“VISION”

A Project Work Synopsis

Submitted in the partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING
IN
CSE (AIML)

Submitted by:
JAHANVI CHAWLA
UID – 18BCS6108

SEJAL GULATI
UID – 18BCS6116

SHIVAM
UID – 18BCS6138

SACHIN KAUSHIK
UID – 18BCS6119

YUVRAJ KOUNDAL
UID-18BCS6114

Under the Supervision of:

MONIKA SINGH

CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413, PUNJAB

SEPTEMBER - 2021

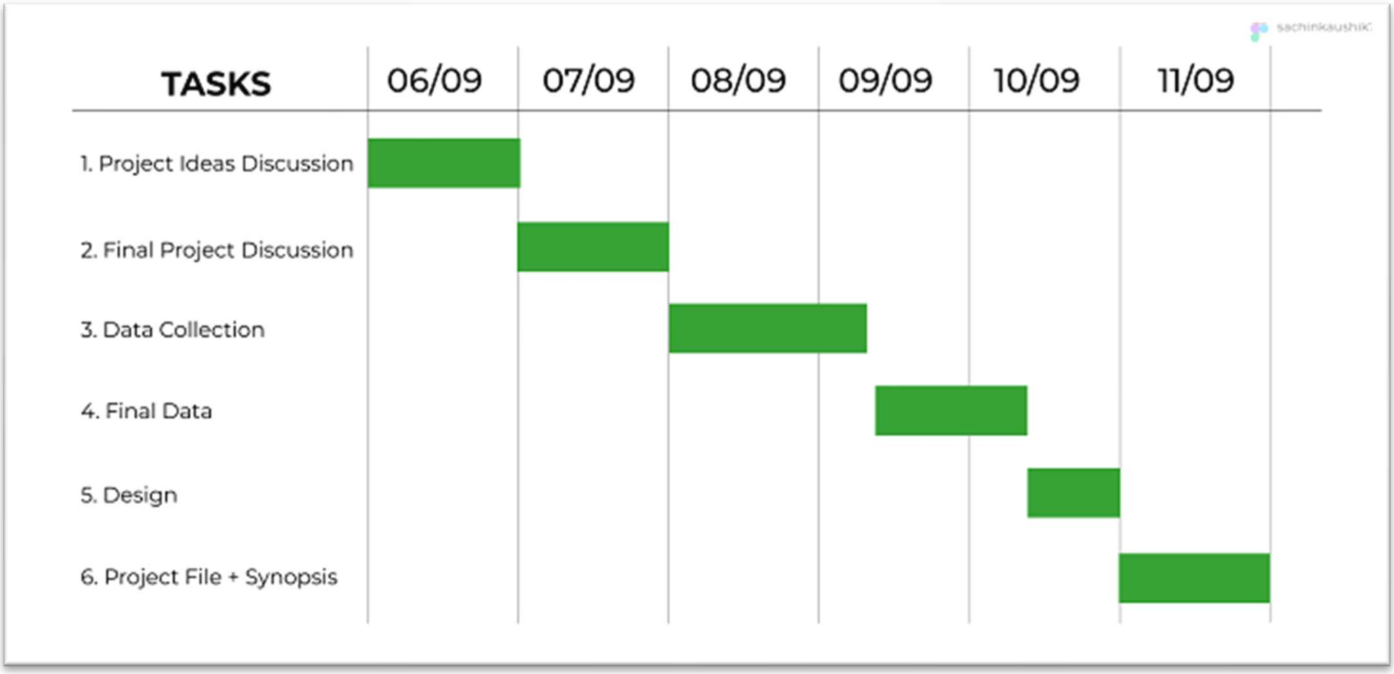
Abstract

Lane detection is a difficult problem to solve. For decades, it has piqued the interest of the computer vision community. Lane detection is essentially a multifeature detection problem that has proven to be difficult for computer vision and machine learning techniques to solve. Despite the fact that various machine learning algorithms are employed for lane detection, they are mostly used for classification rather than feature construction. Modern machine learning algorithms, on the other hand, can be used to find features with high recognition value and have shown to be successful in feature detection tests. However, these strategies have not been fully implemented in terms of lane detection efficiency and accuracy.

