A Project Report

on

Lane Detection In Video for Intelligent Transportation System

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

Bachelor of Technology in Computer Science and Engineering

GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Mr. Dileep Kumar Yadav
Associate Professor

Submitted By

18SCSE1010632 – APOORVA SRIVASTAVA

18SCSE1010458 – ANMOL SHARMA

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GALGOTIAS UNIVERSITY, GREATER NOIDA, INDIA

DECEMBER - 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the project, entitled **" Lane Detection In Video for Intelligent Transportation System "** in partial fulfillment of the requirements for the award of the **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING** submitted in the School of Computing Science and Engineering of Galgotias University, GreaterNoida, is an original work carried out during the period of JULY-2021 to DECEMBER-2021, under the supervision of Mr. Dileep Kumar Yadav, Associate Professor, Department of Computer Scienceand Engineering of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the project has not been submitted by me/us for the award of any other degree of this or any other places.

18SCSE1010632 – APOORVA SRIVASTAVA

18SCSE1010458 – ANMOL SHARMA

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor

(Mr. Dileep Kumar Yadav,Associate Professor)

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of **18SCSE1010632–APOORVA SRIVASTAVA, 18SCSE1010458 – ANMOL SHARMA** has been held on ________and his/her work is recommended for the award of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING**.

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date:

Place:

ABSTRACT

Automatic lane detection to help the driver is an issue considered for the advancement of Advanced Driver Assistance Systems (ADAS) and a high level of application frameworks because of its importance in drivers and passerby safety in vehicular streets. But still, now it is a most challenging problem because of some factors that are faced by lane detection systems like as vagueness of lane patterns, perspective consequence, low visibility of the lane lines, shadows, incomplete occlusions, brightness and light reflection. The proposed system detects the lane boundary lines using computer vision-based technologies. In this paper, we introduced a system that can efficiently identify the lane lines on the smooth road surface. Gradient and HLS thresholding are the central part to detect the lane lines. We have applied the Gradient and HLS thresholding to identify the lane line in binary images. The color lane is estimated by a sliding window search technique that visualizes the lanes. The performance of the proposed system is evaluated on the road dataset. The experimental results show that our proposed method detects the lane on the road surface accurately.Detecting and recognizing objects in unstructured environments is one of the most challengingtasks in computer vision research. We propose an innovative algorithm, called deformable illumination, to address the problem of illumination variance in natural environments.

Lane detection is a critical component of self-driving cars and autonomous vehicles. It is one ofthe most important research topics for driving scene understanding. Once lane positions are obtained, the vehicle will know where to go and avoid the risk of running into other lanes or getting off the road. This can prevent the driver/car system from drifting off the driving lane.

In this project we aim to do a Lane detection pipeline to mimic Lane Departure Warning systemsused in Self Driving Cars. To achieve this, the following steps are taken:

Computed the camera calibration matrix and distortion coefficients of the camera lens used given a set of chessboard images taken by the same camera . Use color transforms, and sobel algorithm to create a thresholded binary image that has been filtered out of unnecessary information on the image . Apply perspective transform to see a “birds-eye view” of the imageas if looking from the sky . Apply masking to get the region of interest, detect lane pixels, Determine the best fit curve for each lane the curvature of the lanes. Project the lane boundaries back onto the undistorted image of the original view. And then, Output a visual display of the lane boundaries and other related information.

“A lane is part of a roadway (carriageway) that is designated to be used by a single line of vehicles, to control and guide drivers and reduce traffic conflicts.”

Table of Contents

List of Figures

List of Tables

Acronyms

ML	Machine Learning
DL	Deep Learning
CV2	Open-Source Computer Vision Library
NUMPY	Numerical Python
RANSAC	Random sample consensus

CHAPTER-1

Introduction

Conferring to information [1], in Great Britain about of 70% all described road accidents are an outcome of driver fault or slow response period. Very recently, many researchers are working on the intelligent vehicles to reduce the road accidents and ensure safe driving. With the advent of technologies, numerous techniques have been developed to notify the drivers about conceivable lane departure or crash, the lane structure and the locations of additional automobiles in the lanes. The lane detection on the road surface has become a very prominent issue as it provides noteworthy symbols in the ambient situation of the intelligent vehicles. It is a challenging task to detect the lane as the input image become noisy due to the variation of the environment. In the very recent years, many of the researches associated with lane detection have been developed based on the use of the sensors, camera, lidar, etc. Clanton et al. [2] proposed a technique for multi-sensor lane departure warning that used roadmap with GPS receiver for road surface lane detection. The authors in [3] suggested a method using lidar with a monocular camera that was able to detect the lane in a real-time environment. The lidar or sensors obtained the data directly from the environment, and the devices are not dependent on the weather circumstances which are the main benefits of these techniques. However, the resolution of GPS is 10-15 m, and the cost of the lidar is comparatively very high, which are the significant disadvantages of the modalities. With the rapid growth of computer vision-based technology, the camera has become more popular that can capture any situation of the environment in any direction. As far as, the researches that have been done using vision-based techniques, the technique in [4] achieved the likely outcomes on the Caltech lane dataset. They introduced mainly three techniques for lane detection. These are the inverse perspective mapping (IPM) that is used to eliminate the perspective effect, image filtering that is used to eliminate noise through candidate lane generation, and lane model fitting that detects the lines on the road images. Besides, this technique concerns the model fitting, but the reliability and efficiency of the system are deserted here. The techniques that have been already proposed have some difficulties to detect lane in case the lanes are not fully visible. Some systems are developed based on the edge detection [5, 6, 18] that are reduced fewer accurate by lane-like noise because of the lens flare consequence. Moreover, the techniques in [7], based on color cues worked well although the illumination changes rapidly. The optical flow [8] can solve the problems that are occurred in the preceding techniques, but it needs high computational power and cannot promisingly work for the road without texture. Furthermore, some techniques have been developed using the concept of a vanishing point for lane detection on the road surfaces. Gabor filters with texture orientation were used in [9] to detect lane. They used an adaptive soft voting technique that estimated a vanishing point and demarcated the confidence rank of the texture orientation. To measure the performance of the lane detection for structured and unstructured roads, a modality using a vanishing point estimation have been introduced in [10]. However, with the change of location, the vanishing point cannot be estimated accurately which is the major drawbacks of these methods. In this paper, we introduced a computer vision-based techniques that can efficiently detect the lanes in any ambient environment. We have mainly used gradient and HLS thresholding for lane detection. For proper mapping, perspective transform has been applied after thresholding. The remaining part of the paper is prepared as follows. Section II presents the on-going as well as previous works that have been continued in this prominent area. The proposed lane detection technique with proper explanation is illustrated in Section III. The experimental results with input and output are depicted in Section IV. Section V finishes the paper. With the rapid development of society, automobiles have become one of the transportation toolsfor people to travel. lane detection is a multi feature detection problem that has become a real challenge for computer vision and machine learning techniques. According to the WHO, each year lives of approximately 1.25 million people cost as result of road traffic crash. Between 20 and 50 million people suffer from non-fatal injuries, which sometimes incur disabilities . Road traffic injuries bring considerable economic losses to victims, their families, and nations as a whole. Therefore, in 2016 many firms or corporation has declared that they were and will participate in the development of the automatic vehicle. Volvo Corporation has promised that by 2020, nobody will be faced seriously accident by one of its new cars by using driving assistance system and warning .

- The lane detection system comes from lane markers in a complex environment and is used to estimate the vehicle’s position and trajectory relative to the lane reliably.
- Lane detection plays an important role in the lane departure warning system.
- The lane detection task is mainly divided into two steps: edge detection and line detection.

1.1 Requirements, Feasibility and Scope/Objective

Objective

The developing recognition of Machine learning plays a significant function in a wide scope of basic applications, for example, information mining, common language preparing, picture recognition, and master frameworks. Since Machine learning gives a potential arrangement in every one of those territories, it is supposed to be a mainstay of future development. AI contains a variety of building calculations make the PC to gain from the information and settling on information-driven choicesjust as expectations. The fast development of AI from the previous few years welcome a huge impact on our everyday life with such instances of AI for insolvency expectation, precipitation determining, climate estimating, self-driving frameworks and optical character recognition and so forth. By consolidating AI approaches with counterfeit knowledge creates a superior outcome. Despite the fact that machine learning shows an excellent presentation, it isn't productive in human data handling frameworks, for example, discourse acknowledgment and PC vision. This can be overwhelmed by the genuine forefront of Machine learning is Deep Learning. Image processing includes some fundamental activities to be specific picture rebuilding/amendment, picture upgrade, image classification, pictures combination, and so forth. Image classification structures a significant piece of picture handling. The target of picture characterization is the programmed assignment of the picture to topical classes. Since 2006, deep structured learning, or all the more usually called deep learning or progressive learning, has risen as a new region of AI research. A few definitions are accessible for Deep Learning; covering one of the numerous definitions from Deep Learning is characterized as: A class of AI methods hat abuse numerous layers of nonlinear data handling for managed or solo highlight extraction and change and for design investigation furthermore, grouping. Computational models of neural networks have been near for quite a while, first model proposed was by McCulloch and Pitts as in . Neural networks are comprised of various layers with each layer associated with different layers shaping the network. A feed-forward neural network or FFNN can be thought of in terms of neural activation and the quality of the associations between each pair of neurons .

1.2 Requirements:-

We are going to research on one Problem that a Lane- Detection using python and OpenCV, so we required to use Principal Component Analysis algorithm, NumPy, Juypter notebook environment or pythonidle, knowledge of Python and its libraries, data cleaning and algorithms, frame differencing, image thresholding.

- PYTHON 3.6
- KERAS
- SCIKITS
- NUMPY
- PANDAS
- LINEAR REGRESSION

1.3 Feasibility Analysis :-

Feasibility studies reflect a project's unique goals and needs, so each is different. However, the tips belowcan apply broadly to undertaking a feasibility study

The driver behavior is the key to safety mobility. The dangerous driving can be categorizes into4 behaviors which are (1) rapid acceleration,

(2) sudden brake

(3) rapid turning and

(4) fast lanechange.

In general, all of the action can be determined from acceleration on the vehicle.

Physically, the acceleration and brake are longitudinal acceleration while turning and lane change are lateral acceleration. Normally, IMU (inertia measurement unit) has been designated to get those data. However, by experiences, the IMU is not convenience to install in the vehiclesespecially as aftermarket additional parts. Previously, the study on conventional navigation system with 1-Hz GPS technology had been confirmed that longitudinal acceleration can be easily detected

The overall objectives of the data collection and analysis efforts were to:

- Understand how driving-related metrics reflect the impairment associated with BAC at 0.05%and 0.10%
- Determine the robustness of these metrics with respect to individual differences such as ageand gender, as well as the roadway situation
- Identify signatures of impairment and develop algorithms to detect alcohol-related impairment
- Compare robustness of metrics and algorithms.

