Road Marking Detection for Vision Based Driver Assistance System

Thota Sridevi[1], Shaik Rahamthulla[2], Joneboyina Pradeep[2], Gurram Chandra Roshini[2]

[1]Associate professor, Department of ECE, K.L.University, Guntur, A.P, India
[2]Project Students, Department of ECE, K L University, Guntur, A.P, India

ABSTRACT

Vision Based Driver Assistance Systems are in-vehicle systems that are designed to increase road safety by helping drivers to become better aware of the road. These systems will provide a better view of road on display to guide the drivers by enhancing the visibility of road in adverse weather conditions, to avoid collisions. The goal of each Driver Assistance System is to make driving easier and safer. We are designing a system, which will give us information about Road markings. Our Motto is to develop an efficient algorithm for Road marking detection, tracking and warning for Vision based Driver Assistance systems in real time. The Algorithm consists of five steps, video grabbing, pre-processing, edge detection, Hough transform and post processing. We are implementing this algorithm to overcome the disadvantages of existing methods.

***Keywords* – Driver assistance system, Edge detection, Hough transform, road markings**

I. INTRODUCTION

Driver Assistance Systems are in-vehicle systems that are designed to increase road safety by helping drivers to become better aware of the road [1]. Vision based systems will provide a better view of road on display to guide the drivers. Few examples of Driver Assistance systems are GPS maps, Advanced Front-lighting System (AFS), 3D In-Dash Visualization, Collision Avoidance systems. Every system is designed with a specific application. Similarly, in this paper we are developing a road marking detection and tracking algorithm for assisting the driver using driver assistance systems. Road markings on the surface of the road are used for the control, warning, guidance or information of road users and to guide advanced driver assistance systems. Already existed methods [2] for road marking detection are, **positive negative gradients algorithm** is able to detect vertical and curved markings. But only one line is processed at a time. Horizontal markings in an image are not detected and spatial continuity of lane markings is not exploited in this extractor. **Steerable filters** are used to achieve spatial continuity by using Gaussian filters. These filters are optimal for one marking width and they are unable to tackle large width range. **Top-hat filter**, which uses convolution at many scales and can handle large width range. Like positive negative gradient extractor this is also dedicated to vertical lane markings. **Global thresholding** is used to extract markings by global gray level thresholding. But it will not work due to variations in lighting conditions within an image. **Local threshold** [3] extraction is used to avoid lighting variations and in this extraction thresholding is done by comparing each pixel with a fixed threshold level. **Symmetrical local threshold** uses threshold level and left, right intensity averages for extraction. In this the processing is done line by line similar to Top-hat filter and positive negative gradient extractors.

To overcome the disadvantages of above specified methods, we are designing an algorithm with five steps, video grabbing, pre processing, Edge detection, Hough transform, post processing. These steps will be explained in preceding sections.

II. BLOCK DIAGRAM

Block diagram of the proposed model consists of '3' units

- Video grabbing (Image acquisition)
    - video acquisition
    - Converting video to frames
- Image processing unit
    - Pre processing
    - Edge detection
    - Hough transform
    - Post processing
- Output or Display unit

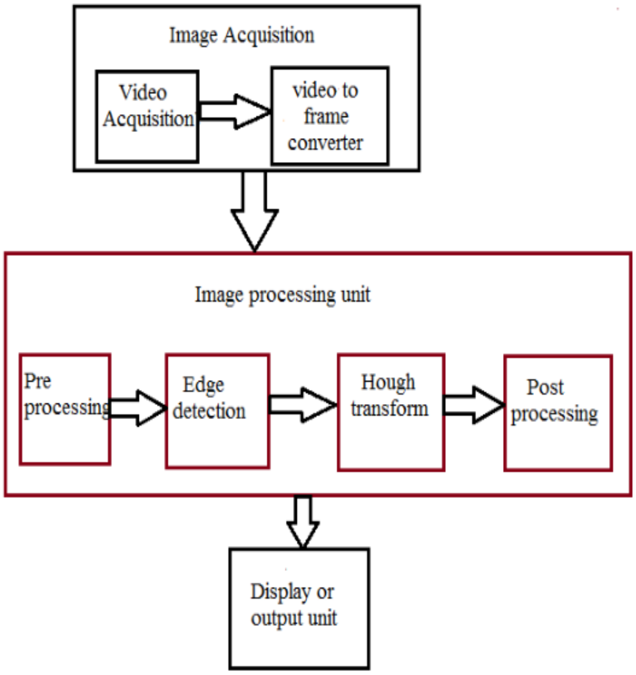

Fig: 2.1 Block diagram for Road marking detection

2.1 Image Acquisition

In this paper, we are considering Highway road video with less traffic as test video. This video is converted to frames for further processing. This converting process is known as Video Grabbing.

