Article

Vision-Based Lane Detection and Tracking under Different Challenging Environmental Conditions

Samia Sultana 1*, Boshir Ahmed 2, Manoranjan Paul 1, Muhammad Rafiqul Islam 1 and Shamim Ahmad 3

1 School of Computing, Mathematics and Engineering, Charles Sturt University, Bathurst, NSW 2795, Australia; sasultana@csu.edu.au (S.S); mpaull@csu.edu.au (M.P.); musilam@csu.edu.au (M.R.I.)
2 Department of Computer Science & Engineering, Rajshahi University of Engineering and Technology, Rajshahi-6204, Bangladesh; boshir@cse.ruet.ac.bd (B.A.)
3 Department of Computer Science & Engineering, University of Rajshahi, Rajshahi, 6205, Bangladesh; shamim_cst@ru.ac.bd (S.A.)

* Correspondence: sasultana@csu.edu.au; Tel.: +61480357073

Abstract: Driving is very challenging when the visibility of a road lane marking is low, obscured or often invisible due to abrupt environmental change which may lead to severe vehicle clash. A large volume of research has been done on lane marking detection. Most of the lane detection methods suffer from four types of major problems: (i) abrupt illumination change due to change in time (day, night), weather, road, etc.; (ii) lane markings get obscured partially or fully when they are colored, eroded or occluded; (iii) blurred view created by adverse weather like rain or snow; and (iv) incorrect lane detection due to presence of other lookalike lines e.g. guardrails, pavement marking, road divider, vehicle lines, the shadow of trees, etc. In this paper, we proposed a robust lane detection and tracking method to detect lane marking considering the abovementioned challenging conditions. In this method, we introduced three key technologies. First, the bilateral filter is applied to smooth and preserve the edges and we introduced an optimized intensity threshold range (OITR) to improve the performance of the canny operator which detects the edges of low intensity (colored, eroded, or blurred) lane markings. Second, we proposed a robust lane verification technique, the angle and length-based geometric constraint (ALGC) algorithm followed by Hough Transform, to verify the characteristics of lane marking and to prevent incorrect lane detection. Finally, a novel lane tracking technique, the horizontally adjustable lane repositioning range (HALRR) algorithm is proposed, which can keep track of the lane position when either left or right or both lane markings are partially and fully invisible for a short period. To evaluate the performance of the proposed method we used the DSDLDE [1] dataset with 1080x1920 resolutions at 24 frames/sec where the video frames containing different challenging scenarios. Experimental results show that the average detection rate is 97.36%, and the average detection time is 29.06msec per frame, which outperformed the state-of-the-art method.

Keywords: Lane Detection and Tracking; Canny edge detector; Optimized intensity threshold range (OITR); Angle and length based geometric constraint (ALGC); horizontally adjustable lane repositioning range (HALRR); Intelligent Vehicles

1. Introduction

The motivation behind the enormous research on advanced driver assistance system, autonomous vehicle, or intelligent transportation system is avoiding vehicle clashes and saving human lives. The lane is a very important part of the road as lots of traffic rules and regulations are based on lane markings. Hence, the performance of advanced driver assistance systems depends on lane marking detection to a great extent. The accuracy and robustness are the two most important properties of lane detection systems. Due to the presence of dramatic variation in the environment of a vehicle roadway, it is very challenging to detect lane. Lane detection is the fundamental operation of different advanced methods.
driver assistance systems (ADASs) such as the lane departure warning system (LDWS) and the lane keeping assistance system (LKAS) [2, 3]. Some successful automotive enterprises, such as Mobileye, BMW, Tesla, etc. have developed their products which include lane departure warnings and lane keeping assistance systems. Either the automotive enterprises or the personal customers have accepted the Mobileye Series ADAS products and Tesla Autopilot for self-driving. Most of the current well-established lane assistance products use the camera as a primary sensor due to its low cost, high accuracy, and high-speed performance.

Aly [4] detected lane marking of the urban street using the RANSAC spline fitting technique. In [5], the algorithm estimated the vanishing point fast and accurately and the estimated vanishing point determined the line segments that belong to the lane marking. Therefore, it can detect the lane under the shadow, but it does not work when confusing pavement marking, or pedestrian crossing exists on the road. In [6] authors used a method that provides strong edges to the lane line in illumination conditions like night time road, yellow or white lamp tunnel, rainy nights, etc. but it does not work well in extreme conditions because they assume that one scene does not have multiple illuminations. [7] reduced the computational complexity by detecting vanishing points and establishing an adaptive region of interest (ROI) and able to detect lanes under illumination like yellow lamp tunnel, sunrise, sunset, rainy, night, etc. But it is unable to detect lanes under strong light reflection, blur lane marks, low sun angle situations, and lane cracks.

In [8], they overcome illumination change effect by detecting vanishing point based on voting map, defining an adaptive ROI and detecting lanes using invariance properties of lane colors. However, their method fails under some extreme conditions like strong light reflection, blur lane marks, low sun angle situations, lane cracks, etc. [9] proposed an algorithm to detect lanes based on the spatiotemporal image. They successfully detected lanes with sharp curvature, lane changes, night roads, obstacles, and lens flare due to using the temporal consistency of lane width on each scanline. Since the detection result depends on the lane width, the system will fail to detect the lane while the lane width increases or decreases on road.

