International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-9, July 2020

An Effective Sharp Curve Lane Detector

A.V.Paramkusam, A.Bharath, B.Ramya, G.Lavanya, B.Karthik

Abstract Sharp bend path location is one of the difficulties of visual condition discernment innovation for self-governing driving. Right now, new hyperbola fitting based technique for bend path recognition is proposed. The strategy for the most part incorporates three sections: extraction, bunching, and hyperbola fitting of path highlight focuses. We analysed our technique with the Bezier bend fitting based, the least squares bend fitting based, the spline fitting based techniques, and a current hyperbola fitting based strategy. Examinations show that our strategy performs superior to these technique.

***Keywords*: Clustering, Hyperbola fitting Structural feature constraint.**

I. INTRODUCTION

Intelligent transport systems vary in technologies applied, but basic management systems such as car navigation, traffic signal control systems, pedestrian's detection, and lane detection.

For lane detection, different methods or techniques are used, such as Hough transform-based, Inverse Perspective Mapping (IPM) based, Threshold based, Edge-based methods. Hough transform method employed to separate the image scrupulous shape features.

This technique will be utilized to identify circles and line shapes. These all methods are used to detect the straight lanes but not the curved lanes. So, for detecting the curved lanes, sharp curve lane identification method is used.

The identification of curve lane is one of the demands in the present technology for autonomous driving.

For curve lane detection we first consider the colour image that colour image is changed to gray intensity picture.

For that gray scale image, we calculate the threshold value, based on this value we calculate the gradient values.

Revised Manuscript Received on July 30, 2020.
***** Correspondence Author
A.V.Paramkusam*, Lendi Institute of Engineering & Technology, JNTUK, Vizianagaram, A.P, India.
A.Bharath, Lendi Institute of Engineering & Technology, JNTUK, Vizianagaram, A.P, India.
B.Ramya, Lendi Institute of Engineering & Technology, JNTUK, Vizianagaram, A.P, India.
G.Lavanya, Lendi Institute of Engineering & Technology, JNTUK, Vizianagaram, A.P, India.
B.Karthik, Lendi Institute of Engineering & Technology, JNTUK, Vizianagaram, A.P, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

II. METHODS

Curve lane detection contains the following methods

- Extraction of feature lane points
- Clustering of feature points
- Fitting of feature points

By using this gradient information, we must extract the lane feature with texture, structure, and color distinctiveness of the lane curve. After extraction process, the clustering will be come into existence. The lane feature points which are extracted in extraction process are clustered within the line characters of the lane in mutually near field of sight and far field of sight. After clustering, the lane feature points were fitted by using hyperbola fitting method.

➢ **Gray scale Processing:**

First, we get the colour features depending on the bright and then extract the edge character. Formula for the extraction process is "I=0.5*(R+G)". Here R&G are the values represent red and green intensity values. The characteristics of white and yellow lane lines are enhanced by this method.

➢ **Gradient calculation**:

Each pixels of a gradient image measures the changes in the intensity value of the same point in the original image.

$$\Delta X\ (i, j) = \sqrt{xi^2 + xj^2}\sqrt{xi^2 + xj^2}$$

In the road images, the pixel of the gray values of a lane line is superior to its area. We consider the x and y directions of the pixel p (i, j) respectively, and calculate as follows:

Fig 1.1 RGB Image of Road.

