



An efficient approach for highway lane detection based on the Hough transform and Kalman filter

Sunil Kumar¹ · Manisha Jailia¹ · Sudeep Varshney²

Received: 1 February 2022 / Accepted: 9 July 2022 / Published online: 28 July 2022
© Springer Nature Switzerland AG 2022

Abstract

The advances in lane detection technologies and computer vision enabled the evolution of lane-keeping systems, driver assistance and lane departure warning in traffic management for road safety. However, it is very challenging to identify and track the lane lines due to improper marking of lane lines and blind turns on the road. The present work proposes effective and efficient vision-based real-time lane markings and tracking lane detection methods for straight and curved lane lines. That can adapt to various environmental conditions. Further, the Hough transform optimization is performed to identify lane lines accurately, and the Kalman filter is employed to track lane lines detected in the ROI by the Sobel operator. The proposed approaches show their significance by achieving real-time response and high accuracy for a vehicle in lane change assistant system on highways. While comparing, the proposed methods show better results in terms of detection rate and processing time for straight lanes and detection accuracy, precision, recall and F1-Score for the curved lanes. The result of processing time and accuracy rate for straight lane detection is 16.7 fps, 96.3%, respectively, and the accuracy, precision, recall and F1-scores for curved lane detection are 97.74%, 98.15%, 97.35% and 97.75% in videos, respectively.

Keywords Canny edge detector · Hough transform · Perspective transform · Sliding window search · Kalman filter

Abbreviations

RGB	Red Green Blue
HSV	Hue Saturation Value
HSL	Hue, Saturation and Lightness
$dst(x, y)$	The pixel coordinate for the transformed image
$src(x, y)$	The pixels in the input image
$P_{k k-1}$	The predicted state covariance
F_{k-1}	State transition matrix
H_k	Measurement transition matrix
G_x	The gain of the Kalman filter
\hat{x}_k^-	Posteriori estimate state
P_k	Covariance matrix of a posteriori estimate error

Introduction

Due to the high population growth rate, the number of vehicles on the road is increasing rapidly. Subsequently, the increment in the number of vehicles increases road accidents. As per the survey, in India, due to road accidents, a person dies every four minutes [1]. Another survey shows that according to the World Health Organization (WHO), every year, 1.25 million people are affected by road traffic crashes [2]. These road accidents are caused by changing lanes to avoid obstacles [3]. In today's world, the passenger's safety and comfort are among the most significant concerns. The fatalities on the road have increased as the number of vehicles also increased every year. Therefore, lane detection is an intriguing and crucial study field for autonomous vehicles. Furthermore, inattention to the road's conditions brought on by driving error or visual blockage is a major factor in many accidents. Consequently, ADAS has been considered one of the significant technologies to minimize the number of accidents, and the lane departure warning systems (LDWS) and the lane-keeping assistance system (LKAS) have relied heavily on lane detection and tracking for advanced driver assistance systems (ADAS) [4–6].

✉ Sunil Kumar
sunildangik107@gmail.com

Manisha Jailia
manishajailia@yahoo.co.in

Sudeep Varshney
sudeep149@gmail.com

¹ Department of Computer Science, Banasthali Vidyapith, Banasthali, Rajasthan, India

² Department of Computer Science & Engineering, School of Engineering & Technology, Sharda University, Greater Noida, India

In recent years, advanced driver assistance systems have received considerable attention. Lane detection is becoming more and more crucial to the advanced driver assistance system (ADAS) for traffic safety. ADAS is one of the indispensable parts of lane detection to provide steady lane-keeping performance by analyzing the images captured by the vehicle's front camera [7, 8]. The development of strategies and procedures for increasing driving safety and reducing traffic accidents has been the subject of extensive research by both individuals and institutions. Road awareness and lane marking detection are two of these methods that are extremely important in assisting drivers in avoiding mistakes. Therefore, straight lane and curved lane detection has been significantly crucial from a safety perspective. Although past lane detection research has addressed both learning-based techniques and handcrafted feature-based approaches. The handcrafted feature extraction is used to carry out several lane line detection techniques used in lane-keeping assistance systems (LKAS) and lane-following assistance systems (LFAS). These techniques are based on a manually designed feature extraction system that supports LKAS and LFAS for extracting the colored features of lanes.

With regard to the aforementioned common ADAS sub-systems, these vision-based techniques are widely used in this research field. The earlier studies that investigate vision-based lane detection systems address four key steps: pre-processing, feature detection, fitting, and tracking. [9–12]. The pre-processing stage is necessary to reduce image noise before moving on to the next steps. It is possible to extract features that identify the lane line edges using the Canny edge detector approach. The Canny edge detector [13] has been widely used because it can recover the edges of the input image without causing interference. Then, the line is determined using the Hough transform [14, 15] rather than lane detection based on the clearly identified edge. However, this takes a lot of time to process. Therefore, the improved Hough transform technique has been incorporated for substantially faster processing times [16]. Despite being relatively reliable in simple road scenarios, handcrafted feature-based algorithms have several issues detecting curving lanes due to the poor lane marking and illumination conditions [17–19]. The line model [20], spline model [21], linear-parabolic model [22], quadratic curve model [23], and hyperbola-pair model [24] are only a few of the methods that have been presented for lane fitting. However, it is challenging to identify geometric information due to camera motion and rapid changes in the road environment, particularly when the lanes are blurry or the road image is noisy. For lane tracking, the Kalman filter is used to track the subsequent region of interest (ROIs) while tracking the identified lane in order to increase processing speed and robustness. On the other hand, the learning-based methods have significantly enhanced the performance of lane detection compared to

handcrafted feature-based methods [25]. However, these techniques have their limits when considering contextual information broadly.

This paper proposes an effective and efficient lane detection method to deal with the above problems: straight and curved lane detection. The main objective of this research is to further improve the accuracy of lane detection by employing the optimized Hough transform and the Kalman filter. The proposed straight lane detection method obtains results in three steps: First, it performs pre-processing of an input image to reduce the noise by applying the Gaussian filter, and then, the edges of lane lines are extracted from the input image by using the Canny edge detector. Also, the required region of interest (ROI) is calculated from the detected edges in the input image. Next, lane lines are obtained from these detected edges by applying the Hough transform, and finally, it uses the optimization of the Hough transform to enhance the accuracy of lane detection.

The curved lane detection method is proposed using the Hough transform, the perspective transform and the Sobel thresholding to detect the edges of lane lines. In addition, the Kalman filter is applied to improve the robustness of the proposed work of curved lane detection. This proposed algorithm comprises the various stages such as pre-processing of image [26, 27], conversion of RGB (Red, Green and Blue) to HSV (Hue, Saturation, Value) and HSL (Hue, Saturation, Light), the perspective transformation applied for converting the 3D image into warped image and the Sobel edge detector algorithm is used to detect edges of the lanes. Furthermore, lane tracking is done by the Kalman filter algorithm [28] to improve the lane detection accuracy rate. The novelty of the proposed work lies in the fact that optimization of the Hough transforms and the lane tracking based on the extended Kalman filter have not been incorporated for the straight and curved lane detection to the best of our knowledge. Apart from this, this work brings a holistic approach for lane detection under different scenarios. The major contribution of this work is simulation and detailed comprehensive discussion on the lane detection technologies. A summary of contribution of this work can be enumerated as below.

- A straight lane detection approach for identifying road boundaries is provided through the use of the Gaussian filter to reduce the input image noise in the pre-processing stage and Canny edge detector for extracting the edges of the lane lines in the input image. Finally, it applied the optimized Hough transform technique to identify both borders of each lane for straight road lane lines. The advantage is that the pre-processing time is significantly reduced, and the performance of the lane detection algorithm is increased.
- An advanced lane detection algorithm is proposed to address the shortcomings of the straight lane detection

method based on the Hough and perspective transform. In this approach, Sobel operator is used to extract the lane lines of the road. And then, Kalman filter is employed to track the road lane lines.

- The performance of proposed method is demonstrated on different sets of images from TuSimple datasets and comparisons with existing lane detection algorithms such as RANSAC + HSV, Gaussian filter + Hough transform, LD + Inverse Perspective, MSER Algorithm, HSV-ROI + Hough, Improved LD + Hough, which obtains the accuracy of straight lane and curved lanes are 96.3% and 97.74%, respectively.

The remainder of the paper is organized as follows. Section 2 discusses the existing lane detection approach used for detecting lane lines. Section 3 describes the proposed methodology of straight and curved lane detection. Section 4 presents the results and discussion for proposed method. Finally, Sect. 5 concludes the research work and enumerates future research directions with the aim to guide future work of lane detection under congested road network.

Related work

In recent years, various lane detection algorithms have been proposed. These algorithms are used to assist the self-drivers in driving the vehicle within a lane. If the vehicle is reaching too close to lane boundaries, then the lane detection system alerts the driver's for self-driving systems to steer the vehicle within lane boundaries. Lane detection is a hot spot in self-driving vehicle systems with rapid development in deep learning technology. Recently, various researchers addressed the lane detection problem and proposed algorithms to solve the problem of lane detection. However, these proposed algorithms are not reliable, and computational time is very high. In addition, straight lane detection and curved lane detection are affected by the structure of the roads. The various recent advances related to lane detection systems are explored in the subsection below.

In this section, the existing literature has been discussed and analyzed to detect straight lane lines and curved lane lines. Xing et al. [32] introduced an algorithm to compare the representative studies for lane detection systems and highlighted the limitation of the lane detection systems. The intelligent vehicle and vehicles' traffic scenes understanding are turning out to be progressively significant, particularly for autonomous vehicles. With the advancement in deep learning technology, various algorithms have been proposed based on deep convolutional networks to find the solution of lane detection.

