Final Project Report

Image Processing Course

Faculty of Computers and Data Science

Department of Intelligent Systems

Team Members:

Fares Hazem

Peter Hany

Instructors:

Dr. Mahmoud Gamal

Eng. Amany

Alexandria, Egypt

Submission Date: 21 December 2024

Abstract

This project investigates the application of advanced image processing techniques, including **noise reduction**, **segmentation**, and **edge detection**, to solve real-world challenges across diverse domains.

In the domain of *autonomous vehicles*, the focus is on **lane detection** in road images, leveraging datasets such as *Carla*. By applying robust algorithms, we aimed to enhance image quality, identify critical regions of interest, and extract meaningful features.

This report delves into the methods, parameters, and challenges encountered throughout the project. It also highlights the solutions implemented and the insights gained, underscoring the transformative potential of image processing in these vital applications.

Keywords: Image Processing, Noise Reduction, Segmentation, Edge Detection, Autonomous Vehicles, Lane Detection

1 Introduction

Lane detection is a crucial component of autonomous driving systems, enabling vehicles to safely navigate roads by identifying and following lane markings. As self-driving technology continues to evolve, accurate lane detection is vital for ensuring safety and efficiency on the road.

This project explores the application of various image processing techniques to solve the challenge of lane detection in road images. Specifically, we apply multiple methods to detect road lanes, including:

- **Noise Reduction:** Eliminating unwanted disturbances to improve image clarity and accuracy.
- **Segmentation:** Isolating lane markings from the surrounding environment for better detection.
- **Edge Detection:** Identifying boundaries of lane markings using techniques like Canny edge detection.

The dataset used for this project comes from *Carla*, a well-known simulation platform for autonomous driving research. We aim to demonstrate how combining these techniques can enhance lane detection performance in varied road conditions.

Report Structure: This report is organized as follows:

1. **Methodology:** Describes the techniques used for lane detection, including noise reduction, segmentation, and edge detection.
2. **Results and Discussion:** Analysis of the lane detection results and evaluation of the techniques' effectiveness.
3. **Conclusion and Future Work:** Summary of findings and suggestions for improving lane detection systems.

2 Methodology

2.1 Noise Reduction

Noise reduction is a crucial step in preparing images for further processing. We applied various techniques to reduce noise while preserving important image features. The following filters were used:

Mean (Average) Filter

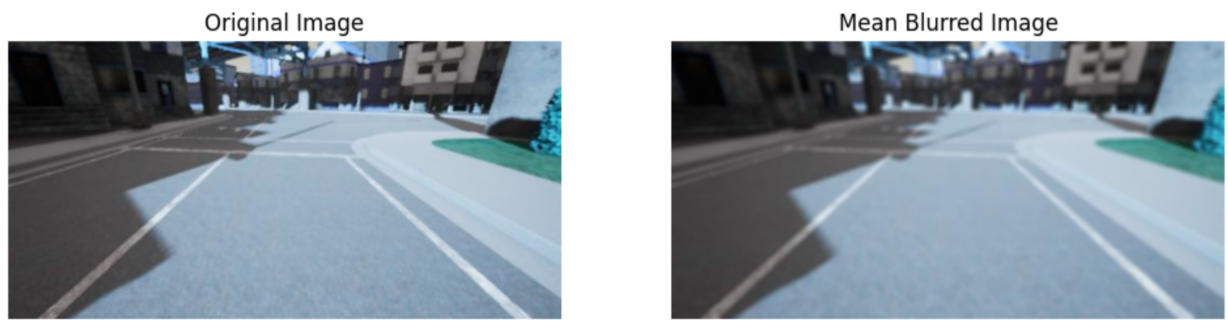

Figure 1: Mean Filter Result.