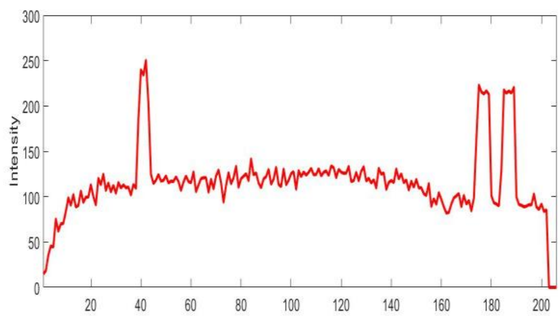

Fig 1.2 Intensity Vs Pixel Graph

$$Xi = \frac{1}{4} f(i, Xi = \frac{1}{4} f(i, j) * \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$Xj = \frac{1}{4} f(i, Xj = \frac{1}{4} f(i, j) * \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Due to these $XiXi$&$XjXj$values we calculate the values of slope of the edge. $Xi\ Xi$ a I-direction and the $Xj\ Xj$ is the j-directions. The changes of the $Xi\ Xi$ and$Xj\ Xj$ are not equal. But the changes of the $Xi\ Xi$ and $Xj\ Xj$ are consistent. The values of the gradient close to the lane edge are large. The side by side edge gradient values are similar. The left edge gradient value of the lane is greater than 0.1 and the right edge gradient of the lane is less than 0.

According to the $Xi\ Xi$ left and right edges of the lane can be determined.

$$\Delta X(i, j) = \begin{cases} \Delta X(i,j), Xi \geq 0 \\ -\Delta X(i,j), Xj < 0 \end{cases} \begin{cases} \Delta X(i,j), Xi \geq 0 \\ -\Delta X(i,j), Xj < 0 \end{cases}$$

The parts of the road and non-road gradient values are very small. When the one pixel of gradient value set to '0'.

$$\Delta X(i, j) =$$

$$\begin{cases} \Delta X(i,j), |\Delta X(i,j)| \geq \Delta Tg \\ 0 |\Delta X(i,j)| < \Delta Tg \end{cases} \begin{cases} \Delta X(i,j), |\Delta X(i,j)| \geq \Delta Tg \\ 0 |\Delta X(i,j)| < \Delta Tg \end{cases}$$

$\Delta Tg\Delta Tg$is a dynamic threshold value. all pixels in the same row of a gradient value expressed in $\Delta Tg\Delta Tg$.

The formula for dynamic threshold value is:

$$\Delta Tg = \frac{1}{n} \sum_{i=1}^{n} |\Delta X(p, q)|$$

$$\Delta Tg = \frac{1}{n} \sum_{i=1}^{n} |\Delta X(p, q)|$$

Here $\Delta X(p,q)\Delta X(p,q)$ represents the gradient points. And p represents the pth column and q represents the qth row.

- **Extraction of lane feature points:**

Based on the edge information, the texture features of the lane are expressed. The gradient value is positive at left edge and the gradient value is negative at the right edge of lane. The acute point is considered as the edge feature points. The feature points located at the left edge are called the increasing edge and feature points which are located at the right edge known as the decreasing edge points.

P (i, j) =

$$\begin{cases} p \uparrow X(i,j) > 0 \text{ and } X(i,j) > X(i \pm 1, j) \\ p \downarrow X(i,j) < 0 \text{ and } X(i,j) < X(i \pm 1, j) \end{cases} \begin{cases} p \uparrow X(i,j) > 0 \text{ and } X(i,j) > X(i \pm 1, j) \\ p \downarrow X(i,j) < 0 \text{ and } X(i,j) < X(i \pm 1, j) \end{cases}$$

Where P (i, j) is the pixel point and X (i, j) represents the gradient. X (i-1, j) denotes gradient at P (i-1, j) . X (i+1, j) represents the gradient at P (i+1, j)

These edges are also called as rising edge & descending edge points. p↓ & p↑ are known as feature pair points. We select the feature points to maintain the original order. We removed According to the formula the value of the rising edges requires the, and the gradient value is larger than the left right side of the pixel gradient values. The complete line must have two edges i.e. left edge and right edge the low pixel points these points are neither in the rising edge nor in the decreasing edge. In this arrangement decreasing edges are adjacent and in front there is rising edge points.

Fig 1.3 Birds Eye View

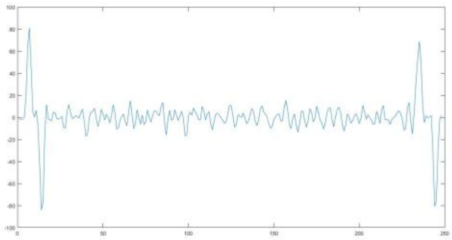

Fig 1.4 Intensity Level of the Pixel

III. STRUCTURAL FEATURE CONSTRAINT:

These feature constraints of the lane line Is manufactured with a certain width. The width can be calculated by the descending and rising edge points of the horizontal co-ordinates. The width as follows:

$w^k w^k = x_m x_m - x_n x_n$ Represents width at structural feature constraints. In the respective image the width of the lane line becomes very small from near too far.