Ko et al. [33] introduced a method based on deep learning for multilane detection to reduce error points. In addition,

the architecture of multilane detection can improve network performance. Liang et al. [34] proposed a LineNet model to detect the lane using the Line Prediction (LP) layer and Zoom module. This proposed model is helpful to understand the position, direction and distance for driving scenarios. Yang et al. [35] proposed a model using recursive neural networks and long short-term memory to improve the model's performance by studying the spatial information in complex scenarios. After analyzing the lane lines structure to support the lane detection systems, Zou et al. [36] proposed a model that combines a recurrent neural network (RNN) and a convolutional neural network (CNN). Datta et al. [4] proposed a straight lane detection technique model to detect lanes. This mainly consists of image pre-processing steps first. It converts the input image into grayscale, detecting the edges using a Canny edge detector, performing a bitwise logical operation on the input image. Finally, the Hough transform detects lines after masking with the region of interest (ROI). Jiang L et al. [37] proposed a method by combining the Hough transform and R-least square method to track the lane using the Kalman filter on straight lane marking.

Dubey et al. [38] presented a model for lane detection in environmental conditions using a Gaussian filter and Hough transform. Ma C et al. [39] proposed a framework to extract the boundary points. It converts RGB image into color space image and then finds the region of interest, but this model fails to detect lane lines for the poor-quality image. Srivastava et al. [40] have presented an improved Canny edge detection algorithm to detect the images' edges by reducing the noises using Gaussian filters. Many challenges have been faced with detecting the accurate lane lines in road lines due to curved lane lines, issues caused by vehicle occlusion, shadow, varying illumination conditions, and improper marking of road lanes [41, 42]. Haselhoff et al. [43] developed a model to detect the road's left lane and right lane with the help of two-dimensional linear filters. This filter could eliminate interference noise during the detection process. Piao, et al. [44] have proposed a model in which Hough transform combines with maximum stable extreme area (MSER) approach for identifying the lane lines by using the Kalman filter to achieve higher accuracy to track continuous lanes of the road. However, when it comes to poor light conditions, it does not achieve accuracy. Jiang, et al. [37] have developed a model to detect straight lane marking by combining the Hough transform and R-least square method using the Kalman filter.

In the proposed work, the Hough transform is used to identify the straight lane lines in the near-field of view, but it fails to identify the curved lane lines on the road. Kalman filter [45] was used to detect curved lane lines in the curved lane proposed work. In this proposed work, image pre-processing is done for removing distortion and noise of the image. Then, original RGB image is converted to HSV (Hue,

Saturation and Value) and HSL (Hue, Saturation and Lightness) color space to extract and identify lanes using color segmentation and strong gradients to filter out the pixel in lane lines of the road. In the next step, applying the perspective transform [46] to determine the exact angle and direction in the road lane line, detecting lane lines using the Sobel operator, which is mainly used to detect the edges. Then, finally, the Kalman filter was introduced for curved lane detection and tracking of lane lines using a parabola model in the far-field of view [47].

Lee et al. [48] proposed a method for lane detection and road recognition based on regression branches of bounding boxes and Hu moments. This method predicts the driving path based on three tasks convolutional neural network (3TCNN). Lee, et al. [49] introduced a deep learning-based method to achieve high performance. However, it is not suitable for lane detection in challenging conditions.

Dubey, [38] introduced a method specially to detect the curvy lanes by using the Hough transform. In this work, Hough transform lines are taken into consideration for curve. Then, the estimation of the weighted centroid of the Hough lines is performed with the help of mean value theorem formula. This method differentiates the two lanes by calculating the slope of tangent. Curved lanes are obtained through slope of tangent, and it can be calculated by Eqs. 1 and 2. But this method is not well suited for adverse conditions as it is based on color thresholding technique.

$$f'(c) = \frac{f(b) - f(a)}{b - a} \Rightarrow \lim_{h \rightarrow 0} \frac{f(x + h) - f(x)}{h} \quad (1)$$

Muthalagu, et al. [50] proposed the minimalistic approach for detection of straight and curved lane lines. For straight lane detection, this approach used the pre-processing along with the color thresholding for enhancement of the image. Then, it applied the Canny edge detector algorithm to detect the edges in the image. After finding the edges in the images, lines can be obtained from these detected edges by applying the Hough transform. The Hough transform is used to convert the extracted lines into parameter space, and finally, it applied the linear regression for calculating the slope of lane lines.

For advanced lane detection, this method used the image pre-processing, the Sobel thresholding, perspective transform to detect the curved lane lines. But these methods are mainly based on the pre-processing step. The computational time for pre-processing step is very high, and it impacts the detection rate. It fails to detect the steep curve on the foreground due to the limitation of this work. Li, et al. [51] proposed a method to extract the edge feature detection using HSV color segmentation and ROI selection in the

pre-processing stage. But it fails to detect the lanes when lines converge each other due to not using the perspective transform. Zheng, et al. [52–55] introduced an improved lane detection method using the Hough transform. In this method, lane lines are identified in the Hough space. However, if any occlusion comes suddenly, it cannot predict the next frame as it is using only the Hough transform. Kim, K. et al. [56] proposed an algorithm to identify the lane lines using RANSAC and color segmentation. But it achieves the lane detection accuracy rate up to 88.66%. Mammeri et al. [57] introduced an algorithm based on a fusion of maximally stable extremal regions (MSER) and progressive probabilistic Hough transform (PPHT) for lane detection and tracking. This method obtained a 92.7% average lane detection rate by tracking lane lines using two Kalman filters. The texture information created by the MSER method is fed into the Hough transform in the detection step, increasing the detection's processing time. Huang et al. [58] proposed a technique for lane detection and tracking lane lines using inverse perspective and the Kalman filter to obtain an accuracy rate of 93.45%. In addition to improving lane detection accuracy, we applied Hough transform optimization, which improves the quality of tracking lane detected lines using the Hough transform and the Canny edge detector for straight lane detection. Similarly, the Hough transform and perspective transform are used to extract lane features, then the Sobel detector is used to extract lane lines, and finally, the Kalman filter is used to achieve lane tracking for the curved lane lines.

Proposed methods

In this paper, an effective lane detection algorithm is proposed for straight lane detection and curved lane detection. The proposed method for straight lane detection is discussed in Sect. 3.1 using the Canny edge detector and the Hough transform technique. The proposed curved lane detection method is discussed in Sect. 3.2 which uses the perspective transform and the Kalman filter to detect and track the lane lines.

Straight lane detection system

In the proposed work, straight lane lines have been detected using Hough transform along with the Canny edge detector. The proposed lane detection method is used to improve the detection rate. The detection rate can be described as the number of frames detected per unit time. The proposed work is divided into two parts first is the image pre-processing to

convert image into grayscale using Gaussian filter. In image processing, Gaussian filter is used to remove the unwanted image noise. Gaussian filter is more powerful and effective for reducing the image noise [56]. As images are the collection of pixels, an image consisting of three-channel with RGB channel, in Red, Green and Blue Channel, each pixel represents three intensity values, while a grayscale image consists of only one channel. The range of each pixel is 0–255 in grayscale image with one intensity value. The grayscale image is processed faster than colored image and second one is the lane detection approach using Canny edge detector operator [59] and Hough transform. In image pre-processing, Gaussian filter is applied to generate the noiseless image for higher accuracy of lane detection. As the Canny edge detector is used in our proposed work for straight lane detection. Therefore, Gaussian filter is used to reduce the noise in the image before detecting the edges. Thus, it improves the results while detecting the edges [50]. Detecting the edges in the entire image using the Canny edge detector, but it clearly detects the edge boundary corresponds to the sharpest changes in intensity gradients that

exceed the maximum value of threshold are identified as bright pixels, the adjacent pixels are identified where the brightness changes rapidly in the image. The small changes and brightness are not traced at all as they are below the lower threshold. Therefore, it is essential to select the ROI (region of interest) [60] of the image using matplotlib function called `region_of_interest` by setting the parameter values height, polygons and masking, then performing the Bit-wise AND operation on ROI image and Canny edge detected image for better result of specific area of lane lines. Hough transform [37] is applied to detect the straight lane from a series of points for tracking and identifying the lane lines. Finally, it performs the optimization of Hough transform by merging these detected lines by Hough transform with color image to display the lines on the lanes. These lines are displayed on lanes with further optimization. Instead of having multiple lines, averaging out their slope and y-intercept into a single line that traces out the lanes. Figure 1 shows the working module of the proposed work for straight lane lines in the road.