Median Filter *(Best Result)*

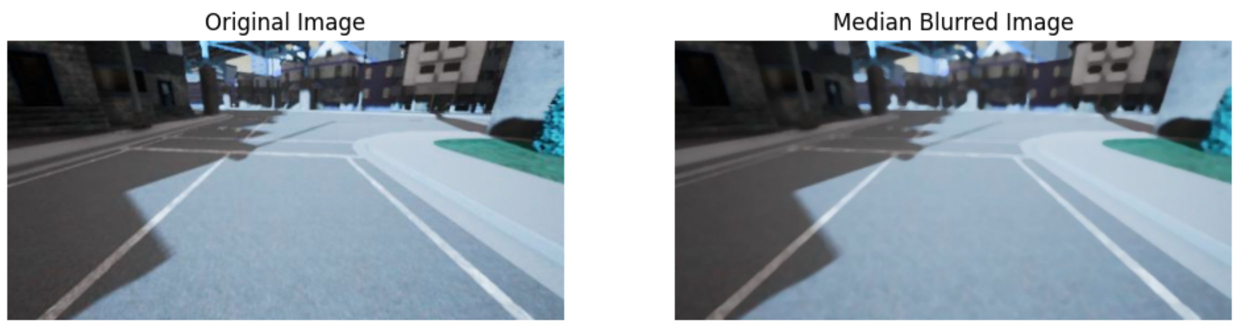

Figure 2: Median Filter Result (Best Result).

Maximum Filter

Figure 3: Maximum Filter Result.

Minimum Filter

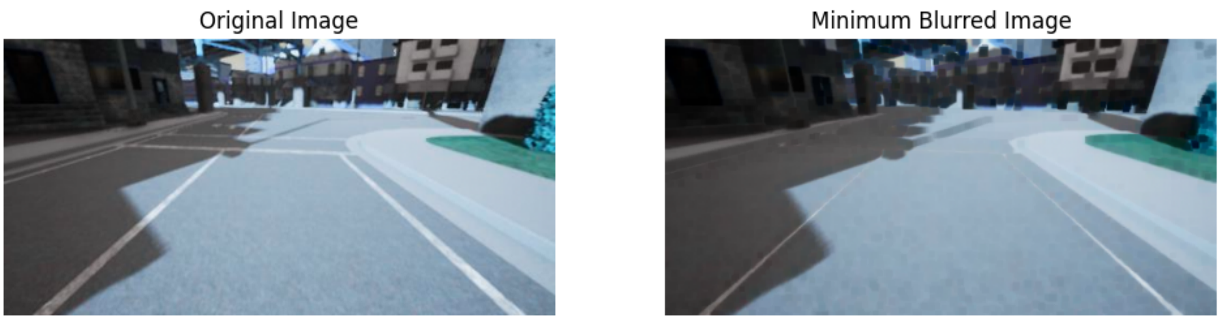

Figure 4: Minimum Filter Result.

Gaussian Blur Filter

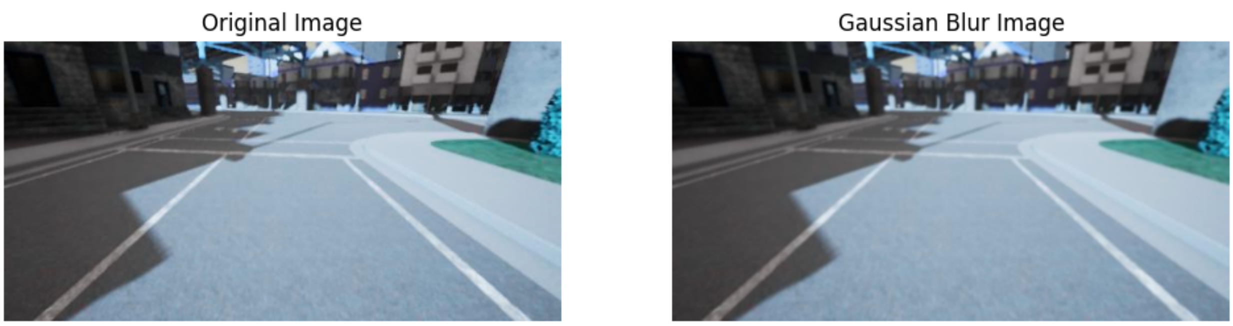

Figure 5: Gaussian Blur Filter Result.