Consider of the nearest picture line is [$w_m^B, w_n^B w_m^B, w_n^B$] and the breadth of the lane is [$w_m^k, w_n^k\ w_m^k, w_n^k$], where $w_m^B w_m^B \leq w_n^B w_n^B, w_m^k w_m^k \leq w_n^k w_n^k$.

$$w_m^k w_m^k = \lambda^k \lambda^k .\ w_m^B\ w_m^B$$

$$w_m^k w_m^k = \lambda^k \lambda^k .\ w_m^B\ w_m^B$$

Calculation of $\lambda^k \lambda^k$ is as follows

$\lambda^k \lambda^k = h^k h^k$/H = K-y0/B-y0. Based on the width range constraint, the breadth at the lane is known, and then some noise feature point pairs are removed. The specific formula is as follows:

$$< p^k p^k \uparrow, p^k p^k \downarrow > = \begin{cases} 1 & w_m^k \leq w^k \leq w_n^k \\ 0 & others \end{cases} \begin{cases} 1 & w_m^k \leq w^k \leq w_n^k \\ 0 & others \end{cases}$$

IV. COLOUR FEATURE CONSTRAINTS:

In the lane line, the gray value is high when it compares to the nearby the gray value road surface, at which is the pixel point of gray value on the lane line is greater than a threshold and is greater than the road threshold value. To define and express this feature, the average gray value to characterize trend is used.

The average gray value is calculated between p↑ and p↓ as follows

avg v = $\frac{1}{N}\sum_{i=m}^{n} f(i, y_k \frac{1}{N}\sum_{i=m}^{n} f(i, y_k)$

Where N is number of pixels between the rising edge point p↑(x_m, y_k x_m, y_k) and descending edge point p↓$(x_n, y_k (x_n, y_k)$. $f(i, y_k\ f(i, y_k)$ represents the gray value of pixel point $P(i, y_k P(i, y_k)$.

We are calculating the average gray value on the left side and the right side of the lane line and it is represented as avg L and avg R respectively. the formulae are as follows

avg L = $\frac{1}{w^k}\sum_{i=m-w^k}^{m-1} f(i, y_k)\frac{1}{w^k}\sum_{i=m-w^k}^{m-1} f(i, y_k)$

avg R = $\frac{1}{w^k}\sum_{i=n+1}^{n+w^k} f(i, y_k)\frac{1}{w^k}\sum_{i=n+1}^{n+w^k} f(i, y_k)$

Here $w^k\ w^k$ is the width of the rising and descending edge points<$p^k\uparrow, p^k\downarrow p^k\uparrow, p^k\downarrow$>.the average gray value can be obtained by using the above three formulas. The following condition must be satisfied for the feature point pair <$p^k\uparrow, p^k\downarrow p^k\uparrow, p^k\downarrow$>.

<$p^k\uparrow, p^k\downarrow p^k\uparrow, p^k\downarrow$>=

$$\begin{cases} 1, avgV > \max(avgL, avgR, avgT) \\ 0, others \end{cases}$$
$$\begin{cases} 1, avgV > \max(avgL, avgR, avgT) \\ 0, others \end{cases}$$

Here avg T is some threshold point which represents the average gray value of the road surface and it represents the bright and dark degree values of the road surface. This value is affected based upon the weather conditions etc. The avg T value can be calculated by setting the region of interest in the road images. The formula is given as

Avg T =

$$\frac{1}{N_{\Omega_{ROI}}}\sum_{p(i,j)\in\Omega_{ROI}} f(i,j)\frac{1}{N_{\Omega_{ROI}}}\sum_{p(i,j)\in\Omega_{ROI}} f(i,j)$$

The feature point pair can be considered as an edge point on the lane line, when it satisfies both structural and colour feature constraints. Then the feature point $p_{lf} p_{lf}$(x, y) can be represented as

$p_{lf} p_{lf}$(X, y) = $\begin{cases} x = (x_m + x_n)/2 \\ y = y_k \end{cases} \begin{cases} x = (x_m + x_n)/2 \\ y = y_k \end{cases}$

V. CLUSTERING OF LANE FEATURE POINT:

In clustering road area divided into two parts.

