**PROJECT REPORT ON**

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



Submitted By:

Rajan

Mamandeep Singh

Nikhil Kumar

Luxman

Manjeet Kumar

**Table of Contents**

Introduction 1

Dataset Description 2

Methodology 3

Results and Discussion 6

Conclusion 8

**Introduction**

This project focuses on the development and deployment of a machine learning model designed for vehicle movement analysis within a college campus. Utilizing the YOLOv5 framework, the model excels in detecting vehicles and recognizing their number plates from both images and video streams. It boasts high accuracy and is versatile enough to be applied across various domains such as traffic management, security systems, and smart parking solutions.

The primary features of this model include:

* **Vehicle Detection**: Accurately identifies different types of vehicles, such as cars, trucks, and motorcycles, in images and videos.
* **Number Plate Recognition**: Effectively extracts and reads number plates from the detected vehicles.
* **High Accuracy**: Delivers robust performance in both vehicle detection and number plate recognition, ensuring reliable results.
* **Versatility**: Capable of being adapted for deployment in diverse environments and applications.
* **Automatic Record Keeping**: Stores the identified number plates in a CSV file for easy record-keeping and future reference.

This model can be applied in various real-life scenarios, including:

* **Traffic Management**: Enabling real-time monitoring and analysis of vehicle movements to improve traffic flow and safety.
* **Security Systems**: Enhancing surveillance capabilities by detecting and identifying vehicles and their number plates, contributing to overall campus security.
* **Smart Parking Solutions**: Facilitating automated management of entry and exit in parking facilities, optimizing the use of space and improving user convenience.

The successful implementation of this model within a college campus can lead to improved traffic regulation, heightened security, and efficient parking management, demonstrating its practical value and adaptability.

**Dataset Description**

In this project, we have made use of the following dataset available freely over the internet:

**1. Stanford Cars Dataset**

The Stanford Cars Dataset is a large collection of images intended for research in the field of image classification. It is particularly useful for tasks involving the detection and recognition of various car models and makes.

**Content:**

* **Images:** The dataset contains 16,185 images of 196 different classes of cars. The images are of high resolution, typically containing a single car per image.
* **Classes:** The 196 classes include a variety of car models spanning different years and manufacturers, providing a comprehensive dataset for fine-grained classification tasks.
* **Annotations:** Each image is annotated with bounding boxes and labels indicating the car's make, model, and year. This facilitates tasks such as object detection, classification, and segmentation.

It can be accessed through the [link to the dataset](https://www.kaggle.com/datasets/jessicali9530/stanford-cars-dataset).

**2. Automatic Number Plate Recognition (ANPR) Dataset**

The Automatic Number Plate Recognition (ANPR) Dataset is designed for research and development in the field of number plate detection and recognition. This dataset is particularly valuable for tasks involving the extraction and reading of license plates from vehicle images.

**Content:**

* **Images:** The dataset comprises thousands of images of vehicles with visible license plates. These images are collected from various sources to ensure diversity in plate design, lighting conditions, and angles.
* **Annotations:** Each image is annotated with bounding boxes around the number plates. Additionally, the dataset includes the text of the number plates for recognition tasks.
* **Variability:** The dataset includes images taken under different lighting conditions, from various angles, and with different plate designs to provide a robust training set for machine learning models.

It can be accessed through the [link to the dataset](https://www.kaggle.com/code/aslanahmedov/automatic-number-plate-recognition).

**Methodology**

Following steps are needed to be followed:

#### Step 1: Set Up the Virtual Environment

To ensure a clean and isolated development environment, we use virtualenv. Follow these steps to create and activate a virtual environment:

1.1. **Install virtualenv**: If virtualenv is not already installed, use the following command:

bash

pip install virtualenv

1.2. **Navigate to your project directory**: Open a terminal or command prompt and navigate to your project directory:

bash

cd path/to/your/project

1.3. **Create a virtual environment**: Create a virtual environment named myenv (or any name of your choice):

bash

virtualenv myenv

1.4. **Activate the virtual environment**: Activate the virtual environment using the appropriate command for your operating system:

* On Windows:

bash

myenv\Scripts\activate

* On macOS/Linux:

bash

source myenv/bin/activate

1.5. **Install dependencies**: With the virtual environment activated, install the necessary dependencies using pip:

bash

pip install package\_name

To deactivate the virtual environment when done, simply use:

bash

deactivate

#### Step 2: Clone the Repository

Clone the project repository that contains the YOLOv5 model and associated files:

bash

git clone ”https://github.com/ultralytics/yolov5.git”

#### Step 3: Install Dependencies

Start the virtual environment created in Step 1 and install all required dependencies. These dependencies can be installed individually using pip, or you can use a requirements.txt file if provided:

bash

pip install -r requirements.txt

If installing individually, use the following commands:

bash

pip install python

pip install numpy

pip install pandas

pip install matplotlib

pip install opencv-python

pip install jupyter

pip install pytesseract

#### Step 4: Open Jupyter Notebook

Once all dependencies are installed, open Jupyter Notebook in the same terminal:

bash

jupyter notebook

This command will launch Jupyter Notebook in your default web browser.

#### Step 5: Run the Notebook

Navigate to the relevant Jupyter Notebook file within the repository and execute all the code blocks sequentially. This will initialize the YOLOv5 model, load the datasets, and run the vehicle and number plate detection algorithms.