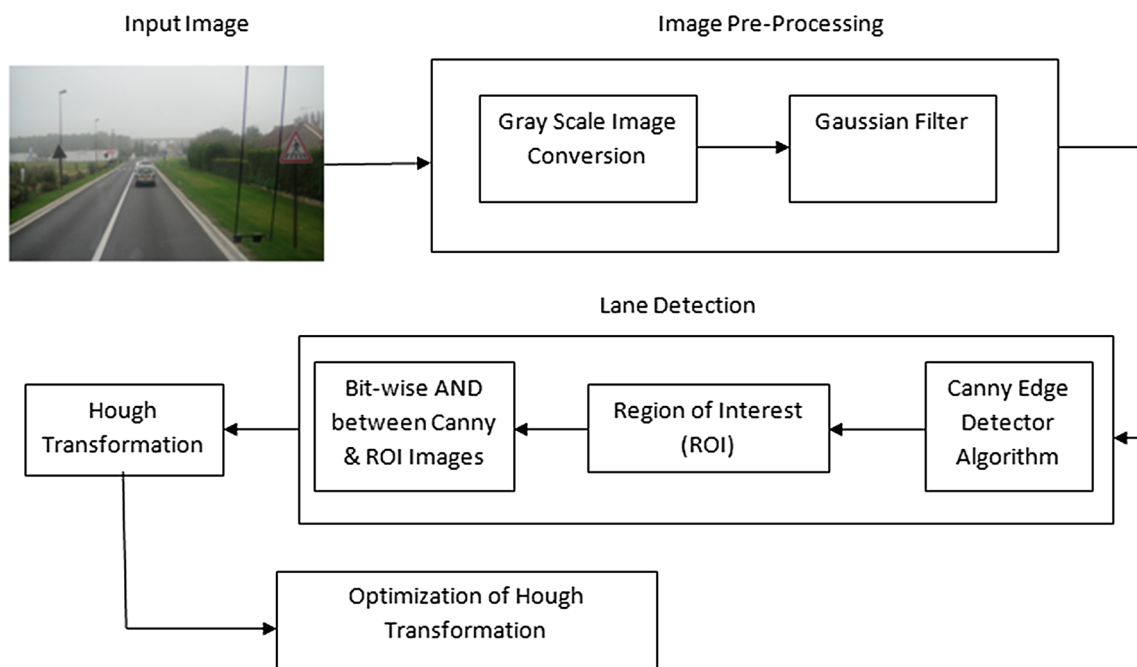


Fig. 1 Framework of proposed straight lane detection

Algorithm-1: Pseudo code for proposed straight lane detection

Input: Considering set of road images $\{x_1, x_2 \dots x_{n-1}x_n\}$ where x is the image and n is number of input images

Output: Detection of road lanes to steer vehicle in right direction

1. Reading Input Image and converting into gray scale image
 2. Applying Gaussian filter to remove the noise from the grayscale image// Producing the blurry image.
 3. Obtain the edge detected image from the blurry image using canny edge detector
 4. Extracting Region of Interest Image by applying matplotlib function.
 5. Performing Bit-wise AND operation on the image received from step 3 and step 4.
 6. Initialize two variables of a line in the image space which can be defined by the Cartesian coordinate system $y = mx + c$ and Polar coordinate system for $r = x \cdot \cos\theta + y \cdot \sin\theta$ and $y = -\frac{\cos\theta}{\sin\theta} \cdot x + \frac{r}{\sin\theta}$ // to finding the lanes of straight lines of road lanes
 7. Detect lane lines using Hough transform
 - (i) Initialize HT (r, θ)
 - (ii) For each edge point in x, y image

for $\theta = 0$ to 180

$r = x \cos\theta - y \sin\theta$

$HT[r, \theta] = HT[r, \theta] + 1$
 - (iii) Calculate the value of (r, θ) where HT [r, θ] is maximum
 - (iv) Line after detection can be represented by the equation $r = x \cos\theta - y \sin\theta$
 8. Optimization of detected lane lines
-

Lane detection

The objective of lane detection is to identify the boundaries of road lines to avoid the vehicle crash and also to guide the driver in the right direction. Detection of lane lines on the road means detecting the image gradients in the form of pixel intensities. To detect sharp changes in large gradient, Canny edge detection algorithm can be applied to detect the true edges for better result in terms of its better performance [59]. Changes in brightness over a series of pixels are the gradients. Steep changes represent strong gradient, and a shallow change represents small gradients. The Canny edge detection algorithm is implemented in four stages. First, it converts the colored input image into grayscale image to increase the contrast of the colors. So that it can detect changes in pixel intensity and then applies the Gaussian filter for reducing the grayscale image noise because gradients are sensitive in Canny algorithm. In the next stage, intensity gradient will be calculated along with the x-axis and y-axis to find out the true edges in the image where horizontal and vertical edges can be identified. Strong pixels can be achieved after non-maximal suppression to get rid of spurious response for detecting the edges and determining the

potential edges by applying threshold and tracking the edges by hysteresis. Figure 8 shows the gradient image in which edge boundary is traced corresponds to the sharp changes in intensity gradients. Gradient is traced when it exceeds the value of high threshold, and such pixels are known as bright pixels of the image. These pixels are the result of rapidly changing brightness identifying nearby pixels in the image.

Region of interest and bit-wise AND of images

Next step is to specify the region of interest by discarding the extra out-side lines of the image received from previous step to get the polygon. Each pixel value is 0 in the black surrounding region, while all the pixel intensity values are 1 in the white polygon region. Straight lane line can be obtained by performing the Bit-wise AND operation between Canny edges detected image and region of interest image. Region of interest reduces the image redundancy and making it more convenient to extract the required lane lines [62]. Required area can be selected from the image to increase the accuracy and speedup the processing time. Figure 9 shows the region of interest image, and Fig. 11 shows the result of Bit-wise AND operation between Canny and ROI image.

Straight lane detection with Hough transform technique

After isolating the region of interest and masking of the image, Hough transform technique is used to identify the lane lines. Hough transform can be used to detect the straight lines, but it also detects incorrect lines [63]. Mostly roads are planned to be straight to drive vehicle easily and steer vehicle for maintaining the constant speed of the vehicle. Assume that there is a line in the road image which is to be detected. In an image, a line can be represented with two coordinate systems: first is Cartesian coordinate system, and second is polar coordinate system. By using Cartesian

coordinate system, straight lines can be represented by expression $y = mx + c$ where m and c are the coordinate pair that represents slope and intercept to which a line is associated. Figures 2 and 3 show the parameter for the Cartesian coordinate system and the polar coordinate system.

In polar coordinate system, a line can be represented by the equation $r = x \cdot \cos \theta + y \cdot \sin \theta$ and $y = -\frac{\cos \theta}{\sin \theta} \cdot x + \frac{r}{\sin \theta}$.

Curved lane detection

In this section, the proposed curved lane detection algorithm for curvy road lane detection has been presented.

Algorithm-2: Pseudo code for proposed curved lane detection

Input: Considering set of road images $\{x_1, x_2, \dots, x_{n-1}, x_n\}$ where x is the image and n is number of input images

Output: pipeline of lane line detection after processing each frame.

1. Calculate $x_{corrected}$ and $y_{corrected}$ radial distortion due to curvature of lens in x, y chessboard image. // calculation of distortion coefficient and camera calibration matrix
2. Removing distortion by applying a correction to the original image.
3. Converting RGB image to HSV and HSL image for color thresholding. // creating a binary image using color transform.
4. Obtaining the polygons on road image by applying Hough transform.
5. Applying perspective transform to rectify binary image and the sobel thresholding to detect road lanes.
6. Track lane lines in real time by Extend Kalman Filter
 - (i) Initialization of variables
 $x_0^- = E [x_0]$ and $P_0 = E [(x_0 - x_0^-)(x_0 - x_0^-)^T]$, Where x_0^- is the initial value of parameter and P_0 is the initial value of co - variance.
 - (ii) Prediction of state and error covariance matrix by Eq. (20).
 $x_{k|k-1}^- = f (x_{k-1}^-)$
 - (iii) Calculate the Kalman Gain by using the Eq. (21).
 - (iv) State Estimation by Eq. (22) where x_k^- denotes the posteriori estimate state
 - (v) Matrix of estimation error covariance by Eq. (24)
7. Apply warp perspective transforms and sliding window search.
8. Calculate the radius of curvature

$$\text{Radius curvature} = \frac{[1+(dy/dx)^2]^{3/2}}{|d^2y/dx^2|}$$

9. Display the result of detection of lane lines on the road.
-

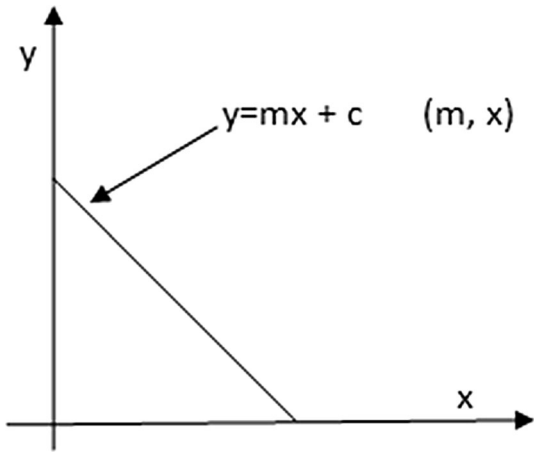


Fig. 2 Cartesian coordinate system parameter

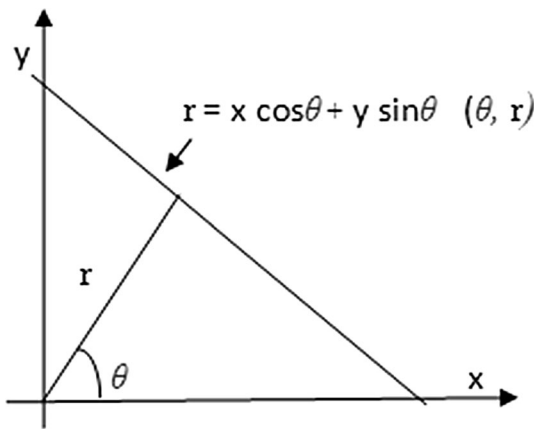


Fig. 3 Polar coordinate system parameter

Camera calibration

Capturing the data from the camera is a collection of images which results in a video. Due to this process, the images may contain distortion [64], which can change the shape and size of the original image. Distortion and occlusions may lead inconsistency in an image. Such distortion and changes in the image affect the accuracy in lane detection. The TuSimple Datasets contain a total of 6408 images of resolution 1280 × 720. Out of 6408 images, the dataset consists of 3626 trained images. Out of these trained images, 358 are for validation. The proposed approach has been tested for 04 different images. The process of camera calibration is set up for capturing the image from videos to overcome the problem of distortions. The camera calibration is applied to images obtained from the video by using the distortion matrix and distortion coefficient. First convert the distorted image to an undistorted image which is

an essential step to improve the accuracy of the lane. Figure 16a, c, e, g, I, k, m, o, q represents the original image, and Fig. 16b, d, f, h, j, l, n, p, r represents the image after removing distortion. Primarily distortion is categorized into radial and tangential distortions. Mathematically, radial distortion can be defined by Eq. (1).

$$\begin{cases} a' = a(1 + m_1r^2 + m_2r^4 + m_3r^6) \\ b' = b(1 + m_1r^2 + m_2r^4 + m_3r^6) \end{cases} \quad (2)$$

where $r^2 = x^2 + y^2$, (a' , b') are the image pixels coordinates.