CHAPTER-2

Literature Survey

Several studies in computer vision have been proposed in the most recent years associated with lane detection. Many institutions have been working for a long time to propose efficient lane detection system. The researches related to this area are drawn briefly as follows. Lee and Moon [11] proposed a real-time lane detection algorithm with a Region of Interest (ROI) that is able to work with the high noise level and response in a shorter time. The system used the Kalman filter and a least square approximation of linear movement for lane tracking operation. The system detects the lane as well as track the lane also. Song et al. [12] presented a system that can detect the lane as well as classify them using the concept of stereo vision for the essence of Advanced Driver Assistance Systems (ADAS). They proposed a model to detect the lane using the idea of Region of Interest (ROI), and for the classification task, they used Convolutional Neural Network (CNN) structure that is trained with the KITTI dataset to classify the right or left lane. However, the system failed to detect the lanes as the disparity image was noisy. Wu et al. [13] designed a lane detection and departure warning scheme by determining the region of interest (ROI) in the region near to the automobile. The ROI is divided into non-overlapping chunks and to get the chunk gradients and chunk angles, two basic masks are developed that decrease the computational complexity. The driving situations are classified into four classes, and the departure system is developed with respect to lane detection outcomes. From the experimental outcomes.

and the departure warning rate is 98.60%. However, it takes comparatively high processing time due to computing the vertical and horizontal gradients. Yoo et al. [14] presented a lane detection technique based on the vanishing point estimation. The system used the probabilistic voting technique to detect the vanishing points of the lane segments at first. The actual lane segments were determined by setting the threshold of the vanishing points as well as the line direction. In addition, to evaluate the lane detection rate a real-time inter-frame similarity scheme was proposed that decrease the false detection rate. As the lane geometry properties do not vary expressively, the real-time assessment scheme was under the postulation. However, the system cannot be worked for shapeless roads. Ozgunalp et al. [15] introduced a vanishing point lane detection technique for multiple curved lanes. The system combined the disparity information with a lane marking technique that is able to estimate the PVP for a non-flat street condition. The redundant information of obstacles is removed by comparing the correct and fitted disparity values. Moreover, the estimation of PVP affected by the outliers at the operation of Least Squares Fitting and sometimes unsuccessful detection of lane happens due to the plus-minus peaks value selection. Piao et al. [16] developed a lane marking technique based on the binary blob analysis. To eliminate the perspective consequence from the road surfaces, the system used vanishing point detection and inverse perspective mapping. The binary blob filtering and blob verification methods are proposed to advance the effectiveness of the lane detection scheme. The outcomes of the system show that the average detection rate for the multi-lane dataset is 97.7%. However, the system failed to perform in a real-time environment. Jung et al. [17] developed a lane marking modality using spatiotemporal images which are collected from the video. The spatiotemporal image created by accruing a set of pixels that are mined on a horizontal scan-line having a static location in every frame along with a time axis. Hough transform is applied to the collected images to detect lanes. The system is very effective for short-term noises such as mislaid lanes or obstruction by vehicles. The system obtained the computational efficacy as well as the higher detection rate. Borkar et al. [18] proposed a lane detection technique based on inverse perspective mapping (IPM). The adaptive threshold technique is used to convert the input image to a binary image, and the predefined lane templates are used to select the lane marker candidates. RANSAC eliminated the outliers, and the Kalman filter tracked the lanes on the road surfaces. Kang et al. [19] introduced a multi-lane detection technique based on the ridge attribute and the inverse perspective mapping (IPM). The features that are extracted have four local maxima as the identical lane is disseminated with a minor variance on the x-axis in the IPM image. The clustering based techniques are used to detect the lanes by clustering the lane attributes nearby every indigenous supreme point. In this paper, the key idea is to firstly detect lanes, then detect vehicles based on the lane and vehicle features to support the warning system, and the driver assistance system. This idea consists of two main steps, the information lane will be detected in the first step. And then, vehicle will be detected inside area among the detected lanes by vehicle features. represents the scheme outline of proposed system. In thescheme outline, we have a notions as a region of interest (ROI). The ROI is defined as an area close to the test vehicle. The full image is used to detect vehicles. There are two phases: 1 – the lane detection on the ROI, and 2 – the vehicles detection based on the detected lanes on the full image. Assume that there are some cases in which vehicle can not detected pr vehicle detection is not accurate. Therefore, we use Kalman filter to track vehicle information with error cases or inaccuracy cases . Intelligent vehicle technologies have strongly developed in recent years because there are many new people who use vehicles, and the number of vehicle accident has increased annually. These technologies utilize some kind of sensor such as Lidar, Radar, and vision sensors. In which, Lidar and radar are only used for obstacle detection, unique vision sensors is used for lane detection and vehicle detection. Detecting lane markings and vehicles enables vehicles to evade collisions and support a warning system. Moreover, today vision sensor are cheaper, smaller, and higher quality than before. In addition to, strong of graphical processing units (GPU) and hardware permits approach for lane detection and vehicle detection to implement in real-time. For these reasons, lane detection and vehicle detection using vision sensor is an interesting topic for researchers and engineer.

CHAPTER-3

Functionality

In this section, the proposed lane detection technique with proper explanation is illustrated. The overall working procedure of the proposed system is presented in Fig. 1. From the Fig. 1, it is shown that image preprocessing steps are taken to remove the noise from the images. The gradient and HLS thresholding are used to detect the lane lines on the images. The perspective transform visualizes the lane line properly. The lane detection dataset is retrieved from KITTI ROAD dataset [20] that contains 289 images. The details description of each step is outlined as follows.

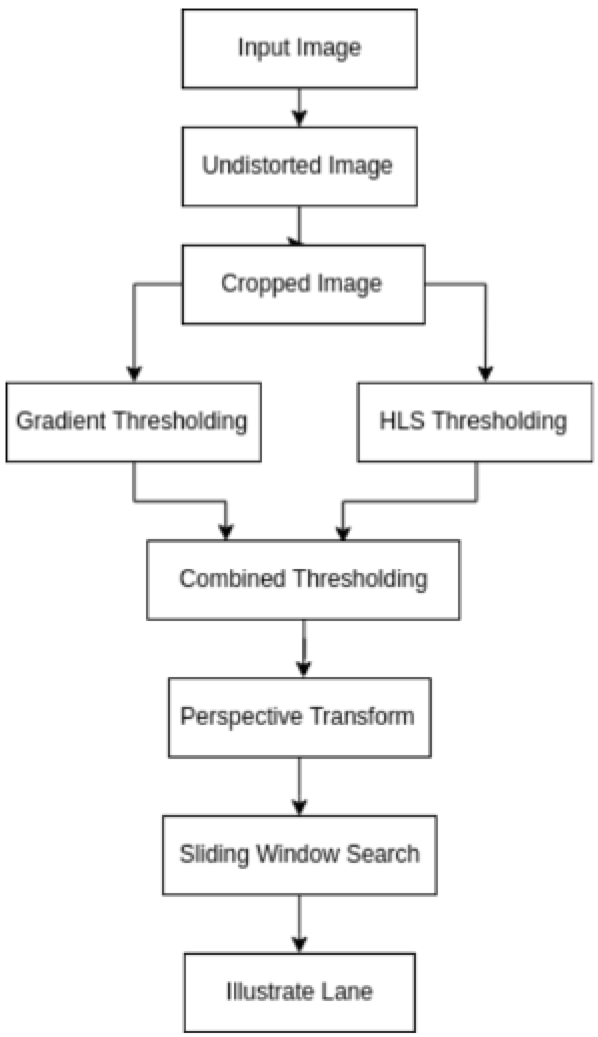


Fig.1 The overall process of the proposed system for lane detection.

A. Preprocessing

Preprocessing stage has a significant characteristic of the lane lines marking. The main purpose of pre-processing is to increase the contrast, eliminate the noise and generate an edge image for the corresponding input image. In this stage, the images are undistorted so that it will restore the straightness of lines, helping to identify lane lines. The variation between the distorted (original) and undistorted images is clear. The curved lines are now straight. The camera matrix and distortion coefficients using chessboard images are calculated in OpenCV. Here, it can be accomplished by gaining the inside corners . within an image and utilizing that information to undistort the image. The foregoing concerns to coordinates in our 2D mapping while the contemporary represents the real-world coordinates of those image points in 3D space (with the z-axis, or depth = 0 for our chessboard images). Those mappings facilitate to attain out how to accurately eliminate distortion on images. The preprocessing steps with input image for the lane detection system are illustrated in it can be seen that the undistorted image is brighter than the original image as it is noise free.

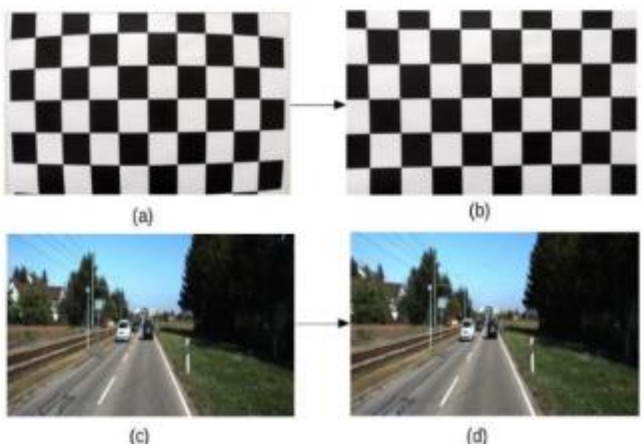


Fig.2

B. Cropped Image

Cropping is a procedure that is used to eliminate the unwanted region from a particular image. During the determination of lane lines, we only need to focus on the regions where we are likely to see the lanes. For this reason, the cropping operation is done and performing the further image processing only in the particular areas of the image. The cropped image to focus the particular region of lanes is illustrated in Fig. 3. We have resized the image to smaller dimensions. This supports with doing the image processing pipeline faster.

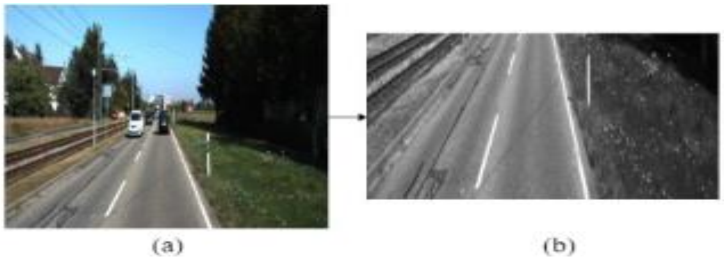


Fig.3 cropped Image for focusing on the region of lanes.

(a) Undistorted image (b) Cropped image.

C. Thresholding

Thresholding [21] is generally used for image segmentation. This method is a kind of image segmentation that separates objects by altering grayscale images into binary images. Image thresholding technique is the most appropriate in images with high stages of contrast. The thresholding procedure can be stated as

T=T[a ,b, p(a ,b), f(a, b)]

where T represents the threshold value, the coordinates points of threshold value are (a, b) and the grayscale image pixels are p (a, b), f (a, b).

i) Gradient Thresholding

Sobel is a kernel for gradient thresholding [22] in both and x and y-axis. Since the lane lines are probably going to be vertical, the more weight on the inclination in a y-axis is given. For proper scaling, total slope esteems and standardized is taken into consideration. The outcome after applying gradient thresholding is illustrated in Fig. 4 .

(ii) HLS thresholding

HLS (Hue Saturation Lightness) [23] color channel is used to handle cases when the road color is too bright or too light. L (lightness) channel threshold diminishes edges formed from shadows in the frame. S (saturation) channel threshold expands white or yellow lanes. H (hue) toward the line colors. The outcome after applying HLS thresholding is depicted in Fig. 4(c). We have combined both of the Gradient and HLS (color) thresholding into one for the final thresholding binary image that improves the overall results of the lane detection process. The output combining the Gradient and HLS thresholding is demonstrated .

