

STEERING ASSIST FOR DRIVERS BASED ON ROAD LANE ARTICULATION

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Technology
in
Computer Science and
Engineering

by

VEDANT BATRA
(18BCE0740)

CSE4019_IMAGE PROCESSING



JUNE, 2020

DECLARATION

I hereby declare that the thesis entitled “Steering Assist for Drivers based on Road Lane Articulation” submitted by me, for the award of the degree of *Bachelor of Technology in CSE* to VIT is a record of bonafide work carried out by me under the supervision of Dr Gladys Gnana Kiruba B.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place : Vellore

Date
:04.06.200

(Vedant Batra)
Signature of the Candidate

ACKNOWLEDGEMENT

I would first like to thank my project advisor Dr Gladys Gnana Kiruba B of the SCOPE school at VIT University, Vellore. The door to Prof. Dr Gladys Gnana Kiruba B office was always open whenever I ran into a trouble spot or had a question about my research or writing. She consistently allowed this paper to be my own work, but steered me in the right the direction whenever he thought I needed it.

I would also like to thank the experts who were involved in the validation survey for this research project: Faculty of the Department, Teachers, Friends. Without their passionate participation and input, the validation survey could not have been successfully conducted..

Finally, I must express my very profound gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Author

Vedant Batra

CONTENTS		Page No.
	Acknowledgement	i
	Table of Contents	Iii
	List of Figures	ix
1	ABSTRACT	1
2	RESEARCH PROBLEMS AND DESIGN OF STUDY	
3	TECHNICAL SPECIFICATION	3
4	METHODOLOGY AND PROCESSING	5
5	CODE IMPLEMENTATION	7
6	PROJECT DEMONSTRATION	8
7	RESULT & DISCUSSION	11
8	REFERENCES	12

Abstract—

Today Road Accidents are a major cause of deaths .The main reason for Road accidents is over speeding and getting lost of lanes thus leading accidents ,forms one of the main causes of impermanence, also

have an economic impact of the owners of vehicles. Most of the automotive industries are developing technology to reduce risk in vehicles. In this way, lane detection systems is important, because from this data is possible to determine risk situations hence reducing accidents. Here we propose a technique for lane detection, based on image processing, which allows identifying the position of lateral or partial lanes. The idea is based on converting the given image to grEy scale image and then tracing the white lines of lanes and getting to a point where maximum of intersections occur and therefore tracing graphic lines hence assisting the driver based on the graphics thus implemented.

1. INTRODUCTION

Steering Assist system is one of the most important feature of the modern vehicles to ensure driver safety and decrease vehicle accident on roads. Apparently, the road lane detection on road boundaries detection is the complex and most challenging tasks.

A vision system using on-board camera looking outwards from the windshield is presented in this paper. The system acquires the front view using a camera mounted on the vehicle and detects the lanes by applying few processes.

The lanes are extracted using Hough transform through a pair of hyperbolas which are fitted to the edges of the lanes. The proposed lane detection system can be applied on painted lanes The proposed system does not require any extra information such as lane width and time to lane crossing

The system can be applied under various situations of changing illumination, and shadows effects in various road types without speed limits.

The system has demonstrated a robust performance for detecting the road lanes under different conditions. .The main motive is Region of Interest(ROI) as compared to previous projects.

RESEARCH PROBLEM AND DESIGN OF STUDY

In this project, we try to learn about the different techniques for Image processing and segmentation using Image Processing methods.

The study on the topic consists of the following phases:-

1.The PRELIMINARY PHASE

In this phase we search and find out the problems faced by the driver in . This phase also consists of gaining an in-depth knowledge of how the information are stored, recent studies/research/survey papers on the information regarding lane detection systems. It majorly focuses on forming the bases of the entire building of the project.

This phase is mainly responsible for broadening our thinking capacities and boundaries on the topics.

2.The RESEARCH PHASE

We aim to develop a robust algorithm and ensure a real-time detection, because we are interested on a real-time implementation. For this purpose, our optimizations will concern image pre-processing, edge detection and line tracking steps.