After visualizing the results, the **Median Filter** was found to provide the best results, effectively reducing noise while preserving lane markings.

2.2 Contrast Enhancement

Contrast enhancement is essential for improving the visibility of important features in images. Two techniques were applied to enhance the contrast:

Linear Contrast Stretching *(Best Result)*

Figure 6: Linear Contrast Stretching Result (*Best Result*).

Histogram Equalization

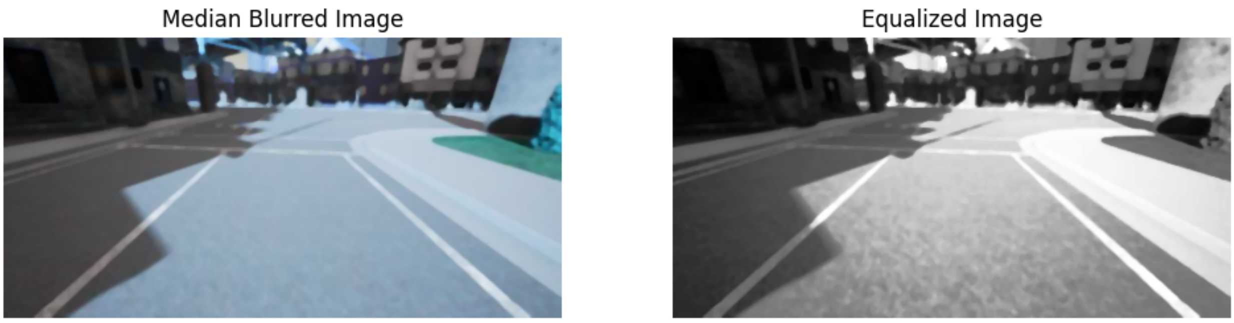

Figure 7: Histogram Equalization Result.

After visualizing the results, it was found that *Linear Contrast Stretching* provided the best results for lane visibility.

2.3 Image Segmentation

Image segmentation is crucial for separating relevant features from the background. We tested several thresholding and region-based methods:

- **Histogram-Based Methods**
- **Region-Based Methods**

Histogram-Based Methods

Simple Thresholding *(Intermediate Result)*

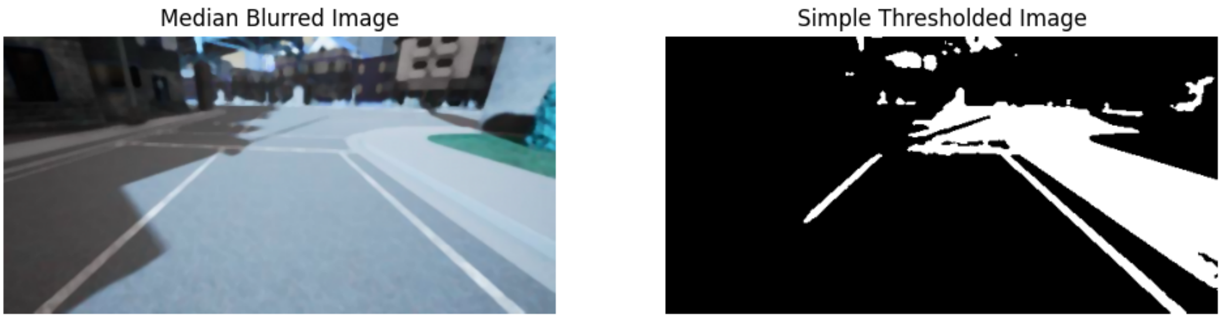

Figure 8: Simple Thresholding Result.

Global Thresholding *(Bad Result)*

Figure 9: Global Thresholding Result (*Bad Result*).