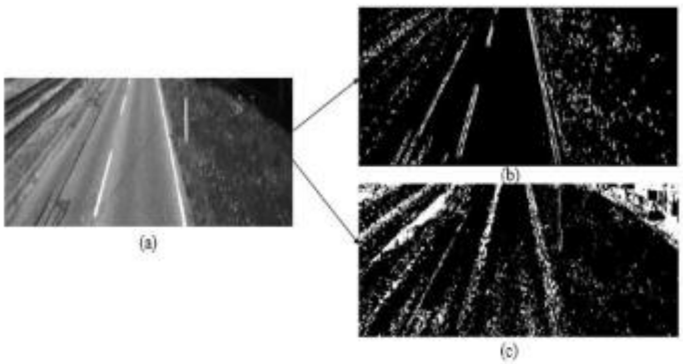


Fig.4. Thresholding applies on the cropped image.

(a)Cropped image (b)Gradient thresholding (c)HLS thresholding

D. Perspective Transform

The perspective transformation [24] is used to convert 3d world image into a 2d image. While undistorting and thresholding help to cover the crucial information, we can additionally divide that information by taking a drake at the part of the image of the road surface. To center in around the road part of the image, we move our point of view to a best down perspective of the street. While we don't obtain any more information from this step, it's enormously easier to isolate lane lines and measure things like curvature from this perspective. The Perspective Transform of the combined thresholding image is shown in Fig. 6 .

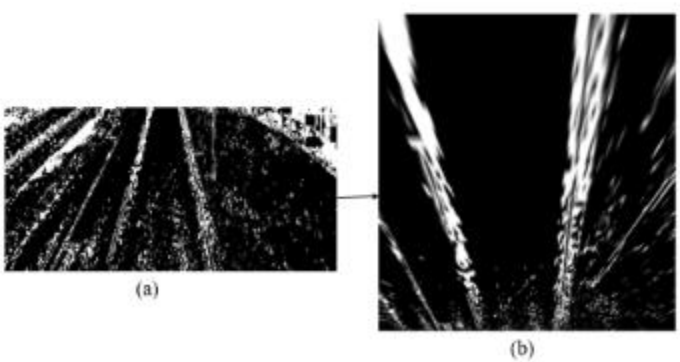


Fig.6. Perspective Transform for viewing to the best down perspective of the road.

(a) Combined thresholding (b) Perspective transform.

E. Sliding Window Search

As the lane lines already detected in an earlier frame, the information is used in a sliding window, placed around the line centers, to detect and track lane lines from bottom to the top of the image. The result of the sliding window search is demonstrated in Fig. 7. This permits us to do a highly qualified search and saves a lot of processing time. To recognize left and right lane line pixels, their x and y pixel positions are used, to fit a second-order polynomial curve:

$$f(y) = ay^2+by+c=0$$

The f(y) is used, rather than f(x) because the lane lines in the warped image are approximately vertical and may have the same x value for more than one y values.

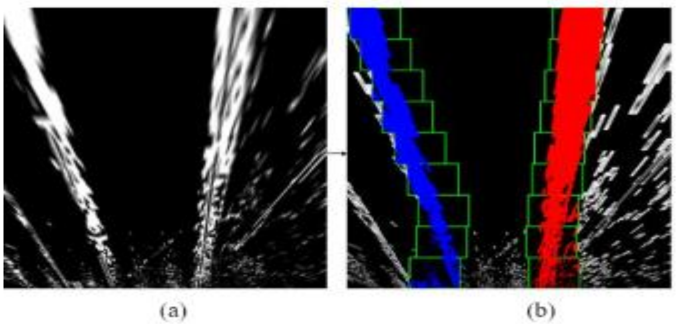


Fig.7. Sliding Window Search for detecting primary stage of lanes from perspective transform image.

(a) Perspective transform image (b) Sliding window search on perspective transform image

F. Illustrate Lane

Starting from the top point of view, the lane lines are easily recognized. A sliding window search distinguishes the lane lines. The green boxes express to the windows where the lane lines are colored. Since the windows search higher, they re-center to the standard pixel position so that they resemble the lines. The shaded lines will be stepped back onto the original image. The result of illustrated lanes from the sliding window search image is shown in Fig. 8.

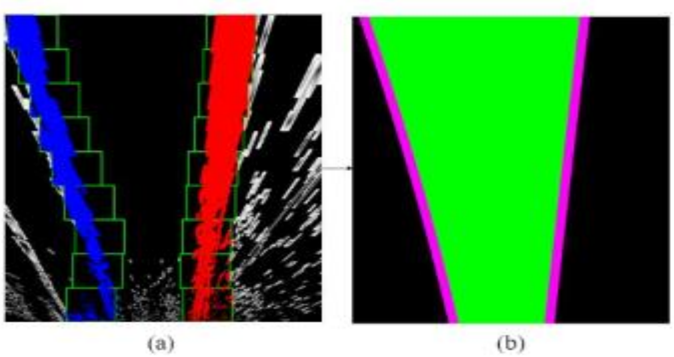


Fig.8. Illustrate lane from sliding window search image.

(a) Sliding window search image (b) llustrated lanes and covered the space with green color.

After illustrating the lane lines, the warp and crop operations are performed for proper visualization of the image. To warp, a 3x3 transformation matrix operation performed. Straight lines will continue straight still after the transformation. To observe this transformation matrix, 4 points on the input image and the corresponding points on the output image are needed. Among those 4 points, 3 of them should not be collinear. Then the images are cropped because, during the determination of recognizing lane lines, we only need to focus on the regions where we are likely to see the lanes. The result of the warped and cropped image is presented oad surface. While the pipeline prepares for a single image, it can easily be applied to processing many images to detect the lane line on the road surface. The final result of the proposed lane detection system is shown in Fig. 9

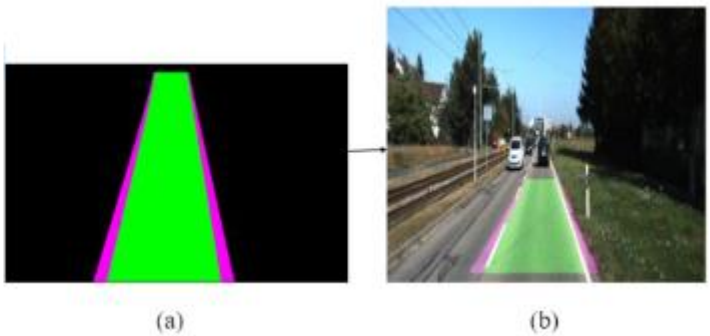


Fig.09. The final result of lane detection.

(a) Warped and cropped image (b) Detection of lanes.

3.1 Analysis, Activity Time Schedule.

Analysis:

The detection and identification of the lane line is to realize the intelligence and automation of the modern transportation system. It has a good application in the fields of advanced parking assistance, lane keeping assistance, off-road warning, and lane change assistance.

Demand analysis is a necessary step in system design. It can not only understand the operating mechanism ofthe system before design but also adjust the design structure in time for the demand. For the lane line identification, the ultimate goal is to accurately and efficiently identify the lane line. For this purpose, do the following needs analysis:

(1) Accuracy

The purpose of intelligent transportation system design is to reduce the rate of urban traffic accidents, so accuracy is the primary requirement of system design. If there is an error or deviation in the detection of thelane line, it may cause the vehicle to head in the wrong direction, which is not conducive to its own driving safety and brings greater safety risks to road safety.

(2) High efficiency

When the vehicle is driving on the road, it must not only be able to accurately follow the direction of the lane line but also be able to maintain a certain speed. This requires both the detection accuracy and the detection inthe lane line identification process efficiently and in real-time.

(3) Memory

In the intelligent public transport system, in addition to the vehicle that can detect the lane line image in realtime, it also has a good memory function, which can effectively store and manage the recorded data information and facilitate investigation and evidence collection after a traffic accident.

(4) Simple design structure

The system design is mainly used in vehicles, and the occupied space should be as small as possible. You canmake full use of 5G communication real-time connection without affecting other functions of the vehicle.

(5) Fully automated

The system is designed to allow the driver to alert and correct the driver's driving errors in an unconscious situation, so it is necessary to be able to fully automate the system during operation.

Activity Time Schedule :-

Phase 1 (Research):

- Choose a topic
- Define the task and prepare a working theory.
- Brainstorm all possible sources
- Locate and evaluate sources for appropriateness for the project.
- Write a Report

Phase 2 (Implementation):

- Research the prerequisites
- Data Preparation/ Pre-processing/ Augmentation
- Model Implementation
- Training
- Evaluation

3.2 **Design** **ARCHITECTURE DIAGRAM**

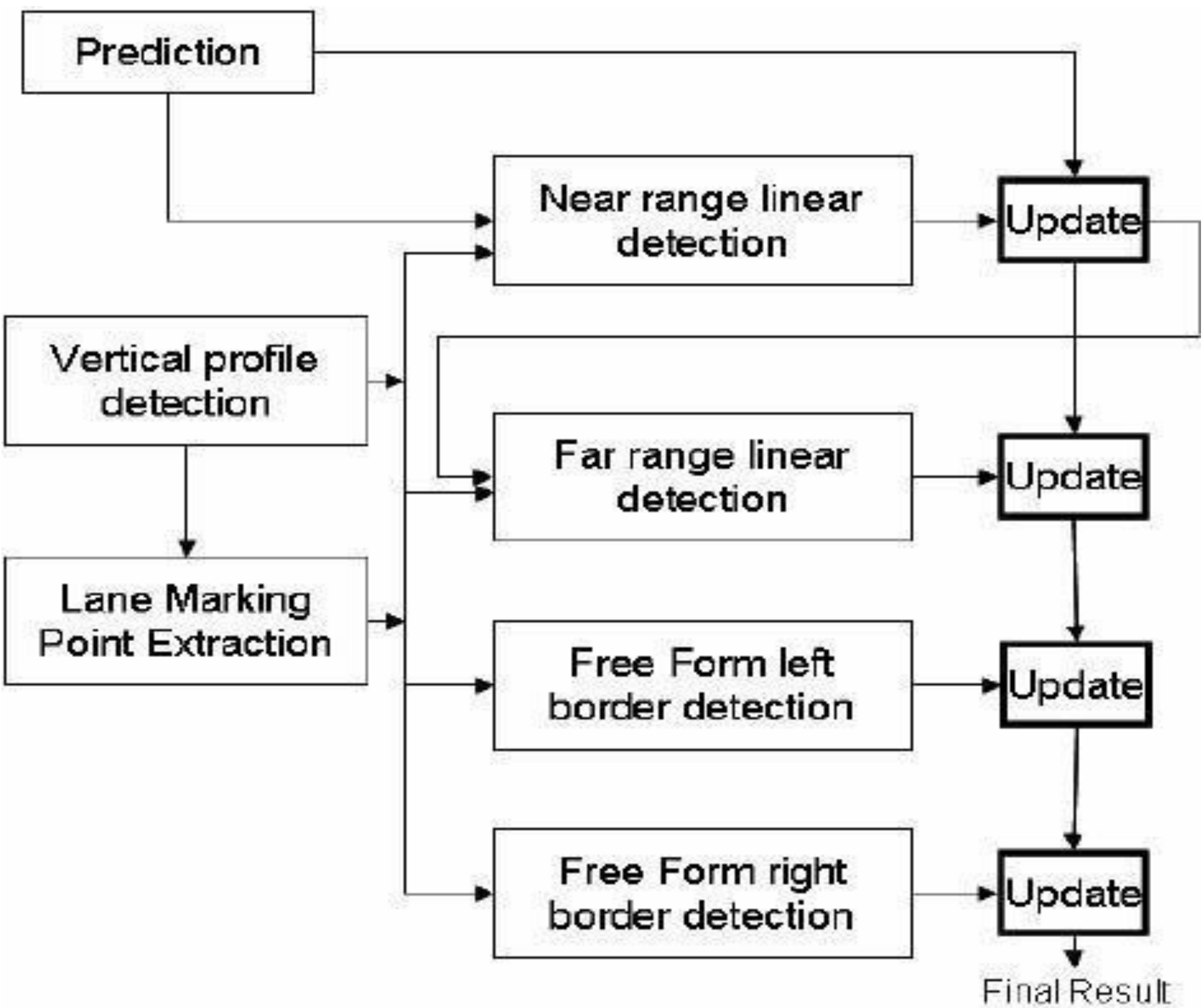


Fig.9

PROBLEM FORMULATION

The Idea Behind Detecting Moving Objects in Videos :
Object detection is a fascinating field in computer vision. It goes to a whole new level when we'redealing with video data. The complexity rises up a notch, but so do the rewards!