Mathematically, tangential distortion can be defined by Eq. (2)

$$\begin{cases} a' = a + [2l_1b + l_2(r^2 + 2a^2)] \\ b' = b + [l_1(r^2 + 2b^2) + 2l_2a] \end{cases} \quad (3)$$

where m_1, m_2, m_3, l_1 and l_2 are distortion coefficients.

Distorted image can be corrected by the calibration of camera and taking the distortion coefficients into consideration.

Image pre-processing with color segmentation

Preprocessing is one of the main features of image processing which plays a significant role in lane detection. Isolation of pixels of a particular area can be extracted using gradient and color segmentation. Pixel gradients and color segmentation can be used to mark the left and right road lane lines. Generally, marking of road lines is to be done in yellow and white color [50]. In the RGB color space, in order to detect lane lines blue channel produces worst result as compared to red color. Red color produces better result in identifying the lane lines. In the HSV color space, result of Hue channel is noisy, but saturation color of HLS results in strong pixels. For value channel in HSV, grayscale image result is much better than lightness channel of HLS. To improve the accuracy of lane detection algorithm, color conversion from RGB to HSV and RGB to HLS, various different techniques have been implemented [63].

RGB to HSV conversion model using color space can be represented as.

$$R' = R/255 \quad G' = G/255 \quad B' = B/255$$

$$y_{\max} = \max(R', G', B') \quad (4)$$

$$y_{\min} = \min(R', G', B') \quad (5)$$

$$Z = y_{\max} - y_{\min} \quad (6)$$

$$V = y_{\max} \quad (7)$$

$$S = \begin{cases} 0, & y_{\max} = 0 \\ \frac{y_{\max}-y_{\min}}{y_{\max}}, & y_{\max} \neq 0 \end{cases} \tag{8}$$

$$S = \begin{cases} 0, & y_{\max} = 0 \\ 1 - \Delta, & y_{\max} \neq 0 \end{cases} \tag{9}$$

$$H = \begin{cases} 0, Z = 0 \\ \left(60 \times \left(\frac{B'-R'}{y_{\max}-y_{\min}}\right) + 120\right) \bmod 360 \\ \left(60 \times \left(\frac{R'-G'}{y_{\max}-y_{\min}}\right) + 240\right) \bmod 360 \\ \left(60 \times \left(\frac{G'-B'}{y_{\max}-y_{\min}}\right) + 360\right) \bmod 360 \end{cases} \tag{10}$$

RGB to HLS conversion model using color space can be represented as

$$L = (y_{\min} + y_{\max})/2 \tag{11}$$

$$S = \begin{cases} \left(\frac{y_{\max}-y_{\min}}{y_{\max}+y_{\min}}\right), & L \leq 0.5 \\ \left(\frac{y_{\max}-y_{\min}}{2-y_{\max}-y_{\min}}\right), & L > 0.5 \end{cases} \tag{12}$$

$$H = \begin{cases} \frac{(G-B)}{(y_{\max}-y_{\min})} & \text{If } R \text{ is maximum} \\ 2 + \frac{(B-R)}{(y_{\max}-y_{\min})} & \text{If } G \text{ is maximum} \\ 4 + \frac{(R-G)}{(y_{\max}-y_{\min})} & \text{If } B \text{ is maximum} \end{cases} \tag{13}$$

Perspective transform and sobel thresholding

Due to failure in curved lane detection, the performance of Hough transform is also degraded. To improve the performance of lane detection, perspective transformation can be used with Hough transform for higher accuracy. The undistorted image is transformed into “birds-eye view” of the road, which results in lane lines relatively parallel to each other [50, 51, 58]. The wrapped image is digitally manipulated any shapes portrayed in the image to correct the distortions. The perspective transform is applied to convert the 3D object image into warped image (bird’s eye view). Lane lines are actually in parallel but appears that they are converging each other due to distortion. The accuracy of lane detection can be affected by these distortions. Converting into wrapped image using perspective transform helps in figuring out the accurate lane line direction. The given 3×3 matrix represents the perspective transform.

$$\begin{bmatrix} t_i x'_i \\ t_i y'_i \\ t_i \end{bmatrix} = M \cdot \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \tag{14}$$

where dst (i) = (x'_i, y'_i) and src (i) = (x_i, y_i), i = 0, 1, 2, 3 are the output image and input image coordinates of the quadrangle vertices.

The given 3×3 matrix M can be transformed in an image as follows:

$$dst(x, y) = src \left(\frac{M_{11}x + M_{12}y + M_{13}}{M_{31}x + M_{32}y + M_{33}}, \frac{M_{21}x + M_{22}y + M_{23}}{M_{31}x + M_{32}y + M_{33}} \right) \tag{15}$$

where dst (x, y) represents the pixel coordinate for the transformed image, and src (x, y) represents pixels in the input image.

Lane tracking based on kalman filter

First lane lines are detected in the road image using Hough and perspective transform, and then, Kalman filter is used to predict the lane markings in the next frame. Kalman filter is used to increase the accuracy and to predict the best possible lane markings. The Kalman filter is used to provide the better lane detection results for curve lanes in the far-field of view [3]. Firstly, Kalman filter initializes the size of state matrix & measurement matrix. Then, transition matrix and error for state are calculated by the Kalman filter. The predicted state is generated by the error for state. The Kalman filter averages the detected lane lines by using the measurement and previous state. That is the reason, the detected lane lines by the Kalman filter are more stable. Kalman filter predicts the current frame based on the previous detected frame of the lane. Therefore, the Kalman filter improves the robustness and detection rate by tracking the border of lane markings to increase the probability for lane markings.

The Sobel detector is based on discrete first-order operator. The neighborhood of adjacent points is influenced by the current pixel point. To calculate the image brightness value by performing the grayscale weighting a differential operation on neighborhood points. The Sobel thresholding achieves better edges by filtering the noise from the images [64]. It can detect the edges as well as their orientations which make the Sobel operator easier due to the gradient magnitude. The Sobel operator uses the kernel convolution to find the 2-D spatial gradient measurement on the images [1]. The processing of x-direction and y-direction is carried out separately. Because lane lines are designed to proceed in vertical directions only. Figure 21 shows the obtained result by applying the Sobel thresholding.

Curve lane detection part is a challenging task in the far-field of view. Data captured from camera may contain noise and uncertainties in the images, so the Kalman filter is the best approach to estimate the lane lines based on prediction estimate of lanes [65]. Parabola equation can be considered for curved line to predict lanes by Kalman filter [3].

Parabola equation is shown by Eq. (15)

$$y = ax^2 + bx + c \tag{16}$$

Three parameters (a, b, c) have been assumed by using three measurement data, where measurement data are representing far section white point coordinates. Parabolic equation can be represented by Eq. (16) and Eq. (17).

$$\begin{cases} y_{k-1} = ax_{k-1}^2 + bx_{k-1}^1 + cx_{k-1}^0 \\ y_k = ax_k^2 + bx_k^1 + cx_k^0 \\ y_{k+1} = ax_{k+1}^2 + bx_{k+1}^1 + cx_{k+1}^0 \end{cases} \tag{17}$$

where x_{k-1} , x_k , x_{k+1} and y_{k-1} , y_k , y_{k+1} are representing the measurement data of the curve line.

$$\begin{bmatrix} x_{k-1}^2 & x_{k-1}^1 & x_{k-1}^0 \\ x_k^2 & x_k^1 & x_k^0 \\ x_{k+1}^2 & x_{k+1}^1 & x_{k+1}^0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} y_{k-1} \\ y_k \\ y_{k+1} \end{bmatrix} \tag{18}$$

In the next equation, three parameters a, b, c have been estimated.

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \left(\text{inv} \begin{bmatrix} x_{k-1}^2 & x_{k-1}^1 & x_{k-1}^0 \\ x_k^2 & x_k^1 & x_k^0 \\ x_{k+1}^2 & x_{k+1}^1 & x_{k+1}^0 \end{bmatrix} \right) \begin{bmatrix} y_{k-1} \\ y_k \\ y_{k+1} \end{bmatrix} \tag{19}$$

Kalman filter can be implemented for curved lane detection by using the matrix form. Where state transition matrix

(F) and measurement transition matrix (H) and. The equation for the same is defined by Eq. 19.

$$[Y_k] = [H_k] [x_{k|k-1}^-] \tag{20}$$

$$w \quad h \quad e \quad r \quad e$$

$$[Y_k] = \begin{bmatrix} y_{k-1} \\ y_k \\ y_{k+1} \end{bmatrix} \dots [H_k] = \begin{bmatrix} x_{k-1}^2 & x_{k-1}^1 & x_{k-1}^0 \\ x_k^2 & x_k^1 & x_k^0 \\ x_{k+1}^2 & x_{k+1}^1 & x_{k+1}^0 \end{bmatrix}, \text{ and } [x_{k|k-1}^-] = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

Equation (20) represents the prediction of state and error covariance matrix used by the Kalman filter. Prediction step and correction of covariance matrix are the two steps used for Kalman filter.

$$P_{k|k-1} = F_{k-1}P_{k-1}F_{k-1}^T + Q_{k-1} \text{ where } F_{k-1} = \frac{\partial f(x_{k-1})}{\partial x} |_{x = x_{k-1}^-} \tag{21}$$

In Eq. (20), $P_{k|k-1}$ represents the predicted state covariance. The correction steps for the Kalman filter are defined by Eqs. (21), (22), (23) and (24).

