**Project Name:** Road Detection System

**Github Link:** https://github.com/projectsforstudents2022/Road\_Detection\_System.git

**Why was this project created?**

Safety precautions and error minimization are crucial if autonomous vehicle driving technology is to advance from the testing stage to a real self-driving car. We will estimate the road width using Python and machine learning algorithms. Additionally, it will indicate the separation from the side of the road and serve as a guidance for passing vehicles. Since the vehicle receives suitable help for passing and parking as well, the trained machine becomes the powerful feature in the autonomous automobile.

**What problem is it solving?**

This project illustrates how to use the lane detection method to delineate the road lanes. To specify the relative location of the vehicle in the designated lane, the width of the road lanes can be explicitly computed. It forecasts the trajectories that were recorded during human management of the vehicle and uses these to generate automatic tags for training a model that predicts pathways using semantics.

**Entire explanation of project**

* **PROPOSED APPROACH**

The data set was gathered across several drives and in a variety of weather conditions. 10 frames per second are recorded by a Pi camera over a 30-minute period. the images gathered by using OpenCV's cv2.resize() function to extract frames from images. After that, masking, gray-scaling, and Gaussian blurring are applied to the remaining image. The image needs to be transformed into a single color scale, sometimes known as grayscale, using edge detection and the Hough transformation. After the image has undergone grayscaling, noise reduction is applied. The image has been blurred or smoothed using a gaussian kernel. It is accomplished using the OpenCv function cv2.GaussianBlur(). It is necessary to specify the kernel's width and height, which must be positive and odd.

All of the aforementioned are combined into a single function called cv2 by OpenCV. In order to obtain the first derivative in the horizontal direction, Canny() for the smoothed picture is then filtered using a Sobel kernel in both the horizontal and vertical directions. A linear line equation is represented in the parametric form using the Hough Transform. If a shape is represented mathematically, it can be detected. HoughLines P is an improvement on the Hough Transform known as Probabilistic Hough Transform. The many lines that the Probabilistic Hough Transform discovered are not lane lines on the road. Two filtering techniques filtering using slope value and filtering using intercept value are utilized to eliminate the undesired lines. The y-intercept value for each produced line is determined to eliminate false positives.

Algorithm for creating next word prediction model :

**Step 1:** Import Libraries & Load Dataset

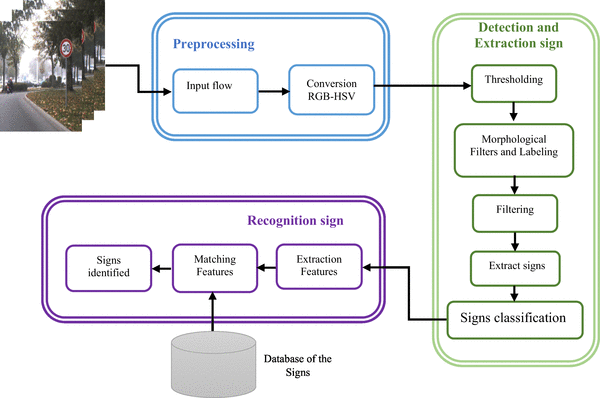
**Step 2:** Data Preprocessing

**Step 3:** Gaussian Blur

**Step 4:** Canny Edge Detection

**Step 5:** Train Model

**Step 6:** Testing & Visualization

* **DATA FLOW DIAGRAM**
* **RESULT**



* **CONCLUSION**

The issue of automated overtaking on highways was resolved. In order to create an automated driving system for autonomous highway overtaking, a test protocol for highway overtaking assistance was created. The autonomous driving system was tested virtually while the designed test regimen was analytically validated using mathematical equations. The outcomes of the simulation were discovered to be consistent with the desired host vehicle behavior.