We can perform super useful high-value tasks such as surveillance, traffic management, fightingcrime, etc. using object detection algorithms. Here's a GIF demonstrating the idea.

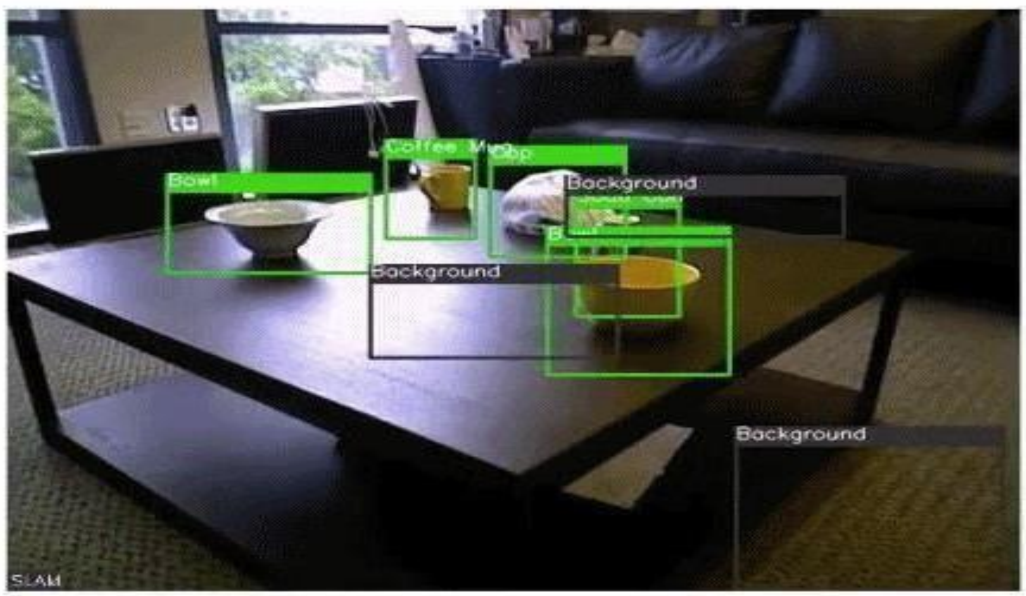


Fig 10.

There are a number of sub-tasks we can perform in object detection, such as counting the number of objects, finding the relative size of the objects, or finding the relative distance between the objects. Allthese sub-tasks are important as they contribute to solving some of the toughest realworld problems.

The task that we wish to perform is that of real-time lane detection in a video. There are multiple ways we can perform lane detection. We can use the learning-based approaches, such as training adeep learning model on an annotated video dataset, or use a pre-trained model.

However, there are simpler methods to perform lane detection as well. I will show you how to doit without using any deep learning model. But we will use the popular OpenCV library in Python.

Given below is a frame from the video that we will be work:

Fig.11

As we can see in this image, we have four lanes separated by white-colored lane markings. So, to detect a lane, we must detect the white markings on either side of that lane. This leads to the key question – how can we detect the lane markings?

There are so many other objects in the scene apart from the lane markings. There are vehicles on the road, road-side barriers, street-lights, etc. And in a video, a scene changes at every frame. Thismirrors real-life driving situations pretty well.

So, before solving the lane detection problem, we have to find a way to ignore the unwanted objectsfrom the driving scene.

One thing we can do right away is to narrow down the area of interest. **Instead of working with the entire frame, we can work with only a part of the frame.** In the image below, apart from thelane markings, everything else has been hidden in the frame. As the vehicle would move, the lanemarkings would fall more or less in this area only:

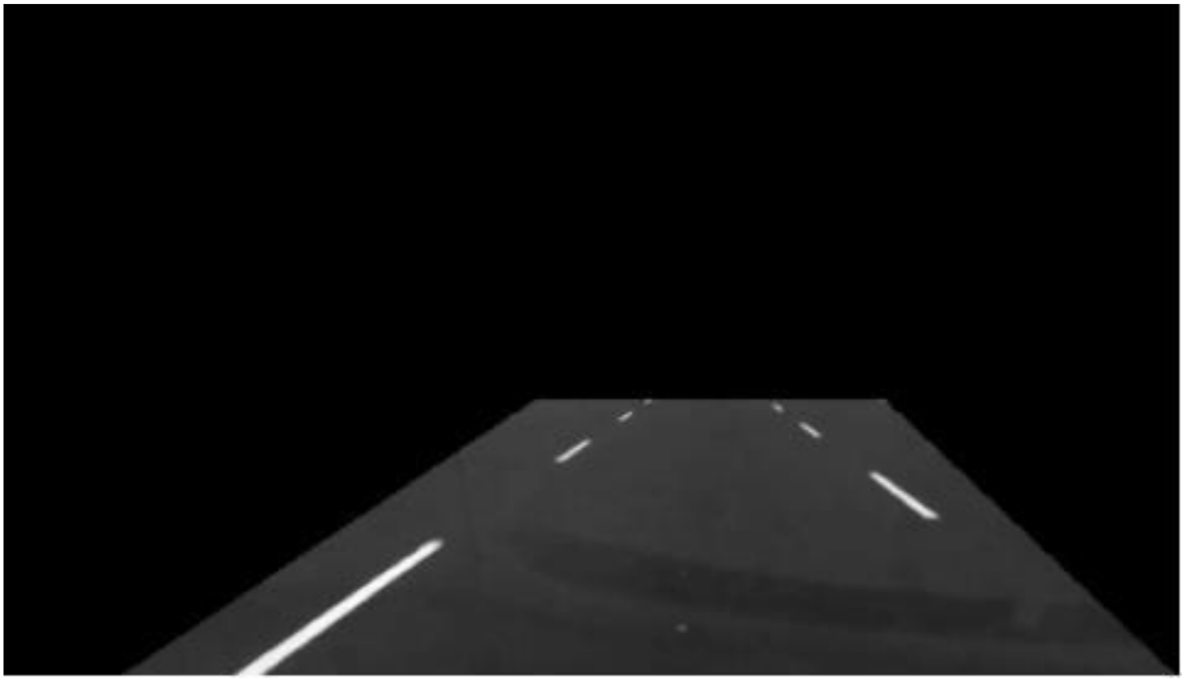


Fig 12.

Lane Line Fitting Evaluation and Curvature Radius Calculation :-

The RANSAC algorithm uses the continuous iterative method to obtain the optimal fit curve. Therefore, determining an effective evaluation criterion for lane line fitting is necessary. The comparison of the errorrate magnitude between the fitting curve and the real lane line determines the best lane line fitting result.

When the distance from all valid points in the area around the fitting curve to the fitting curve was the smallest, the error between the fitting curve and the real lane line was minimised, and the lane line fitting evaluation score was thehighest. The following evaluation score (Score) is defined as the reciprocal of the mean of the relative distance between all data points in the valid point set Q and the fitting curve:

$$\text{Score}=n\sum i=1n||(xt)i-(xsp)i||$$

where i=0,1,2,…,n, and *n* is the number of all data points in the valid point set Q.

The curvature of the current lane line was used to express its curve degree. The reciprocal of the curvature isthe curvature radius, which was used in this study to represent the curvature (in *m*). The curvature calculation of the third-order B-spline curve is more complicated than that of the simple curve. In this study,the original curve was reduced in order, the first and second derivative curves re-obtained by first- and second-order reduction, respectively. In the two-dimensional plane, let the coordinate corresponding to the first derivative be (x',y') and the coordinate corresponding to the second derivative be (x'',y''). The curvature of the lane line is defined as follows:

$$K=x'y''-y'x''[(x')2]3/2$$

WHAT IS FRAME MASK:-

Here, a frame mask is nothing but a NumPy array.When we want to apply a mask to an image, we simplychange the pixel values of the desired region in that image to 0, or 255, or any other number.
Given below is an example of image masking.
The pixel values of a certain region in the image have been set to 0:

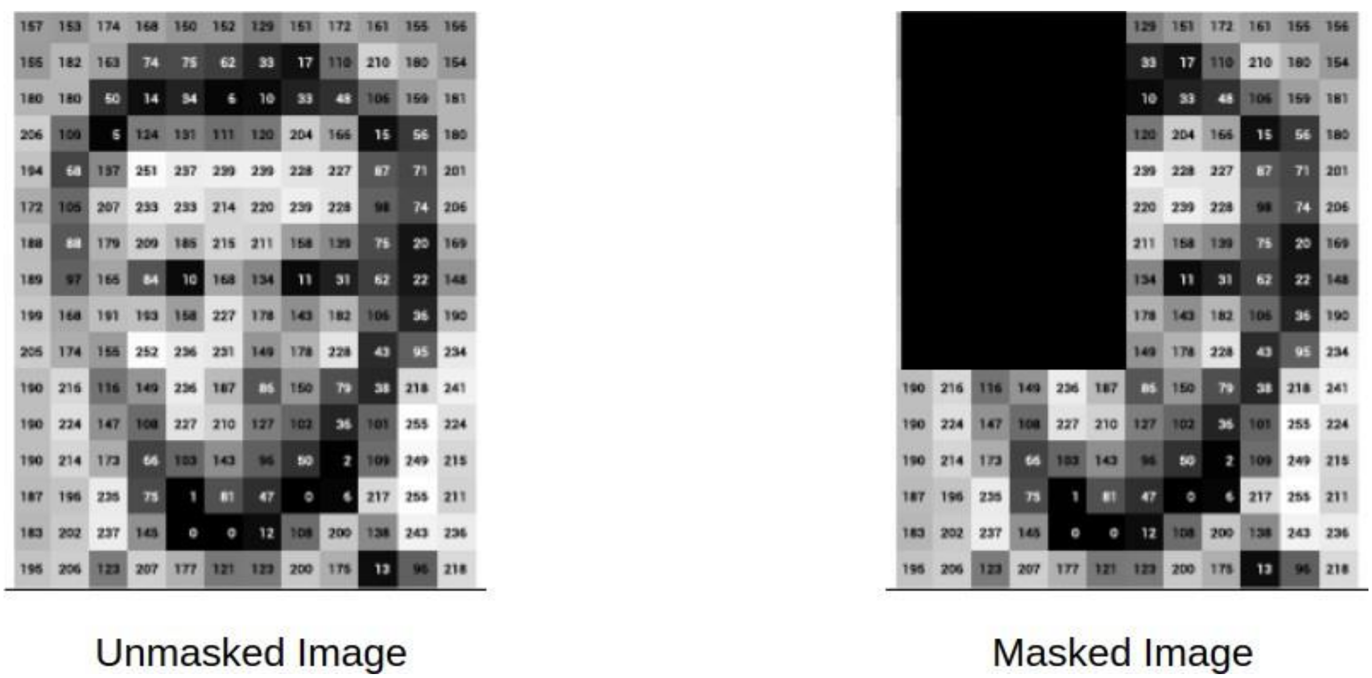
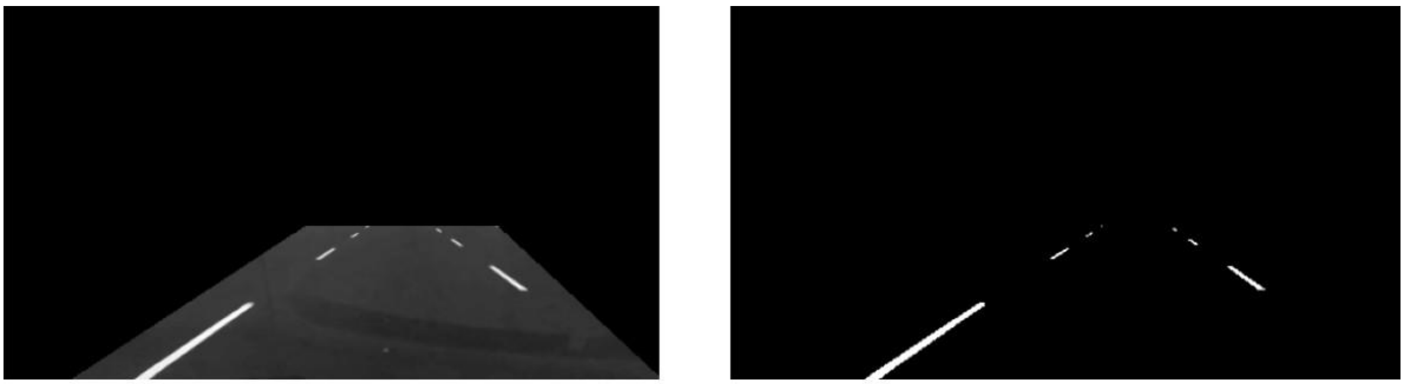