$$G_x = P_{k|k-1}H_k^T(H_kP_{k|k-1}H_k^T + R_k)^{-1}, \text{ where } H_k = \frac{\partial h(x_k)}{\partial x} |_{x = x_{k|k-1}^-} \tag{22}$$

$$x_k^- = x_{k|k-1}^- + G_k(G_k - h(x_{k|k-1}^-)) \tag{23}$$

In Eq. (21) and (22), G_x represents the gain of the Kalman filter, x_k^- denotes the posteriori estimate state, and P_k denotes covariance matrix of a posteriori estimate error Eq. (25).

$$\begin{bmatrix} a_{k+1} \\ b_{k+1} \\ c_{k+1} \end{bmatrix} = \begin{bmatrix} a_k \\ b_k \\ c_k \end{bmatrix} + G_k \left(\begin{bmatrix} y_{k-1} \\ y_k \\ y_{k+1} \end{bmatrix} - \begin{bmatrix} x_{k-1}^2 & x_{k-1}^1 & x_{k-1}^0 \\ x_k^2 & x_k^1 & x_k^0 \\ x_{k+1}^2 & x_{k+1}^1 & x_{k+1}^0 \end{bmatrix} \begin{bmatrix} a_k \\ b_k \\ c_k \end{bmatrix} \right) \tag{24}$$

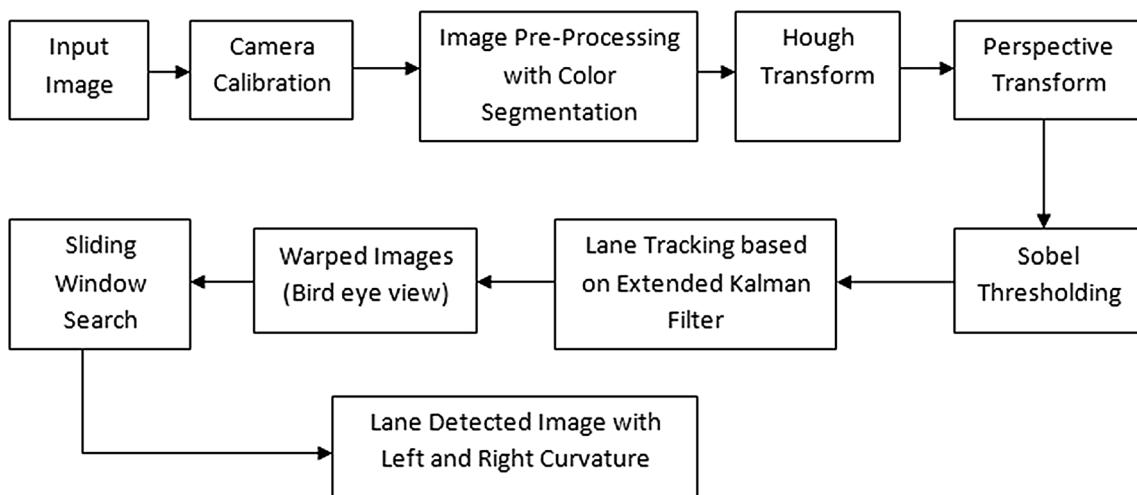


Fig. 4 Framework of proposed curved lane detection

Table 1 Metrics of performance evaluation parameters

Performance metrics
$Accuracy(ACC) = \frac{TP+TN}{TP+TN+FP+FN}$
$Recall(R) = \frac{TP}{TP+FN}$
$FalseNegativeRate(FNR) = 1.00 - R$
$Precision(P) = \frac{TP}{TP+FP}$
$FalsePositiveRate(FPR) = 1.00 - P$
$F1score = \frac{2P}{RP + R}$
$MeanAverage Precision(mAP) = \frac{1}{N} \sum_{i=1}^N AP_i$

$$P_k = (1 - G_k H_k) P_{k|k-1} \tag{25}$$

Sliding window search

Left lane and right lanes are accurately detected by identifying region of interest and applying the perspective transform. Sliding window search is applied to detect the left lane and right lane of the road [66–68]. Figure 26a, b shows the results of left lane and right lane by perspective transform and sliding window search.

The literature survey discussed in the previous section shows that there are various methods for straight and curved lane detection for identifying lane lines. Though, these methods produce better accuracy results for lane line detection.

However, the weaknesses of the existing methods have also been improved for better outcomes, such as lane detection accuracy in various weather conditions. The proposed methods use the optimization of the Hough transform and the extended Kalman filter to enhance the performance of the lane detection further. The proposed methods have been simulated on the TuSimple datasets to analyze the performance of the lane detection. The comparisons have been made with existing methods to demonstrate the robustness and effectiveness of the proposed methods in different environments. These methods show a significant improvement in the results of lane detection compared to the existing methods.

Results and discussion

The proposed algorithms for straight lane and curved lane lines have been implemented in Python programming language using Open CV library.

Performance evaluation metrics for proposed work

The effective and accurate extraction of lane features is the most significant part of lane detection system. The process of lane analysis is influenced by the accuracy estimation of lane feature extraction. The inaccurate lane detection of road model system will not work for real-time lane time detection system. The impact of incorrect lane features will affect the performance of lane detection system. Therefore, the analysis of the accuracy for lane detection system is needed



Fig. 5 Input image



Fig. 6 Converted grayscale



Fig. 7 Gaussian filter image

to ensure the reliability of the lane detection system, and it also improves the performance of the lane detection system.

TuSimple dataset images are used for testing the proposed algorithms Fig. 4. With respect to the detected lane characteristics in an input image frame, the TP, TN, FP, and FN denote the number of true positives, true negatives, false positives, and false negatives, respectively. These are the most common metrics for the evaluation of performance in the image processing Table 1.

Straight lane detection

In the previous section, the proposed lane detection algorithm for straight lane line of the road has been discussed. The simulation has been carried out on the road image to find the straight lane lines. Figure 5 is used as an input image, and then, it is converted to grayscale image; the output after conversion from input image to grayscale image is enhanced image. Figure 6 represents the grayscale converted image. Noise can be removed by applying the Gaussian filter for better result. Figure 7 is showing the result after applying the Gaussian filter. In the second stage, edge extraction is done using Canny edge detector. Figure 8 is the result of Canny edge detection algorithm, which detects the edges of boundary of an object. Figure 9 shows the result of Canny edge detector with color segmentation to indicate the lanes clearly. But adaptive area is to be selected to identify the lanes of the road line, and the resultant image after selecting the region of interest is shown in Fig. 10. The Bit-wise AND operation is performed to find the lane lines of the road, and the same is shown in Fig. 11. Finally, the Hough transform is used to identify the straight lane lines of the road. Figure 12 shows the detected lines using Hough transform on black image. Further, the result of lane detected on black image is blended to an original color image by adding weight to the sum of color image with the line image. Figure 13 shows the result of lane detection on to the original color image by blending the lane detected images on black image.

Figure 14 shows the final output of lane detection using Hough transform, which is the optimized result received from the previous step. The final output is obtained by averaging the lane lines detected by the Hough transform.

The lane detection accuracy of the proposed algorithm while compared with the existing traditional algorithm gives better performance. Table 2 shows the comparison of the proposed algorithm with the existing approaches, and the same is shown graphically in Fig. 15a, b.

Curved lane detection

Identifying the curvy lane lines on the road is the most challenging tasks. This process involves the basic pre-processing, Hough transform, Sobel thresholding, perspective transform and finally the Kalman filter to track the lane lines. The simulation results for this proposed work are divided