The purposes for the construction of the ROI (Region of Interest)are the reduction of processing time and the reduction of memory space required for each video frame. Moreover, we limit the number of memory access. Then, the RGB colour image is converted to an intensity image and the rest of operations are per-formed on intensity image.. Finally, we threshold the intensity image to obtain road lines

3. LITERATURE SURVEY AND REVIEW

Several different image processing algorithms have been developed such as Likelihood of Image Shape (LOIS) algorithm , B-Snake algorithm and others using a feature-based method to extract features in an image (e.g. edges),with different techniques . A feature-based method has the advantage to other to researchers a large choice of the kind of techniques to be used in each step of the algorithm and then

contribute on its optimization. However, it has two major problems. The rest issue of lane detection is the amount of calculation required. Thus, for real-time application, the performances and the processing time of the algorithm proposed above can be improved by using other operators for edge detection on the one hand and for line tracking on the other hand. The second issue is the difficulty of identifying lane boundary lines, because a real road image can contain other

candidate edges (e.g. noise edges). That's why; it should be a judicious choice of an edge detection technique to reduce the number of edge points in the output image of this step. Indeed, the edges must be detected well for Hough transform to be efficient.

4.The IMPLEMENTATION PHASE

It is truly said that the true knowledge comes with the use of a tool.

After gaining the THEORETICAL KNOWLEDGE upto this point, we aim to implement the theory into practicality. This way we would learn more about the problem and the tools and its usages.

The tools that would be implemented would help us to gain knowledge on the inner functioning of the tools. In-depth knowledge of image processing tools will help us expand the boundaries of our understanding from how the tool works to how it is used.

Thus, this phase will focus on concentrating to gain knowledge about the Image Processing tools available in this field.

5. METHODOLOGY AND PROCESSING OF THE IMAGE

5.1 Hardware Requirements

- Camera
- Live Video Capturing Instrument

5.2 Software Requirements

- Open CV
- Python
- Pip package
 1. Scikit-learn
 2. Num py
 3. Matplot lib

5.3 Processing of the Image

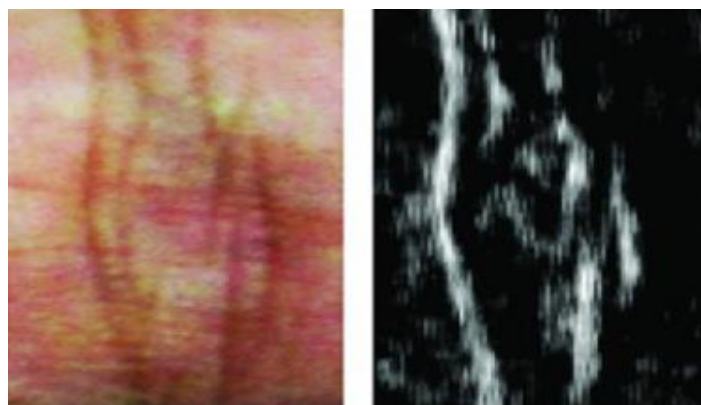
1[1]. Grayscale Image

Grayscale Image Basically we convert the given image to grayscale image because it consists of black and white intensity value no coloured values therefore reducing memory utilization because in real world memory plays a major role .Another major reason for

conversion to Grey scale being for faster processing and execution because it is easy to process Grey scale image as compared to coloured image .So for above reasons the given image is converted to Grey scale image.

2.[2] Gaussian Blur

The given Gray scale image is passed through the gaussian blur, the main reason being i.e to reduce noise present in the image . In our case the reasons for noise being fog, sand ,rain , water and etc other natural factors can result in making the image noisy i.e reducing the amount of the details required for further processing, i.e it is important to pass the given image through gaussian blur i.e smoothening the image.



3[3] Canny edge detection

Basically an Edge is formed when we go through higher to lower intensities or vice versa. This edge detection technique is basically used to distinguish road lanes from other non-useful stuff hence reducing the memory utilization as well as faster processing. In the edge detection system we are using Canny inbuilt function of cv2 library .The function automatically identifies edge when an image is passed through it.

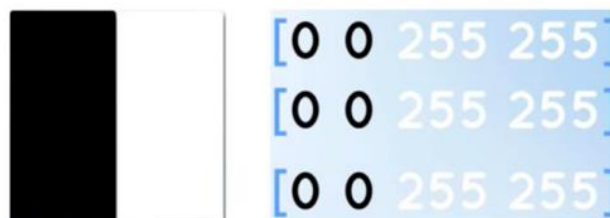


Fig 5.1

4. [4] Region of interest

In the road lane detection system we are basically interested in road lanes. So we design a specific area to be considered while detecting road lanes .This helps in faster execution and less memory utilization.

Coming to the area to be considered can be designed according to the vehicle size as well as to the gap between the right and left lane and many other factors.



Fig 5.2

5. [5] Hough Transform

The Hough transform is a feature extraction technique used in image analysis, computer vision, and digital image processing. The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure.

We see that when the **point (2,12)** is passed through different lines we obtain a straight line on the Hough space.