Fig 13.

IMAGE THRESHOLDING :-

In this method, the pixel values of a grayscale image are assigned one of the two values representing black and whitecolors based on a threshold value. So, if the value of a pixel is greater than a threshold value, it is assigned one value,else it is assigned the other value.



Masked Image   Image after Thresholding

Fig 14.

As you can see above, after applying thresholding on the masked image, we get only the lane markings inthe output image. Now we can easily detect these markings with the help of Hough Line Transformation.

HOUGH LINE TRANSFORMATION

Hough Transform is a technique to detect any shape that can be represented mathematically.

For example, it can detect shapes like rectangles, circles, triangles, or lines. We are interested in detectinglane markings that can be represented as lines.

HOUGH LINE TRANSFORMATION IN OPENCV

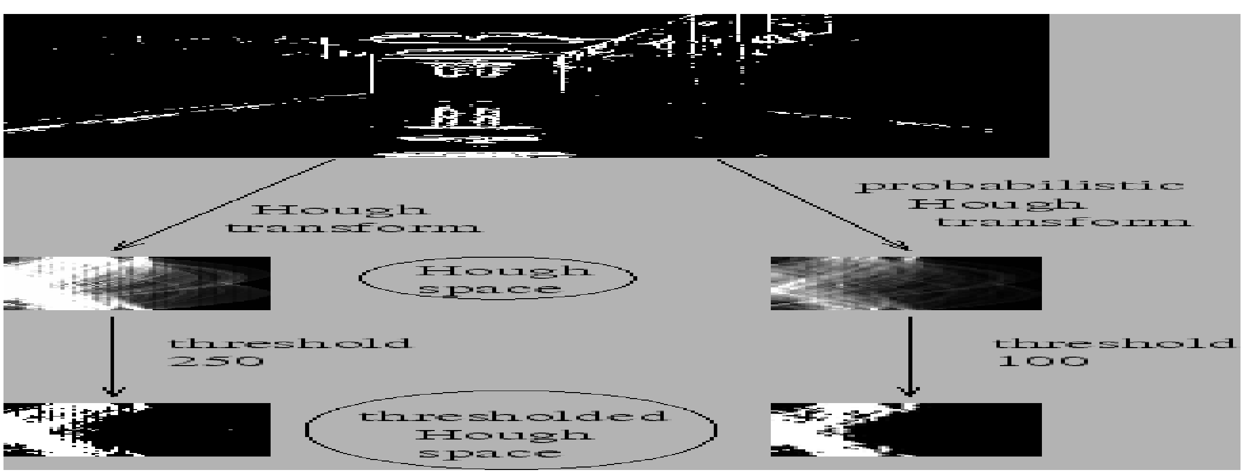
Everything explained above is encapsulated in the OpenCV function, cv2.HoughLines(). It simply returns anarray of (ρ, θ) values. is measured in pixels and is measured in radians. First parameter, Input image should be a binary image, so apply threshold or use canny edge detection before finding applying hough transform. Second and third parameters are and accuracies respectively. Fourth argument is
the threshold, which means minimum vote it should get for it to be considered as a line. Remember, number of votes

depend upon number of points on the line. So it represents the minimum length of line that should be detected.

```
import numpy as np

img = cv2.imread('dave.jpg')
gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY) edges =
cv2.Canny(gray,50,150,apertureSize = 3)

lines = cv2.HoughLines(edges,1,np.pi/180,200) for rho,theta in
lines[0]:
    a = np.cos(theta) b =
    np.sin(theta)x0 = a*rho
    y0 = b*rho
    x1 = int(x0 + 1000*(-b))y1 = int(y0
    + 1000*(a)) x2 = int(x0 - 1000*(-b))
    y2 = int(y0 - 1000*(a))

    cv2.line(img,(x1,y1),(x2,y2),(0,0,255),2)

cv2.imwrite('houghlines3.jpg',img)
```

PROBABILISTIC HOUGH TRANSFORM :-

In the hough transform, you can see that even for a line with two arguments, it takes a lot of computation. Probabilistic Hough Transform is an optimization of Hough Transform we saw. It doesn't take all the pointsinto consideration, instead take only a random subset of points and that is sufficient for line detection. Just we have to decrease the threshold. See below image which compare Hough Transform and Probabilistic Hough Transform in hough space.



Applying Hough Line Transformation on the image after performing image thresholding willgive us the below output:

Fig.15

Lane Detection Based on Road Driving Video :-

To verify the recognition performance of the lane detection algorithm in complex working conditions anddynamic environments, road driving videos were used to carry out the simulation test experiment in this work. Such videos in the experiment were collected by using a real-time vehicle-mounted camera, and thelane lines were detected for different road conditions, such as highway, mountain road and tunnel road.

Furthermore, the classical sliding window search method was utilised to test the lane detection algorithm dynamicallyin the simulation test experiment. Firstly, a pixel histogram was generated by the pixels whose grey value was not equal to 0, and the peak value in the pixel point set was obtained (the area where the lane line exists). Secondly, the sliding window was used to accumulate from bottom to top. When the number of pixels in the window exceeded the set threshold number, the mean value was taken as the centre of the next sliding window. In addition, by using the sliding window search method, the offset of the vehicle relative to the centre position of the road could be estimated.

