Date: Feb 20, **2025**

**Project: Lane Line Detection on the Road**

# Problem Statement

# Modern transportation systems are evolving rapidly, with a significant shift toward automation and intelligent driving assistance technologies. One of the critical tasks in autonomous vehicles is the ability to detect road lanes accurately to ensure safe navigation. Human drivers rely heavily on lane markings for proper alignment, turning, and overtaking. However, automating this capability presents several challenges due to variations in lighting, weather conditions, faded markings, shadows, and curved roads.

# The objective of this project is to design and implement a computer vision-based lane detection system that can process video input and identify lane lines in real-time. The system must handle varying road conditions and detect both white and yellow lane markings. By applying image processing techniques such as color space conversion, edge detection (Canny), region of interest masking, and line detection using the Hough Transform, the system aims to provide a reliable solution for identifying lane boundaries on roads. This project serves as a foundational step toward developing advanced driver assistance systems (ADAS) and autonomous navigation tools.

# Abstract

This project implements a lane detection system using computer vision techniques. It processes video frames to identify lane lines on the road using edge detection, color filtering, and the Hough Transform. The goal is to develop a fundamental understanding of visual processing techniques used in self-driving car systems.

# Introduction

Lane detection is a crucial component of driver assistance and autonomous driving systems. By identifying the road boundaries, the system can assist in vehicle control, warning systems, and navigation. This project leverages traditional computer vision techniques to detect lane markings from road video footage.

# Tools & Libraries Used

- Python

- OpenCV

- NumPy

- Matplotlib

- MoviePy

# Theory Behind the Project

## 1. Canny Edge Detection

Canny Edge Detection is used to detect edges in the image where the pixel intensity sharply changes. It helps isolate the road lines from the rest of the image.

## 2. Color Masking

Lane lines are usually white or yellow. By applying a color filter in the RGB, HSV, or HLS space, we isolate relevant colors to make edge detection more accurate.

## 3. HSV and HLS Color Space

HSV (Hue, Saturation, Value) and HLS (Hue, Lightness, Saturation) are color spaces that separate chromatic content from intensity. This helps in better color filtering under varying lighting conditions.

## 4. Region of Interest

Not all parts of the image are useful. A polygonal mask selects only the region in front of the car where lanes are expected.

## 5. Gaussian Smoothing

This helps reduce noise and smoothens the image for better edge detection. It uses a kernel (matrix) of odd size.

## 6. Hough Transform

The Hough Transform is used to detect straight lines by transforming points in image space to a parameter space. It helps identify lane lines after edge detection.

# Code Explanation

The program loads a video, processes each frame by converting color spaces, filtering colors, detecting edges, and then applying Hough Transform to detect lines. The lines are drawn back on the image and returned as video.

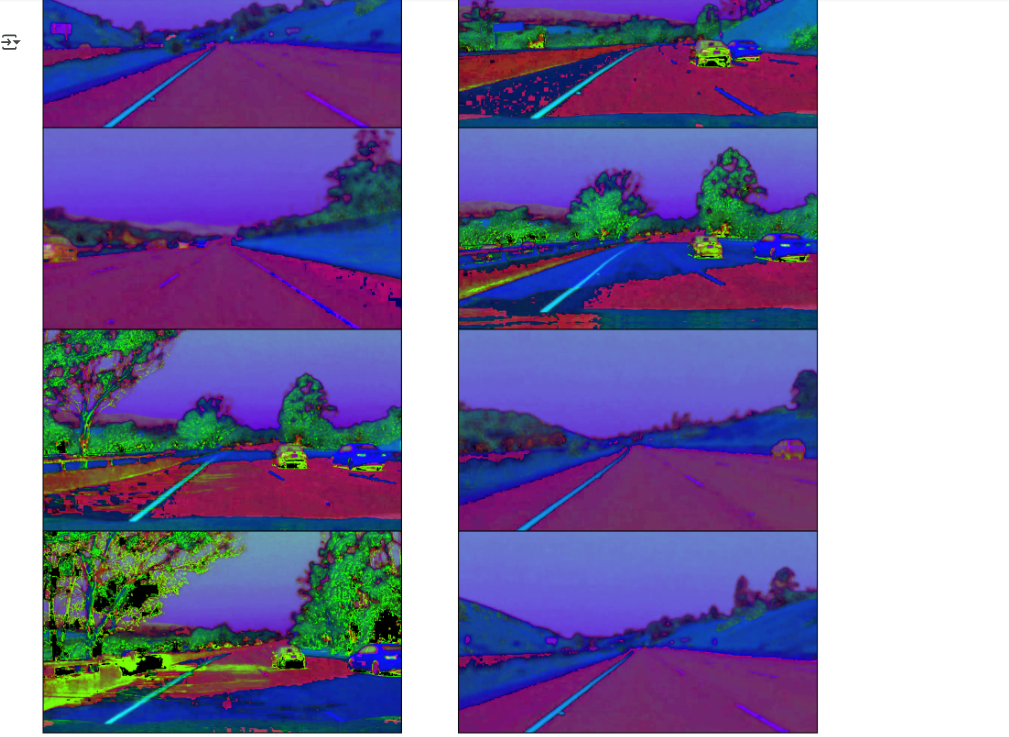
Functions like `select\_rgb\_white\_yellow`, `convert\_hsv`, `convert\_hls`, and `apply\_smoothing` prepare the image. The `LaneDetector` class uses a deque to smooth the lane prediction over multiple frames using a buffer queue.

