CAR LANE DETECTION

A PROJECT REPORT

Submitted by

NAME OF THE CANDIDATES
(Register No)
MADA SAI SURYA - 19MIM10095
G. NAGA SAI PRANEETH - 19MIM10095
KADIYALA MEGHNATH - 19MIM10097
ABHISHEK SWAMI - 19MIM10117

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Specialization in

Artificial intelligence and machine learning

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

VIT BHOPAL UNIVERSITY

KOTHRI KALAN, SEHORE
MADHYA PRADESH - 466114

May 2020

BONAFIDE CERTIFICATE

Certified that this project report titled **"CAR LANE DETECTION"** is the bonafide work of "**MADA SAI SURYA (19MIM10095), G. NAGA SAI PRANEETH (19MIM10044), KADIYALA MEGHNATH (19MIM10097), ABISHEK SWAMI (19MIM10117)"** who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported here does not form part of any other project / research work on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

PROGRAM CHAIR
Dr.V.Pandimurugan
Assistant Professor
School of AI & ML division
VIT BHOPAL UNIVERSITY

PROJECT GUIDE
Dr. Rajasoundaran
Assistant Professor
School of AI & ML division
VIT BHOPAL UNIVERSITY

ACKNOWLEDGEMENT

First and foremost I would like to thank the Lord Almighty for His presence and immense blessings throughout the project work.

I wish to express my heartfelt gratitude to **Dr.V.Pandimurugan**, Head of the Department, School of Aeronautical Science for much of his valuable support and encouragement in carrying out this work.

I would like to thank my internal guide **Dr. Rajasoundaran**,for continually guiding and actively participating in my project, giving valuable suggestions to complete the project work.

I would like to thank all the technical and teaching staff of the School of Aeronautical Science, who extended directly or indirectly all support.

Last, but not the least, I am deeply indebted to my parents who have been the greatest support while I worked day and night for the project to make it a success.

LIST OF FIGURES

ABSTRACT

Driver support systems are one of the most important features of modern vehicles to ensure driver safety and decrease vehicle accidents on roads. Apparently, the road lane detection or road boundaries detection is the most complex and challenging task. It includes the localization of the road and the determination of the relative position between vehicle and road. A vision system using an 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 both painted and unpainted roads as well as curved and straight roads in different weather conditions. The proposed system does not require any extra information such as lane width, time to lane crossing and offset between the center of the lanes. In addition, camera calibration and coordinate transformation are also not required. The system was investigated 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.

TABLE OF CONTENTS

INTRODUCTION

Chapter - 1 : Introduction

During the driving operation, humans use their optical vision for vehicle maneuvering. The road lane marking acts as a constant reference for vehicle navigation. One of the prerequisites to have in a self-driving car is the development of an Automatic Lane Detection system using an algorithm. Computer vision is a technology that can enable cars to make sense of their surroundings. It is a branch of artificial intelligence that enables software to understand the content of an image and video. Modern computer vision has come a long way due to the advances in deep learning, which enables it to recognize different objects in images by examining and comparing millions of examples and cleaning the visual patterns that define each object. This means that a driverless car might crash into a truck in broad daylight, or worse, accidentally hit a pedestrian. The current computer vision technology used in autonomous vehicles is also vulnerable to adversarial attacks, by manipulating the AI's input channels to force it to make mistakes. For instance, researchers have shown they can trick a self-driving car to avoid recognizing stop signs by sticking black and white labels on them.