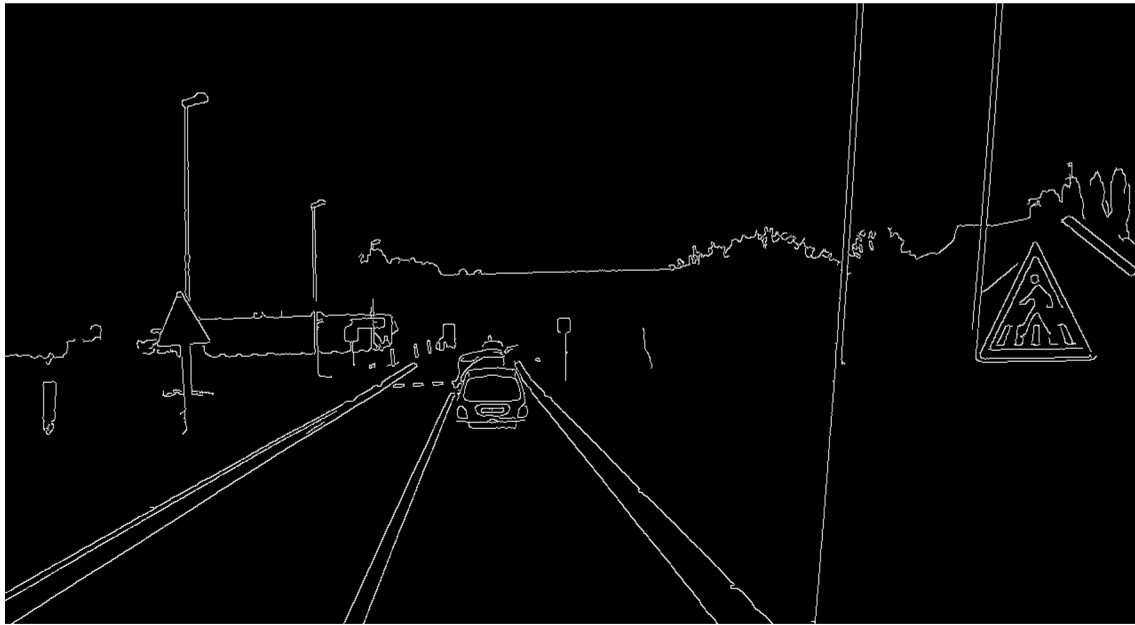


Fig. 8 Canny edge detection algorithm image

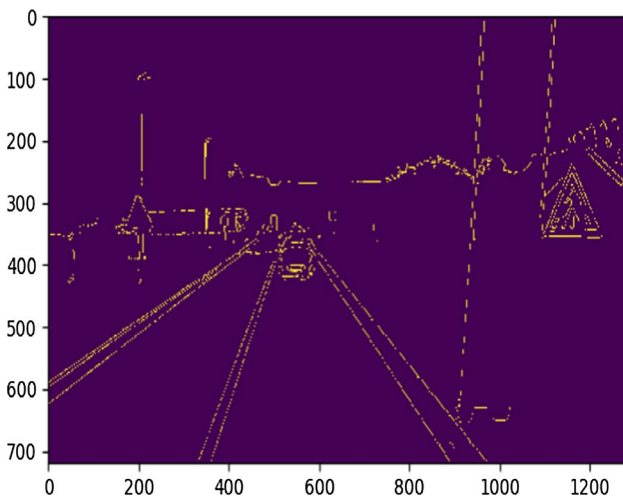


Fig. 9 Color space image along with X- and Y-Axis

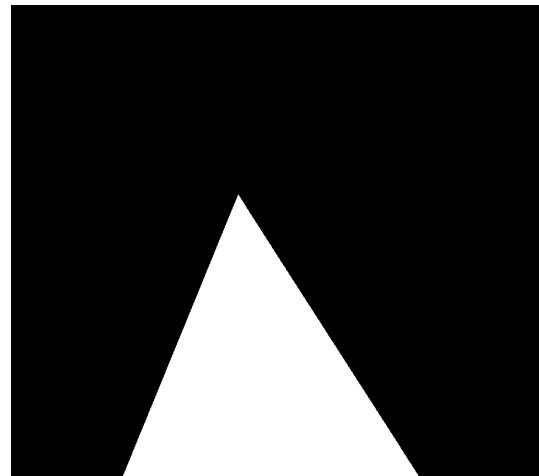


Fig. 10 Region of interest extraction (ROI)

into two parts, pre-processing the image by removing the distortion and by adding HSV and HSL color conversion to improve the accuracy of the lane detection. Figure 16a, c is the original images. Figure 16b, d illustrates the result of undistorted images, and figs. 17, 18 depict the results of color transform images during basic pre-processing. In the next step, draw the polygons on the lane lines to achieve the better result by using the Hough transform. Figure 19a, c shows the result of polygon image using Hough transform and as shown in Fig. 19b, d the result of unwarped images using perspective transform. Figure 20 shows the result of

Sobel edge detected images of unwarped images. Figure 21 is the result of lane lines using extended Kalman filter for identifying and tracking the lane lines. Figure 22 shows the warped image of the original image. At the end, left lane and right lane of the road can be identified by sliding window search. Figure 23a, b shows the result of perspective and sliding window search method, respectively. Finally, Fig. 24 shows the outcome of the lane detection of the proposed algorithm along with the left and right curvature. The radius of curvature is also calculated to compute the vehicle position and visualize the direction of the curve lane lines.