**Dependencies**

Below is a list of essential packages required for the project. These should be included in your requirements.txt file or installed individually:

python numpy pytesseract

pandas matplotlib

opencv-python jupyter

**Contributory Segmentation of Project:**

**Preliminary Stage (Requirement Gathering and Analysis):** Carried by **Nikhil Kumar**

Building the project begun with some fixed objective. Having a proper roadmap on what and how to do and that too with the specified time period was needed. So, it was necessary to verify at each stage that we go with the correct order and aligned with the objectives. Any changes as and when found critical, were made.

**Stage 1 (Dataset Gathering):** Carried by **Rajan**

The task involved searching for various datasets across the internet. Keeping limited GPU support in all of our laptops, we could not process a large and very detailed image. The task thus required to collect datasets with images extensive enough and with enough classes to train the model upon.

**Stage 2 (Data Preprocessing and Model Building):** Carried by **Mamandeep Singh**

The task involved verifying the annotations of the images and labelling them wherever needed. To help it, labelImg was used. Model building involved testing various models so as to obtain optimal results. The model yoloV5 was finally chosen considering its high accuracy with real time data steam.

**Stage 3 (Model Training and Recording data for Analysis):** Carried by **Luxman**

The processed images were fed into the yoloV5 model for training on the image dataset. The model with optimal weights was given finally given a go. The moto of the project ends with collecting data for analysis. So, the lines of logic for storing the data (identified number plates) in CSV files is also written in this phase.

**Stage 4 (Documentation):** Carried by **Manjeet Kumar**

Documentation and report making required understanding of the flow of the project. It was only with proper communication among the members of the group that all of the document with just the right amount of information could be made. Also, tools like ChatGPT and QuillBot were helpful in proper phrasing of the sentences at times.

**Results and Discussion**

The vehicle movement analysis system was successfully developed and tested within the college campus environment. The primary objectives were to detect vehicles, recognize number plates, and store the data for further analysis. The results are summarized as follows:

1. **Vehicle Detection**:
   * The YOLOv5 model showed high accuracy in detecting vehicles from images and video streams.
   * The model correctly identified various types of vehicles including cars, , and motorcycles.
   * The detection accuracy was maintained across different lighting conditions and angles.
2. **Number Plate Recognition**:
   * The number plate recognition system was able to accurately extract and read number plates from detected vehicles.
   * The recognized number plates were stored in a CSV file for record-keeping and further analysis.
3. **Performance Metrics**:
   * **Precision**:
   * **Recall**:
   * **F1 Score**:
4. **Real-Time Processing**:
   * The system was capable of processing video streams in real-time, making it suitable for applications in traffic management and surveillance.

The results show that our system is effective for use on a college campus, providing reliable vehicle and number plate detection. Here are some key descriptions:

1. **Model Performance**:
   * The YOLOv5 model's high accuracy and efficiency make it great for real-time applications. It can detect vehicles under various conditions, showing it is robust.
   * The number plate recognition system was also accurate but had some trouble with plates that were partially covered or very dirty. We can improve this by using better image processing techniques.
2. **Application in Traffic Management**:
   * The system's ability to process in real-time allows for effective traffic monitoring and management. By analysing vehicle movement patterns, the campus administration can improve traffic flow and reduce congestion.
   * Automatically recording number plates can help monitor parking violations and manage access control.
3. **Security Enhancement**:
   * Integrating this system into campus security can significantly improve surveillance. It can help identify vehicles and track their movements, identifying unauthorized entries and potential security threats.
   * The stored data can be used to analyse vehicle movements during specific times.
4. **Smart Parking Solutions**:
   * The system can automate entry and exit management in parking facilities, making parking operations more efficient. By identifying and recording number plates, it ensures only authorized vehicles access restricted areas.
   * This leads to better use of parking spaces and a better experience for students, faculty, and visitors.
5. **Future Work**:
   * Future improvements could focus on better number plate recognition in difficult conditions, like low light or bad weather.
   * Expanding the dataset to include more diverse vehicle types and number plate formats can make the model more versatile.
   * Adding features like vehicle speed estimation and anomaly detection can provide more comprehensive traffic management and security solutions.

**Conclusion**

The vehicle movement analysis system based on the YOLOv5 model is an extremely successful tool for monitoring vehicle movements, improving security, and optimizing parking on a college campus. Its excellent detection and number plate recognition accuracy ensures that traffic flow is reliably monitored and managed. The system's real-time processing capability enables fast response to traffic circumstances, minimizing congestion and increasing overall campus safety.  
  
By incorporating this device into the campus security architecture, surveillance capabilities are considerably improved. The automatic recognition and recording of vehicle details helps to monitor unwanted or suspicious activity, resulting in a safer campus environment. The stored data can also be used to do thorough post-event analysis.

The system's application in parking management is especially useful. It automates the entrance and leave operations, guaranteeing that only authorized vehicles can enter restricted areas, maximizing parking space use and improving user experience.   
  
Overall, the YOLOv5 model's dependability and efficiency make it a viable option for a variety of campus applications. Its design enables for future improvements, such as greater recognition in difficult settings and the inclusion of new capabilities like speed estimates. This makes the vehicle movement analysis system a useful and versatile tool for both present and future requirements.