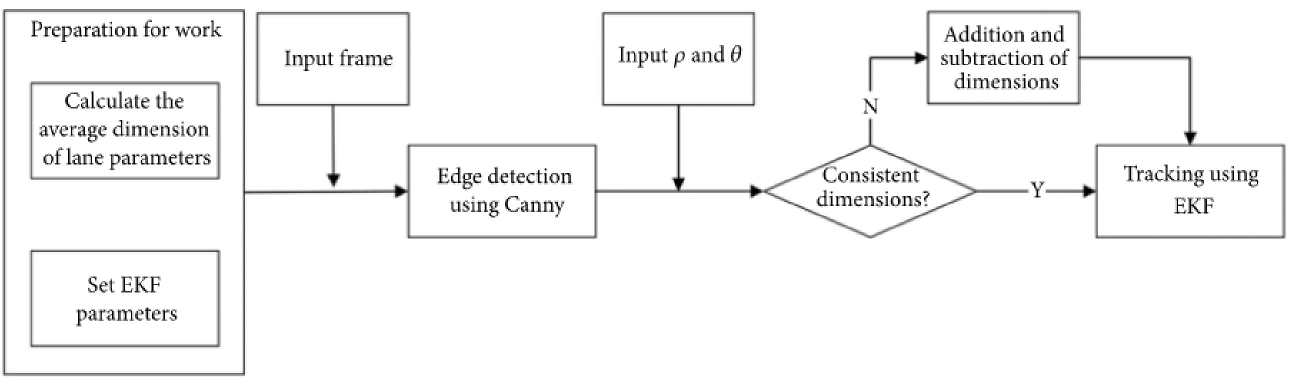
Chapter - 2 : Motivation for the work

Road accidents are the main causes for the sudden death in this world. Even though we have many good and advanced techniques in this world, we are left over with something to make it better than before. There are chances from different angles. Road lane detection and object detection is also another important way that we can improve the safety in roads.

In intelligent transportation systems with improved technologies, the vehicles are made more sophisticated with better infrastructure. But the way to move on the roads by means of lane and object detection aspect is neglected by many automobile companies and the ways to improve these aspects does not change from many years. Lane detection and object detection plays vital roles for accidents. For human vision and human intelligence the task of lane detection and object detection changes due to variations in the road conditions. Sometimes it is very easy to detect with the human eyes but in some conditions due to external effects the human intelligence has detection problems. In our project we have motivated to improve the intelligent vehicle assistance with improving the SNR quality for lane and object detections as an important aspect to avoid the road accidents and improving the safety on roads.

Chapter - 3 : Aims and Objectives

The aim of this thesis is to avoid accidental deaths and provide better safety on roads, by use of advanced technologies in the driving assistance system. The clear idea of the important aims and objectives of this thesis is explained with the help of block diagram.



Fig(1):Lane detection Flow chart

LITERATURE SURVEY

Chapter - 1 : Introduction of Literature Survey

At present, drivers' vision is obstructed in extreme weather conditions like, heavy rainfall, foggy conditions, snow fall and night travel along hilly terrain. During such situations the head lamp and the fog lamps are only of little use. Yet the drivers are forced to maneuver the situation with certain consciousness and leave the results for benefit of doubt. This also limits the speed.

There are many steps in detecting lanes on a road, first comes the camera calibration. Cameras use curved lenses to form an image, and light rays often bend a little too much or too little at the edges of these lenses. Images can be undistorted mapping distorted points to undistorted points such as a chessboard. This distortion correction is then applied to raw images, convert images to grayscale, apply gradients and finally apply deep learning. Then perspective transform is applied to the binary image from the bird's eye view.

We find this method from one of the research papers of Tseng et al. [2005] where he gave a lane marking detection algorithm by using geometry information and modified Hough transform.

Chapter - 2 : Methodology

2.2.1 Canny Edge Detection:

The most widely used edge detection technique in Computer Vision (J Canny 1986).The goal of edge detection is to identify the boundaries of objects within images.This is used to try and find regions in an image where there is a sharp change in intensity. We can recognize an image as a matrix or an array of pixels. A pixel contains the light intensity at some location in the image. Each pixel's intensity is denoted by a numeric value that ranges from 0 to 255, an intensity value of zero indicates no intensity if something is completely black whereas 255 represents maximum intensity, something being completely white.A gradient is the change in brightness over a series of pixels. A strong gradient indicates a steep change whereas a small gradient represents a shallow change.

The outline of white pixels corresponds to the discontinuity in brightness at the points that strengthen the gradient. This helps us identify edges in our image since an edge is defined by the difference in intensity values in adjacent pixels.

And wherever there is a sharp change in intensity (rapid change in brightness) i.e., wherever there is a strong gradient, there is a corresponding bright pixel in the gradient image. By tracing out all these pixels, we obtain the edges. We're going to use this concept to detect the edges in our road image.

2.2.1.1 Algorithm for Canny Edge Detection:

- Applying a Gaussian filter for noise removal and image smoothening.
- Computing the intensity gradients for all the pixels in the image.
- Applying a process called “non-maximum suppression”to avoid unauthentic response to edge detection.
- Applying a double-threshold categorization to evaluate edges, and determine the potential ones
- Evaluating edges by categorization: completing the detection of edges by removing all the other edges that are in the low category or are weak but not associated (close to or connected) to edges in the high category.

2.2.1.2 Gaussian Blur:

Each of the pixels for a grayscale image is described by a single number that describes the brightness of the pixel. In order to smoothen an image, the typical answer would be to modify the value of a pixel with the average value of the pixel intensities around it. Averaging out the pixels to reduce the noise will be done by a kernel.The kernel is like a distributed number (arrays) that run across the image and sets each pixel value to the weighted average of its neighbor pixels,thus smoothing the image.