We present a new preprocessing and ROI selection method. The main purpose is to extract white features using the HSV color transformation, add preliminary edge feature detection in the preprocessing step, and then pick ROI based on the proposed preprocessing. The lane is detected using this new preprocessing method.

TABLE OF CONTENTS

GANTT CHART


TASKS
06/09
07/09
08/09
09/09
10/09
11/09
1. Project Ideas Discussion
2. Final Project Discussion
3. Data Collection
4. Final Data
5. Design
6. Project File + Synopsis

1.
INTRODUCTION

1.1 Problem Definition

Lane detection is a prominent topic in machine learning and computer vision, and it's been used in smart vehicle systems. The lane detection system gathers data from lane markers in a complex environment and uses it to accurately predict the vehicle's position and trajectory in relation to the lane. Automobiles have been one of the primary modes of mobility for people as society has progressed. There are an increasing number of automobiles of various types on the small route. Every year, as more automobiles enter the road, the number of people killed or injured in car accidents rises. The question of how to drive safely when there are a lot of cars on the road and the roads are narrow has become a hot topic. Lane departure warning (LDW), Lane Keeping Assist (LKA), and Adaptive Cruise Control (ACC) are examples of advanced driver assistance systems that can help people analyze their current driving environment and provide appropriate feedback for safe driving or notify the driver in dangerous situations. This type of auxiliary driving system is predicted to improve with time.

1.2 Problem Domain

The difficulty in predicting the road traffic environment, however, is a bottleneck in the development of this system. According to the findings, the likelihood of accidents is significantly higher than typical in the complex traffic environment when vehicles are many and speeds are excessive. The key perceptual signals of human driving in such a complex traffic condition are road color extraction and texture recognition, as well as road boundary and lane marking. Lane detection, on the other hand, is critical in the lane departure warning system. The task of lane detection is split into two parts: edge detection and line detection. In lane detection, line detection is just as crucial as edge detection. When it comes to line detection, we normally use one of two ways: feather-based methods or model-based methods.

In this paper, we suggest a lane recognition method that is suitable for a wide range of complicated traffic conditions, particularly where road speeds are excessive. We first preprocessed each frame image before selecting the processed images' region of interest (ROI). Finally, for the ROI area, we only needed an edge detection vehicle and line detection.

2.
LITERATURE REVIEW

We provided a new preprocessing method and ROI selection mechanism in this approach. First, we transformed the RGB color model to the HSV color space model and retrieved white features from the HSV model during the preprocessing stage. Simultaneously, preliminary edge feature detection is added to the preprocessing step, and the ROI area is chosen from the area below the picture based on the proposed preprocessing. Existing preprocessing methods simply conduct operations like greying, blurring, X-gradient, Y-gradient, global gradient, thresh, and morphological closure when compared to existing approaches. Furthermore, there are numerous methods for determining the ROI area. Some of them select the ROI region based on the lane's edge feature, while others select the ROI area based on the lane's color feature. These existing technologies do not give reliable and timely lane information, making lane identification more difficult. Experiments in this research show that the suggested strategy outperforms the existing preprocessing and ROI selection methods in lane detection.