Adaptive Thresholding *(Intermediate Result)*

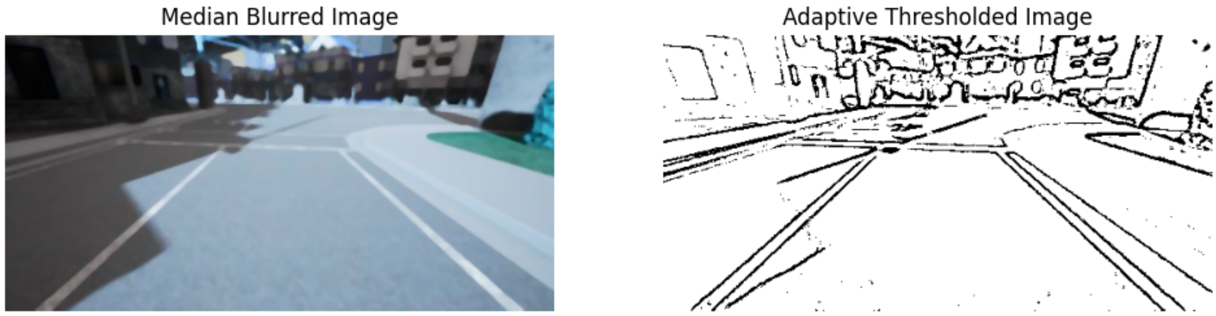

Figure 10: Adaptive Thresholding Result (*Intermediate Result*).

Otsu Thresholding *(Bad Result)*

Figure 11: Otsu Thresholding Result (*Bad Result*).

Region-Based Methods

Clustering *(Bad Result)*

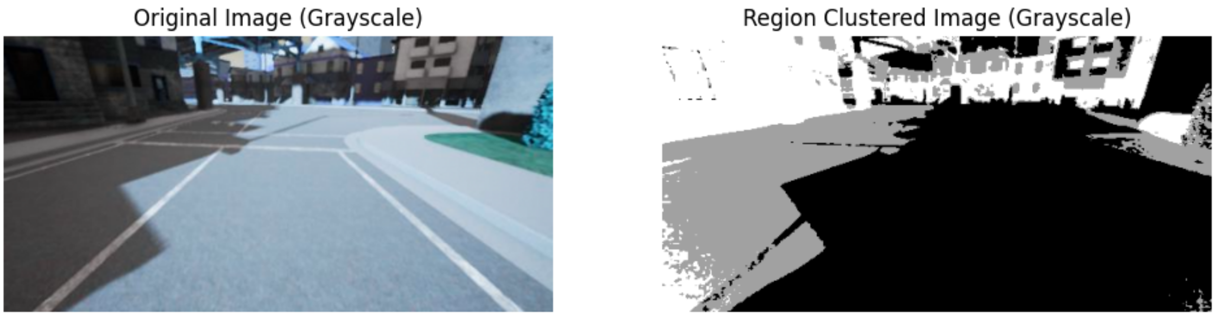

Figure 12: Clustering Result (*Bad Result*).

Growth Region *(Bad Result, Needs Optimization)*

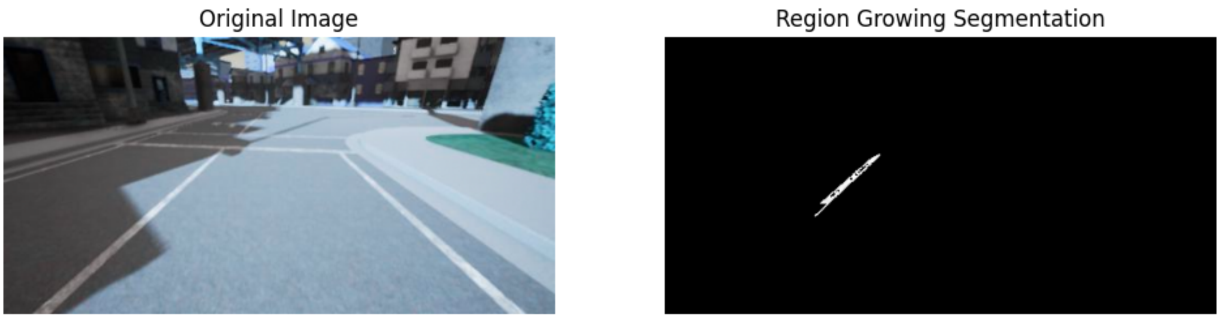

Figure 13: Growth Region Result (*Needs Optimization*).