2.2.1.3 Edge Detection:

An edge corresponds to a region in an image where there is a sharp change in the intensity/colour between adjacent pixels in the image. A strong gradient is a steep change and vice versa is a shallow change. So in a way we can say an image is a stack of matrices with rows and columns of intensities.

This means that we can also represent an image in 2D coordinate space, x axis traverses the width (columns) and y axis goes along the image height (rows). Canny function performs a derivative on the x and y axis thereby measuring the change in intensities with respect to adjacent pixels. In other words we are computing the gradient in all directions. It then traces the strongest gradients with a series of white pixels.

1. The low_threshold, high_theshold allow us to isolate the adjacent pixels that follow the strongest gradient.
2. If the gradient is larger than the upper threshold then it is accepted as an edge pixel, if it's below the low threshold then it is rejected.
3. If the gradient is between the thresholds then it is accepted only if it's connected to a strong edge.
4. Areas where it's completely black correspond to low changes in intensity between adjacent pixels whereas the white line represents a region in the image where there is a high change in intensity exceeding the threshold.

2.2.2 Region Of Interest:

The dimensions of the image are chosen which will contain the road lanes and mark it as our region of interest or the triangle. Then a mask is created which is the same as the dimension of the image which would essentially be an array of all zeros. Now we fill the triangle dimension in this mask with the intensity of 255 so that our region of interest dimensions are white.

We need to print the pixel representation of the masked image.The binary representation of 0=0000 & for 255=11111111.

Apply the masked_image on to the canny image to only show the region of interest.

We do this by a bitwise AND operation with the canny image and the mask which will result in our final region of interest.

2.2.3 Hough Transform:

A camera initially captures an image and then extracts the region of interest from the image input. Then the image is converted to the gray - scale image. In order to minimize the processing time The Hough Transform is then used to detect the lanes of straight roads and curved roads.

We know that the Straight line in a 2-D graph is represented by **Slope : y = mx + b**. m and b are two intercepts of the equation and if we find change in x and change in y we can get a point that is represented in Hough space. Now in Hough Space we have a point and if we try to draw straight lines, we can draw n number of lines through that point.

However there is one line that is consistent with both points. We can determine that by looking at the point of intersection enough space because that point of intersection in Hough Space and that point of intersection represents the M and B values of a line consistent with crossing both the points. Now in order to identify the lines, we will first split our Hough space into a grid. Each bin inside the grid corresponds to the slope and y intercept value of the line.

For every point of intersection in a Hough Space bin we're going to cast a vote inside of the bin that it belongs to. The bin with the maximum number of votes will be our line. But as we know that the slope of a vertical line is infinity. So to express vertical lines, we will use polar coordinates instead of cartesian coordinates.

Chapter - 3 : Research issues/observations from literature Survey:

Of the many major issues facing the graduate student, a primary one is the identification of a research problem. Problems may arise from real-world settings or be generated from theoretical frameworks. The source of research problems will vary according to the experience of the person contemplating an investigation, but it is generally agreed that the process begins with a question or need.

SYSTEM ANALYSIS

Chapter - 1 : Introduction of System Analysis

This chapter gives the information regarding analysis done for the proposed system. System Analysis is done to capture the requirement of the user of the proposed system. It also provides the information regarding the existing system and also the need for the proposed system. The key features of the proposed system and the requirement specifications of the proposed system are discussed below.

Chapter - 2 : Existing system

The images from the cameras are used to detect three lanes, and detect vehicles. In the lane detection, the line detection is used. Moreover, a Kalman filter is used to track the detected vehicle. Finally, the relative speed between the detected vehicle and the test is computed in this work.