Improve the efficiency and accuracy of real-time lane detection using enhanced lane detection technology. Typically, the lane detection module is split into two parts:

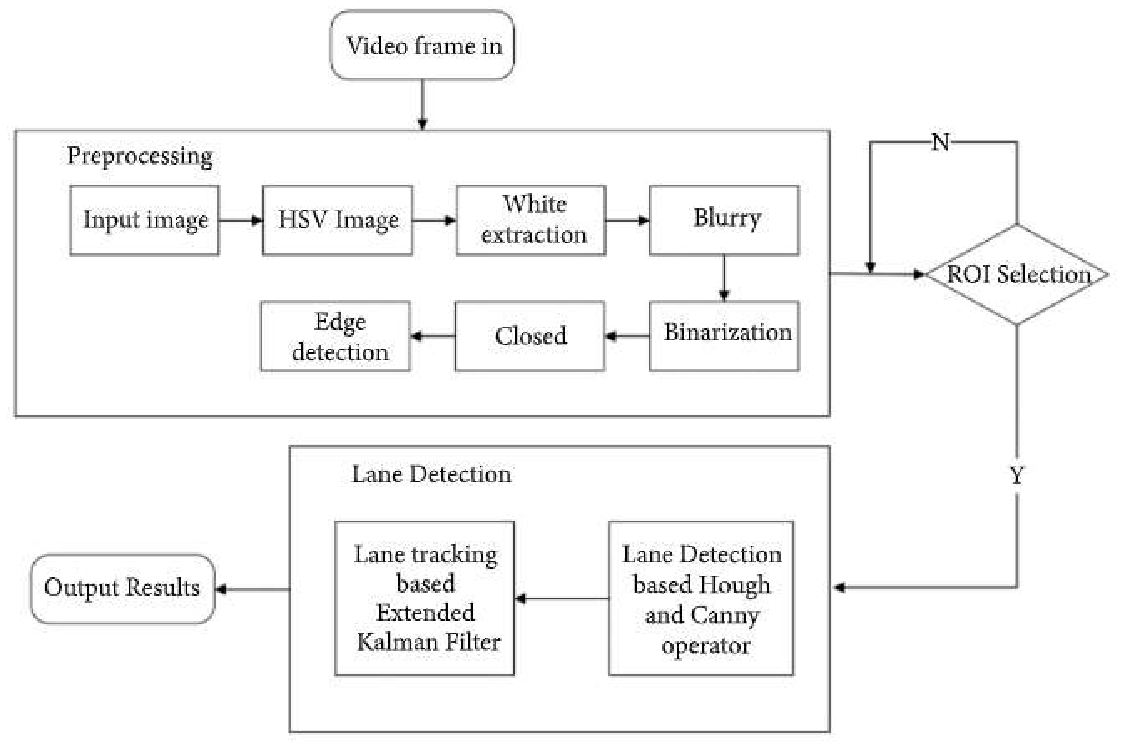
- picture preprocessing
- line lane detection model creation and matching

Figure 1 depicts the overall diagram of our proposed system, with the key contributions of this research being the lane detecting blocks. Reading the frames in the video stream is the first step. The image preparation module is entered in the second stage.

What sets us apart from the competition is that during the preprocessing step, we not only process the image but also extract color and edge features. After extracting the color information of the image, we need to smooth the image with a Gaussian filter to limit the influence of noise in the motion and tracking process. After that, binary threshold processing and morphological closure are used to create the image. These are the approaches for preprocessing that have been mentioned.

Figure 1

Block diagram of proposed methods.



The adaptive region of interest (ROI) in the preprocessed image is then selected. Lane detection is the final phase. The Canny operator is used to detect the edge of the lane line, followed by the Hough transform to detect the line lane. Finally, to detect and track lane lines in real time.

3.
PROBLEM FORMULATION

The already existing problem in automated vehicles is perceiving lane such that they never tend to cross over to the opposite side or to turn to an illegal lane. Hence, this very project helps the automated i.e. the self-driving vehicles to keep track of the lane and keep a easy flow of the traffic. Since, it could not be possible with the help of tools such as Machine Learning and Computer Vision, an extensive use of such technology had been used in it to bring out the maximum accuracy and hence, the safety and reliability of our project.