In [10], they proposed a vision-based, integrated framework based on spatio-temporal incremental clustering coupled with curve fitting and Grassmann manifold learning for lane detection, tracking and road surface marking detection. In [1], the algorithm that processes a gradient cue and a color cue together and a line clustering with scan-line tests to verify the characteristics of the lane markings. They successfully detected lane under various weather conditions. However, they can’t detect lane markings when lanes are invisible due to streetlight reflection and the also failed to detect yellow colored lane markings when lane marking get obscured due to the appearance of yellow colored background for nighttime streetlights or for a tunnel with yellow lights. Initially, in [11] we have addressed only the rainy weather challenges. We successfully detected lane under different rainy weather challenges. But we did not consider the other adverse weather like snow or nighttime challenges.

2. Contribution

To overcome these difficulties mentioned above we proposed a robust lane detection and tracking system. To reduce the illumination effects and to enhance the lane marking edges bilateral filter was used for the first time in the lane detection system in our previous paper [12]. So, in this paper, we implemented the bilateral filter on the gray-scaled input images to smooth as well as preserve lane edges. Next, we approached an optimized intensity threshold range (OITR) in the edge detection stage, which improves the performance of the canny edge detector to enhance and detect the edges of low-intensity (colored, eroded, or blurred) lane markings. To remove unwanted noisy line edges area, we have selected a region of interest using an isosceles trapezoid-shaped mask placed in the middle of the horizontal axis and two third of the vertical axis excluding the lower third to avoid the car hood edges. To detect all the lines inside region of interest, Hough
Transform is used. The lines detected by Hough Transform are candidate lane lines (CLL). Here we have separated candidate left lane (CLL) lines from candidate right lane (CRL) lines using slope-based constraint. We proposed a robust lane verification technique, the angle and length-based geometric constrain, to verify the characteristics of lane marking and prevent incorrect lane detection. Finally, a novel lane tracking method, the horizontally adjustable lane repositioning range is introduced, which can keep track of the lane position. In summary, the main contributions of the proposed method are:

- We approached an optimized intensity threshold range (OITR) in the edge detection stage, which improves the performance of canny to enhance and detect the edges of low-intensity (colored, eroded, or blurred) lane markings.
- We proposed a robust lane verification technique, the angle and length based geometric constraint (ALGC) algorithm, to verify the characteristics of lane marking.
- We introduced a novel lane tracking method, the horizontally adjustable lane repositioning range (HALRR) to keep track of the lane position when either left or right or both lane markings are partially or fully invisible due to erosion or occlusion for a short period.

3. Organization

This paper is organized as follows: Section 4 provides a brief description of our proposed algorithms. In section 4.1 performance is evaluated by 30 video clips which consist of more than 33 thousand frames and compared with 4 other recent works based on detection rate, processing time and CPU configuration. In Section 3.3, our method is compared with other recent works in terms of different challenges, time, accuracy, and methodology. Finally, we will conclude our work in Section 4.

4. Methodology

Here, we divided the proposed lane detection and tracking method into four major parts- i) pre-processing, ii) feature extraction and iii) lane detection and iv) lane tracking. The overview of the proposed lane detection method has been shown in figure 1.

![Figure 1: Block diagram of proposed lane detection and tracking method](image)

In the pre-processing stage, the extracted video frames were gray-scaled, and the bilateral filter is used to remove noises as well as preserve the edges. Next, In the feature extraction stage, edges were detected by the canny edge detector improved by OITR and the position of lines was extracted by Hough transform. After that, in the lane detection
stage, a robust lane verification technique, angle and length based geometric constraint is proposed which successfully verifies candidate lane lines. Finally, we introduced a robust lane tracking technique, horizontally adjustable lane repositioning range proposed which predicts the lane location of the present frame using the information of lane location of the previous frame.

4.1 Pre-processing

Pre-processing is an important part of the lane line detection procedure. The purpose of pre-processing is to enhance the feature of interest and reduce noise. We have pre-processed input video frames by gray-scale conversion and noise filtering.

4.1.1 Gray-Scale Conversion

Color image has three channels red, green, and blue that means each pixel has three values. To execute the proposed edge feature-based method, we only need the intensity information. So, the color image has been converted to grayscale image which has only one value of each pixel from 0 to 255. Because of three different colors have three different wavelengths, human eye is most sensitive to green, while the lowest to blue, hence a weighted average method is used for conversion. The gray scale value is calculated by the equation (1) [13],

\[ G = ((0.3 \times R) + (0.59 \times G) + (0.11 \times B)) \] (1)

4.1.2 Noise Filtering

Smoothing or noise filtering is the simplest way to denoise an image. To carry out smoothing operation, Gaussian [14], mean or median [15] filters are used. In [16], to preserve the feature of interest and to remove unwanted clutter the image was filtered by median filter and to enhance the grayscale image, image histogram has been used. To remove different lighting effects, at preprocessing stage adaptive threshold is performed. Adaptive thresholding is performed using Otsu’s algorithm. It is observed that Bilateral filter improved the detection rate 10% compared to gaussian filter [12]. Therefore, in this paper, we used bilateral filter to smooth the image.

Bilateral filter is a nonlinear, edge preserving and noise reducing image filter. There is no need to employ any edge sharpening technique after smoothing because bilateral itself preserves the strong edges besides smoothing. The bilateral filter is the weighted average of neighborhood pixels, which is same as Gaussian convolution. The difference is that the bilateral filter considers the difference in intensity value with the neighbors to preserve edges while smoothing. For a pixel to influence another pixel, it should not only occupy a nearby position but also have a close intensity value to that pixel [17, 18]. In an input image \( I_p \), is the coordinate of centered pixel, \( q \) is the coordinate of the current pixel to be filtered, \( I_p \) and \( I_q \) is the intensity value of pixels \( p \) and \( q \) respectively. The equation of bilateral filter, \( I_{BF} \) can be expressed as follows,

\[ I_{BF} = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\|p - q\|)G_{\sigma_i}(|I_p - I_q|)I_q \] (2)

