARCHITECTURE DESIGN



TECHNOLOGIES USED

1. Python: Core programming language for the project.
2. OpenCV: Computer vision library for image processing and video analysis.
3. TensorFlow Lite: Deep learning frameworks for object detection and tracking models.
4. Tesseract: OCR for license plate recognition.
5. Matplotlib: Data visualization for insights and reports.
6. Edge AI Framework: (e.g., TensorFlow Lite)for deployment on edge devices.

TEAM MEMBERS AND CONTRIBUTION

- ***Vaibhav Kumar Bhardwaj (2106081):***

I spearheaded the design of the OCR model, taking a hybrid approach that effectively utilizes the capabilities of both Tesseract and TensorFlow Lite. This innovative architecture ensures efficient and accurate license plate recognition. Furthermore, I played a crucial role in the data extraction pipeline, implementing methods to detect images, extract recognized license plate numbers, and meticulously ensure the resulting CSV file is free of redundancies.

- ***Shouvik Ghosh (2106067):***

My passion for model training fueled my significant contribution to the object detection model. I began by leveraging TensorFlow to build a robust detection model, meticulously aligning it with the project's requirements. To ensure efficient deployment on edge devices, I then utilized the TensorFlow Lite converter to optimize the model for resource-constrained environments. Furthermore, I played a key role in data extraction. By leveraging Gemini, I extracted a dummy dataset representing authorized campus vehicles and integrated timestamping functionality to capture the detection time of each license plate.

TEAM MEMBERS AND CONTRIBUTION

- ***Priyanshi Tiwari (22053452):***

Beyond model development and data extraction, I actively participated in translating raw data into insightful visualizations. By leveraging the Python libraries Matplotlib and Seaborn, I helped extract valuable information about vehicle movement patterns and parking lot allocation throughout the analyzed period. These insights were then incorporated into a well-structured presentation and report, effectively consolidating the collective work of the team for clear understanding and evaluation.

CONCLUSION

1. In conclusion, this project successfully demonstrates the potential of edge AI in analyzing vehicle movement within a college campus.
2. By leveraging computer vision and machine learning techniques, we were able to develop a system that accurately detects and tracks vehicles, analyzes traffic patterns, and monitors parking occupancy in real-time.
3. The insights derived from this system can significantly contribute to campus planning, traffic management, and security initiatives.