IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 2, June 2023

Lane Line Detection using OpenCV

Sonali R[1], Lokeshwari[2], Meena R[3]
UG Students, Department of Computer Science and Engineering[1,2]
Faculty, Department of Computer Science and Engineering[3]
Prathyusha Engineering College, Chennai, India

Abstract: *Many modern transportation systems use lane detecting systems, which are crucial components. Nevertheless, because of the various road conditions, it is a challenging task to implement. Humans employ their optical vision during vehicle operating to manoeuvre the vehicle. The creation of an open cv-based automatic lane detection system is one of the requirements for a self-driving car. Computer vision is a technology that enables cars to understand their surroundings and recognise the things in the video image. It is a subfield of artificial intelligence that enables software to comprehend the information of images and videos. The front of the vehicle has a camera that records the view of the road and can identify lane markings. There have been presented a number of techniques for locating lane markings on the road.*

Keywords: Lane System

I. INTRODUCTION

A marked lane is a section of the road that can be used by one line of cars to guide and manage traffic so that collisions between vehicles are minimised. One of the fundamental roles of Intelligent Transportation Systems (ITS) is to increase safety and save human lives. Both motorists and pedestrians must maintain lane discipline. Finding the lane markings is the system's objective. Its objective is tomake the environment safer and the traffic situation better. A crucialelement of both self-driving automobiles and generic computer vision is lane line identification. To eliminate the possibility of entering another lane, this idea is utilised to describe the route for self-drivingcars. We will locate the lines in the lane markings on the road whereautonomous vehicles must travel using Python computer vision techniques. This will be a crucial component of autonomous vehicles because they should not cross their lanes or drive in the opposite direction to prevent accidents. Accurate detection of lane roads is essential for lane recognition and departure warning systems. The technology will instruct the vehicles to prevent collisions and sound an alarm when a vehicle crosses a lane line. These clever solutions always ensure safe transportation. The Road Transport Offices can utilise this system to monitor and report driver negligence and irresponsibility on the road while also tracking the driver's performance.

II. LITERATURE REVIEW

In this project, we utilised OpenCV and Python, one of the most popular computer languages for this task, to identify lane lines in photos. The term "OpenCV" refers to a module with many practical tools for image analysis. OpenCV is an acronym for "Open-Source Computer Vision." For a reliable assessment of the vehicle's position in relation to the lane, a lane recognition system must be able to recognise all kinds of road markers and filter them. Numerous programming languages, including Java and Python, are supported by OpenCV. It can evaluate photos and movies to identify items, people, and even a person's handwriting.

The paper takes into account two different neural network types and expands the concept of deep learning in lane detection by creating a multitask deep CNN. The effectiveness of the CNN and RNN detectors in detecting lanes is further demonstrated. Several approaches, including the Hough Transform, Canny edge detection algorithm, and bilateral filter, are used to detect road markings and road boundaries. These are how they all primarily function as:

HOUGH TRANSFORM

In image analysis and digital image processing, the Hough transform is a method for feature extraction. To determine an image's precise location or grasp its geometrical structure, it is frequently used to detect circles, ellipses, and lines. The Hough transform is a perfect tool for a self-driving automobile to recognise lane lines because of its capacity to recognise forms. In the edge detected image, we utilise the Hough Transform to choose the lane lines. Straight lines can be found using the Hough Transform.

In essence, counting the intersections between curves can be used to locate a line. The line represented by an intersection will have more points if there are more curves intersecting it. The first counterargument is the clipped gradient image of lane lines that was produced earlier. The resolution of the Hough accumulator array— the grid designed to recognise most intersections—is specified by the second and third inputs. The threshold value, which is the fourth input, determines the bare minimum of votes necessary to detect a line. The Hough Transform accomplishes this. It keeps track of the points on each curve where the image's points intersect. It defines it to be a line if there are more intersections than a predetermined threshold.

CANNY EDGE DETECTION

The concept of edge detection is to find locations in a digital image where the brightness of the image abruptly changes. The sharp transitions in image brightness are grouped into a series of curved line segments known as edges. By identifying points in a digital image with discontinuities—basically, abrupt changes in the image brightness—edge detection is an image processing approach. The edges (or boundaries) of the image are those regions where the brightness of the image fluctuates dramatically. The ensuing effort of deciphering the information contents in the original image may consequently become significantly easier if the edge detection phase is successful. The ability to extract such perfect edges from moderately complex real-world photos is not always attainable, though. For this project, Canny Edge Detection was chosen. A multiple stage approach is used by the Canny edge detector, an edge detection technique, to find edges in images. Its objective is to determine the best edge detection.