In Fig 2., the performance of Bilateral filter has been compared to gaussian and median filter. The input images are smoothed by three different filters to remove noise edges. However, to compare the performance of these filters, canny edge detector is applied on the filtered image to detect both noise edges and lane edges inside the smoothed images. In fig. 2(a) the input image contains flare of light and the reflection of light on wet road which severely affected the visibility of right-side lane marking. In fig. 2(b) the input image includes flare of headlight and taillight from cars, the presence of rain drops on the windshield and the reflection of a navigation device on windshield. These issues severely affected the visibility of both left and right lane markings. Figures 2(c), 2(d) are the output images of the gaussian filter and figures 2(e) and 2(f) are the output of the median filter.
Both filters failed to remove the noises mentioned above. In fig. 2(g) and 2(h) it is clearly visible that bilateral filter removes most of the noises along with keeping the lane edges.

Figure 2. Comparison among Gaussian, Median and Bilateral filters, (a), (b) input images, white ellipses indicate the source of noises, (c), (d) Gaussian filtered binary edge images, (e), (f) median filtered binary edge images, red rectangles indicate the noisy lane edge area, (g), (h) bilateral filtered binary edge images, green rectangles indicate almost clear lane edge area

4.2 Feature Extraction

At feature extraction stage, particular lane features such as edge, texture, length, width, or color etc. are identified. In the case, when the illumination conditions drastically change and the view becomes blur and obscured specially in the rainy or snowy weather, it is tough to discriminate the road and the lane by using color or texture feature. Edge-based feature is more robust than color-based features in various illumination condition
and adverse weather condition. Therefore, in this method, we considered the edge feature for lane detection method.

4.2.1 Edge Detection and ROI Selection

There are lots of edge detection techniques like Sobel, Canny, Prewitt, Roberts etc. [15] applied a fuzzy method for lane detection and canny to get a better edge detection. In [18], the performance of canny and Sobel has been compared and experiment shows that the canny is better than Sobel. Canny is a multi-step algorithm. At first, noise is removed by gaussian. Then, the edge gradient and direction are determined. Next, an edge thinning technique named non-maximum suppression has been applied. Finally, the candidate edges are detected and connected by dual-threshold method.

This dual threshold method uses two thresholds \( [T_u, T_l] \) where \( T_u \) denotes upper intensity threshold value and \( T_l \) denotes lower intensity threshold value, to find the edges of interest. The edge pixels above the upper limit are accepted as edges and edge pixels below the threshold are rejected. Pixels in-between upper and lower threshold are considered only if they are connected to pixels of upper threshold. The ratio between the upper and lower threshold is recommended as 3:1.

However, it is challenging to choose the exact intensity threshold range for different varying lighting conditions. Specially it is difficult to detect the edges of colored lane (yellow, blue etc.) line due to the low intensity difference between road surface and lane marking. These colored lanes become almost invisible due to blur view while it’s raining or snowing. Therefore, we proposed an optimized intensity threshold range (OITR) which improves the performance of canny operator to detect lane edges of colored, obscured, and blurred lane marking due to varying lighting conditions e.g., day, night, rainy, snowy etc. We have selected the upper intensity threshold value, \( T_u = 30 \) which is quite low because in low light or blur visibility condition the intensity of lane edge pixels becomes low. We have chosen the lower intensity threshold value, one third of upper threshold that is \( T_l = 10 \).

In fig 3(a) there are two input frames where each scene has following multiple challenges: i) the view is blur due to rainy weather, ii) the left lanes are yellow and iii) wipers obscure the lane view. In figure 3(b) and 3(c) the intensity threshold ranges that have been applied correspondingly \([50,10]\) and \([45,15]\). In both cases the lane edges detected partially. In figure 3(d) the intensity threshold range \([30,10]\) is applied and in this case most of the lane edges detected.

![Figure 3](image-url)

Figure 3. Comparison among three different intensity threshold range to detect yellow colored lane edges, (a) input images, (b) Implementation of intensity threshold range, \([50,10]\), (c) Implementation of intensity threshold range \([45,15]\), red ellipses indicate the failure of edge detection, (d) Implementation of intensity threshold range \([30,10]\), yellow ellipses successfully detected most of the yellow-colored lane edges

We experimentally observed that this is the most optimized intensity threshold range (OITR) for all colored lane lines and for all lighting conditions as it can handle multiple
challenges mentioned above. However, OITR will keep lots of noisy edge pixels along with lane edge pixels. Selecting region of interest can resolve this issue.

It is noticeable that the lane line information lies in the lower half of the image. So, we do not need the whole image to process for lane detection. Most of the videos inside datasets are captured by positioning the camera in a middle place behind the windshield. Road is always located in front of the vehicle because vehicle moves in a forward direction [15]. Therefore, it is recommended to select ROI at the lower side of image [14-16]. We have selected a region of interest using an isosceles trapezoid-shaped mask placed in middle of the horizontal axis and two third of vertical axis excluding the lower third to avoid the car hood edges. Isosceles trapezoid is trapezoid in which the base angles are equal and therefore the left and right-side lengths are also equal. This mask efficiently keeps the lane line edges while removing other noisy edges for different datasets. In Figure 4(d) the output image of region of interest selection has been shown.