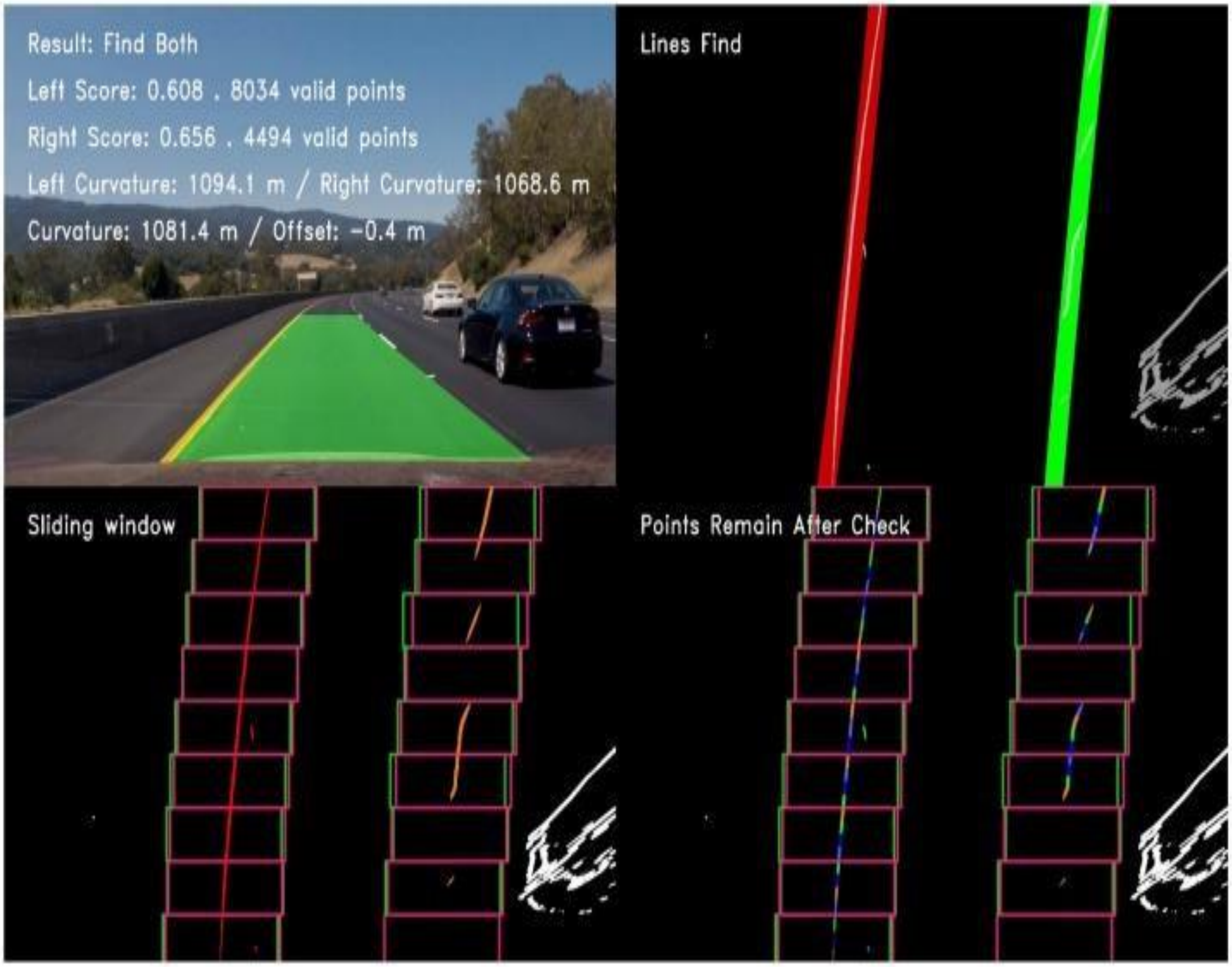


Fig. Lane Detection Based on the Tusimple Dataset :-

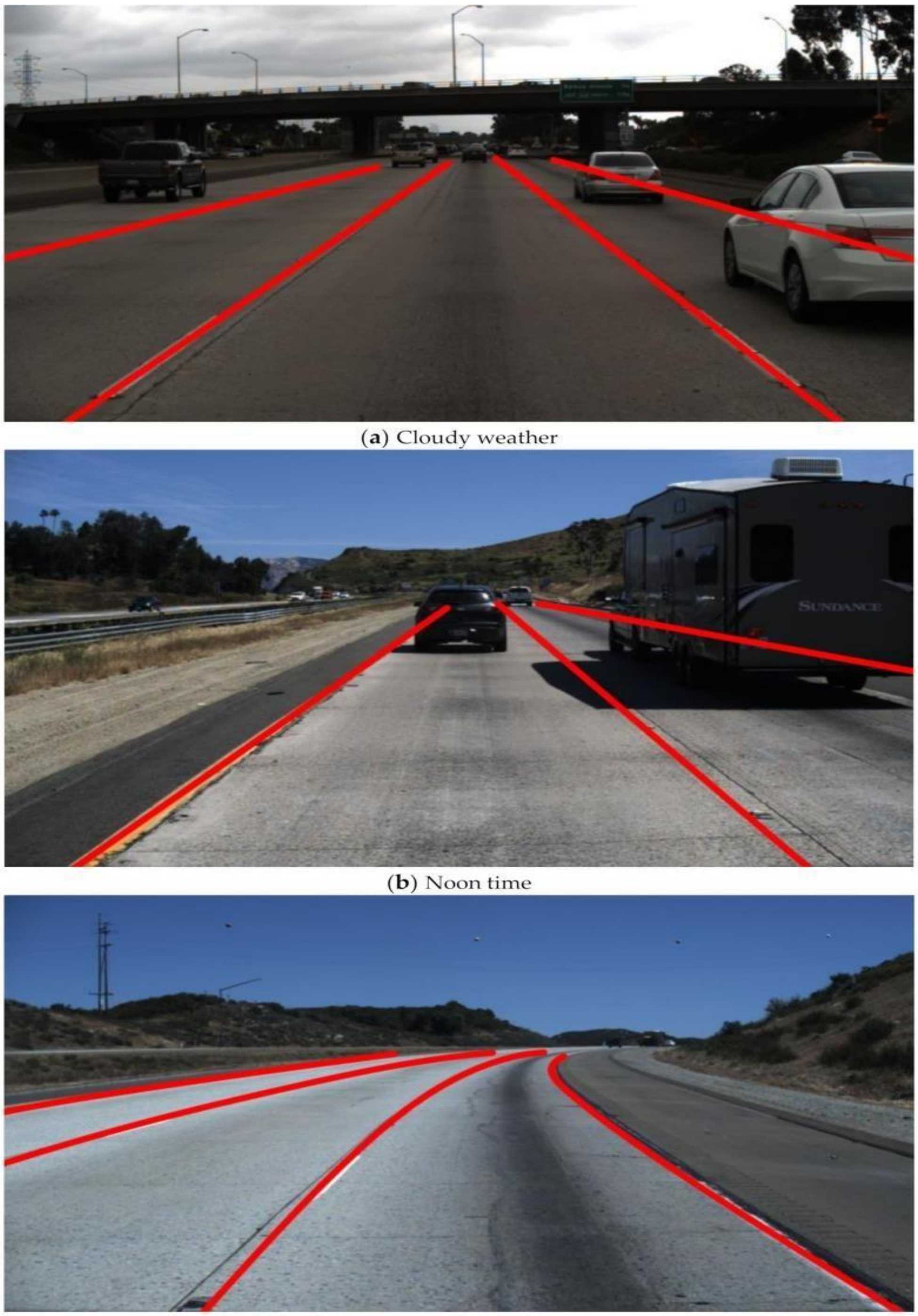
Lists the lane detection results based on road driving videounder complex road conditions. :-

The lane detection results based on road driving video under complex road conditions.

Video Sequence Number	Complex Road Conditions	Total Frames	Detected Frames	Misdetected and Missed Frames	Accurate Recognition Rate (%)	Average Processing Time (ms)/ Frame
1	Highway	1535	1522	13	99.15	20.8
2	Tunnel Road	1763	1736	27	98.47	21.6
3	Mountain Road	1470	1438	32	97.82	22.1
Total	-	4768	4696	72	98.49	21.5

In order to further test the comprehensive performance of the proposed algorithm under a variety of complexroad conditions, in addition to the road driving videos for simulation test experiments, the Tusimple dataset (general-purpose benchmark dataset) was also used for lane detection. The Tusimple dataset consists of 3626 training images and 2782 testing images, and each sequence image comprising 20 consecutive frames taken in 1s, of which the first 19 frames are unmarked and the 20th frame is labelled with lane ground truth. The images in the dataset can be roughly divided into four conditions, including different weather conditions, daytime, lanes number (2 lanes/3 lanes/4 lanes or more), and traffic environments 150 images representing different conditions were randomly extracted from the Tusimple dataset separately in this work,and the lane detection experiments were carried out using the proposed algorithm.

Fig.16 shows the lane detection



(**a**) Cloudy weather

(**b**) Noon time

(**c**) Four lanes

3.3 TECHNOLOGY USED: DEEP LEARNING

- Deep learning is a class of machine learning algorithms that uses multiple layers to progressively extract higher level features from the raw input.

- Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.

- OpenCV is a cross-platform library using which we can develop real-time **computer vision applications**. It mainly focuses on image processing, video capture and analysis including features like face detection and object detection.

- Keras is a minimalist Python library for deep learning that can run on top of Theano or TensorFlow. It makes implemention of deep learning models as fast and easy as possible

- NumPy is a python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices.

- Linear regression is the most simple and popular technique for predicting a continuous variable. It assumes a linear relationship between the outcome and the predictor variables.

- Deep learning is a class of machine learning algorithms that uses multiple layers to progressively extract higher level features from the raw input.

- Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structureand function of the brain called artificial neural networks

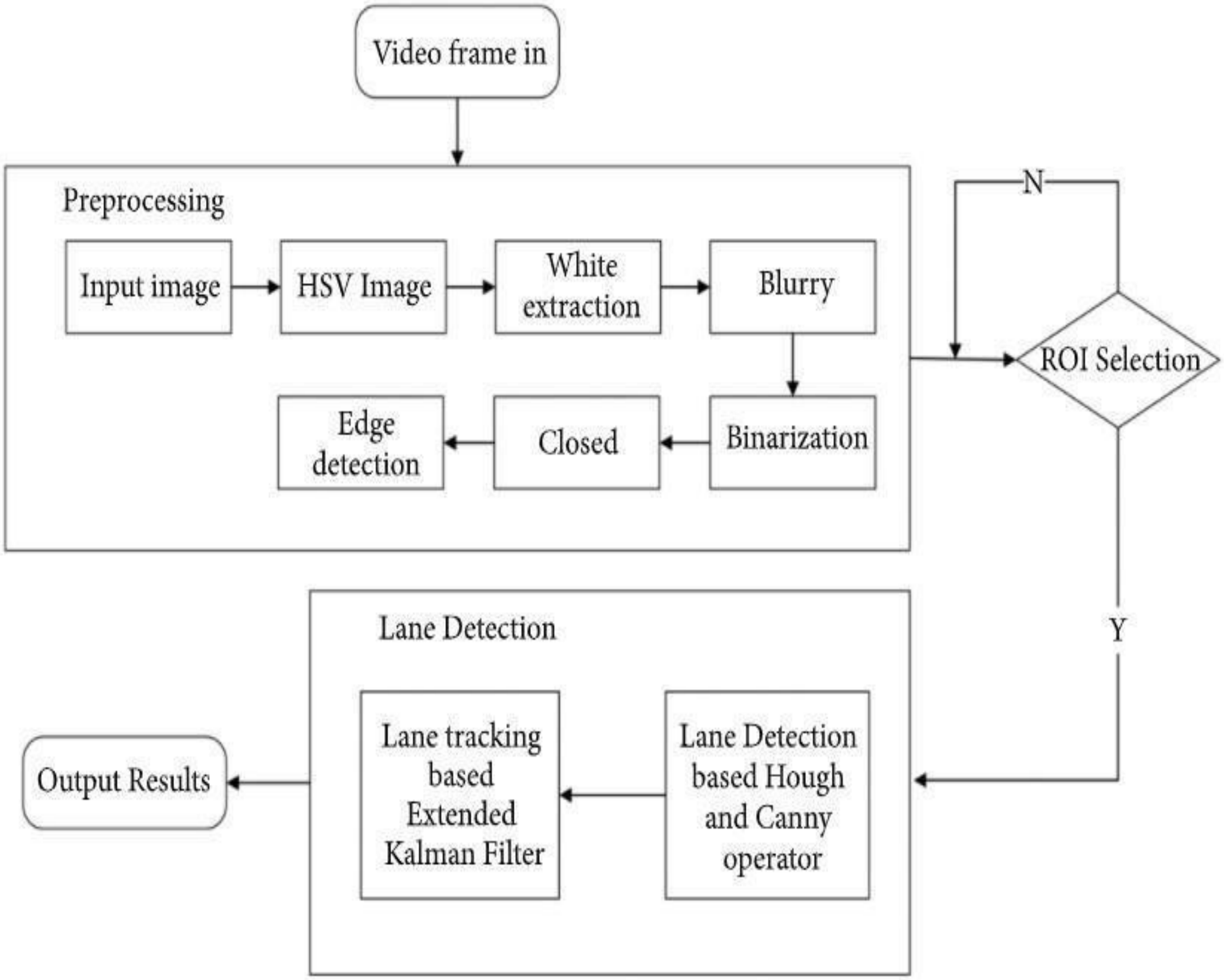
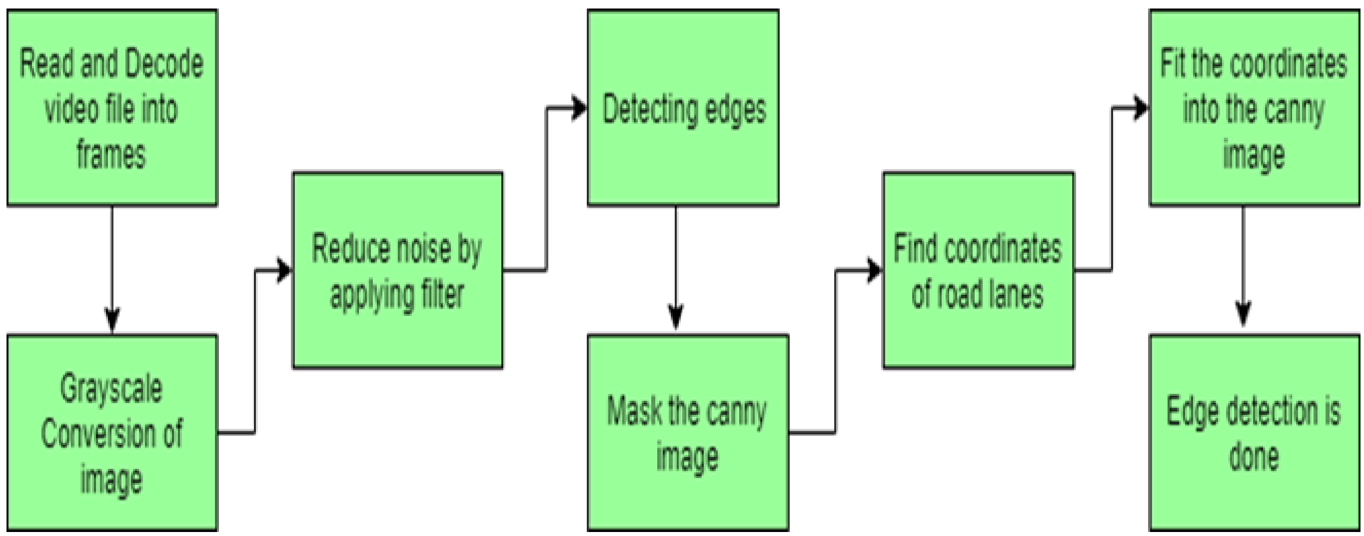


