Problem Statement

PROBLEM STATEMENT 13:

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

OBJECTIVE:

The primary objective of this project is to develop an Edge AI-based solution that can analyse vehicle movement in and out of a college campus using data from cameras capturing vehicle photos and license plates. The solution should provide insights on vehicle movement patterns, parking occupancy, and match vehicles to an approved vehicle database.

Solution

Our solution provides an automated and efficient system for monitoring and analyzing vehicle entries and exits within a designated area using state-of-the-art machine learning and optical character recognition (OCR) technologies. The system comprises the following key components:

1. Video Surveillance with YOLO Model:

We deployed a YOLO (You Only Look Once) model to continuously monitor video feeds from the surveillance cameras. The YOLO model is specifically trained to detect license plates on vehicles entering and exiting the area. This real-time detection capability ensures accurate and prompt identification of all vehicles.

Solution

2. License Plate Recognition with OCR:

Upon detecting a license plate, we employ Tesseract OCR and EasyOCR technologies to extract the license plate numbers. These OCR tools convert the image data of the license plate into readable and storable text data, ensuring high accuracy in recognizing various license plate formats.

3. Database Management:

The recognized license plate numbers are recorded in a centralized database. The system maintains two primary tables:

- **Records Table:** This table logs the entry and exit times of each vehicle. When a vehicle is detected entering the area, an entry timestamp is recorded. Upon detection of the vehicle's exit, the corresponding entry record is updated with the exit timestamp, effectively noting the duration of stay.

Solution

- **Customer Table:** This table stores additional information about the vehicles and their owners, facilitating a comprehensive view of customer interactions and history.

4. Data Visualization with Streamlit App:

To provide insightful and user-friendly access to the collected data, we have developed a Streamlit app. The app offers a variety of data visualizations and analytics, including:

- **Average Duration of Stay:** Calculates and displays the average time vehicles spend within the monitored area.
- **Peak Entry and Exit Times:** Identifies and visualizes the times of day when vehicle traffic is at its highest.
- **Vehicle Frequency:** Tracks the number of vehicles entering and exiting the area over specific periods.

Solution

- **Daily/Weekly Trends:** Analyzes and displays patterns in vehicle traffic over days and weeks.
- **Duration Distribution:** Visualizes the distribution of stay durations for all recorded vehicles.
- **Utilization Rates:** Shows the occupancy rates and utilization of the monitored area.
- **Entry/Exit Patterns:** Provides detailed insights into the entry and exit behavior of vehicles.

Features Offered

1. Real-Time Vehicle Detection:

Utilizes the YOLO (You Only Look Once) model for immediate identification of vehicles and license plates in video feeds.

2. High-Accuracy License Plate Recognition:

Employs Tesseract OCR and EasyOCR technologies to extract license plate numbers with high precision from detected vehicles.
Detailed Database Management

3. Maintains a structured database with:

Records Table: Logs entry and exit times of vehicles, tracking duration of stay.
Customer Table: Stores detailed information about vehicles and their owners.

Features Offered

4. Comprehensive Data Visualization:

Features a Streamlit app for visualizing and analyzing data, including:

- **Average Duration of Stay:** Displays the average time vehicles spend within the monitored area.
- **Peak Entry and Exit Times:** Identifies and visualizes times of highest vehicle traffic.
- **Vehicle Frequency:** Tracks the number of vehicles entering and exiting over specified periods.
- **Daily/Weekly Trends:** Analyzes vehicle traffic patterns on a daily and weekly basis.
- **Duration Distribution:** Shows the distribution of vehicle stay durations.
- **Utilization Rates:** Highlights the occupancy rates and usage of the monitored area.
- **Entry/Exit Patterns:** Provides insights into patterns of vehicle entry and exit behavior.

Process flow

Vehicle Detection:
Input: Video feed from surveillance camera.
Action: YOLO model analyzes the video to detect vehicles and license plates.
Output: Detected license plate images.

License Plate Recognition:
Input: Images of detected license plates.
Action: Apply Tesseract OCR and EasyOCR to extract text from license plate images.
Output: Extracted license plate numbers.

Process flow

Database Update:

Input: Extracted license plate numbers.

Action: Check if the vehicle is already in the database.

If Not: Record entry time in the Records Table and add vehicle details to the Customer Table.

If Yes: Update exit time in the Records Table if the vehicle exits.

Output: Updated database records.