Split & Merge *(Bad Result, Needs Optimization)*

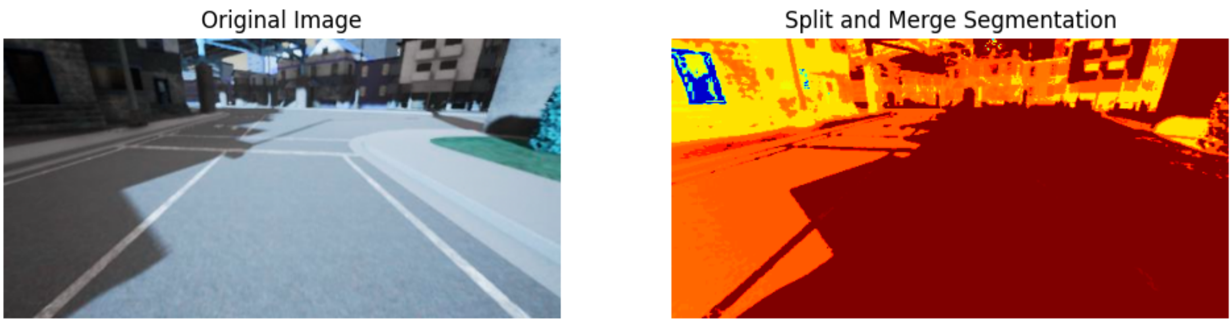

Figure 14: Split & Merge Result (*Needs Optimization*).

2.4 Edge Detection

Edge detection is essential for detecting lane boundaries and other important features. We compared several methods:

Sobel Filter *(Intermediate Result)*

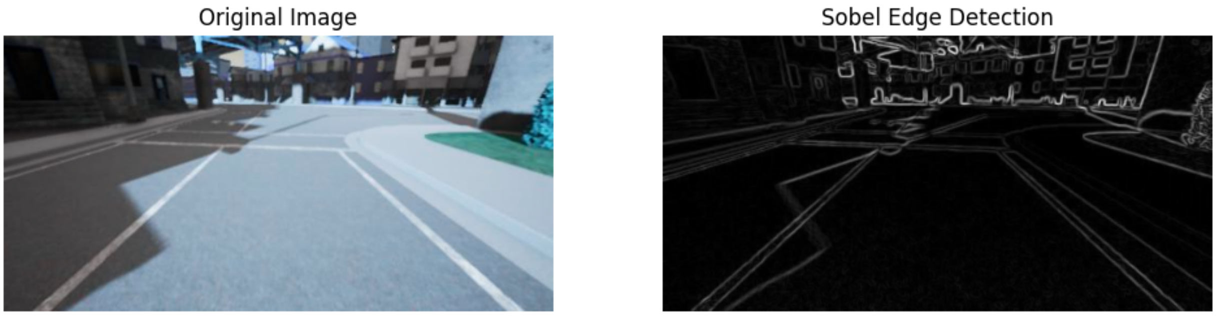

Figure 15: Sobel Filter Result.

