IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

REAL TIME ROAD LANE DETECTION SYSTEM

Prof. Asmeeta Mali Guide,
Komal Shyam*1, Vedant Khule*2, Yukta More*3, Sangram Shinde*4 Students
Department of Computer Engineering
Dr. D. Y. Patil Institute of Technology Pimpri, Pune, India

Abstract: Road accidents have been rising alarmingly, and one of the main reasons of these accidents is a driver's distraction. As a result, several technical advancements have lately been developed in the field of road safety. Technological advancements should be implemented to prevent mishaps and stay safe. Utilizing lane detection systems, which work by identifying lane borders on the road and warning the driver if he shifts to an improper lane marking, is one way to go about this. Many modern transportation systems that use advanced technology include lane detecting systems. Despite the fact that it can be challenging to achieve due to the many road conditions one may encounter, particularly when driving at night or during the day. The front of the automobile has a camera that records the view of the road and can identify lane markings. The technique employed in this study separates the video image into a number of smaller images and creates image features for each one, which are then used to identify the lane markings on the road. There have been provided a number of techniques for finding lane markers on the road.

Index Terms - Lane Detection, Computer Vision, Canny Edge Detection, Gaussian Blur, Region of Interest, Hough Transform.

I. INTRODUCTION

As traffic in cities increases, road safety becomes more and more important. Most accidents on the avenues are caused by people exiting lanes against the rules. Most of these are the result of the driver's disorganized and slow actions. Both automobiles and pedestrians must maintain lane discipline when using the road.

A type of technology that enables cars to understand their surroundings is computer vision. It is a subset of artificial intelligence that aids in the comprehension of image and video input by software. Finding the lane markings is what the technology aims to do. Its objective is to make the environment safer and the traffic situation better. The proposed system's functionality can range from showing the position of the road lines to the bot on any outdoor display to more complex applications like lane switching in the near future to lessen traffic-related concussions.

Accurate lane detection is important for lane recognition and departure warning systems. Vehicles with the predicting lane borders system give instructions to avoid collisions and sound an alarm when a vehicle crosses a lane boundary. The lane borders need not always be clearly apparent for these intelligent systems to ensure safe travel; in some cases, poor road conditions, insufficient paint used to designate the lane boundaries, and other circumstances can make it challenging for the system to accurately detect the lanes. Other variables may include environmental impacts like fog brought on by constant lightning conditions, day and night conditions, shadows produced by objects like trees or other vehicles, or streetlights. These elements make it difficult to tell a person apart from a road lane in the background of a picture that has been captured.

In order to solve the problems brought up previously as a result of lane border changes. With the primary objective of reducing the frequency of accidents, the algorithm utilized in this work tries to recognize lane markings on the road by giving the system a video of the road as an input. The installation of a system in cars and cabs can prevent accidents brought on by careless driving on the roadways. It will guarantee the security of the kids riding school buses. Additionally, the performance of the driver may be monitored, and Road Transportation Offices may use the system to track and document driver negligence and inattention on the road.

II. ASSUMPTION AND DEPENDENCIES

Assumptions:

When creating the project plan, the following presumptions were made:

1. In order for a road to be constructed, it must meet the standards set by the highway authority
2. Roads should be of good quality in order to ensure safety on the road
3. It is assumed that the weather will be sunny and there will be adequate lighting on the road during the event
4. In addition, it should be noted that the system that is being used to detect lanes is sufficiently fast, and there is also no limit as to how much memory can be allocated.
5. It is important to be able to see the lane paint on the road
6. A yellow lane is used to delineate the beginning of the left side of the road
7. A white lane is used to delineate the Right Side of the Road