(1) NFOV (near field of view)

(2) FFOV (far field of view)

Coming to NFOV feature points, it is based on the straight-line detection. The best and effective straight-line detection method is Hough transformation. the principle of this method is every edge point in the map is transformed to all possible lines, when the Hough transformation is done, arrived line segments on the same lane. These lane segments are represented by L {L1, L2, L3... Lk …Ln}. In the NFOV there is straight is straight lane segments and it has two end points

$p_1^k(x_1^k, y_1^k)$ and $p_2^k(x_2^k, y_2^k)$

$p_1^k(x_1^k, y_1^k)$ and $p_2^k(x_2^k, y_2^k)$. The angle between the horizontal line is ɵ and range of this angle is [0, π/2]. NFOV represents by a straight line. Due to the straight line the same lane line segments are merged. If we want to know whether the lines are merged or not, first we consider the two arbitrary line segments $L_i L_i$ and $L_j L_j$ .the similarities between $L_i L_i$ and $L_j L_j$ are represented as $S_{ij} S_{ij}$. The similarities between $L_i L_i$ and $L_j L_j$ calculated based on the distance similarity and direction similarity. The similarities between $L_i L_i$ and $L_j L_j$ calculated as

$$s_{ij} = \frac{\mu d_{ij}}{w} s_{ij} = \frac{\mu d_{ij}}{w} + \frac{2\lambda ɵ_{ij}}{\pi} \frac{2\lambda ɵ_{ij}}{\pi}$$

And the distance similarities and directions similarities calculated as:

$$d_{ij} = (d_U + d_L)/2 d_{ij} = (d_U + d_L)/2$$

$$ɵ_{ij} = |ɵ_i - ɵ_i|\ \ ɵ_{ij} = |ɵ_i - ɵ_i|$$

The horizontal distance of $l_i l_i$ and $l_j l_j$ are represented as $d_v d_v$ and $d_l d_l$ in both upper and lower boundaries of NFOV.$ɵ_i ɵ_i$&$ɵ_j ɵ_j$ are the horizontal angles of $l_i l_i$&$l_j l_j$.

For every line segment we calculate the similarity values and as well as the clustering is done for all the line segments by similarity. After the calculation of similarity values, we must generate a new line by performing the LS straight line fitting method for all the line segments. This new line can be taken as the selected lane line. These lane lines are approximated as both parallel and straight lines. The vanishing points of these lines are same. The interference line segment is removed by processing the candidate line segment.

First of all, we can determine the values of vanishing point v ($x_v, y_v x_v, y_v$) and for each candidate lane line we calculate the lateral coordinates ($x_k x_k$). By using these two values, we calculate the lateral distance $d_x^k\ d_x^k$. It is the distance between the lateral coordinates and the vanishing point. The formula is given as where $x_k x_k$ lane line and it is given by

$$x_k x_k = x_1^k x_1^k + \frac{y_v - y_1^k y_v - y_1^k}{K \qquad K}$$

K is determined as K= $\frac{y_2^k - y_1^k y_2^k - y_1^k}{x_2^k - x_1^k x_2^k - x_1^k}$

The distance d_x^k d_x^k checks that the candidate line segment l_k l_k is a lane line or not. Depending on this judgment we can use distance error ε_d ε_d. When the distance error ε_d ε_d is within the range, then the l_k l_k is considered to be as a lane line, can be filtered. If the candidate line segment is within the range, then these are all can be taken into the red box. The line segments within the red box are considered as the lane, the line segment l_d l_d was not the lane line because it is out of the red box.

Clustering of FFOV Feature Points based on Neighborhood Search:

The FFOV contains both the straight lane and curved lane feature points. By using the point clustering method, we cannot cluster the curved lanes. So, neighborhood feature points method is introduced for clustering curved lanes.