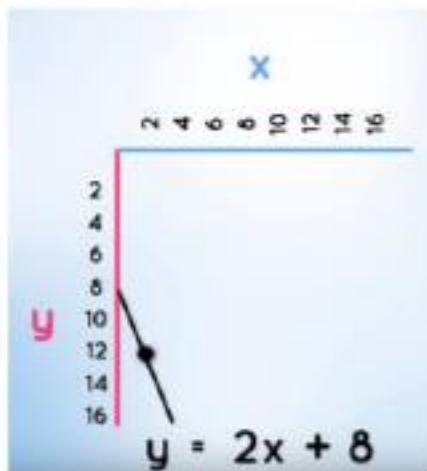


Fig 6.1

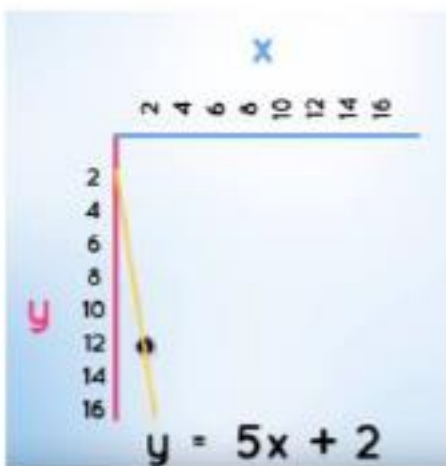


Fig 6.2

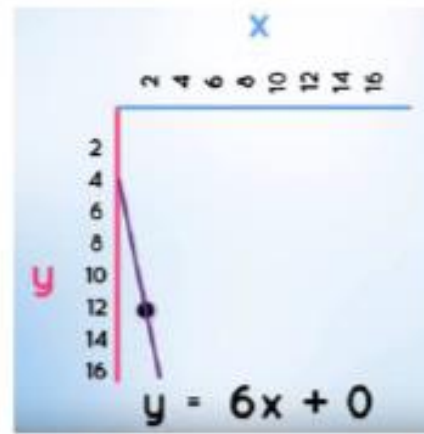


Fig 6.3



Fig 6.4

Now considering different point on the same line we obtain different lines on the Hough space therefore on intersection point is the point or more specific line of best fit, for example in the above case (4,4) is the intersection point and hence the line of best fit is $y=4x+4$. We convert the given coordinates to polar coordinates for more convenience although the concept is the same, Here also the intersection occurs hence helping us to get the line of best fit. The Results are clearly visible as follows :