4.
METHODOLOGY

4.1 Preprocessing

Preprocessing is a key aspect of image processing as well as lane detection. Preprocessing can help minimize the algorithm's complexity, minimizing the time it takes to run the programme later. The video input is a RGB-based color image sequence obtained from the camera. Many studies use various picture preprocessing approaches to increase the accuracy of lane detection.

A typical picture preparation approach is to smooth and filter graphics. The basic goal of filtering is to reduce image noise and improve the image's effect. For 2D images, low-pass or high-pass filtering can be utilized; low-pass filtering (LPF) is useful for denoising, and image blurring and high-pass filtering (HPF) are used to determine image boundaries. An average, median, or Gaussian filter could be employed to execute the smoothing procedure. Xu and Li employ a median filter to filter the image and then utilize an image histogram to enhance the grayscale image in order to preserve detail and remove unnecessary noise.

4.2 Color Transform

Color model transform is an important aspect of machine vision, and it's also crucial for lane detection in this paper. The actual road traffic environment, as well as the intensity of light, all cause noise that obstructs color detection. We can't tell the difference between white lines, yellow lines, and vehicles in the backdrop. The RGB color space used in the video stream is particularly sensitive to light intensity, therefore processing light at different periods has an undesirable effect. The RGB sequence frames in the video stream are color-converted to HSV color space images in this article. Figures 5(a) and 5(b) depict RGB and HSV color spaces, respectively.

HSV (hue, saturation, and value) is an abbreviation for hue, saturation, and value. When compared to other colours, the values of white and yellow colours in the V-component are particularly brilliant and easy to extract, providing a good foundation for the following colour extraction. Experiments reveal that colour processing in the HSV space is more reliable when it comes to detecting individual targets.

Figure 5

Two images of different colour spaces. (a) RGB and (b) HSV colour transform.