4.2.2 Line Position Extraction

Hough transform is a feature extraction method for detecting shapes such as circles, lines, etc. in an image by applying a voting procedure [19]. In our proposed method, this algorithm is used to identify the position of the lines created by the edges. Two points of each line \((x_1, y_1)\) and \((x_2, y_2)\) are the final output of the Hough Transform. Thus, position of all the lines inside the region of interest are being extracted by Hough Transform. Since among these extracted lines only one left and one right line would be the lane lines, these lines are called candidate lane (CL) lines. In figure 4(a), three sample frames has been shown as input. In figure 4(b), 4(c), 4(d) and 4(e) all the corresponding output images for bilateral filtering, canny edge detection, region of interest selection and Hough Transform has been shown respectively.
Figure 4. (a) input frames containing lane marking view affected by cracked road, shiny yellow light effect and snow-covered blurry effect (from left to right); (b) bilateral filtering; (c) OITR based canny edge detection (d) region of interest selection by isosceles trapezoid-shaped mask; (e1) lane line detection by Hough transform

4.3 Lane Detection

To detect the left lane and right lane boundary simultaneously in each frame is called lane detection. In this paper, lane detection task is divided into following steps: a) dividing candidate lane lines (CLL) into left and right-side lines, b) lane verification.

4.3.1 Dividing CLL into left and right side

Hough transform successfully detected all the lines inside region of interest. From figure 4(e) we can see that Hough Transform has detected lots of noisy lines including lane lines. All these lines are called candidate lane (CL) lines as among them lane lines would be detected. Here we have separated candidate left lane (CLL) lines from candidate right lane (CRL) lines using slope-based constraint. Slope, \( m \) is the ratio of vertical change to horizontal change of a line. A line has a positive slope if \( y \) increases along with \( x \) and on the other hand slope is negative if \( y \) decreases along with \( x \) increases. The left lane line always has a negative slope and right lane line always has a positive slope. By using equation (3) the slope is calculated. The slope-based constraint has been set by equation (4). So, lines with slope \( m > 0 \) identified as CRL lines, slope, \( m < 0 \) as CLL lines and slope \( m = 0 \) \& \( m = \infty \) as false positive.

\[
\text{slope, } m = \frac{\Delta y}{\Delta x} = \frac{y_2 - y_1}{x_2 - x_1} \quad (3)
\]

\[
Lane = \begin{cases} 
 \text{Right; } & m > 0 \\
 \text{Left; } & m < 0 \\
 \text{False; } & m = 0 
\end{cases} \quad (4)
\]

4.3.2 Lane Verification

Sometimes false lane detection happens due to presence of other lookalike lines e.g., guardrails, pavement markings, road divider, traffic signs, zebra crossing, car hood, reflection of stuff (mobile, navigation device, video or image capturing device) kept inside the car, vehicle lines, the shadow of trees etc. which have line like structure. Therefore, lane verification is important to perform to filter out these confusing lines. It is noticeable that, from frame to frame, the vertical position of lane lines almost remains unchanged, but the horizontal position (along x-axis) of lane lines gets changed. So, we have set a range of angle formed by lane lines with x axis to verify the horizontal position.
An angle which measures less than 90° is called acute angle. The left lane forms an acute angle (anti-clockwise) with x axis. An angle that measures greater than 90° and less than 180° is called obtuse angle. The left lane forms an obtuse angle (anti-clockwise) with x axis. Therefore, a range of angle is defined for both left and right lane considering these two types of angles. So, the angle between the left lane and x-axis (anti-clockwise) is defined as $\theta_l$ and the angle between the right lane and x-axis (anti-clockwise) is defined as $\theta_r$. According to the definition of acute and obtuse angle, the range of $\theta_l$ should be less than 90 degree and the range of $\theta_r$ should be greater than 90 degree and less than 180 degrees.

In case of left side lane, to minimize the range and to maximize the noisy line removal, we have selected the mid value between 0 to 90° i.e., 45° as borderline. We considered a variable $c$ for the position of camera. So, the lower value is defined by subtracting $c$ from 45° and the upper value is defined by adding $c$ with 45°. Finally, the range of angle ($\theta_l$) of left lane formed with x axis is $\left[\frac{\pi}{4} - c, \frac{\pi}{4} + c\right]$. On the other hand, for right side lane, we have selected the mid value of an obtuse angle range i.e., 135° as borderline. So, the lower value is defined by subtracting $c$ from 135° and the upper value is defined by adding $c$ with 135°. Finally, the range of angle ($\theta_r$) of right lane formed with x axis, is $\left[\frac{3\pi}{4} - c, \frac{3\pi}{4} + c\right]$. We experimentally observed many datasets with a variety of challenging situations and found that above consideration of angle range performs best. In [20], they have determined the angular range for left lane (25, 75) and the angular range for right lane (105, 155) which is wider than our range. As range of angle defined in our method is narrower than their method. Therefore, our method can eliminate more confusing lines than their method. All the angles, $\theta$ is calculated by the equation (5) and the mathematical representation of angle range has been shown in equation (6).

$$\theta = \tan^{-1}m$$  \hspace{1cm} (5)

$$\text{angle} = \begin{cases} \theta_l, & \frac{\pi}{4} - c \leq \theta_l \leq \frac{\pi}{4} + c \\ \theta_r, & \frac{3\pi}{4} - c \leq \theta_r \leq \frac{3\pi}{4} + c \end{cases}$$  \hspace{1cm} (6)

Applying angle-based constraints we get filtered candidate left lane (FCLL) lines and filtered candidate right lane (FCRL) lines. However, this angle-based constraint can’t remove some confusing lines which form angles with x axis within the angle range or which are parallel to lane lines. For example, guard rails, road divider, road curb, bridge railing.
wall edges of road tunnel, other vehicles, shadow of tree or building, pavement marking parallel to lane etc., these objects create confusing lines because of their angle and position. To avoid this type of misdetection we have proposed length-based geometric constraint.