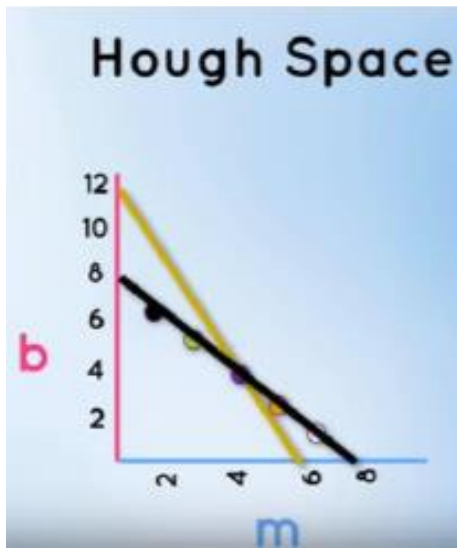


Fig 7.1

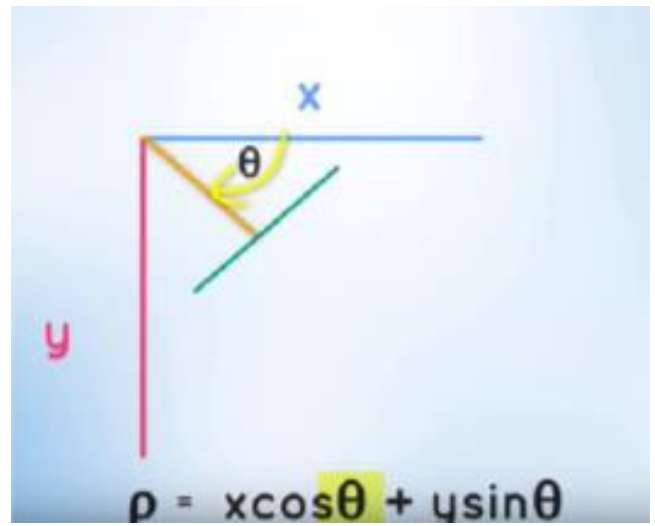


Fig 7.4

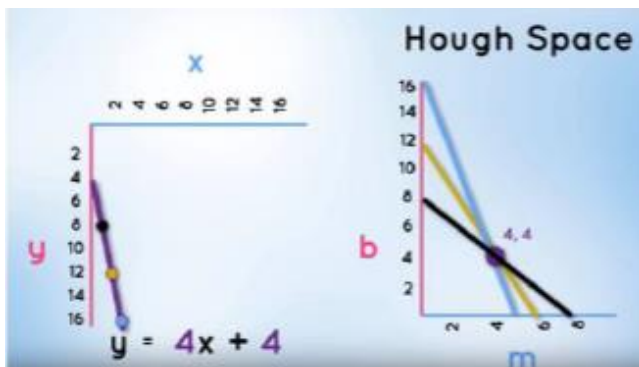


Fig 7.2

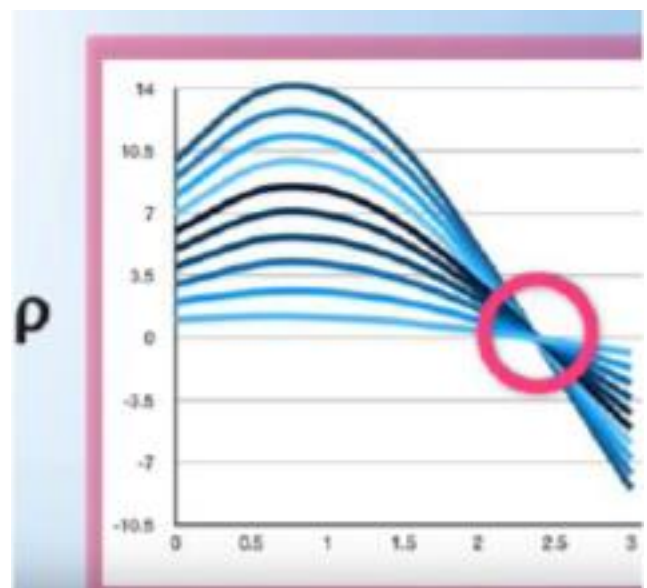


Fig 7.5

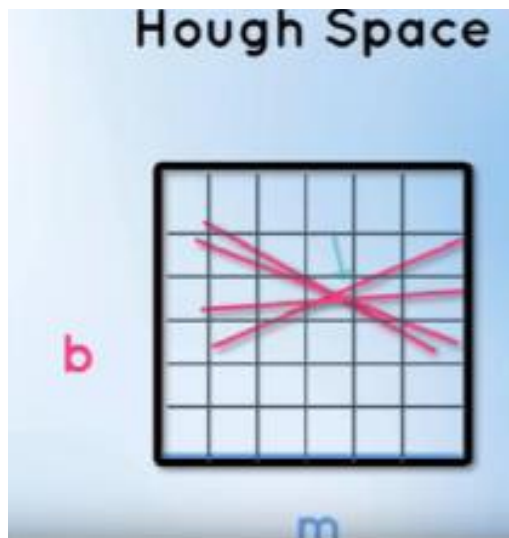


Fig 7.3

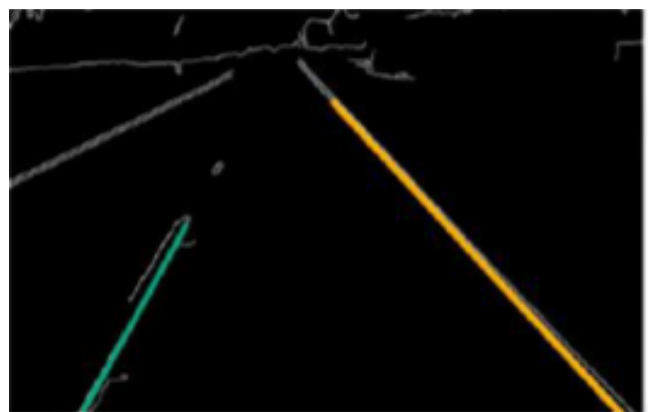


Fig 7.5

5. Coding Phase Using Python 3

```
import cv2
import numpy as np
import matplotlib.pyplot as plt

def make_coordinates ( image , line parameters ) :
    slope , intercept = line parameters
    y1 = image . shape [ 0 ]
    y2 = int ( y1 _ ( 3 / 5 ) )
    x1 = int ( ( y1 _ intercept ) / slope )
    x2 = int ( ( y2 _ intercept ) / slope )
    return np . array ( [ x1 , y1 , x2 , y2 ] )

def averageslopeintercept ( image , lines ) :
    leftfit = [ ]
    rightfit = [ ]
    for line in lines :
        x1 , y1 , x2 , y2 = line . reshape ( 4 )
        parameters = np . polyfit ( ( x1 , x2 ) , ( y1 , y2 ) , 1 )
        slope = parameters [ 0 ]
        intercept = parameters [ 1 ]
        if slope < 0 :
            leftfit . append ( ( slope , intercept ) )
        else :
            rightfit . append ( ( slope , intercept ) )
    leftfitaverage = np . average ( leftfit , axis = 0 )
    rightfitaverage = np . average ( rightfit , axis = 0 )
    print ( leftfitaverage , 'left' )
```

```
print ( rightfitaverage , 'right' )

leftline = make_coordinates ( image , leftfitaverage )
rightline = make_coordinates ( image , rightfitaverage )
return np . array ( [ leftline , rightline ] )

def canny ( image ) :
    gray = cv2 . cvtColor ( image , cv2 . COLOR_RGB2GRAY )
    blur = cv2 . GaussianBlur ( gray , ( 5 , 5 ) , 0 )
    canny = cv2 . Canny ( blur , 50 , 150 )
    return canny

def regionofinterest ( image ) :
    height = image . shape [ 0 ]
    polygons = np . array ( [ [ ( 200 , height ) , ( 1100 , height ) ] ] )
    mask = np . zeros_like ( image )
    cv2 . fillPoly ( mask , polygons , 255 )
    masked_image = cv2 . bitwise_and ( image , mask )
    return masked_image

def display_lines ( image , lines ) :
    line_image = np . zeros_like ( image )
    if lines is not None :
        for line in lines :
            x1 , y1 , x2 , y2 = line . reshape ( 4 )
            cv2 . line ( line_image , ( x1 , y1 ) , ( x2 , y2 ) , ( 255 , 0 ) )
    return line_image

image = cv2 . imread ( 'testimage . jpg' )
lane_image = np . copy ( image )
#canny image = canny ( lane_image )
#cropped image = regionofinterest ( canny image )
#lines = cv2 . HoughLinesP ( cropped image , 2 , np . pi / 180 , 100 ,
, minLineLength = 40 , maxLineGap = 5 )
#averagedlines = averageslopeintercept ( lane_image )
```

```