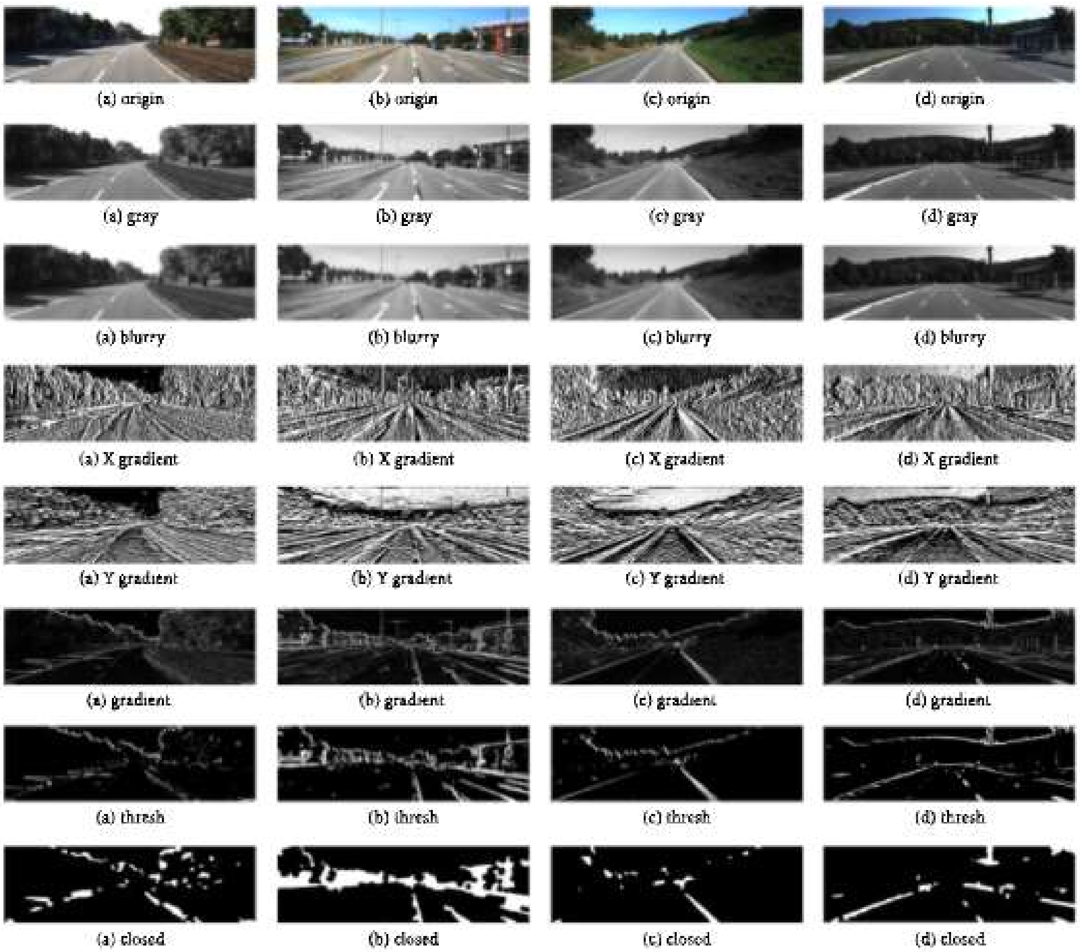
(a) RGB

(b) HSV

4.3 Basic Preprocessing

The video will be preprocessed for a huge number of frames. Individual photos are gray-scaled, blurred, and have their X-gradients, Y-gradients, global gradients, thresh of frame, and morphological closure determined. During the preprocessing phase, an adaptive threshold is established to account for various illumination situations. After that, we delete the spots in the binary conversion image and do the morphological closing procedure. Basic preprocessed frames aren't likely to be very effective in removing noise. Although preliminary lane information can be collected following the morphological closure, the results show that there is still a significant amount of noise.

Basic preprocessing of sample frames (a), (b), (c), and (d).