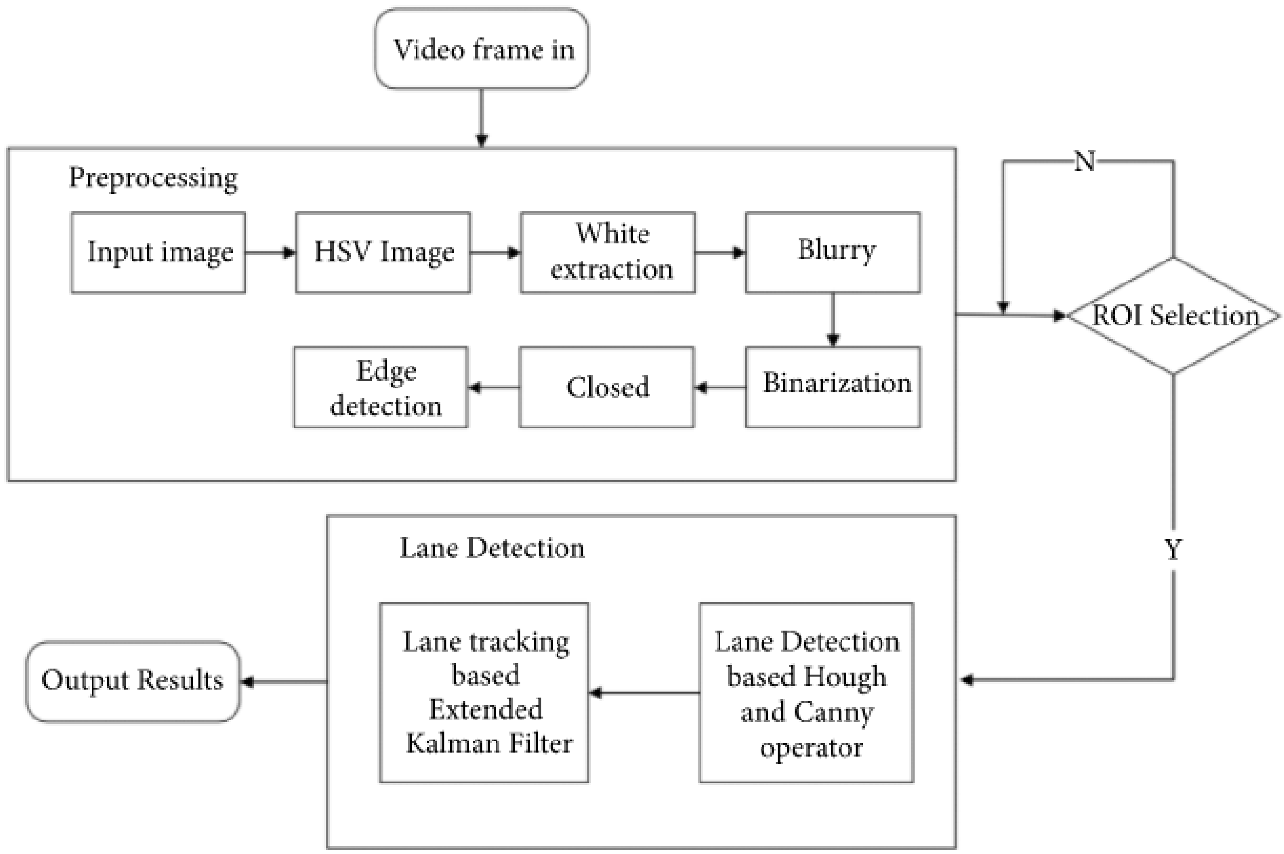
Chapter - 3 : Proposed system

The lane detection module is usually divided into two steps: image preprocessing and the establishment and matching of line lane detection models.

The first step is to read the frames in the video stream. The second step is to enter the image preprocessing module. What is different from others is that in the preprocessing stage we not only process the image itself but also do colour feature extraction and edge feature extraction. In order to reduce the influence of noise in the process of motion and tracking, after extracting the colour features of the image, we need to use a Gaussian filter to smooth the image. Then, the image is obtained by binary threshold processing and morphological closure. These are the preprocessing methods mentioned in this paper.

SYSTEM DESIGN AND IMPLEMENTATION

Chapter - 1 : System Architecture



Fig(2):System Architecture

Chapter - 2 : IMPLEMENTATION

The algorithm is implemented using Python and OpenCV computer vision library and the following steps describe the implemented pipeline in order of execution:

1. **Reading test images:** It simply reads all the images in the designated directory "test_images" one at a time in alphabetical order.
2. **Converting the color test image to grey:** this is done using the OpenCV function **"gray= cv2.cvtColor(lane_image,cv2.COLOR_RGB2GRAY)"**.

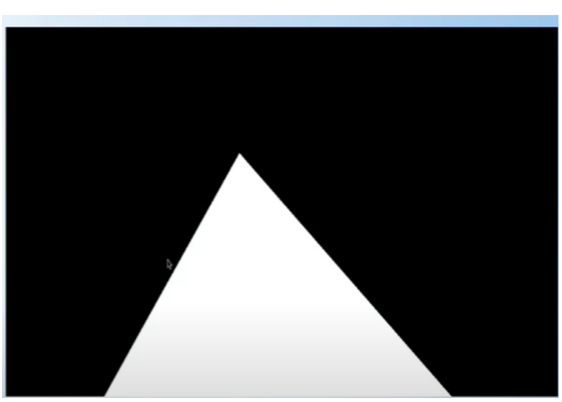
3. **Filtering the noise:** this is done using the OpenCV function "GaussianBlur" which executes the Gaussian filter algorithm. A kernel size of 5 has been selected.
4. **Edges Detection and Extraction:** this is done using the OpenCV function "Canny" which executes the well known Canny algorithm.The several parameters for the algorithm operation that have been selected after careful tuning and many trial and errors sessions.