Prewitt Filter *(Intermediate Result but Very Noisy Image)*

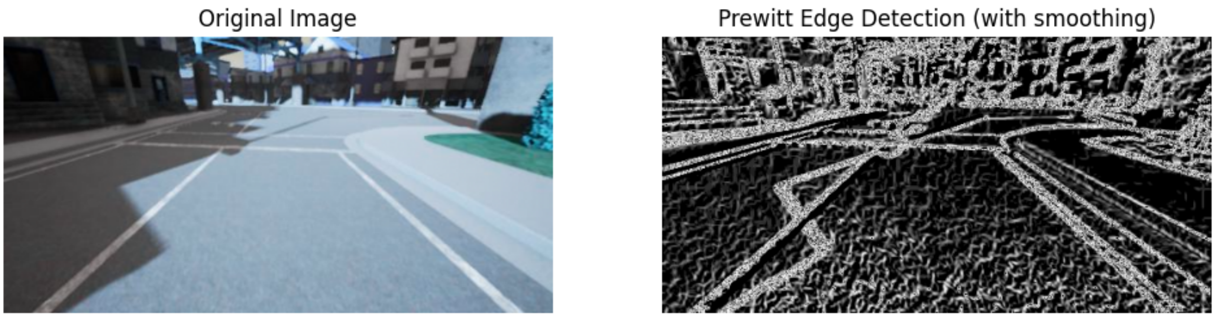

Figure 16: Prewitt Filter Result (Very Noisy Image).

Canny Edge Detection with Morphological Operations *(Best Result)*

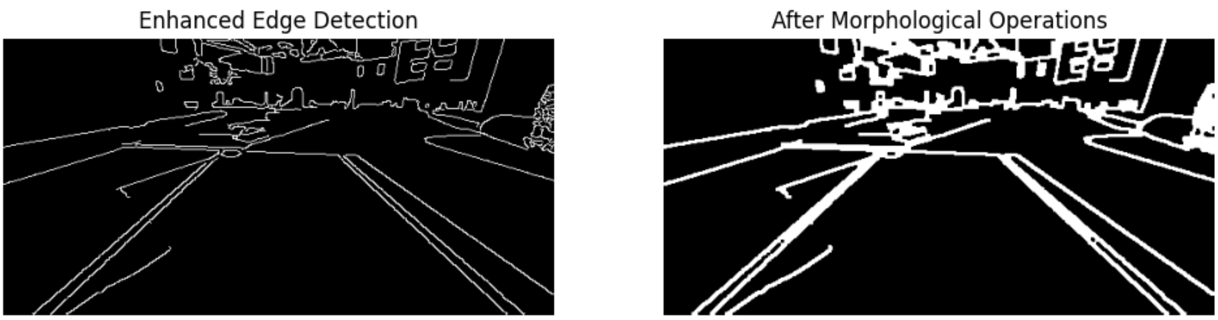

Figure 17: Canny Edge Detection with Morphological Operations (Best Result).

3 Final Lane Detection Results

The final lane detection results were achieved by combining noise reduction, contrast enhancement, segmentation, and edge detection. Below are the images showing the original image with keypoints, birdseye image, and thresholded output.

On Carla Dataset:

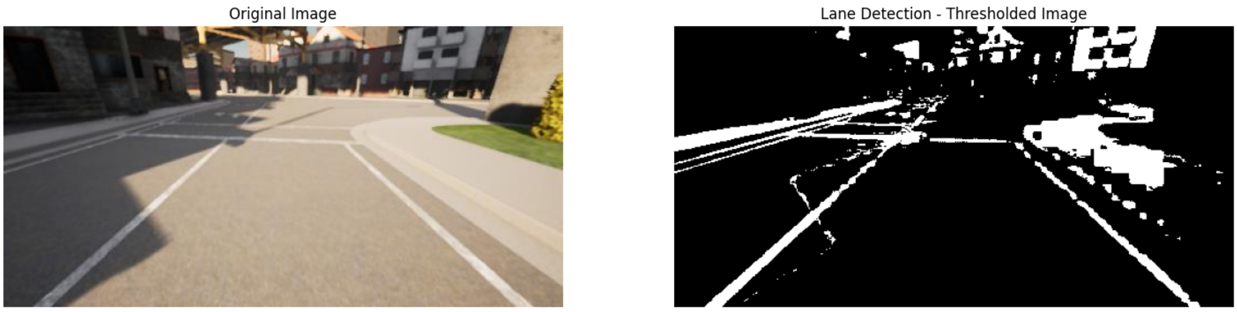

Figure 18: HSV Thresholding Result.