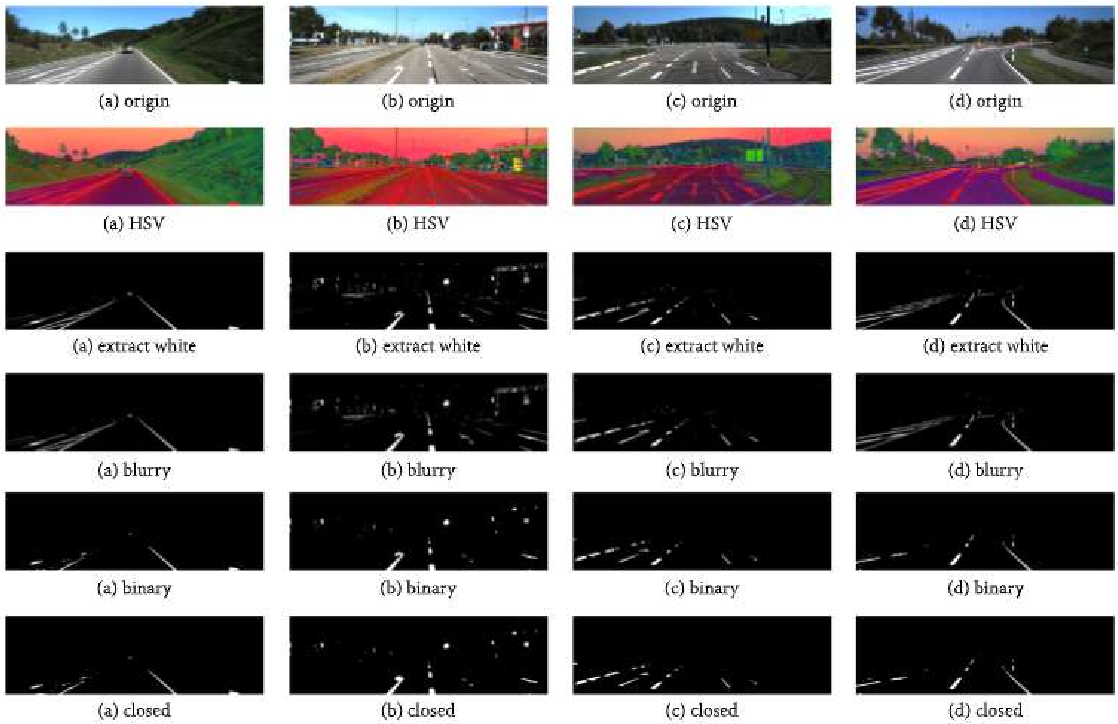


4.4 Color Extraction

We add a feature extraction module to the preprocessing stage to improve the accuracy of lane detection. The goal of feature extraction is to maintain any lane-related features while removing nonlane-related ones. This work focuses on extending feature extraction to color.

We add the white feature extraction after the image greying and color model conversion, and then do the traditional preprocessing operations one by one. The color extraction process presented in this paper is depicted in Figure below.

Adding white extraction in preprocessing of sample frames (a), (b), (c), and (d).

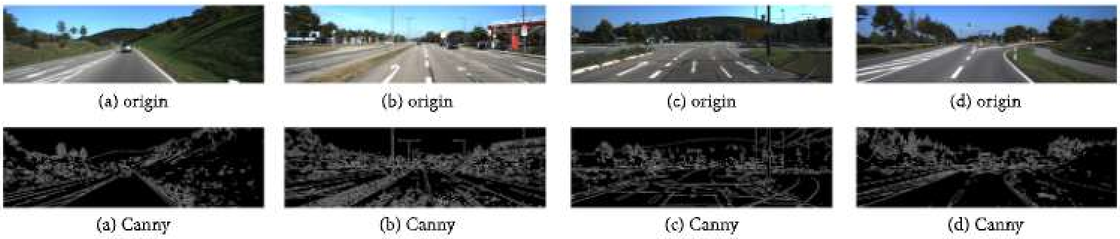


4.5 Edge Detection

Edge detection two times in a row; the first time is to extract a wide range of edges throughout the full frame image. The edge detection is done again after the lane identification and ROI selection in the second. This detection enhances lane detecting accuracy even more. Using the updated Canny edge detection technique, this portion primarily performs overall edge detection on the frame image. The following are the concrete steps of Canny operator edge detection:

- We smooth the image (preprocessed image) with a Gaussian filter, then calculate the gradient magnitude and direction with the Sobel operator.
- The non-maximal value of the gradient amplitude must then be suppressed.
- Finally, we must detect and link edges using a double-threshold technique. Figure displays the image after Canny edge detection was used to extract it.

Canny edge detection of sample frames (a), (b), (c), and (d).

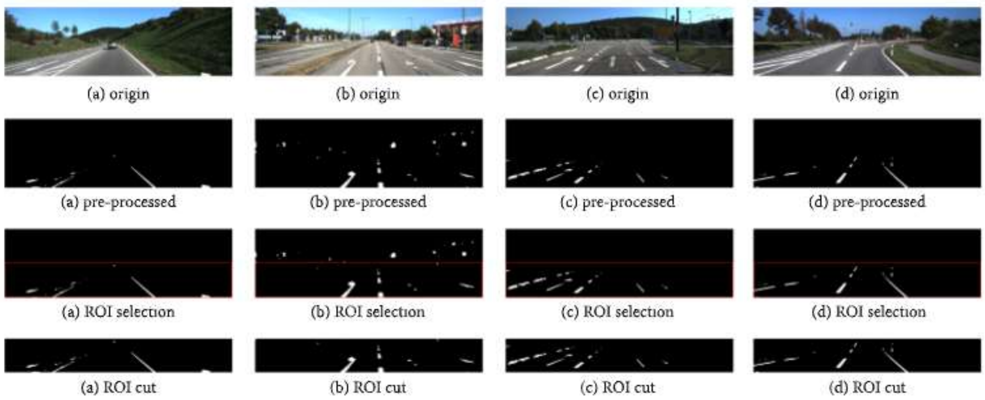


4.6 ROI Selection

We can observe that the obtained edge comprises not only the required lane line edges, but also other unnecessary lanes and the edges of the surrounding fences after edge detection by Canny edge detection. To get rid of these unnecessary edges, identify a polygon's visual area and only save the visible region's edge information. The camera is fixed in relation to the car, and the car's relative location to the lane is similarly fixed, so the lane is essentially maintained in a fixed area in the camera.