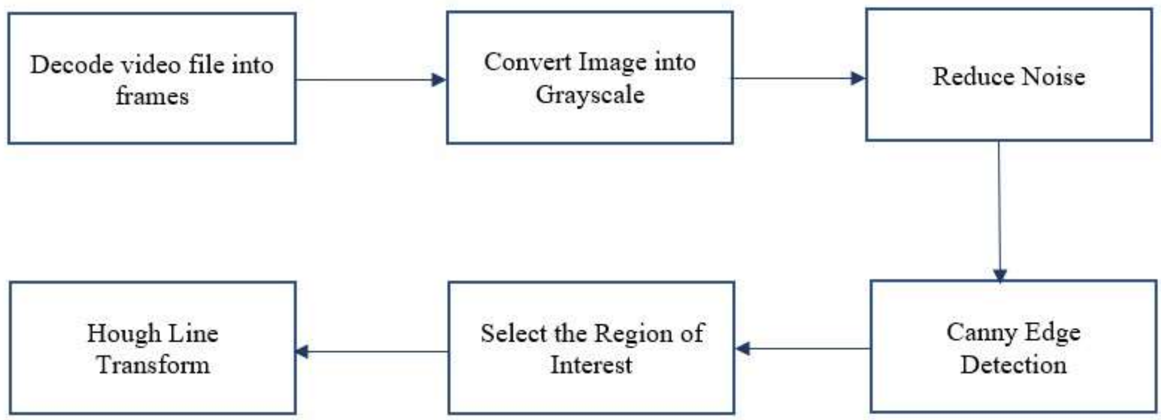
Dependencies:

Dependencies in resource-based planning:

1. High GPU computing power is needed for data training, hence training time is closely related to system raw power.
2. The output of the camera installed on the moving vehicle will significantly influence the outcome of the detection.
3. The accuracy of the lane detection model will be significantly impacted by lane visibility in the weather. Dependencies in logical planning

a. Python libraries need to be pre-installed for this module to function.
b. It's important to update all operating system files.
c. The system must have enough memory to run modules

III. SYSTEM ARCHITECTURE



Steps to be followed:

1. **Decode Captured Video:** We will use the Video Capture object to capture the video, and after initializing the capture, each video frame is decoded.
2. **Grayscale Conversion:** The video frames are in RGB format, however because processing a single channel image is quicker than processing a three-channel colored image, RGB is transformed to grayscale.
3. **Reducing Noise:** Before continuing, it is essential to do image smoothening because noise can produce misleading edges. This technique is carried out using a Gaussian filter.
4. **Canny Edge Detection:** It calculates the gradient of our blurred image in all directions and tracks the edges with significant intensity variations.
5. **Region of Interest:** Only the area covered by the road lane will be considered in this stage. Here, a mask is made that has the same dimensions as our image of the road. Additionally, between each pixel of our clever image and this mask, a bitwise AND operation is carried out. Finally, it hides the clever image and reveals the area of interest that was outlined by the polygonal contour of the mask.
6. **Hough Line Transform:** Straight lines can be found using the Hough Line Transform. Here, the probabilistic Hough Line Transform is utilized, and the output is represented by the identified lines' extremes.

IV. METHODOLOGY

A. The Canny Edge Detection Technique:

The goal of edge detection is to identify item borders in images. In an image, a detection is used to look for regions where the intensity sharply varies. It is possible to identify an image as a matrix or a collection of pixels. The amount of light existing at a particular location in an image is represented by a pixel. A value of zero implies that something is completely black, while a value of 255 indicates that something is completely white. Each pixel's intensity is represented by a numeric value ranging from 0 to 255. A gradient is a pattern of pixels with different brightness's. While a little gradient denotes a shallow shift, a high gradient denotes a steep change.

Additionally, there is a bright pixel in the gradient image anywhere there is a strong gradient, or wherever there is an abrupt change in intensity (rapid change in brightness). By tracing out each of these pixels, we may obtain the edges. The edges in our road image will be found using this idea.

We will load and read our image into an array.

We then convert the image to grayscale.

B. Gaussian Blur:

Each pixel in a grayscale image is represented by a single number that represents its brightness. Changing a pixel's value to match the average value of the around pixels' intensities is a typical way to smooth an image. In order to lower noise, a kernel will average the pixels. Our entire image is smoothed down using this kernel of normally distributed numbers (np.array ([[1,2,3],[4,5,6], [7,8,9]]), with each pixel's value set to the weighted average of its neighbors.