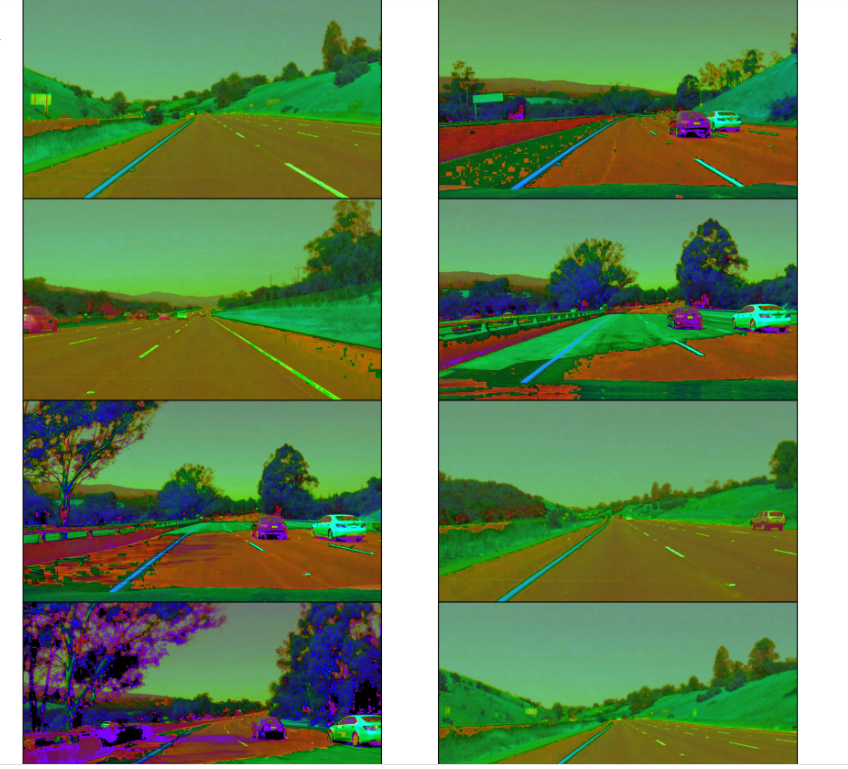
# Results

**Color Selection**

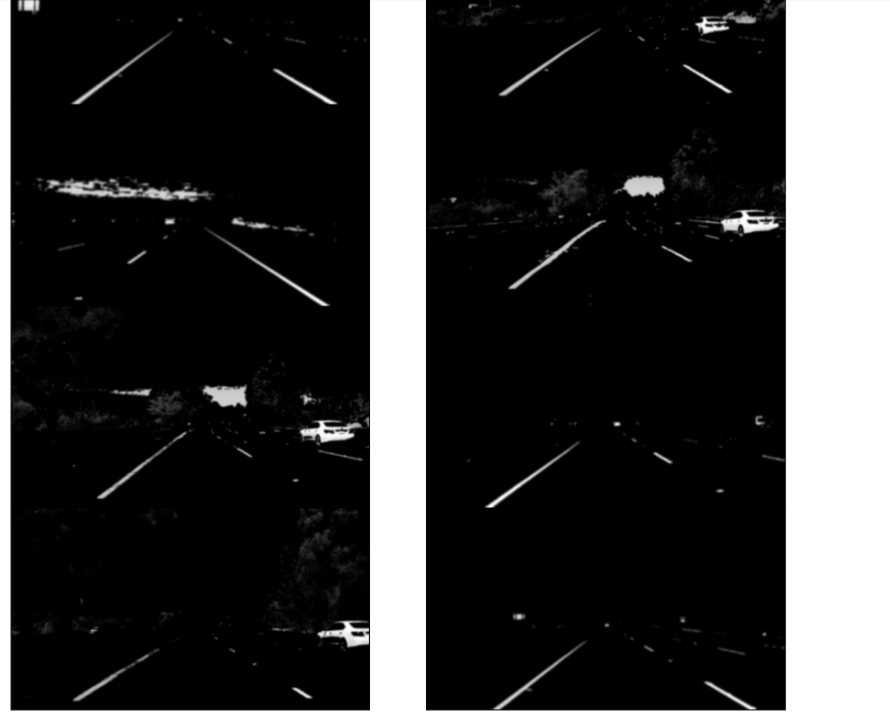


**HSL and HSV Color Space**

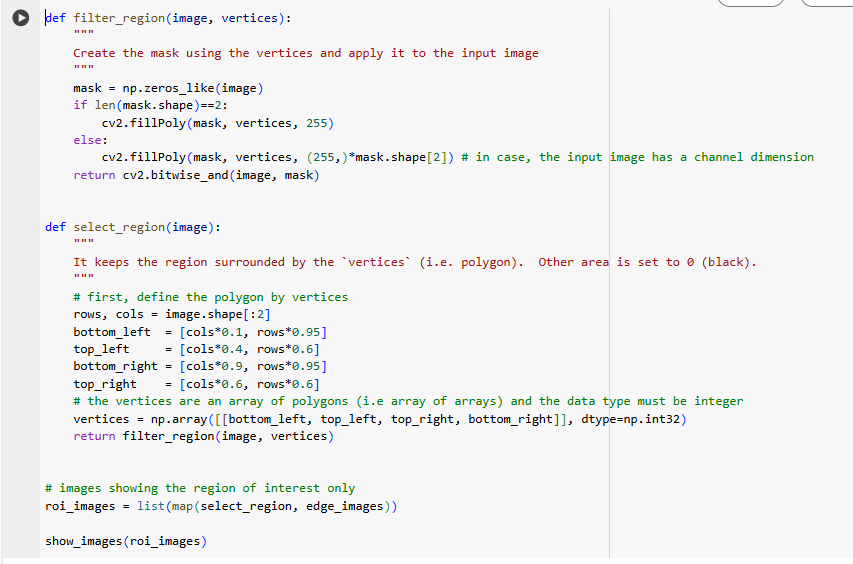


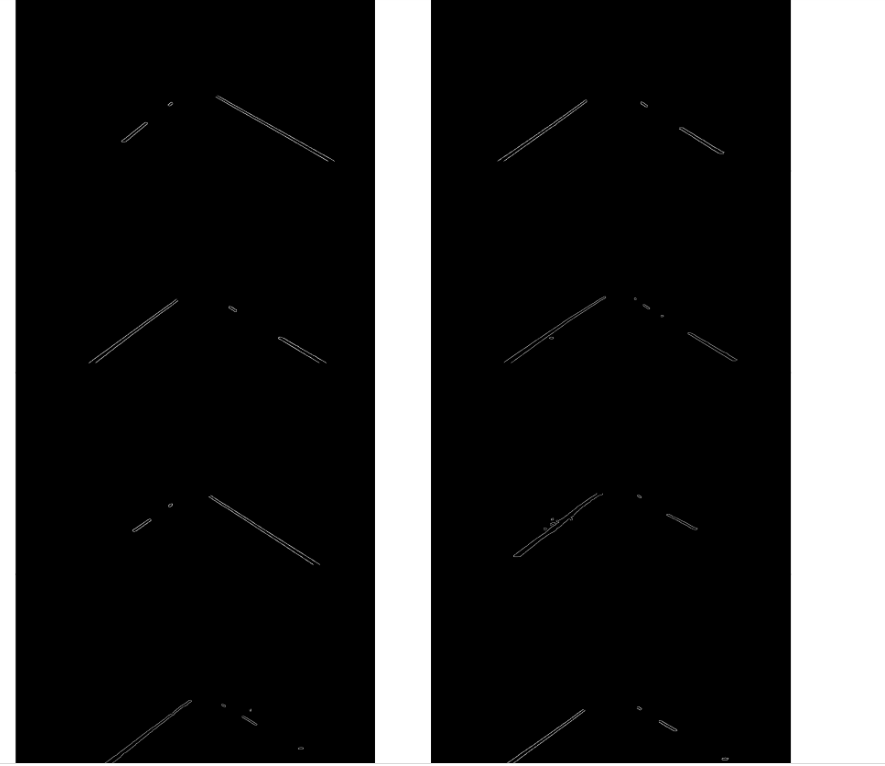


**Gaussian Smoothing (Gaussian Blur)**

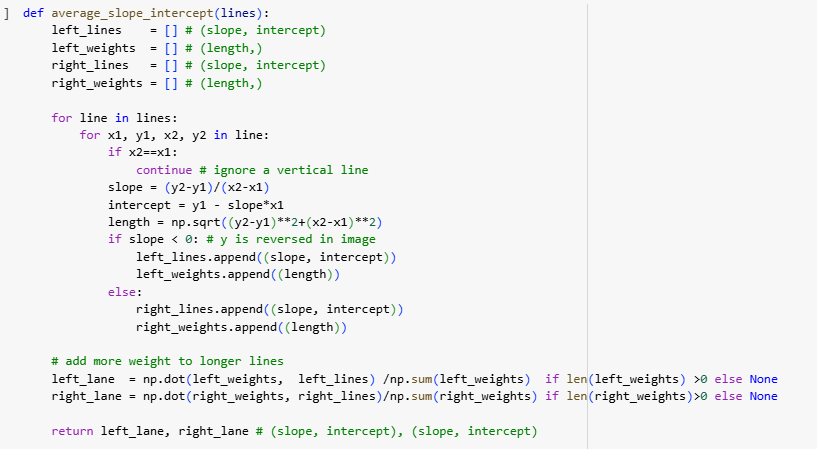