We can use an adaptive area of interest (ROI) on the image to reduce image redundancy and algorithm complexity. The input image is only set on the ROI area, and this strategy can improve the system's speed and accuracy. The standard KITTI road database is used in this paper. Each frame of the vehicle's running video is divided into two pieces, with one-half of the lower section of the image frame serving as the ROI area. The ROI selection of sample frames (a), (b), (c), and (d) treated by the suggested preprocessing is shown in Figure. After being processed by the proposed preprocessing method, the images of the four separate sample frames were able to substantially display the lane information, although the upper half of the image contains a lot of nonlane noise in addition to the lane information. As a result, the ROI area was chopped out of the lower half of the image (one-half).

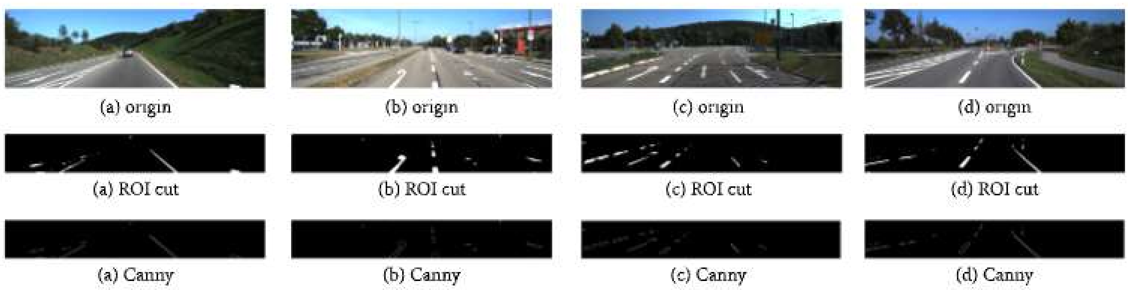
ROI selection of sample frames (a), (b), (c), and (d) based on the proposed preprocessing.



4.7 Feature Extraction

For lane detection, feature extraction is critical. Canny transform, Sobel transform, and Laplacian transform are some of the most frequent edge detection methods. We chose Canny transform because it is superior. Following the proposed ROI selection, we performed Canny edge detection, as seen in the figure.

Canny edge detection of sample frames (a), (b), (c), and (d) after the proposed preprocessing.