C. Edge Detection:

An edge is a spot in a picture where the contrast between adjacent pixels in the image drastically varies. A considerable gradient is one that is steep, as opposed to one that is shallow. An image can be compared to a matrix with rows and columns of intensities in this manner. With the x axis travelling the width (columns) and the y axis traversing the image height, a picture can also be represented in 2D coordinate space (rows). The Canny function computes a derivative on the x and y axes to determine the brightness difference between adjacent pixels. In other words, we are figuring out the gradient (or change in brightness) in every direction. It then traces the strongest gradients with a series of white pixels.

The low threshold and high threshold functions can be used to distinguish between neighbouring pixels that adhere to the strongest gradient. If the gradient exceeds the upper threshold, it is accepted as an edge pixel; if it falls below the lower level, it is discarded. Only if the gradient is connected to a strong edge and falls within the parameters is it permitted. The entirely black areas correspond to low fluctuations in intensity between adjacent pixels, whereas the white line shows a spot in the image where there is a substantial shift in intensity above the threshold.

D. Region of Interest:

The image's size was selected to show the lanes on the roads and to highlight the triangle as our area of attention. Then, a mask that has the same dimension as the image and is essentially an array of all zeros is created. In order to render the dimensions of our region of interest white, we will now fill the triangle dimension in this mask with 255. To obtain our final region of interest, we will now perform a bitwise AND operation on the clever picture and the mask.

E. Hough Transform:

Now, we use the Hough transform method to identify the lane lines in the image by looking for straight lines. The following formula describes a straight line: y= mx + b. The slope of the line is just a climb over a run. The line can be represented as a single dot in Hough Space if the y intercept and slope are specified. There are many lines, each with a different 'm' and 'b' value, that can pass through this dot. There are numerous lines that can cross each point, each with a distinct slope and y intercept value. However, a single line runs between the two locations. By examining the point of intersection in sufficient space, which reflects the 'm' and 'b' values of a line that crosses both places in Hough Space, we may determine this. We must first create a grid in our Hough space before we can identify the lines. The slope and y intercept values of the line are represented by a bin in the grid. The bin to which a point of intersection in a Hough Space belongs will cast a vote for that point. The container with the most votes will be where we draw our line. On the other hand, a vertical line has an infinitely steep slope. Therefore, we shall utilize polar coordinates rather than cartesian coordinates to express vertical lines. With the help of the while loop and the video capture function, the same methodology can now be used to read each frame of a movie that has many frames or images.

V. CONCLUSION

Using the OpenCV library and techniques like the Canny Function, we were able to detect edges. Once we had a zero-intensity mask made, we used the bitwise method to map our area of interest. Following that, the lane lines in the image were located and detected using the Hough Transform technique. Polar coordinates were used since a slope for the vertical and horizontal lines in Cartesian coordinates could not be determined. Our zero-intensity image was combined with the original image to create a final representation of the lane lines.

REFERENCES

1. Lane Detection using lane boundary marker network with road geometry constraints: Hussam Ullah Khan, Afsheen Rafaqat Ali, Ali Hassan, Ahmed Ali, Wajahat Kazmi, Aamer Zaheer - 14 May 2020

2. Deep reinforcement learning based lane detection and localization: Zhiyuan Zhao, Qi Wang, Xuelong Li - 6 November 2020
3. Robust lane detection and object tracking: Muhammad Shafique, Muhammad Fahim, Prashant Pydipogu - JUN 2013
4. A Machine Learning approach for Detecting and Tracking Road Boundary Lanes: Satish Kumar Satti, Dr K Suganya Devi, Prasenjit Dhar, Dr P Srinivasan - August 2020
5. ROAD LANE LINE DETECTION USING OPENCV: Dr. A.K. Madan, Divyansha Jharwal - 05/May-2022
6. Vision Based Road Lane Detection System for Vehicles Guidance: Othman O Khalifa, Md. Rafiqul Islam, Abdulhakam.AM. Assidi, Aisha-Hassan Abdullah and Sheroz Khan - May 2011
7. Real-time Robust Lane Detection and Warning System using Hough Transform Method: Prajakta R. Yelwande, Prof. Aditi Jahagirdar - 26-08-2019