Low error rates and the ability to filter undesirable data from useful data should be the main criteria. The second objective is to maintain as little variance as possible between the raw image and the processed image. Multiple replies to a grasp are eliminated by the third criterion. The smart edge detector initially smooths the image to remove noise based on these criteria.

GAUSSIAN BLUR (GAUSSIAN SMOOTHING)

In image processing, Gaussian blur is sometimes referred to as Gaussian smoothing. It is the result of applying a Gaussian function to blur an image. A 2-D convolution operator known as the Gaussian smoothing operator may be used to 'blur' images and remove noise and detail. In this way, it is similar to the mean filter, but it makes use of a unique kernel that simulates the shape of a Gaussian ('bell- shaped') hump. This kernel operates with normal distribution and has certain unique characteristics.

Graphics software frequently employs the Gaussian Smoothing effect, which is typically employed to lower image noise. In reality, the smoothing algorithm's "kernel" reflects the form of the function that is used to calculate the average of the nearby points. To blur and eliminate extraneous detail and noise, the Gaussian filter is used.

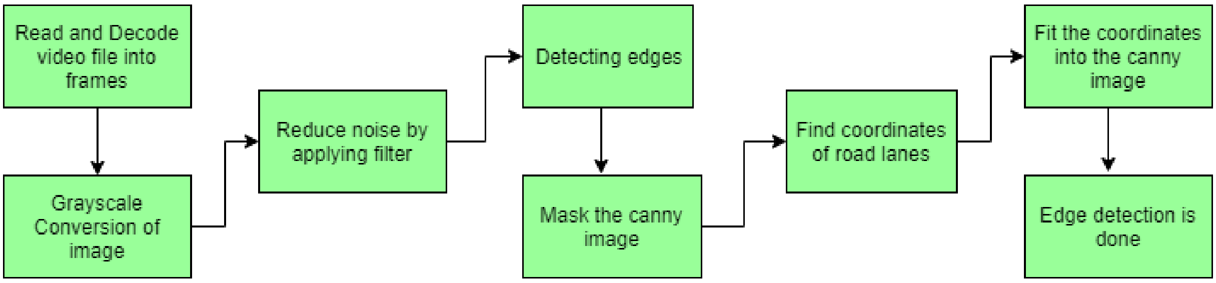
The conventional convolution process will be used to accomplish the Gaussian smoothing. Compared to the specific Image, a convolution mask is much smaller. Each square of pixels is moved over the image matrix by the kernel one at a time.

A kernel with a Gaussian form, or a normal distribution curve, is referred to as a Gaussian kernel. We will first convert the image to grayscale before applying the Gaussian filter so that it will be simpler for us to discern between the results of the clever edge detector. We have developed a pipeline for lane detection now that the following methods and features have been established. Our input image will go through each of the necessary processes listed above using this function.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 2, June 2023

III. METHODOLOGY

The project focuses on lane line detection in a picture using Python and OpenCV. OpenCV, which stands for "Open-Source Computer Vision," is a software collection that contains a number of crucial tools for image analysis. Colour selection, region of interest selection, grey scalability, Gaussian smoothing, Canny Edge Detection, and Hough Transform line detection are the available techniques



Color Selection

First, let's choose some colours. Consider the following example: Lane lines are typically white, and we know that the RGB value of white is (255,255,255). Red, green, and blue thresholds will be used to construct a colour threshold in this case, and these values will beused to populate the RGB threshold. Red, Green, and Blue (R, G, B)have minimal values in this vector.

Region Masking

Assuming that the front-facing camera used to capture the image is fixedly installed on the vehicle, the lane markings should always be visible in the same general area of the picture. Only take into accountpixels for colour selection using this criterion in the area where we anticipate seeing lane lines.

Gaussian Blur

The pre-processing phase of gaussian blur (also known as gaussian smoothing) is used to remove noise from images. This method allows us to maintain the image's most pronounced edges while removing many other detected edges.