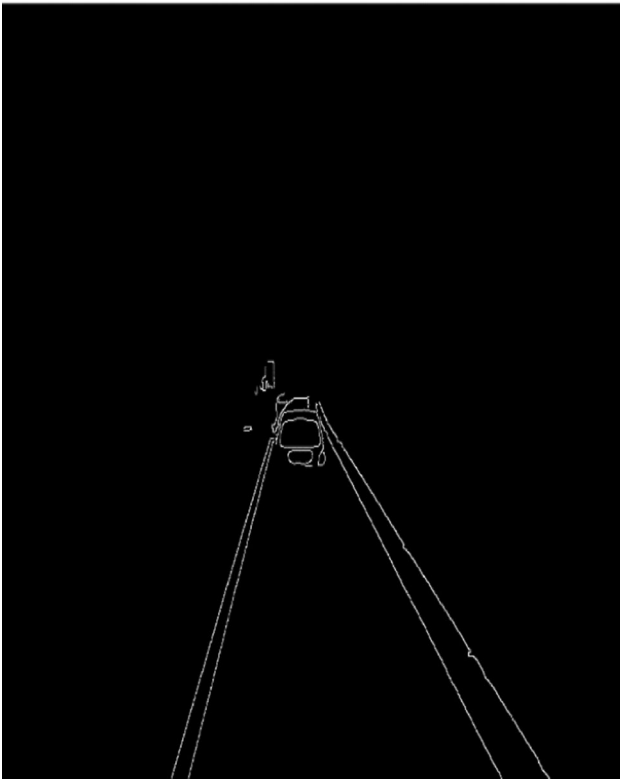


Fig. 11 Result of lane lines by ROI

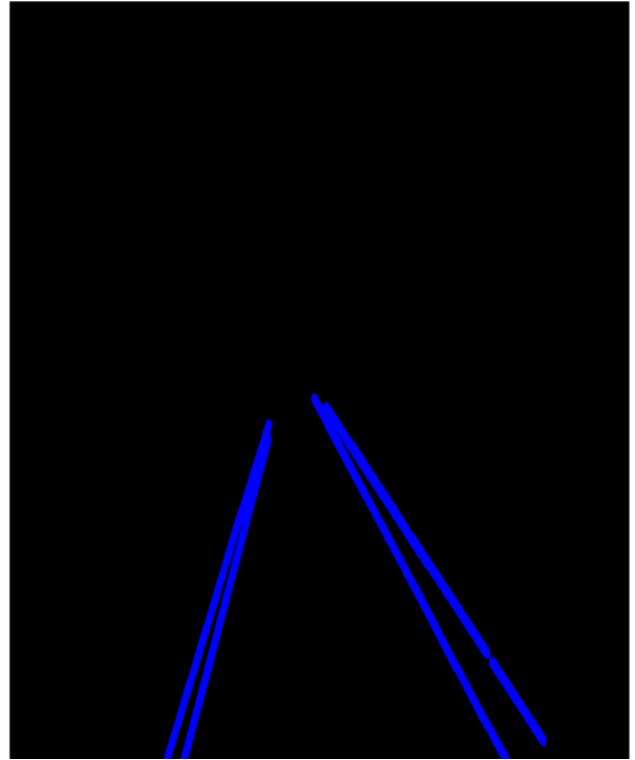


Fig. 12 Result of Hough transform technique

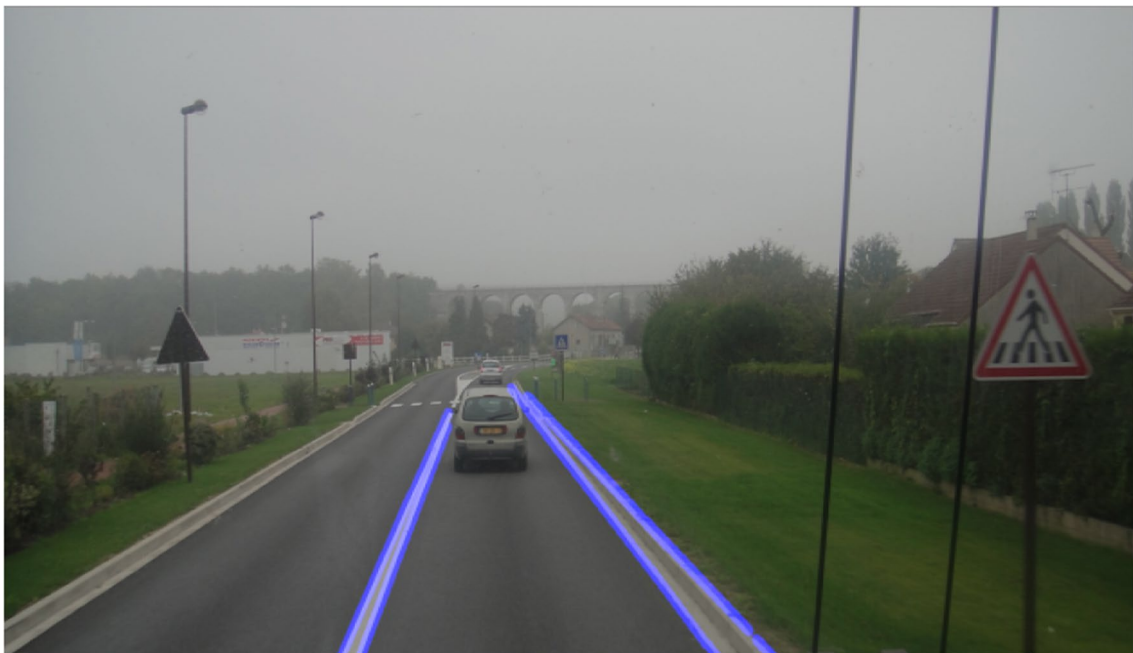


Fig. 13 Lane detection using Hough transform technique

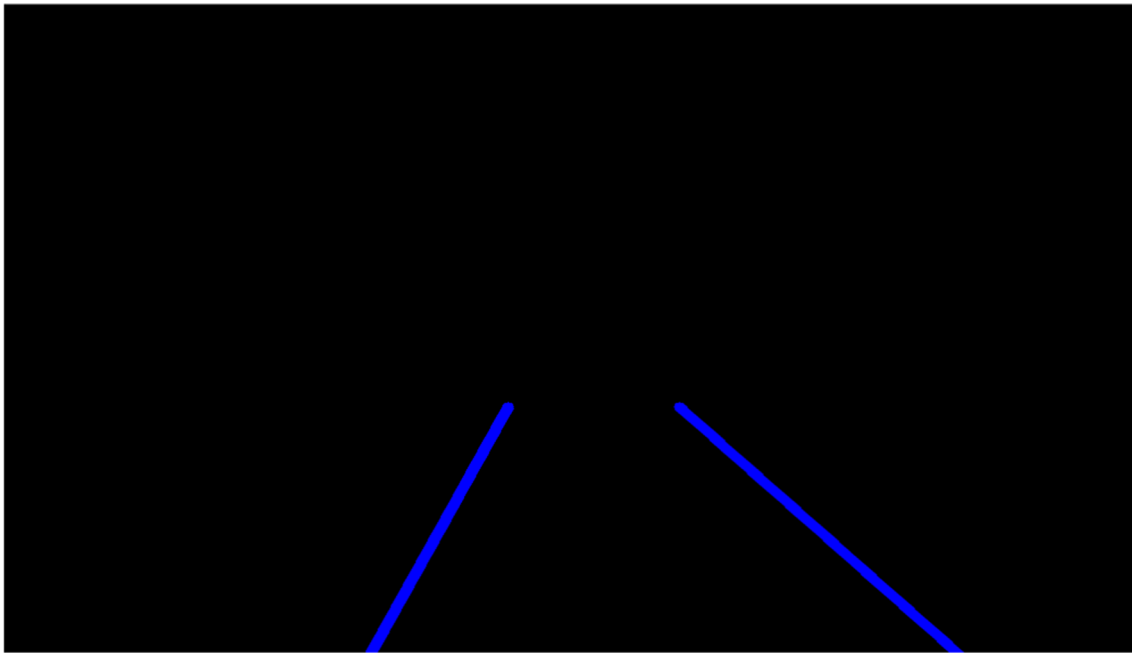


Fig. 14 Road lane lines detection by Hough transform

Table 2 Comparison of processing time and accuracy rate

Parameter	Hough transform technique [28]	Proposed algorithm
Processing time	15.3 Fps	16.7 Fps
Detection accuracy rate	95.7%	95.7%

Figures 25, 26 and 27 show the lane detection results under different scenarios Table 3.

The parameters used for the calculation of accuracy and performance of proposed curved lane detection algorithm are accuracy, precision, recall and F1 score. The simulation

results based on performance evaluation parameters have been compared with the existing traditional algorithms to detect lane boundaries of the road lane. The comparative analysis and performance are illustrated in, and the same is shown graphically in Fig. 28.

The main findings of the proposed work are the results obtained after validating the actual road datasets under different environments. In the TuSimple dataset, the road scenes are visualized through the Hough transform and perspective transform techniques which track the lane lines effectively. Table 2 and 4 show that the performance of the proposed methods gives better results than the existing traditional methods that we have compared in the result and discussion section. The straight lane detection method

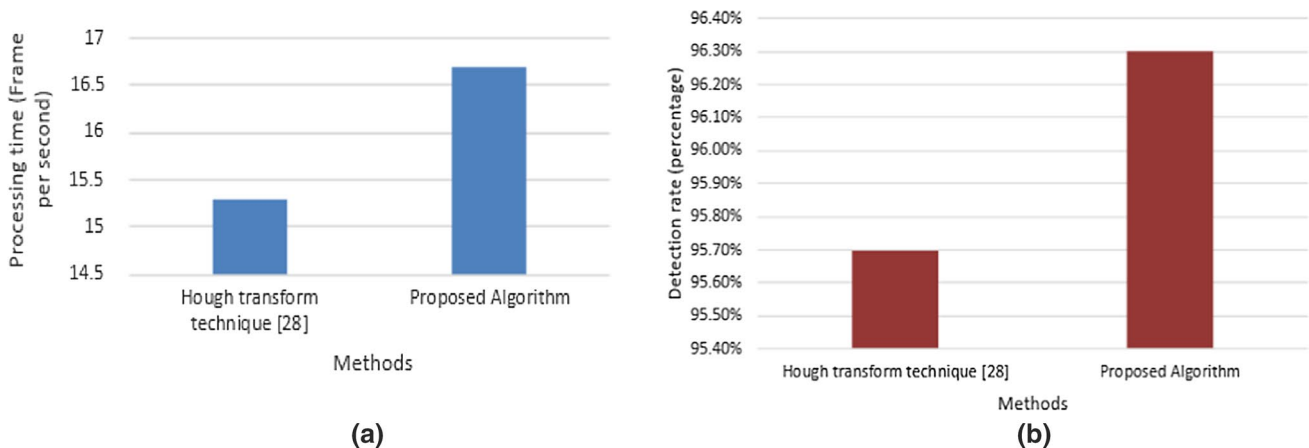


Fig. 15 (a), (b) The comparative analysis of accuracy of detection rate and processing time

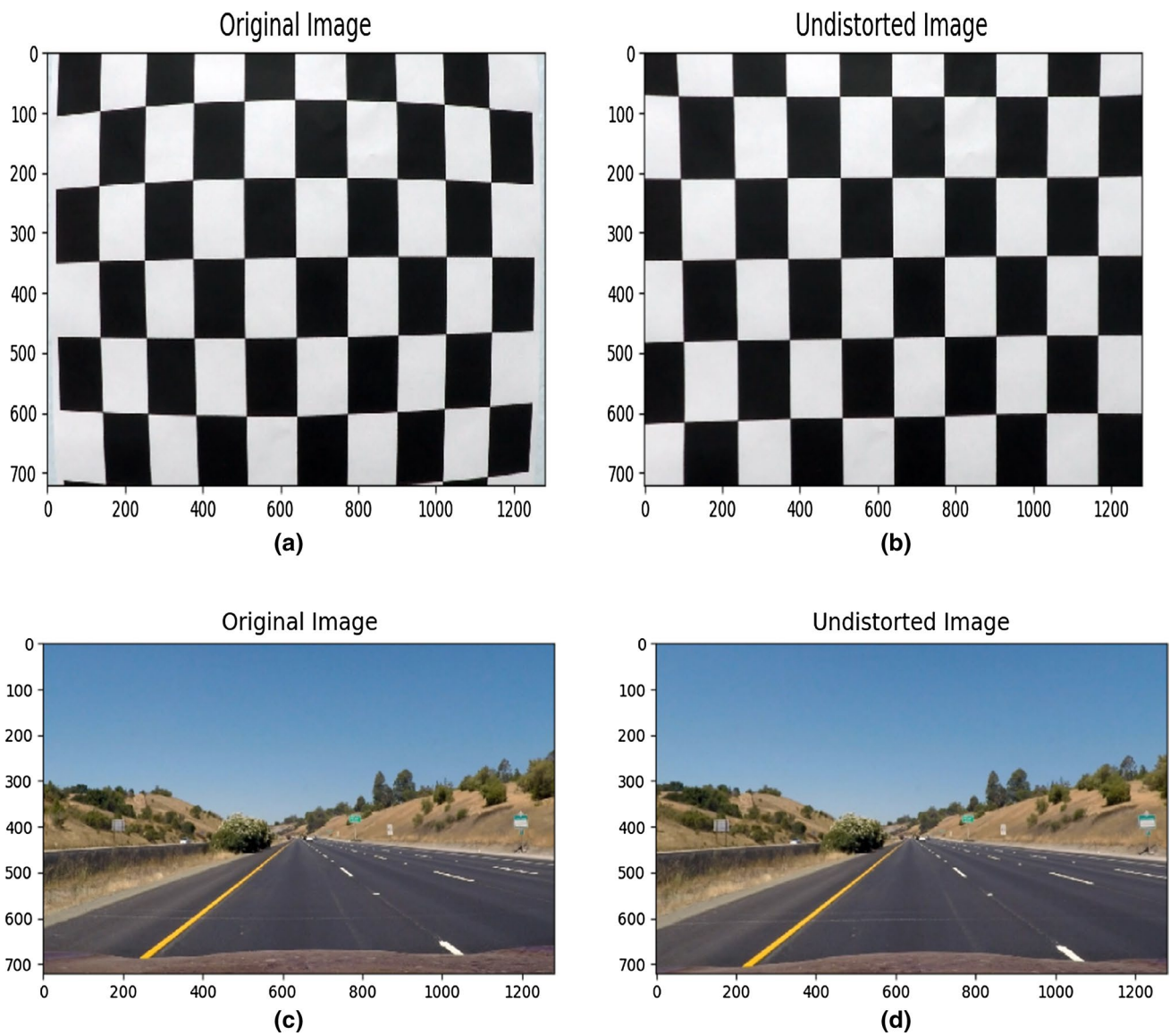


Fig. 16 The original images and corresponding undistorted images

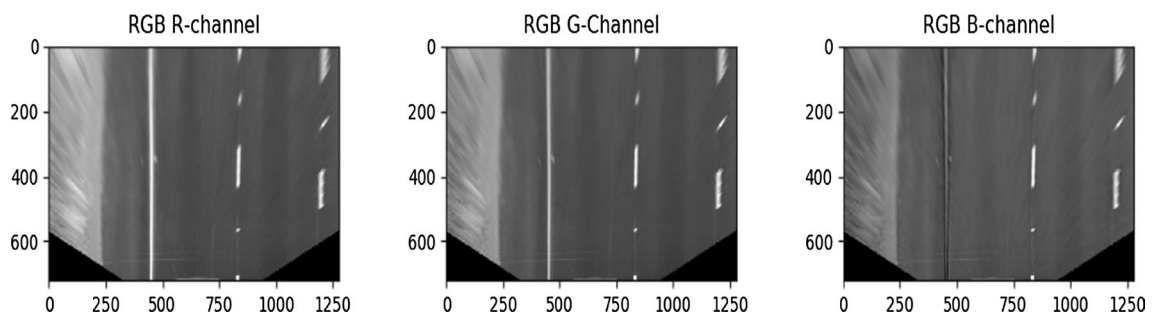


Fig. 17 The result of RGB Red, Green and Blue channel

additionally performs the optimization of the Hough transform, and the curved lane detection method tracks the lane lines using the Kalman filter to improve the accuracy of the proposed method. The simulation results show the

robustness of the proposed methods in different illuminations, on highways, and on curvy roads. The effectiveness of these lane detecting approaches has long been employed to provide lane departure warning, lane-keeping, and driver