Usually, the camera is mounted on the middle of the car and the left and right lane is located at the immediate front of the car. So, the lane line is visible as the longest line among all other confusing lines. In some cases, the lane line may not be the longest line, but due to the trapezoid shaped ROI, part of other long lines such as lines of guard rail, road divider, road curb, bridge railing, road tunnel wall etc. is eliminated. In the case of dashed lane line, Hough transform turned the dashed short lines into one long line. So, we choose length parameter for final verification. We took certain number of frames and measured the length of different lines created by lane lines, guard rail, vehicle edge, parallel other lane, road crack, tunnel wall edges etc. which form angle with x axis within the angle range and calculated the average length of different type of lines.

In figure 6, we showed a comparison between the length of lane marking line and other lines e.g., guard rails, vehicle edges, road crack edges, tunnel wall edges, arrow pavement, wiper etc. The metrics indicates that length of lane line is longer than any other lines. We observed that more than 95 percent cases lane lines are longer than any other lines in each frame. The longest left line is the true left lane line, and the longest right line is the true right lane line. Using equation 7, the length of each FCLL line has been calculated and using equation 8, the length of each FCRL line has been calculated. The line with maximum length among FCLL lines has been selected as left lane line (equation 9) and the line with maximum length among FCRL lines has been selected as right lane line (equation 10).

\[
\text{length}_{left} (FCLL) = \sqrt{((x_{l_1} - x_{l_2})^2 + (y_{l_1} - y_{l_2})^2)} \\
\text{length}_{right} (FCRL) = \sqrt{((x_{r_1} - x_{r_2})^2 + (y_{r_1} - y_{r_2})^2)} \\
\text{lane}_{left} = \max (\text{length}_{left}) \\
\text{lane}_{right} = \max (\text{length}_{right})
\]

![Figure 6](image)

*Figure 6.* Comparing length of lane lines with other confusing lines e.g., Guardrail, vehicle edges, parallel lanes, road crack edges, tunnel wall edges, arrow pavement, wiper etc.

### 4.4 Lane Tracking

To estimate and predict the position of the lane markings of the next frame using the lane marking position of previous frame is called lane tracking. Lane tracking is implemented to follow the change of lane position. Sometimes challenging conditions such as rain, snow, reflection of road lamp on wet road, overexposed sunlight, or shiny tunnel light wear out the lane marks and the presence of raindrop on windshield and snow on the road might also affect the visibility of road marking severely. Therefore, it is very
important to develop a lane tracking system that can detect lane marking even when they are partially or fully invisible for short period of time. For lane tracking, Kalman filter [21, 22], extended Kalman filter [23], Annealed particle filter [24], and super-particle filter [25] are used. A lane line position \( P \) can be easily defined by two points \((x_1, y_1)\) and \((x_2, y_2)\). Between two consecutive video frames, there will not be much deviation as there is temporal and spatial continuity between frame sequences.

It is noticeable from the video frame sequences that; the vertical position of a lane remains almost same but the horizontal position of a lane changes remarkably. Considering these issues, we have proposed a robust lane tracking technique named horizontally adjustable lane repositioning range (HALRR) of lane. In this method, we have defined an adjustable range along \( x \) axis which will adjust the range with respect to the lane position of previous frame. If the lane line position of previous frame is \( P_p(x_{p1}, y_{p1}, x_{p2}, y_{p2}) \) and the lane line position of next frame is \( P_n(x_{n1}, y_{n1}, x_{n2}, y_{n2}) \), the \( P_n \) would be located either at left side of \( P_p \) or at right side of \( P_p \). As, the position of lane does not change vertically, \( y_{p1} = y_{n1} \) and \( y_{p2} = y_{n2} \).

We observed that the deviation of lane position along \( x \) axis (both at left and right side) is less than 6 percent of the width of the image. If the deviation of the lane position is \( d \), the range of \( x_{n1} \) is \( R_1 \), the range of \( x_{n2} \) is \( R_2 \), \( w \) is the width of the image, then the expression of \( d \), \( R_1 \), \( R_2 \) is as follows:

\[
d = w \times \frac{z}{100}; z<6
\]

\[
R_1 = [(x_{p1} - d), (x_{p1} + d)]
\]

\[
R_2 = [(x_{p2} - d), (x_{p2} + d)]
\]

This technique can effectively increase the lane detection rate. After employing lane tracking the system can detect lane even after changing the lane or in the presence of confusing lines that are parallel to the lane lines or longer than lane lines or while the lane line is fully or partially invisible or unpainted for short period of time. This procedure takes only 0.99ms per frame whereas Kalman filter takes 2.36ms per frame [19]. We have shown how the lane tracking system can adjust while changing lane in figure 7. Here, a lane changing scene has shown by picking six frames from a rainy-day video from frame no. 515 to 677. From figure 7(a) to 7(e) the lane change happens, and the tracking system can keep track of the lane. After shifting to the left lane completely, the tracking method can automatically adjust the lane line in the figure 7(f).

![Figure 7. Automatic lane tracking while lane changing](image)
In figure 8 we have shown some challenging conditions where lane markings are partially or fully eroded, dashed, unpainted or invisible due to some difficulties like, occlusion by wiper, blurred view due to heavy rain, darker road view without streetlamp, overexposed headlight from opposite direction car, shiny effect and flare of light created by streetlamp reflection on the windshield. All these difficulties were handled by our proposed lane tracking method.
5. Results

We performed experiments using the Spyder (python 3.7) environment on an Intel core i5, 2.30GHz CPU equipped with 8 GB RAM. We implemented our proposed method on 33323 frames from DSDLDE [1] dataset, which provides 1080x1920 resolutions at 24
frames/sec. This dataset is captured in USA and Korea in which the videos include different challenging scenarios such as rain, snow, abrupt light change while tunnel entry and exit, illumination variation captured in day and night. Experiments performed on 30 video clips of 594x1056 resolution which consist of more than 33 thousand frames.