# lane image=image if we use like this then changes w
image=display_lines( lane image , averaged_lines
#combo image=cv2 . addWeighted( lane image , 0 . 8 , line
) #cv2 . imshow("result" , combo image )
#cv2 . waitKey(0)
cap=cv2 . VideoCapture ( "test2 .mp4")
while ( cap . isOpened ( ) ) :
, frame=cap . read ( )
canny image=canny ( frame )
cropped image=region_of_interest( canny image )
lines=cv2 . HoughLinesP( cropped image , 2 , np . pi /180
, minLineLength=40,maxLineGap=5)
averaged_lines=averageslopeintercept( fr
# lane image=image if we use like this then changes w
g e=display_lines( frame , averaged_lines )
combo image=cv2 . addWeighted( frame , 0 . 8 , line ima g
) cv2 . imshow( "result" , combo image )
cv2 . waitKey( 1 )

```

6. Implementation Video

For the reference purpose and as a successful running proof of the project, the screen recording of the implementation of project is attached as a google drive link :

https://drive.google.com/file/d/11bUC2AcaMRHmX6ndFbG_TGlsvgwxkuk5/view?usp=sharing

7. Results and Conclusion

Therefore we have used all the six concepts of our model flow and finally instead of image we can consider a video with different frames and the algorithm works well for video as well .The main thing that our project is based is on reducing high memory and time utilization .In our project we have reduced the two.

On conclusion we conclude that our projects aims at improving the current lane detection system which required high memory utilization as well as high execution time. Also the system is suitable to work in noisy image capture situations . The main motive of this project being ROI which has reduced

memory utilization and also gaussian blur and other important things helping recover a good image for good results. So our system solves the problems as in the projects or articles previously mentioned.

Implementation Snapshots:-



8. Future Scope

In future work the main part is that the execution time can be reduced in order to make it more effective and real life useable . Our main focus is on the basic idea of lane detection and it can also be improved and further getting the faster results .

9. References

1. A. Takahashi, Y. Ninomiya, M.O.M.N., Yoshikawa, N.: Image processing technology for rear view camera: Development of lane detection system (2003)
2. Ajit Danti, Jyoti .Y. Kulkarni, P.: A real-time road boundary detection and vehicle detection for Indian roads, international journal of applied information systems, issn : 2249-0868, volume 5 no.4 (2013)
3. Y. Wang, E.K.T., Shen, D.: Lane detection and tracking using b-snake, image and vision computing, 22, p269-280 (2004)