Fig. 11 Complete architecture

MERITS INVOLVED:

- Gives assistanceand details to pedestrians and drivers
- Uniformity of the markings is an important factor in minimizing confusion and uncertaintyabout their meaning
- Allows vehicular drivers to drive safely
- gives us a warning when the vehicle is veering off the lane without signaling.
- also plays an important role in advanced driver assistant system.
- helps the drivers in the driving process.
- Advanced driver assistance system consists of collision avoidance system, blind spot systemand many more systems.
- In lane detection there are many approaches that are applied like feature based and model based.
- Feature based approach are used to detect edges and model based approach is a type of curve

Working of Project



- Capturing and decoding video file: We will capture the video using VideoCaptureobject and after the capturing has been initialized every video frame is decoded
- Grayscale conversion of image: The video frames are in RGB format, RGB is converted to grayscale because processing a single channel image is faster thanprocessing a three-channel coloured image.
- Reduce noise: Noise can create false edges, therefore before going further, it's imperative to perform image smoothening. Gaussian filter is used to perform thisprocess.
- Canny Edge Detector: It computes gradient in all directions of our blurred image andtraces the edges with large changes in intensity.
- Region of Interest: This step is to take into account only the region covered by the road lane. A mask is created here, which is of the same dimension as our road image.Furthermore, bitwise AND operation is performed between each pixel of our canny image and this mask. It ultimately masks the canny image and shows the region of interest traced by the polygonal contour of the mask.
- Hough Line Transform: The Hough Line Transform is a transform used to detectstraight lines.

DESCRIPTION OF PROJECT MODULES:

It's time to implement this lane detection project in Python! I recommend using Google Colab because ofthe

computation power that will be required for building our lane detection system.

Let's first import the required libraries:

```
import os
import re
import cv2
import numpy as np
from tqdm import tqdm_notebook
import matplotlib.pyplot as plt
```

Read Video Frames

```
# get file names of frames col_frames
= os.listdir('frames/')
col_frames.sort(key=lambda f: int(re.sub('\D', '', f)))

# load frames col_images=[]for i in
tqdm_notebook(col_frames):
img = cv2.imread('frames/'+i)
col_images.append(img)
```

Let’s plot one of the frames:

```
# specify frame index idx
= 457

# plot frame plt.figure(figsize=(10
,10))
plt.imshow(col_images[idx][:,:,0], cmap= "gray") plt.show()
```

Frame Mask Creation:

Our region of interest is in the shape of a polygon. We want to mask everything except thisregion. Therefore, we first have to specify the coordinates of the polygon and then use it toprepare the frame mask:

```
# create a zero arraystencil =
np.zeros_like(col_images[idx][:,:,0]) #specify coordinates of
the polygon
polygon = np.array([[50,270], [220,160], [360,160], [480,270]])

# fill polygon with ones
cv2.fillConvexPoly(stencil,polygon,
# plot polygon plt.figure(figsize=(10,10))
plt.imshow(stencil, cmap= "gray")plt.show()
```

```
# apply polygon as a mask on the frame
img = cv2.bitwise_and(col_images[idx][:,:,0],
col_images[idx][:,:,0], mask=stencil)

# plot masked frame
plt.figure(figsize=(10,10))
plt.imshow(img, cmap= "gray")
plt.show()
```

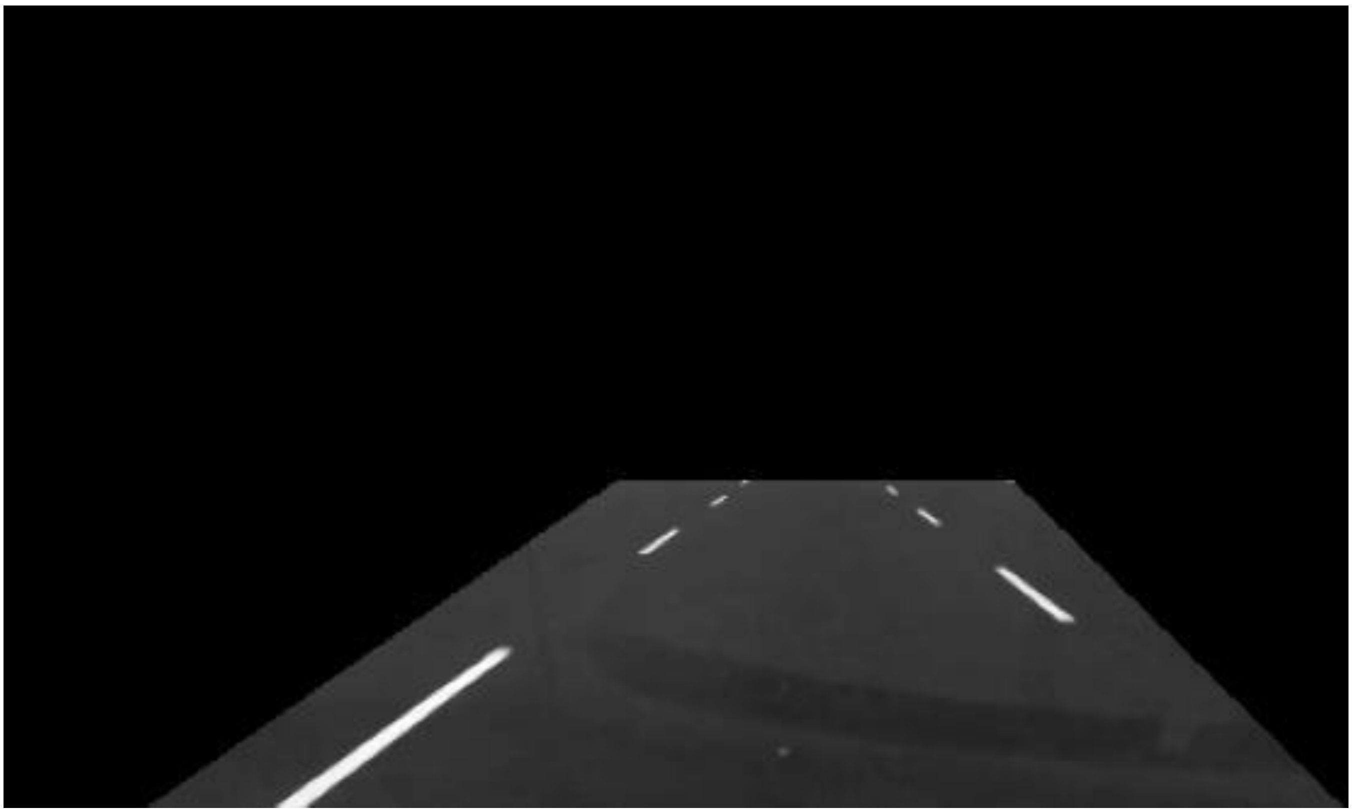


Image Pre-processing

We have to perform a couple of image pre-processing operations on the video frames todetect the desired lane. The pre-processing operations are:

Image Thresholding

Hough Line Transformation

Image Thresholding:

```
# apply image thresholding
ret, thresh = cv2.threshold(img, 130, 145, cv2.THRESH_BINARY)
# plot image
plt.figure(figsize=(10,10))
plt.imshow(thresh, cmap= "gray") plt.show()
```

Hough Line Transformation

```
lines = cv2.HoughLinesP(thresh, 1, np.pi/180, 30, maxLineGap=200)
# create a copy of the original frame dmy
= col_images[idx][:,:,0].copy()

# draw Hough lines
for line in lines:   x1,
y1, x2, y2 = line[0]
  cv2.line(dmy, (x1, y1), (x2, y2), (255, 0, 0), 3)

# plot frame
plt.figure(figsize=(10,10))
plt.imshow(dmy, cmap= "gray") plt.show()
```

Now we will apply all these operations on each and every frame. We will also save the resultantframes in a new directory:

```
cnt = 0 for img in
tqdm_notebook(col_images):

  # apply frame mask
  masked = cv2.bitwise_and(img[:,:,0], img[:,:,0], mask=stencil)

  # apply image thresholding
  ret, thresh = cv2.threshold(masked, 130, 145, cv2.THRESH_BINARY)

  # apply Hough Line Transformation
  lines = cv2.HoughLinesP(thresh, 1, np.pi/180, 30, maxLineGap=200)
dmy = img.copy()

  # Plot detected lines   try:
for line in lines:       x1,
y1, x2, y2 = line[0]
      cv2.line(dmy, (x1, y1), (x2, y2), (255, 0, 0), 3)

    cv2.imwrite('detected/'+str(cnt)+'.png',dmy)
     except TypeError:
cv2.imwrite('detected/'+str(cnt)+'.png',img)
   cnt+=1
```

Video Preparation

```
# input frames path
pathIn= 'detected/'

# output path to save the video
pathOut = 'roads_v2.mp4'

# specify frames per second
fps = 30.0
from os.path import isfile, join

# get file names of the frames
files = [f for f in os.listdir(pathIn) if isfile(join(pathIn, f))]
files.sort(key=lambda f: int(re.sub('\D', '', f)))
```

Next, we will get all frames with the detected lane into list:

```
frame_list = []

for i in tqdm_notebook(range(len(files))):
    filename=pathIn + files[i]
    #reading each files
```

```
img = cv2.imread(filename) height,
width, layers = img.shapesize =
(width,height)

#inserting the frames into an image array
frame_list.append(img)
```

Finally, we can now combine the frames into a video by using the code below:

```
# write the video
out = cv2.VideoWriter(pathOut,cv2.VideoWriter_fourcc(*'DIVX'), fps, size)

for i in range(len(frame_array)):#
    writing to a image array
    out.write(frame_array[i])

out.release()
```

Lane Detection stage in Line-based Lane Detection Algorithm:

```
// Calculating score of each line
1. for i ← 0 to NUMBER OF ROLLS - 1 do
2. y start point = 0
3. y end point = image height
4. x start point = mean + Gaussian sample
5. x end point = mean + Gaussian sample
6. slope = (x end point − x start point) / image height
7. for j ← 0 to image height - 1 do
8. x current = x start point + j · slope
9. score += intensity value at(j, x current)
10. end
11. end
// Sorting lines
12. sort(vector of lines.begin(), vector of lines.end(), compare score)
13. return line with highest score
```

Libraries:

Numpy:

NumPy stands for ‘Numerical Python’ or ‘Numeric Python’. It is an open source module of Python whichprovides fast mathematical computation on arrays and matrices. Since, arrays and matrices are an essential part of the Machine Learning ecosystem, NumPy along with Machine Learning modules like Scikit-learn, Pandas, Matplotlib, TensorFlow, etc. complete the Python Machine Learning Ecosystem.