Our proposed method has solved lots of highly harsh conditions abrupt illumination change at daytime i.e. darkness due to shadow, tunnel entry, extreme sunlight, white out on tunnel exit etc., abrupt illumination change at nighttime i.e. flare of light from other vehicle’s headlight or taillight, reflection of road light on wet road, tunnel entry, tunnel exit etc., low contrast between lane marking and road surface when the lane is colored or eroded or the color of road light or tunnel light is same as lane color, incorrect lane detection due to presence of lines similar to lane i.e. guardrails, pavement marking, road divider, lines created by road crack, vehicle lines etc., detection interruption due to lane change, blurred view due to presence of heavy rain or snow, occlusion due to presence of wiper, raindrop, reflection of any stuff kept inside the car i.e. mobile, navigation device, video or image capturing device on the windshield.

Figure 9 shows the successful lane marking detection and tracking in different harsh driving conditions. In Fig. 9 (a), (b), (c), inside the tunnel, the lane is almost invisible because the light color and the lane marking color are same. In fig. 9 (d) the light reflection from a device kept inside and the flare of headlight and taillight on the windshield made the view hazy. In fig. 9(e) blur view created by heavy rain and in fig. (f) lanes are invisible due to streetlight reflection in rainy night. In fig. 9(g) and 9(h), heavy snow and dark tunnel entry in a snowy day has been shown respectively. In fig. 9(i) highway in a snowy night has been shown.

![Figure 9. Examples of successful lane marking detection under harsh driving conditions; (a) daytime dark tunnel, (b) nighttime shiny yellow light tunnel, (c) nighttime shiny white light tunnel, (d) inside and outside reflection of light on car windshield, (e) blur view due to heavy rain, (f) reflected streetlight on wet road in rainy night, (g) daytime heavy snow, (h) heavy snow tunnel entry, (i) lane like edges created by nighttime snow](image-url)
All these successful detections verify the robustness of the proposed algorithm. In table 1, we have shown the lane marking detection rate under different time, weather, and challenging conditions.

**Table 1. Detection rates of proposed method in different challenging conditions**

| Time     | Weather | Condition/Challenges                     | number of frames ($f_t$) | incorrectly detected frames ($f_i$) | Detection rate, ($f_t - f_i$)/ $f_t$ (%) | Average detection rate (%) |
|----------|---------|------------------------------------------|-------------------------|-------------------------------------|------------------------------------------|---------------------------|
| Clear    | Day     | Shadow, eroded lane marking, bumpy road  | 4187                    | 72                                  | 97.77                                    | 98.3                      |
|          | Day     | Tunnel                                    | 700                     | 0                                   | 100                                      |                           |
|          | Day     | Cracked road                               | 1122                    | 33                                  | 97.06                                    |                           |
|          | Rainy   | Light rain                                | 1463                    | 20                                  | 98.63                                    |                           |
|          | Rainy   | Heavy rain with traffic and lane change    | 2522                    | 130                                 | 94.85                                    | 96.9                      |
|          | Rainy   | wet road inside tunnel                     | 500                     | 16                                  | 96.80                                    |                           |
|          | Rainy   | Hilly and curved highway                   | 1351                    | 18                                  | 98.67                                    |                           |
| Snowy    | Day     | Light snow, straight road                  | 2308                    | 16                                  | 99.30                                    |                           |
|          | Day     | Heavy snow, eroded lane marking, curved road, tunnel | 2092 | 191 | 90.86 | 95.3 |
| Clear    | Night   | Highway                                   | 2455                    | 10                                  | 99.51                                    |                           |
|          | Night   | White tunnel                               | 900                     | 8                                   | 99.11                                    | 99                        |
|          | Night   | Yellow tunnel                              | 2100                    | 35                                  | 97.61                                    |                           |
|          | Night   | Light rain and clearly visible lane marking| 1463                    | 0                                   | 100                                      |                           |
|          | Night   | Light rain and traffic faded and unpainted lane marking, reflection of road lamp on wet road | 1235 | 6 | 99.51 |
|          | Night   | Flare of headlight and taillight of car from same and opposite direction | 1260 | 35 | 97.22 |
| Rainy    | Night   | Extreme dark (no road lamp) and bumpy road, reflection of taillight, lane change, Reflection of navigation device and reflection of shiny bridge light on windshield, white tunnel | 548 | 16 | 97.08 | 96.5 |
|          | Snowy   | Dark and curvy road                        | 1464                    | 85                                  | 94.2                                     | 94.2                      |
|          | Total and average                          | 33323                   | 880                                  | 97.36%                                 |                           |

So, in the clear daytime the average lane detection rate is 98.07% which includes different conditions like shadow, eroded lane marking, lane occluded by traffic and pavement marking, cracked road, tunnel etc. In the rainy day, the average lane detection rate is 96.85% which includes conditions like heavy rain with traffic jam, lane change, wet road inside tunnel, hilly and curved highway. In the snowy day, the average lane detection rate is 95.29% which also includes different conditions like eroded lane marking, curved road, tunnel etc. In the clear night, the average detection rate is 98.3% which includes different conditions like city road with traffic, white and yellow colored light tunnel etc. In the rainy
night, the average detection rate is 96.47% which includes some hard conditions like, faded and unpainted lane marking, reflection of road lamp on wet road, flare of headlight and taillight of car from same and opposite direction, Extreme dark (no road lamp) and bumpy road, reflection of taillight, lane change, Reflection of navigation device and reflection of shiny bridge light on windshield, white tunnel etc. In snowy night, the average detection rate is 94.19% where the visibility of lane is very poor due to snow fall at night. The overall detection rate in different weather conditions is more than 94% and the average detection rate is 97.36%.