On Video Found Online:

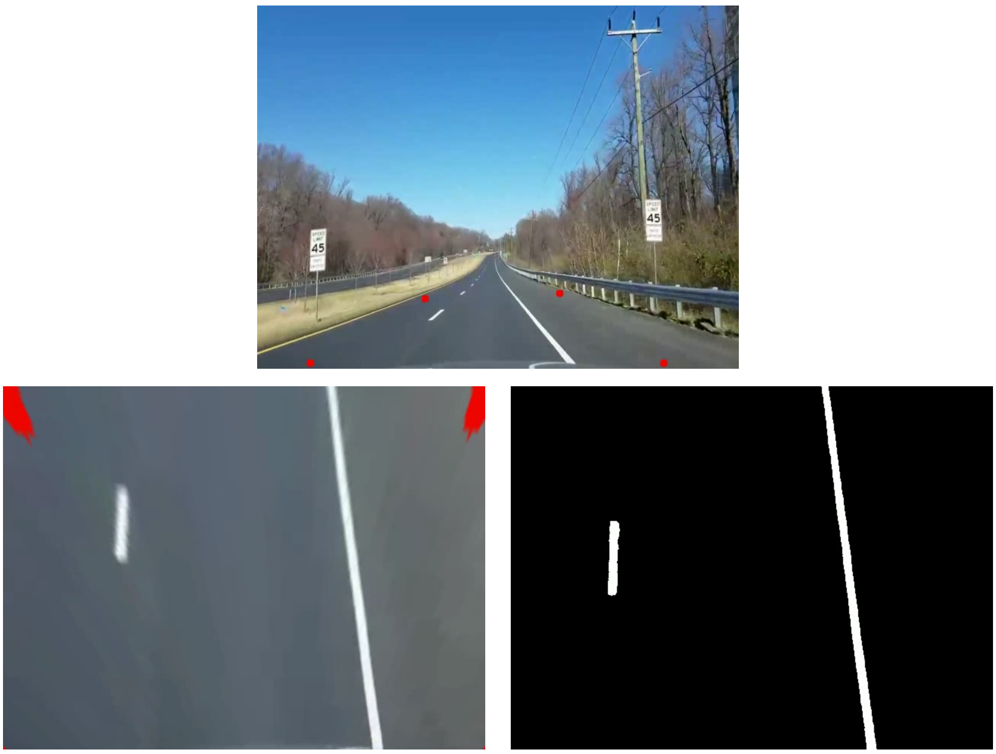

Figure 19: Lane Detection Results: Comparison of HSV Thresholding, Original Image with Keypoints, Birdseye Image, and Thresholded Output.

4 Discussion

- **Challenges faced:**
  - Image processing techniques are quite **old** and **hard to apply** because they require careful **selection**, **tuning**, and **optimization** of parameters to achieve the best results. This process can be **time-consuming**.
  - **Lighting issues** in the dataset made the processing more difficult, especially when applying filters to the images. A **cleaner dataset** would have been preferable for better results.
- **Solution applied:** We focused on **HSV Thresholding**, which proved to be more efficient than the traditional methods. **Tuning** the parameters for HSV is simpler and faster.
- **Analysis of results:** The **HSV Thresholding method** provided a **better result**, especially when compared to other image processing methods that require more complex fine-tuning.

5 Conclusion and Future Work

- **Summary of findings:**
  - The image processing techniques used, though effective, require a lot of **manual tuning** and can be **time-consuming**.
  - **HSV Thresholding** emerged as a more **efficient** and **easier-to-tune** method, giving more consistent results with less effort.
- **Suggestions for future work:**
  - The use of **AI** could lead to more **consistent** and **generalized results** across different datasets and lighting conditions.
  - Once the best solution is found, it should be **applied to the entire dataset** to improve accuracy and reliability.

6 References

- **Cited works:**
  - Aditya Singh's lane detection project on GitHub: https://github.com/Aditya-Singh-SSJ2/Lane_Detection_Sliding_Windows/tree/main
- **Dataset:**
  - Lane Line Detection Dataset (Kaggle): https://www.kaggle.com/datasets/hyunkunkookminuniv/lanelinedetection