One of the important attributes of a NumPy object are:

Ndim: displays the dimension of the array

Shape: returns a tuple of integers indicating the size of the array

Size: returns the total number of elements in the NumPy array

Dtype: returns the type of elements in the array, i.e., int64, character

Itemsize: returns the size in bytes of each item

Reshape: Reshapes the NumPy array

Pandas:

Pandas is one of the most widely used python libraries in data science. It provides high-performance, easyto use structures and data analysis tools. Unlike NumPy library which provides objects for multidimensional arrays, Pandas provides in-memory 2d table object called Dataframe. It is like a spreadsheet with column names and row labels.

Some commonly used data structures in pandas are:

Series objects: 1D array, similar to a column in a spreadsheet

DataFrame objects: 2D table, similar to a spreadsheet

Panel objects: Dictionary of DataFrames, similar to sheet in MS Excel

Pandas Series object is created using pd.Series function. Each row is provided with an index and by defaults is assigned numerical values starting from 0. Like NumPy, Pandas also provide the basic mathematical functionalities like addition, subtraction and conditional operations and broadcasting.

OpenCv

OpenCV-Python is a library of Python bindings designed to solve computer vision problems. **cv2.imread()** method loads an image from the specified file. If the image cannot be read (because of missing file, improper permissions, unsupported or invalid format) then this method returns an emptymatrix.

Syntax: cv2.imread(path, flag)
Parameters:
path: A string representing the path of the image to be read.
flag: It specifies the way in which image should be read. It's default value is **cv2.IMREAD_COLOR**
Return Value: This method returns an image that is loaded from the specified file.

cv2.IMREAD_COLOR: It specifies to load a color image. Any transparency of image will be neglected. It is the default flag. Alternatively, we can pass integer value **1** for this flag.
cv2.IMREAD_GRAYSCALE: It specifies to load an image in grayscale mode. Alternatively, we can pass integer value **0** for this flag.
cv2.IMREAD_UNCHANGED: It specifies to load an image as such including alpha channel. Alternatively, we can pass integer value **-1** for this flag.

linear_model :-

Sklearn.**linear_model**.ElasticNet. Elastic-Net is a linear regression model trained with both l1 and l2 -norm regularization of the coefficients. Notes. From the implementation point of view, this is just plainOrdinary Least Squares

Ransac :-

RANSAC (Random Sample Consensus) algorithm. **RANSAC** is an iterative algorithm for the robust estimation of parameters from a subset of inliers from the complete data set................................. score(X, y) : Returns the

mean accuracy on the given test data, which is used for the stop criterion defined by stop_score .

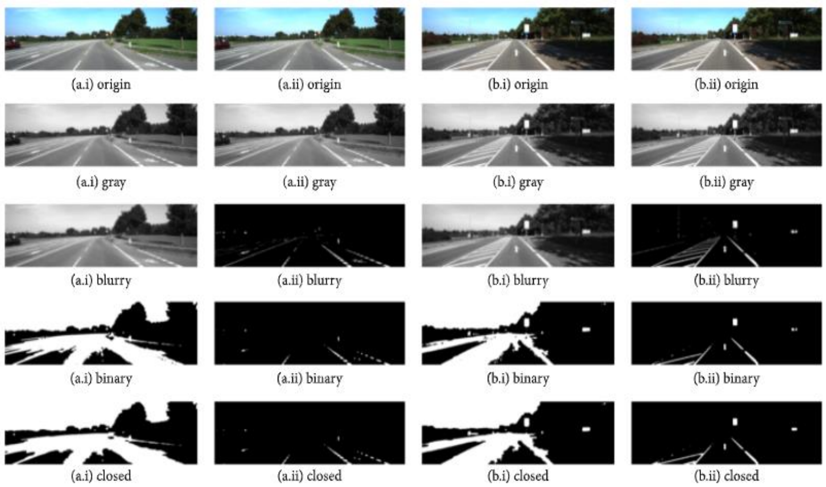
is a predictive modeling tool widely used in the image processing field for cleaning datasets from noise. RANSAC could be used as a "one stop shop" algorithm for developing and validatingQSAR models, performing outlier removal, descriptors selection, model development and predictions for test set samples using applicability domain.

CHAPTER-4

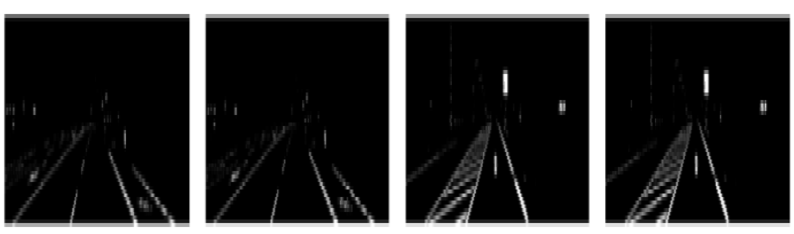
Results and Discussion

Figure shows the working of four edges of pictures. Packaging (I) and blueprint (ii) are ready by essential working without utilizing extraction of white part), and packaging (iii) and framework (iv) are taken care of through the recommended preprocessor (white component extraction. Diagram (ii) and framework (iv) which are ready by proposed usefulness can show the pathline. In any case, there are loads of white development in layout (I) and blueprint (iii), and it is difficult to distinguish way lines. Thusly, the crucial usefulness of the edge doesn't work outstandingly for way ID. Considering these, we can add HSV concealing change in the pre working stage. Subsequently this, separate the white features of the edge before the hazy view, to achieve a prevalent revelation affect and improve perspective on acknowledgment accuracy.

As shown in Fig(a) and (b) , pictures show white features that are extracted from video, respectively.



As appeared in Fig., outline (a) and outline (b) are extricated white highlights, separately.



Most investigation analysts clearly execute return on initial capital investment assurance onto the main picture. In this exploration, another return on initial capital investment assurance method is proposed. Examinations mirror that the proposed return for money invested decision can improve the precision and capability of way ID. Figures show the return for capital invested assurance of white part. It will in general be observed that return for capital invested decision of white component can't exactly recognize the zone of way line, which will unavoidably convey an exceptional mistake.

CHAPTER 5

Conclusion and Future Scope

We covered an essential methodology for way area. This proposed a way distinguishing proof estimation for astute vehicles in jam-packed road conditions. First thing, changing over the mangled picture and using the superposition edge estimation subordinate onto the Sobel head and the HSL concealing space for the edge area, ethereal viewpoint on the way was gained with the assistance of using extraction of return for money invested and talk perspective change.

Additionally, the RANSAC estimation was gotten to fix the bends for the way lines dependent on the solicitation B-spline twist model, and the fitting appraisal and the curve range figuring onto the twist was finished immediately. At last, by using the road driving video under complex road In regards to way ID accuracy and computation monotonous, the proposed way revelation estimation had clear positive conditions. It was useful for tremendously overhauling the driving security of adroit vehicles in the real driving conditions and suitably meeting the continuous objective essentials of sharp vehicles and accepted a huge part in keen vehicle driving assistance.

References

[1] “Reported road casualties Great Britain 2009,” Stationery Office Dept. Transp., London, U.K., 2010.

[2] J. M. Clanton, D. M. Bevly, and A. S. Hodel, "A Low-Cost Solution for an Integrated Multisensor Lane Departure Warning System," in IEEE Transactions on Intelligent Transportation Systems, vol. 10, no. 1, pp. 47-59, Mar. 2009 .

[3] Q. Li, L. Chen, M. Li, S. Shaw and A. Nüchter, "A Sensor-Fusion Drivable-Region and Lane-Detection System for Autonomous Vehicle Navigation in Challenging Road Scenarios," in IEEE Transactions on Vehicular Technology, vol. 63, no. 2, pp. 540-555, Feb. 2014.

[4] M. Aly, "Real-time detection of lane markers in urban streets," in Proc. IEEE Intelligent Vehicles Symposium, Eindhoven, pp. 7-12, 2008.

[5] H. Yoo, U. Yang, and K. Sohn, “Gradient-enhancing conversion for illumination-robust lane detection,” in IEEE Transactions on Intelligent Transportation Systems, vol. 14, no. 3, pp. 1083–1094, Sep. 2013.

[6] J. Deng, J. Kim, H. Sin, and Y. Han, “Fast lane detection based on the B-Spline fitting,” International Journal of Research in Engineering and Technology, vol. 2, no. 4, pp. 134–137, 2013.

[7] sung-Ying Sun, Shang-Jeng Tsai, and V. Chan, "HSI color model based lane-marking detection," in Proc. IEEE Intelligent Transportation Systems Conference, Toronto, Ont., pp. 1168-1172, 2006.

[8] R. Ke, Z. Li, J. Tang, Z. Pan and Y. Wang, "Real-Time Traffic Flow Parameter stimation From UAV Video Based on Ensemble Classifier and Optical Flow," in IEEE Transactions on Intelligent Transportation Systems. no.99, pp. 1-11, Mar. 2018.

[9] H. Kong, J. Audibert and J. Ponce, "General Road Detection From a Single Image," in IEEE Transactions on Image Processing, vol. 19, no. 8, pp. 2211-2220, Aug. 2010

[10] P. Moghadam and J. F. Dong, "Road direction detection based on vanishing-point tracking," in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, pp. 1553-1560, 2012.

[11] C. Lee and J. H. Moon, "Robust Lane Detection and Tracking for Real-Time Applications,"

in IEEE Transactions on Intelligent Transportation Systems, vol. no.99, pp.1-6, Feb. 2018.

[12] W. Song, Y. Yang, M. Fu, Y. Li and M. Wang, "Lane Detection and Classification for Forward Collision Warning System Based on Stereo Vision," in IEEE Sensors Journal, vol. 18, no. 12, pp. 5151-5163, Jun. 2018.

[13] C. Wu, L. Wang and K. Wang, "Ultra-low Complexity Block-based Lane Detection and Departure Warning System," in IEEE Transactions on Circuits and Systems for Video Technology, Feb. 2018.

[14] J. H. Yoo, S. Lee, S. Park and D. H. Kim, "A Robust Lane Detection Method Based on Vanishing Point Estimation Using the Relevance of Line Segments," in IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 12, pp. 3254-3266, Dec. 2017.

[15] U. Ozgunalp, R. Fan, X. Ai and N. Dahnoun, "Multiple Lane Detection Algorithm Based on Novel Dense Vanishing Point Estimation," in IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 3, pp. 621-632, March 2017.

[16] J. Piao and H. Shin, "Robust hypothesis generation method using binary blob analysis for multi-lane detection," in IET Image Processing, vol. 11, no. 12, pp. 1210-1218, Dec. 2017.

[17] S. Jung, J. Youn, and S. Sull, "Efficient Lane Detection Based on Spatiotemporal Images," in IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 1, pp. 289-295, Jan. 2016.

[18] A. Borkar, M. Hayes, and M. T. Smith, "A novel lane detection system with efficient ground truth generation," in IEEE Transactions on Intelligent Transportation Systems vol. 13, no. 1, pp. 365–374, Mar. 2012.

[19] S. Kang, S. Lee, J. Hur and S. Seo, "Multi-lane detection based on accurate geometric lane estimation in highway scenarios," in Proc. IEEE Intelligent Vehicles Symposium, Dearborn, MI, pp. 221-226, 2014.

[20] J. Fritsch, T. Kühnl and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in Proc. 16th International IEEE Conference on Intelligent Transportation Systems, pp. 1693-1700, 2013.