4.8 Lane Detection

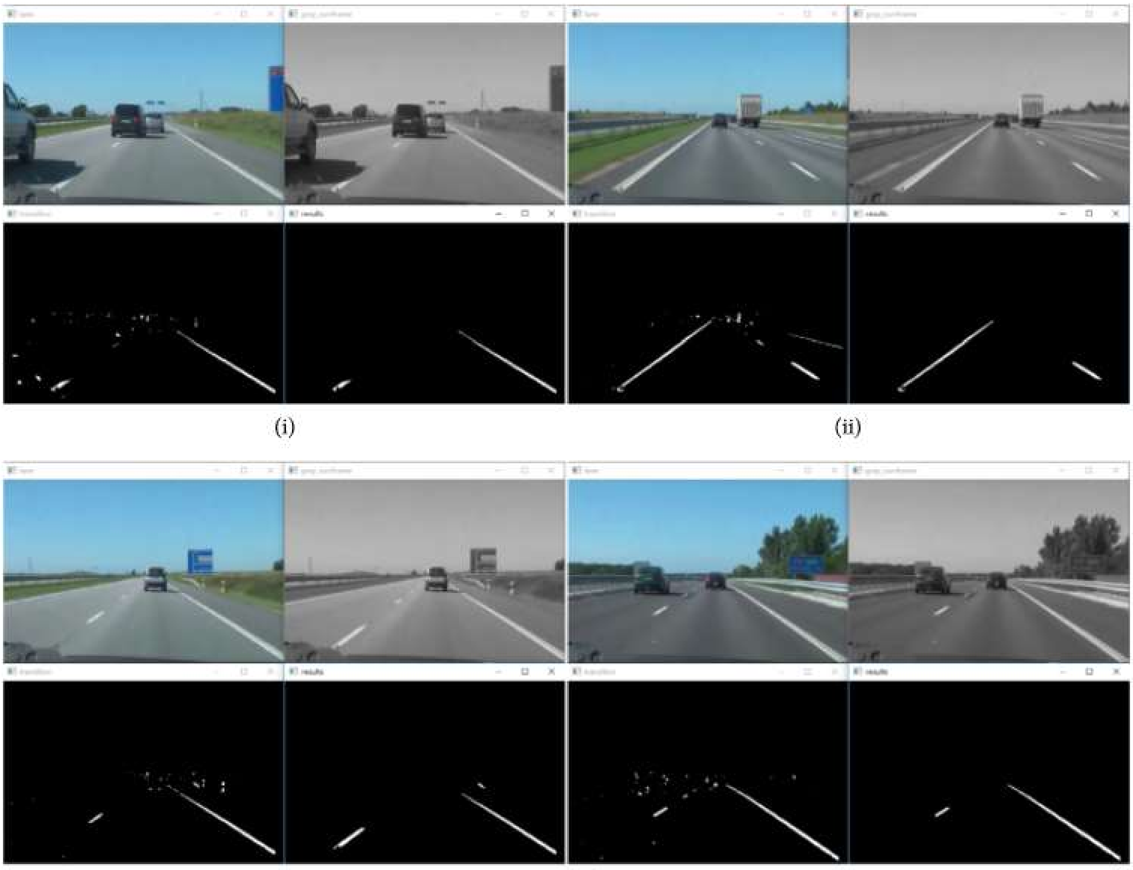
Lane detection is divided into two types: lane edge detection and linear lane detection. This part provides the fundamental lane detection functions and conducts lane detection based on the proposed ROI selection and better preprocessing.

Lane detection methods include feature-based and model-based approaches. In this study, the method-based feature is utilised to detect the colour and edge features of lanes in order to increase lane detection accuracy and efficiency.

Straight lane detection can be accomplished in two ways. One option is to draw lane lines in the corresponding area of the original image using the Hough line detection function provided by the OpenCV package, which is widely used for image processing. Self-programming is the alternative option. The ROI area is traversed in the header file to perform line detection for a given range of angles.

The video shows both approaches, with the first method running faster. We chose the first technique (Hough line function in the OpenCV package) to run faster for linear detection because this post focuses on the accuracy and efficiency of lane detection. Furthermore, the Hough transform is employed to extract lane line parameters in each frame of the image series for lane detection because it is noise-insensitive and can analyze straight lines well.

Different moments (i), (ii), (iii), and (iv) in one video.



(i) (ii) (iii) (iv)

5.
CONCLUSION AND FUTURE SCOPE

In a nutshell, this project will successfully be able to perceive lanes and would be providing a smooth flow of automated vehicles. The future scope of its potential is that it would be used in almost every self-driving vehicle such that it increases its accuracy and use domain over the time. In the near future, eventually every vehicle would be integrated with an automated driving feature and for that to be possible our project i.e., '**VISION**' would be playing an important role.

Not limited to this, with an extensive reach of it, it could be applied to public transport vehicles so that they can be automated to drive themselves.