Starting from the seed point the gray scale value and gradient values are performed. Then, the gradient value is determined by the direction of the lane. The search is described, for the local search similarity of gray scale value used.

VI. HYPERBOLA FITTING:

In the hyperbola fitting first, we calculate the key point. Connection among the direct line and curved shape is known as the key point. By using least square method the curve is integral, and using the hyperbola the curve line is fitted. The feature points of data set on lane line L_k L_k is

$P_k(p_1^k, p_2^k, p_3^k, \ldots, p_n^k)$, $P_k(p_1^k, p_2^k, p_3^k, \ldots, p_n^k)$,and the

p_{key}^k p_{key}^k = arg_{key} arg_{key} min $\{\frac{1}{key}\sum_{i=1}^{key}\varepsilon_i^l \frac{1}{key}\sum_{i=1}^{key}\varepsilon_i^l + \frac{n \quad n}{key key}\}$

Where $\varepsilon_i^l \varepsilon_i^l$ is the distance ith feature point for exact surface point of the straight line l_{key}^k.Here l_{key}^k resultant obtained by performing LS method point

p_k^l p_k^l (p_1^k p_1^k , $p_2^k p_2^k$, p_3^k p_3^k, p_4^k p_4^k, p_5^k p_5^k,...... p_{key}^k p_{key}^k).

The parameter $\frac{n \quad n}{key key}$ indicates a high point generated in a directly line. Subsequent to the key point calculation, we obtained the feature points of the data set and now the curve correct is needed. For curve right, we have preferred the hyperbola fitting method since the image is road it curve shape meaning hyperbola. The equation for representation for its hyperbola is given as

$$x - x_v \qquad x - x_v \qquad = \qquad \frac{k}{y - y_v} + k^l(y - y_v) \qquad \frac{k}{y - y_v} + k^l(y - y_v)$$

Here ($x_v x_v$,$y_v y_v$) is the coordinate point of the vanishing point and the straight line. l_{key}^k having the slope of $k^l k^l$. K is curvature parameters of the hyperbola. Depending upon the k value the curve of lane is positioned in some specific direction. If K=0, the curve of lane is positioned as line. If K>0, the lane line is positioned in rightward direction. If K<0, the lane line signifies it is positioned in leftward direction.

K value is given as

$arg_K arg_K \min\{\sum_{i=key}^{n}\varepsilon_i^c \sum_{i=key}^{n}\varepsilon_i^c\}$

Here the distance between the feature point and the hyperbola $l_{HC}^k l_{HC}^k$ is represented as $\varepsilon_i^c \varepsilon_i^c$. The error sum is minimized based upon the K value.

VII. RESULTS AND ANALYSIS:

Here, we analysis the lane detection by using different algorithms. And also introduce the best platform for lane detection. All algorithms are fully verified, and then perform the comparative experiments.

Based on the bunching consequences of path line Highlight focuses, the Bezier cubic bend, the LS cubic bend, the spleen fitting, and the hyperbola are completed for break down favorable circumstances and detriments of all correct strategy. At long last, for checking the power and actual execution calculation, it is investigated exploratory outcomes with various natural circumstances, for example, light, climate, and Vehicle unsettling influence, and checked the executing time of all images.

VIII. EXPERIMENTAL PLATFORM AND EVALUATION:

This experiment implemented in MATLAB. The exhibition assessment of path line discovery can be isolated into emotional assessment and target assessment.

Since the abstract assessment can't precisely depict the discovery exactness of the calculation, we utilized a normally target assessment technique: identification exactness.

The precision A was utilized to assess the impact of path line location. This experiment implemented in MATLAB.