2.2 Image processing unit

Each frame is preprocessed if its quality is poor. Generally applied preprocessing techniques are Smoothing techniques. Frames may contain the unnecessary information like trees, sky etc. This information is cropped from the image in preprocessing. After pre-processing Canny Edge detection is employed to the image which will give the fine details of image. Hough transform is applied to the resultant image to detect the road markings as lines. If any extra lines were occurred they will be removed in post processing.

2.3 Output or Display unit

Generally used display units for driver assistance systems are visual Monitors, Warning Alarms or Combination of both. In this paper, we are displaying results on visual monitors.

III. PROPOSED ROAD MARKING ALGORITHM

3.1 Algorithm

```
Video is converted to frames
For each frame
{
    Find the edges in the image by canny method.
    For all pixels in edge detected image
        If the pixel is an edge
            For all angles [0 180]
            Calculate the Rho value
            Increment the position (Rho, theta) in
            Hough matrix
    Locate  the highest values in Hough matrix
    Draw lines for located values
}
Convert output frames (images) to video.
```

for edge detection are Gradients or Second order derivatives. Mostly used operators like Robert, Sobel and Prewitt are derived from gradients [4]. In this paper we are using sobel operator in canny edge detection.

3.2.1 Canny edge detection

Canny detection algorithm consists of five steps

- Gaussian Smoothing
- Gradient Analysis
- Non-maximum Suppression
- Hysteresis Thresholding
- Edge Linking

Step1*:* First step involves smoothing the image [4] by convolving with the Gaussian filter. Smoothing is to reduce the noise within an image

Gaussian smoothing function is given by

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \text{.............. (1)}$$

Where σ is standard deviation
2-D convolution operation is

$$I'(x,y) = \sum_{k=-N}^{N} \sum_{l=-N}^{N} g(k,l) I(x-k, y-l) \quad \text{........................ (2)}$$

Where g (k, l) = Convolution Kernel
I(x, y) = Original image

The larger the width of Gaussian mask, the lower is the detector's sensitivity to noise.

Step2: This step is to find edge strength by taking the gradient of the image using Sobel operator. The sobel operator uses a pair of 3x3 convolution masks [5], one estimating gradient in X-direction and one in Y-direction

+1	+2	+1
0	0	0
-1	-2	-1

Gy

-1	0	+1
-2	0	+2
-1	0	+1

Gx

Magnitude of edge strength is given by [4]: $|G| = \sqrt{Gx^2 + Gy^2}$ …………………………..……. (3)

Angle of orientation of the edge is given by: $\theta = \tan^{-1}(Gy/Gx)$………………………………. (4)

Step3: Once the edge direction is known, the next step is to relate the edge direction to one of the nearest four possible directions (0, 45, 90,135 degrees) [6]. Along the edge direction it finds the local maxima and suppresses the remaining pixels. This step is to maintain the single pixel thin edges and to minimize false edges.

Step4*:* In this step we will keep two thresholds high, low to detect strong and weak edges. Two thresholds are used to allow the little intensity variations near threshold level [4]. Strong edges are considered as true edges and weak edges may or may not be true.

Step5*:* Any pixel that is connected to the strong edge and that have a value greater than low threshold are also selected as edge pixel [6].

Performance of canny algorithm depends on the adjustable parameters σ and the threshold values.

3.3 Hough transform

Hough transform is a technique used to find shapes like lines, circles, ellipses, etc that are mathematically expressed in a binary digital image. In our project, we considered only straight lines [7]. Edge detected binary image is given as input to the Hough transform.