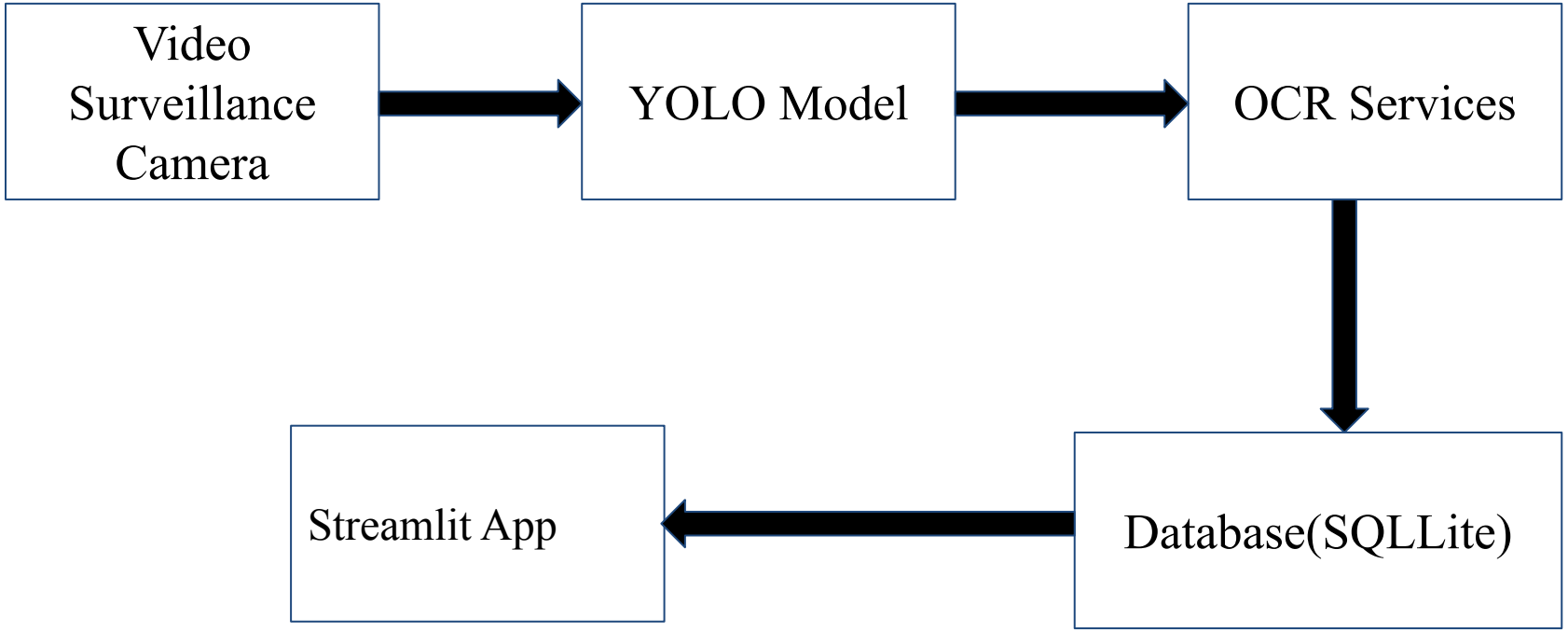
Data Analysis and Visualization:

Input: Data from the database (entry/exit times, vehicle details).

Action: Generate various analytics and visualizations using the Streamlit app.

Output: Dashboards and reports displaying insights like average duration of stay, peak times, and vehicle frequency.

Architecture Diagram



Technologies used

- Python: Primary language used for developing the YOLO model, OCR processing, and Streamlit app and Pandas.
- SQL: For querying and managing the database.
- YOLO (You Only Look Once): Deep learning model for real-time object detection, used for detecting vehicles and license plates in video feeds.
- Tesseract OCR: Open-source OCR engine for extracting text from images of license plates.
- EasyOCR: Another OCR tool for text recognition, used to complement Tesseract OCR for better accuracy and coverage.
- SQL Database: For storing and managing records of vehicle entries, exits, and customer details.

Technologies used

- Streamlit: Python library for creating interactive web applications and dashboards to visualize and analyze data.
- Matplotlib: Python plotting library used within the Streamlit app to create various data visualizations.
- OpenCV: For image and video processing tasks.
- TensorFlow: Frameworks for training and running the YOLO model

Team members and contribution

1. Swetha Balaji (Team Lead):
   Developed a web interface using Streamlit, including the design and implementation of all data visualizations.
2. Kamaleshwar V:
   Handled data collection, trained the YOLO model, and integrated Streamlit with the database.
3. Prabakaran G S:
   Managed the database requirements for the project.

Conclusion

The Vehicle Movement Analysis and Insight Generation System uses Edge AI, real-time video processing using YOLO, and OCR technologies to automate vehicle monitoring on campus. It captures vehicle data, stores it in a database, and analyzes it to provide insights into traffic patterns. This data is then visualized through a Streamlit web app, helping to improve parking management within the campus security.