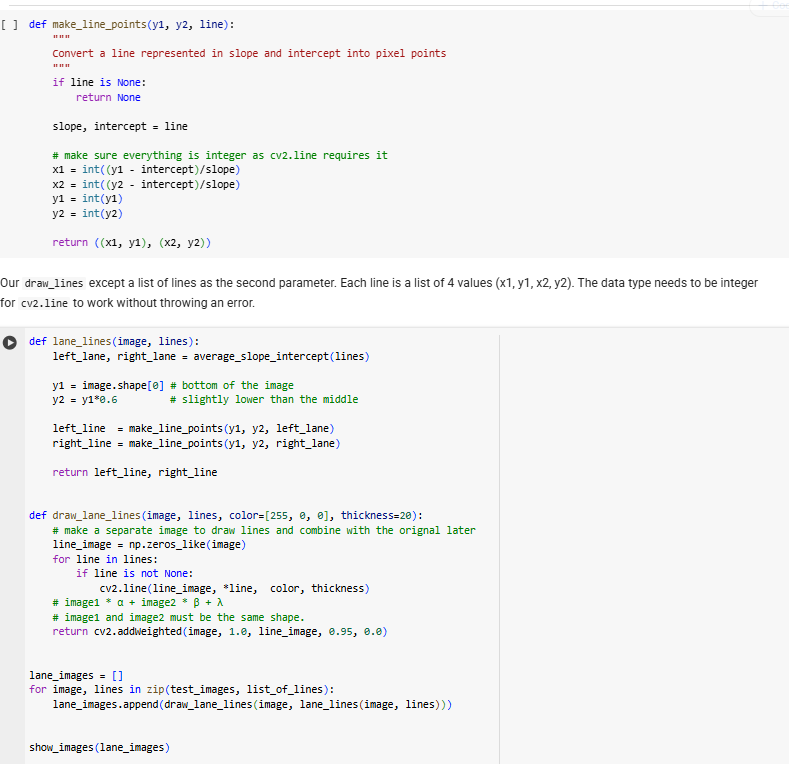
Region of Interest Selection





**Final and Hough Transformation applying**

****

****

****

The system successfully detects and overlays lane lines on video frames. It handles straight roads well but may struggle on curved or faded lanes. The processing is done in real-time using MoviePy.

# Conclusion

This project demonstrates the effectiveness of traditional computer vision techniques for lane detection. By combining edge detection, color filtering, and Hough Transform, the system achieves robust results for straight roads.