General equation for a straight line is

$$y = m*x + c \quad \ldots\ldots (5)$$

Where m= slope and c = y-intercept

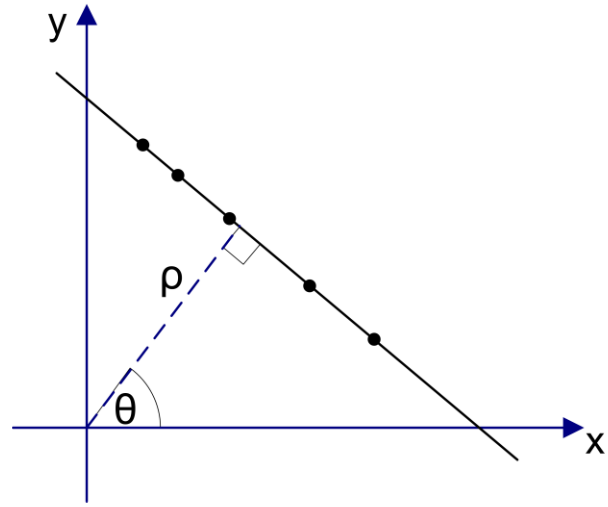

Fig 3.1: Rho 'ρ' and theta 'θ' representation of a straight line.

But in representation of (m, c) parameter space a disadvantage is that, both m and c goes towards infinity when the line becomes more and more vertical. So we are going for another expression of line with some parameters that have limited boundaries [5]. If 'ρ' is the distance from the origin to the line along a vector perpendicular to the line and the angle 'θ' is the angle between the x-axis and the 'ρ' vector.

The line equation is written as:

$$\rho = x*\cos(\theta) + y*\sin(\theta) \quad \ldots\ldots (6)$$

Boundaries of (ρ, θ) are $\rho \in [-D,D]$ where D is the diagonal of the image and $\theta \in [0, 180]$ in degrees. Using the above equation each pixel in spatial co-ordinates is represented with a unique parameter set (ρ,θ) in the Hough space. The result of the Hough transform is stored in a matrix called as Accumulator. Matrix dimensions are 'ρ' values and 'θ' values along two dimensions. The element with the highest value tells what line is most represented in the input image.

IV. RESULTS

Fig 4.1: input image

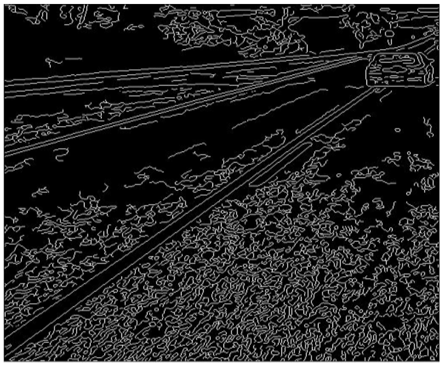

Fig 4.2: Edge detected image

Fig4.3: Image after indicating road markings

V. CONCLUSION & FUTURE SCOPE

In this paper, we have presented an efficient algorithm with edge detection and Hough transform for road marking detection. We concentrated mainly on straight lines and the results produced gave the solution for the problems occurred with the other filters and existed methods. This application is specifically for highways or less traffic areas where this system is exactly required.

This work can be extended to all types of road shapes. And this can be extended to tracking also and this can be applied to design a real time system for driver assistance.

ACKNOWLEDGEMENTS

The authors like to express their thanks to the management and department of ECE, K L University for their support and encouragement during this work.

REFERENCES

[1] Prof. Ferit Küçükay, Janine Berghol, “Driver Assistant Systems”

[2] Thomas Veit, Jean-Philippe Tarel, Philippe Nicolle and Pierre Charbonnier, P (2008), “Evaluation of Road Marking Feature Extraction” published in *IEEE transactions on image processing, vol. 15, no. 3*, 2006

[3] **Alberto** Broggi, “**A** Massively Parallel Approach to Real-Time Vision-Based Road Markings Detection” Published in *IEEE*.

[4] John canny, P (1986), “A Computational Approach to Edge Detection” published in *IEEE*.

[5] R. C. Gonzalez and R. E. Woods, *Digital image processing* (2nd edition, Prentice-Hall, 2002)

[6] Ehsan Nadernejad, Sara Sharifzadeh, P (2008), “Edge Detection Techniques Evaluations and Comparisons”

[7] Bryan S. Morse, Brigham Young University, “Segmentation (Edge Based, Hough Transform)”

[8] N. Aggarwal and W. C. Karl, “Line detection in images through regularized Hough transform,” *presented at the Conf. Information Science Systems*, 1999.

[9] Hong Shan Neoh, Asher Hazanchukp, P (2005), “Adaptive Edge Detection for Real-Time Video Processing using FPGAs” published in *ALTERA*

[10] Sergei Azernikov, P (2008), “sweeping solids and manifolds”

[11] Thomas B. Moeslund. Image and video processing. August 2008