The proposed method is compared with other methods that have been reported recently and have shown relatively better performance. In table 2, the performance of the proposed algorithm is compared with that of the previous works [1, 8, 9, 11, 19] in terms of detection rate and processing time per frame.

Table 2. Result comparison of the proposed method with other methods

| Detection rate (%) | Performance |
|--------------------|-------------|
|                    | Day        | Night      | Processing time(frame/msec) | Platform, CPU, Image resolution |
|                    | Clear      | rain       | Snow        | Clear      | Rain       | Snow        |
| Son [8]            | 93.1       | 89         | -           | 94.3       | 93         | -           | 33          | PC(unknown)   |
| Jung[9]            | 88-98      | -          | -           | 87.7       | -          | -           | 39          | PC, 2.8GHz, 4GB, 640X480 |
| Lee[1]             | 99         | 96.9       | 93          | 98         | 93.6       | 92.2        | 35.4        | ARM A9 @800 MHz, 1920X1080 (PC, 3.4GHz) |
| Marzougui [19]     | 90-93.5    | 90.20      | -           | 96.4       | -          | -           | 21.54       | Core i7 2630QM, 8GB, 640X360 |
| Sultana[11]        | -          | 94.12      | -           | -          | -          | -           | 30.24       | PC, core i5, 2.30GHz, 8GB, 594x1056 |
| Proposed           | 98.3       | **96.9**   | **95.3**    | **99.03**  | **96.5**   | **94.2**    | **29.06**   | PC, core i5, 2.30GHz, 8GB, 594x1056 |

Son [8] utilized the Cb component of YUV color space to detect the yellow lane markings. The binary intensity value of a pixel and the Cb value are combined using OR operation. Both the intensity and the yellow value are below the threshold when the yellow lane markings have eroded or occluded. As the proposed method is edge feature-based method, here the yellow lane marking was detected by applying an optimized intensity threshold range (OITR) in the canny edge detection stage. OITR improve the performance of canny in the case of detecting the edges of colored lane. This reduce the computational time and can detect edges of lane marking those are eroded or occluded by rain or snow. Jung’s [9] lane detection algorithm attempts to find vertical lines and depends on the road width. So, it failed to detect lane when the width of the lane is either increasing or decreasing. Additionally, the spatiotemporal image depends on vehicles’ speed, which will affect the detection results. When both the left and right lane is completely missing, this method can’t detect lane. In proposed method, the detection algorithm is not dependent on either width of the lane or speed of the vehicle and able to detect lane while both lane marking is completely missing because lane tracking was applied.

Lee [1] proposed an adaptive method for detecting colored lane markings using the YUV color model. They added the U-V values and grayscale intensity to boost the values of yellow lane marking. They used Kalman filter to track lane markings. However, this
method fails to detect yellow lane markings while the lane markings become obscured due to the appearance of yellow colored background for nighttime streetlights or for a tunnel with yellow lights. In our method, we proposed OITR in the edge detection stage which improves the performance of canny edge detector to enhance and detect the colored or eroded lane marking edges. Usually Kalman filter takes 2.36 ms to track lane marking at each frame [19]. In our proposed method, we proposed horizontally adjustable lane repositioning range (HALRR) algorithm to track lane markings which takes only 0.99 ms to process per frame. So HALRR is faster than Kalman filter. Sultana [11] proposed a lane detection and tracking method which can overcome different rainy weather challenges in the daytime but it does not work during the lane is being changed. In our method, we addressed and overcome all possible real-life challenges.

The proposed method yields better result than other method, although the video datasets are different. Table 2 shows that the detection rates of the proposed algorithm are outstanding to those of the other works, and more challenges related to light, road and weather conditions are explored in this work than the other state of the art works. Especially, video clips of roads at night with snow did not get good result in other works. The processing time is even less than any other algorithms. The results showed that the proposed method provides much faster computation speed than the other algorithms with powerful and faster PC configuration. However, the proposed algorithm failed to detect lane markings when snow creates lane like lines and when tunnel edges create lane like lines as shown in Figure 10. There exist some cases of misdetection of lane marking. It could be improved by adding more parameters in verification stage.

![Figure 10. Example of failed lane marking detection, red line indicates missed lane marking](image)

6. Conclusion

In this paper, we have addressed all kind of real-life environmental challenges to detect road lane markings and categorized them into four types: first, abrupt illumination changes due to change of time, weather, road etc., second, lane markings get obscured partially or fully when the lane markings are colored, eroded or occluded, third, blurred view created by adverse weather(rain/snow), fourth, incorrect lane detection due to presence of lane like confusing lines. We have proposed a robust method to detect road lane marking under all these challenging real-life environmental conditions. An optimized intensity threshold range (OITR) is proposed in edge detection stage, which improves the performance of canny. A robust lane verification technique, angle and length based geometric constraint (ALGC) algorithm is proposed, to verify the characteristics of lane marking. Finally, a novel lane tracking method, horizontally adjustable lane repositioning range (HALRR) algorithm is introduced to keep track of the lane position when either left or right or both lane markings are partially or fully invisible due to erosion or occlusion for short period of time. The proposed algorithm shows better performance for road conditions with noisy components than others. The proposed algorithm is verified by 30 video clips consist of various challenges. The computation time satisfies the real-time operation. The detection rate and the computation time for the proposed method are
compared with those of other works and it is manifested that the proposed method is superior to them.