International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-9, July 2020

Fig1. 5: RGB of the Road image

Fig1.6: Birds View

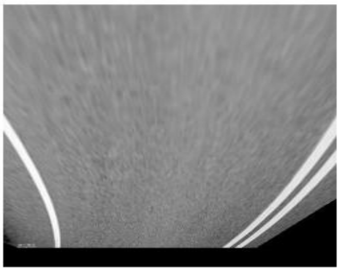

Fig1. 7: gray image of the birds view

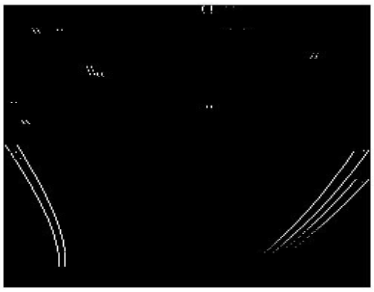

Fig 1.8: output image

IX. CONCLUSION:

The curve path detection in autonomous vehicles play vital rule in field of self-driving cars and mobile robots. Mainly this technique is used in the hill areas where dead curves are present to increase the safety and avoiding the accidents the curve path detection in autonomous vehicles are necessary. This paper develops a method to identify the lane with the help of top view road picture transformation method. The precise line in the whole road image lane is detected next the top angle is portioned by two sections that are near road picture and far road picture. In fact, a straight-line identification is implemented with Hough transformation at cutting-edge a close image section. Whereas, the propose method integrates an LSM and parabolic model for detecting at the distant road picture section.

REFERNCES:

1. C. Rose, J. Britt, J. Allen, and D. Bevly, “An integrated vehicle navigation system utilizing lane-detection and lateral position estimation systems in difficult environments for GPS,” IEEE Trans. Intell. Transp. Syst., vol. 15, no. 6, pp. 2615–2629, Dec. 2014.
2. . V. Gaikwad and S. Lokhande, “Lane departure identification for advanced driver assistance,” IEEE Trans. Intell. Transp. Syst., vol. 16, no. 2, pp. 910–918, Apr. 2015.
3. J. Wang, T. Mei, B. Kong, and H. Wei, “An approach of lane detection based on inverse perspective mapping,” in Proc. IEEE Int. Conf. ITSC, Oct. 2014, pp. 35–38.
4. R. K. Satzoda and M. M. Trivedi, “Drive analysis using vehicle dynamics and vision-based lane semantics,” IEEE Trans. Intell. Transp. Syst., vol. 16, no. 1, pp. 9–18, Feb. 2015.
5. S.-N. Kang, S. Lee, J. Hur and S.-W. Seo, “Multi-lane detection based on accurate geometric lane estimation in highway scenes,” in Proc. IEEE IV Symp., Jun. 2014, pp. 221–226
6. 6. J. Huang, H. Liang, Z. Wang, k Y. Song and Y. Deng, “Lane marking detection based on adaptive threshold segmentation and road classification,” in Proc. IEEE Int. Conf. ROBIO, Dec. 2014, pp. 291–296.
7. G. Liu, S. Li and W. Liu, “Lane detection algorithm based on local feature extraction,” in Proc. IEEE CAC, Nov. 2013, pp. 59–64.
8. V. S. Bottazzi, P. V. K. Borges and J. Jun, “A vision-based lane detection system combining appearance segmentation and tracking of salient points,” in Proc. IEEE IV Symp., Jun. 2013, pp. 443–448.
9. H. Yoo, U. Yang and K. Sohn, “Gradient-enhancing conversion for illumination robust lane detection,” IEEE Trans. Intell. Transp. Syst., vol. 14, no. 3, pp. 1083–1094, Sep. 2013.
10. H. Tan et al., “A novel curve lane detection based on improved river flow and RANSA,” in Proc. IEEE Int. Conf. Intell. Transp. Syst., 2014, pp. 133–138.
11. J. Wang and X. An, “A multi-step curved lane detection algorithm based on hyperbola-pair model,” in Proc. IEEE Int. Conf. Automat. Logistics, 2010, pp. 132–137.
12. R. Cimurs, J. Hwang, and I. H. Suh, “Bezier curve-based smoothing for path planner with curvature constraint,” in Proc. IEEE Int. Conf. Robot. Comput., 2017, pp. 241–248.
13. C. Li et al., “Multi-lane detection based on RMFP for self-driving in urban traffic scenes,” in Proc. Int. Conf. Sens., Mechatronics Automat., 2016, pp. 692–703.