   Low threshold=50;High threshold=150

Fig(3):After applying the Canny function

5.**The Identification of the region of interest:** This is implemented by masking a trapezoidal area in an image with edges detected to produce an image which has only lines corresponding to road lane lines.

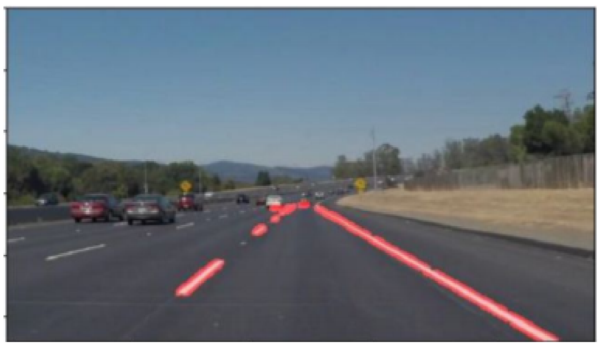
" poly= np.array([[(200,height),(1100,height),(550,250)]])"vertices of the region

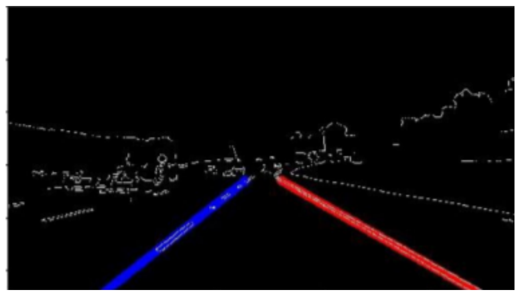


Fig(4):the image of the mask

Fig(5): the masked_image

6.**Detecting Straight Lines:** straight lines are identified using the Hough algorithm in polar coordinates using the OpenCV HoughLinesP() function. The resultant Hough line segments are depicted in . Moreover, additional functionality is added to not only calculate Hough line segments but also to draw extrapolated lines on the original image.The left lane line is always drawn in blue and the right lane-line is drawn in red. The parameters that have been found, after careful and recursive tuning, and used in the Hough algorithm.





Fig(6):Hough line segments **Fig(7):The image after drawing lane lines**

Chapter - 3 : Hardware/Software interface

To get the most accurate results with the state-of-the-art in lane detection. This model must be tested on NVIDIA GTX Titan X hardware to reach 20 frames per second performance. Jupyter Notebook or Google Colab software is enough to run our model.

FUTURE ENHANCEMENT AND CONCLUSION

Chapter - 1 : Future Enhancement

This model can be updated and tuned with more efficient mathematical modelling, whereas the classical OpenCV approach is limited and no upgrade is possible as the approach is not efficient. It is unable to give accurate results on the roads which do not have clear markings present on the roads. Also it cannot work for all climatic conditions This technology is increasing the number of applications such as traffic control, traffic monitoring, traffic flow, security etc.

Chapter - 2 : Conclusion

In the methodology, we made use of the OpenCV library and its functions such as the Canny Function through which we achieved edge detection. Then we prepared a mask of zero intensity and mapped our region of interest by performing the bitwise operation. Then we used the Hough Transform technique that detected the straight lines in the image and identified the lane lines. We made use of the polar coordinates since the Cartesian coordinates don't give us an appropriate slope of vertical and horizontal lines. Finally, we combined the lane image with our zero-intensity image to show lane lines.

REFERENCES

[1] Jay Hoon Jung, Yousun Shin, YoungMin Kwon (2019). “Extension of Convolutional Neural Network with General Image Processing Kernels”.

[2] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi (2016). “You Only Look Once: Unified, Real-Time Object Detection”.

[3] Mayank Singh Chauhan, Arshdeep Singh, Mansi Khemka, Arneish Prateek, Rijurekha Sen (2019). “Embedded CNN based vehicle classification and counting in non-laned road traffic”.

[4] Tejas Mahale, Chaoran Chen, Wenhui Zhang (2018). “End to End Video Segmentation for Driving: Lane Detection for Autonomous Car”