**Author Contributions:** Conceptualization, S.S., and M.R.I.; methodology, S.S.; software, S.S.; validation, S.S.; formal analysis, S.S., and S.A.; investigation, S.S., and B.A.; resources, S.S. and B.A.; data curation, S.S.; writing—original draft preparation, S.S.; writing—review and editing, B.A., M.F. and M.R.I.; visualization, S.S.; supervision, B.A.; project administration, B.A.; funding acquisition, M.P. and M.R.I. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** Data “DSDLDE v.0.9. Video Clips for Lane Marking Detection, [online]” obtained from [1] are available at https://drive.google.com/file/d/1315RV7ciL3nRvU5SCXM_4meR2MyI/view?usp=sharing.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**

1. C. Lee and J.-H. Moon, "Robust lane detection and tracking for real-time applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 12, pp. 4043-4048, 2018.
2. J. E. Gayko, "Lane departure and lane keeping," *Handbook of intelligent vehicles*, vol. 1, pp. 689-708, 2012.
3. C. Visvikis, T. Smith, M. Pitcher, and R. Smith, "Study on lane departure warning and lane change assist systems," *Transport Research Laboratory Project Rpt PPR*, vol. 374, 2008.
4. M. Aly, "Real time detection of lane markers in urban streets," in *2008 IEEE Intelligent Vehicles Symposium*, 2008: IEEE, pp. 7-12.
5. B. Beniligiray, C. Topal, and C. Akinlar, "Video-based lane detection using a fast vanishing point estimation method," in *2012 IEEE International Symposium on Multimedia*, 2012, pp. 348-351.
6. H. Yoo, U. Yang, and K. Sohn, "Gradient-enhancing conversion for illumination-robust lane detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1083-1094, 2013.
7. G. Liu, S. Li, and W. Liu, "Lane detection algorithm based on local feature extraction," in *2013 Chinese Automation Congress*, 2013: IEEE, pp. 59-64.
8. J. Son, H. Yoo, S. Kim, and K. Sohn, "Real-time illumination invariant lane detection for lane departure warning system," *Expert Systems with Applications*, vol. 42, no. 4, pp. 1816-1824, 2015.
9. S. Jung, J. Youn, and S. Sull, "Efficient lane detection based on spatiotemporal images," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 1, pp. 289-295, 2015.
10. A. Gupta and A. Choudhary, "A framework for camera-based real-time lane and road surface marking detection and recognition," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 4, pp. 476-485, 2018.
11. S. Sultana and B. Ahmed, "Lane Detection and Tracking under rainy weather challenges," in *2021 IEEE Region 10 Symposium (TENSYMP)*, 2021: IEEE, pp. 1-6.
12. S. Sultana and B. Ahmed, "Robust Nighttime Road Lane Line Detection using Bilateral Filter and SAGC under Challenging Conditions," in *2021 IEEE 13th International Conference on Computer Research and Development (ICCRD)*, 2021: IEEE, pp. 137-143.
13. A. Borkar, M. Hayes, and M. T. Smith, "A novel lane detection system with efficient ground truth generation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 365-374, 2011.
14. P.-Y. Hsiao, C.-W. Yeh, S.-S. Huang, and L.-C. Fu, "A portable vision-based real-time lane departure warning system: day and night," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 2089-2094, 2008.
15. J.-G. Wang, C.-J. Lin, and S.-M. Chen, "Applying fuzzy method to vision-based lane detection and departure warning system," *Expert systems with applications*, vol. 37, no. 1, pp. 113-126, 2010.
16. H. Xu and H. Li, "Study on a robust approach of lane departure warning algorithm," in *2010 2nd International Conference on Signal Processing Systems*, 2010: IEEE, pp. V2-201-V2-204.
17. S. Paris, P. Kornprobst, J. Tumblin, and F. Durand, *Bilateral filtering: Theory and applications*. Now Publishers Inc, 2009.
18. S. Vijayarani and M. Vinupriya, "Performance analysis of canny and sobel edge detection algorithms in image mining," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 1, no. 8, pp. 1760-1767, 2013.
19. M. Marzougui, A. Alasiry, Y. Kortli, and J. Baili, "A Lane Tracking Method Based on Progressive Probabilistic Hough Transform," *IEEE Access*, vol. 8, pp. 84893-84905, 2020.
20. T. M. Hoang, H. G. Hong, H. Vokhidov, and K. R. Park, "Road lane detection by discriminating dashed and solid road lanes using a visible light camera sensor," *Sensors*, vol. 16, no. 8, p. 1313, 2016.
21. S. K. Gehrig, A. Gern, S. Heinrich, and B. Woltermann, "Lane recognition on poorly structured roads-the bots dot problem in California," in *Proceedings. The IEEE 5th International Conference on Intelligent Transportation Systems*, 2002: IEEE, pp. 67-71.
22. A. Borkar, M. Hayes, and M. T. Smith, "Robust lane detection and tracking with ransac and kalman filter," in *2009 16th IEEE International Conference on Image Processing (ICIP)*, 2009: IEEE, pp. 3261-3264.
23. B. Özcan, P. Boyraz, and C. B. Yiğit, “A monoSLAM approach to lane departure warning system,” in 2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2014: IEEE, pp. 640-645.

24. H. Zhao, Z. Teng, H.-H. Kim, and D.-J. Kang, “Annealed particle filter algorithm used for lane detection and tracking,” Journal of Automation and Control Engineering, vol. 1, no. 1, pp. 31-35, 2013.

25. B.-S. Shin, J. Tao, and R. Klette, “A superparticle filter for lane detection,” Pattern Recognition, vol. 48, no. 11, pp. 3333-3345, 2015.