Canny Edge Detection

A multi-stage method is used by the Canny edge detector, an edge detection operator, to identify a variety of edges in images. Its goal isto identify the best edge detection method. 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. The thresholds utilised for edge detection arelow threshold and high threshold

Hough Transform

One method for isolating features of a specific shape inside a picture is the Hough transform. Its objective is to identify straight lines in the image in order to determine the lane lines. The image is initially turned into a binary image using some type of thresholding for the purpose of detecting lines in images, and the positive or appropriate examples are then added to the dataset. A point in "gradient vs. intercept" space is created by the Hough transformation from a "x vs. y" line. In Hough space, lines and points in the image will match.

Thus, the intersection of two lines in Hough space corresponds to the intersection of a line in Cartesian space.

IV. EXISTING SYSTEM

The current system only allows for use on ideal road conditions, like a runway. Since Simulink Edge Detection, which is implemented in MATLAB, was the edge detection method previously used, this could not be used on regular roads. The Hough transform is the secondary factor in the current system. To detect objects in a single dimension of an image, space is only used for angle rotation and has a very small road dataset.

V. PROPOSED SYSTEM

In our proposed system, we substitute Canny Edge Detection for Simulink Edge Detection, a more contemporary and effective implementation that uses Python rather than MATLAB. Python's support for quicker execution of mathematical operations makes it possible for the Canny Edge Detection approach to utilise them.

Python is a scripting and statistical modelling language. Second, we employ Hough Transform Space to recognise 3-Dimensional objects, which is more rapid and precise than 1-D object detection.

VI. FUTURE WORKS

Because we use modular implementation, updating algorithms is straightforward, and work on the model can be continued in the future. We add the model's pickle file to the necessary places so that it can be easily transferred to items. As a result, compiling the entire huge code may be easily avoided. By imagining a future in which the road can be seen in the dark or at night, we can further develop the concept. In daylight, the colour recognition and selection process are highly effective. Adding shadows will make things a little noisier, but it will not be as rigorous as driving at night or in low light (e.g., severe fog). Furthermore, this study can only identify lanes on bituminous roads; loamy soil roads, which are common in Indian villages, are not included. This project can be enhanced to detect and prevent accidents on roads with loamy soil that are present in communities. Therefore, this project can be enhanced further to identify the roads with loamy soil that are present in the villages and stop accidents. As we did with Python, we can also use the forthcoming, simple, and understandable programming language Julia to implement our project. The most anticipated project, self-driving cars, depend heavily on this road detection system.

VII. CONCLUSION

We make navigational decisions while driving by using our vision. Our constant reference for where to steer the vehicle is the pattern of lines on the road that the model has identified as the locations of the lanes. Also automatically performed is this steering. Lane detecting systems will become increasingly popular as a result of all the advancements being made in the field of autonomous vehicles. Naturally, one of the first things we would like to do when creating a self-driving car is to use an algorithm to automatically recognise lane lines. The area of interest for road detection (ROI) needs to be adaptable. Results for edge-canny detection and gaussian blur are improved using the Hough transform. For rural road photos, there is a need for a high-quality dataset, which will expand the project's potential application areas. The luxury of computerised driving will entice users to test out newer technology. The next significant development in transportation technology seems to be driverless cars. The software is still being developed and upgraded as autonomous car technology advances. Although the idea of driverless cars was the beginning, radio waves, cameras, sensors, and other semi- autonomous features will soon be developed. This will ease traffic and improve safety by allowing for quicker reflexes and fewer mistakes. Finally, in order to display lane lines, we combined the original image with our zero-intensity image.

REFERENCES

[1] Bar Hillel, Aharon, et al. "Recent progress in road and lane detection: a survey." Machine vision and applications 25.3 (2014): 727-745.

[2] Kim, ZuWhan. "Robust lane detection and tracking in challenging scenarios." IEEE Transactions on intelligent transportation systems 9.1 (2008): 16-26.

[3] Kaur, Gurveen, and Dinesh Kumar. "Lane detection techniques: A review." International Journal of Computer Applications 112.10 (2015).

[4] Wang, Yue, Dinggang Shen, and Eam Khwang Teoh. "Lane detection using spline model." Pattern Recognition Letters 21.8 (2000): 677-689.
[5] Assidiq, Abdulhakam AM, et al. "Real time lane detection for autonomous vehicles." 2008 Inte