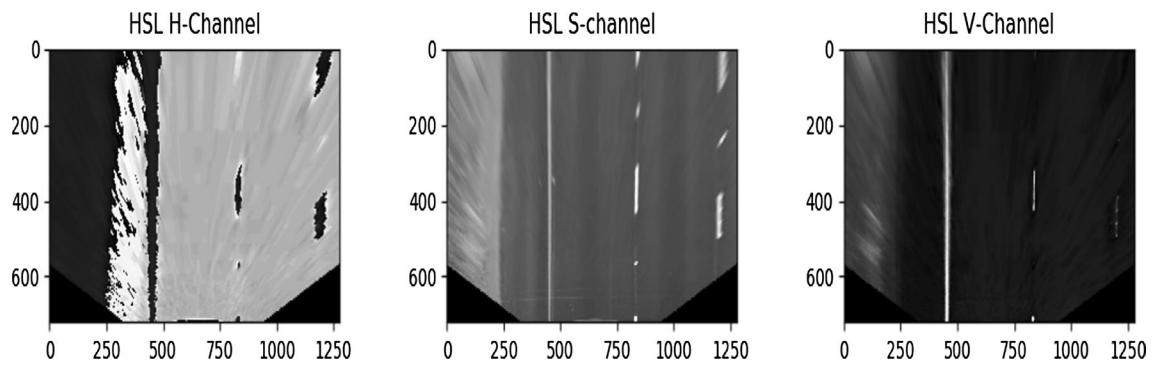


Fig. 18 The result of HSL Hue, Saturation and Lightness channel

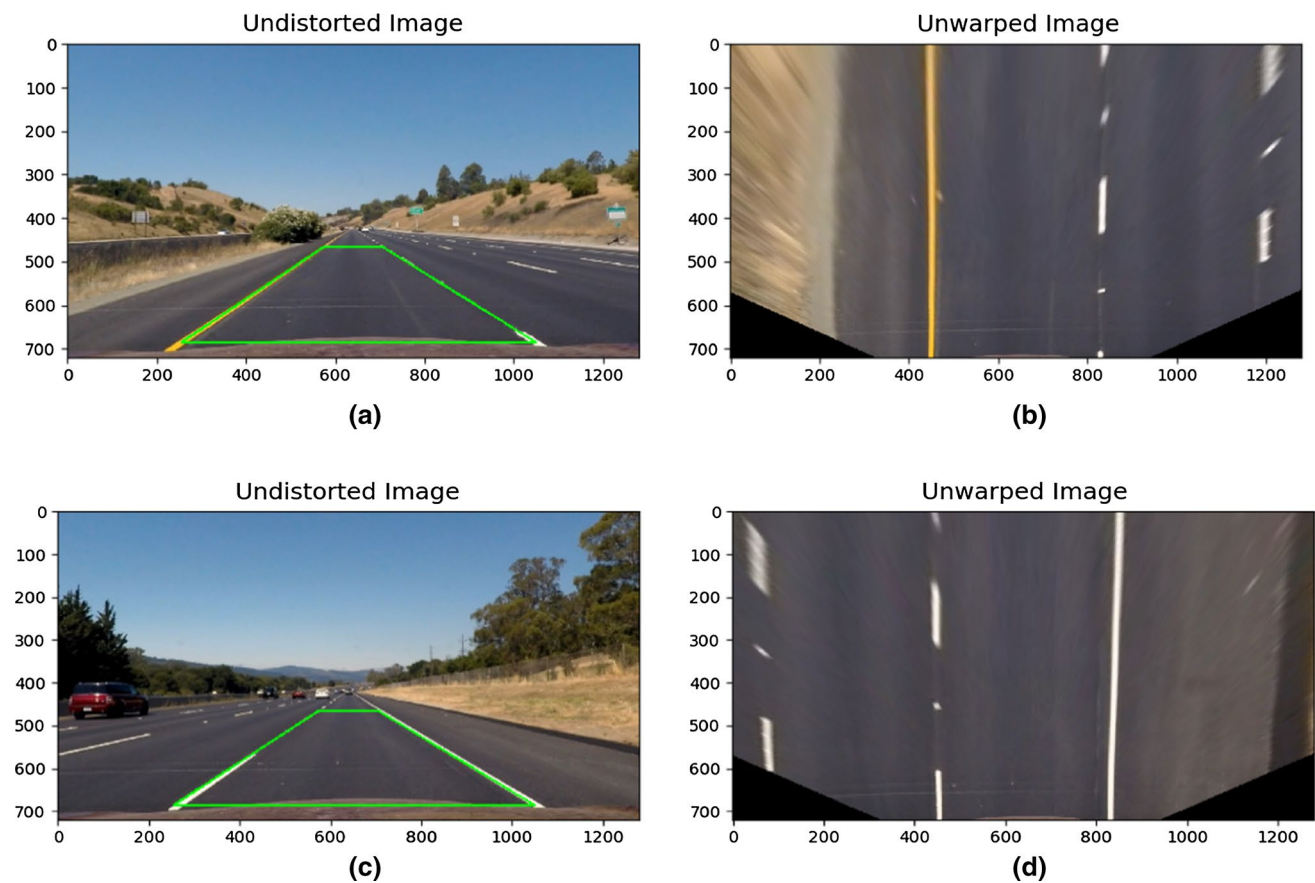


Fig. 19 The result of polygon and corresponding unwarped image using perspective transforms

assistance systems for intelligent vehicle systems. Consequently, this kind of algorithm has attracted a lot of interest. However, the accuracy of the proposed methods still can be enhanced for the congested network.

Conclusion and future work

Lane detection has been an active research area in the recent past because of its potentially widespread areas of applications such as driver assistance, lane departure warning, and lane-keeping systems. This work proposes efficient and effective lane detection methods that facilitate straight and curved lane detection for roads. The presented paper

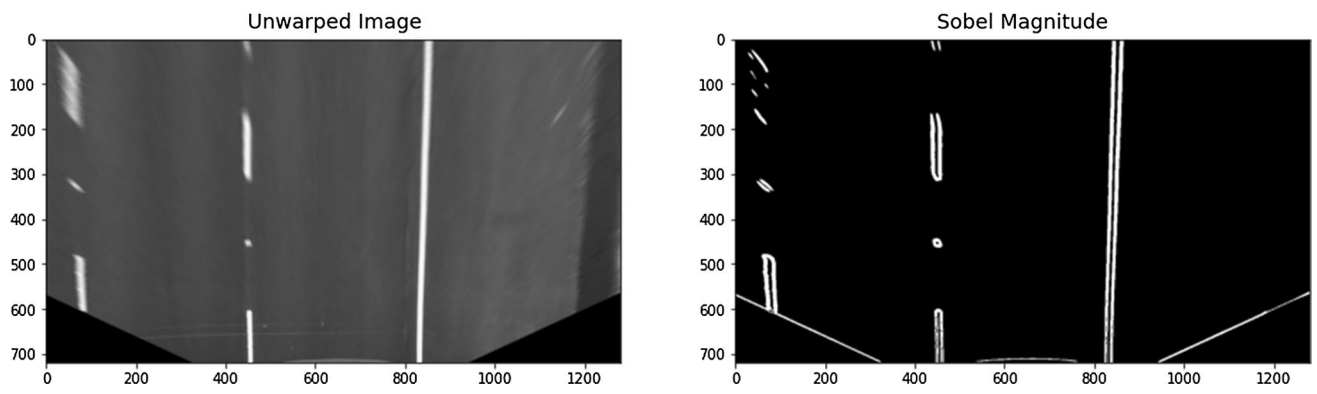


Fig. 20 The result of unwarped image and lane detection by Sobel algorithm

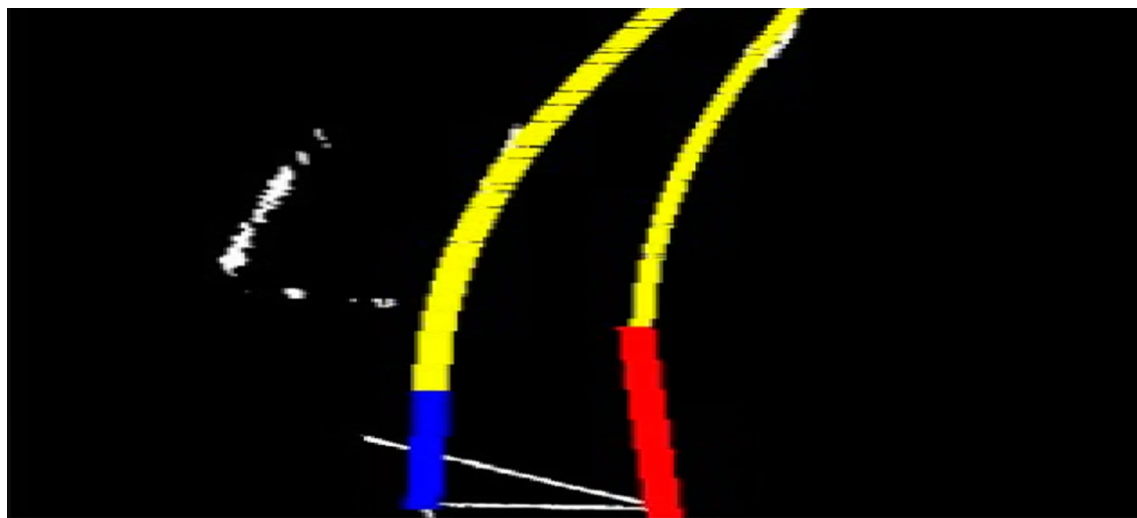


Fig. 21 Curve lane detection using Kalman filter

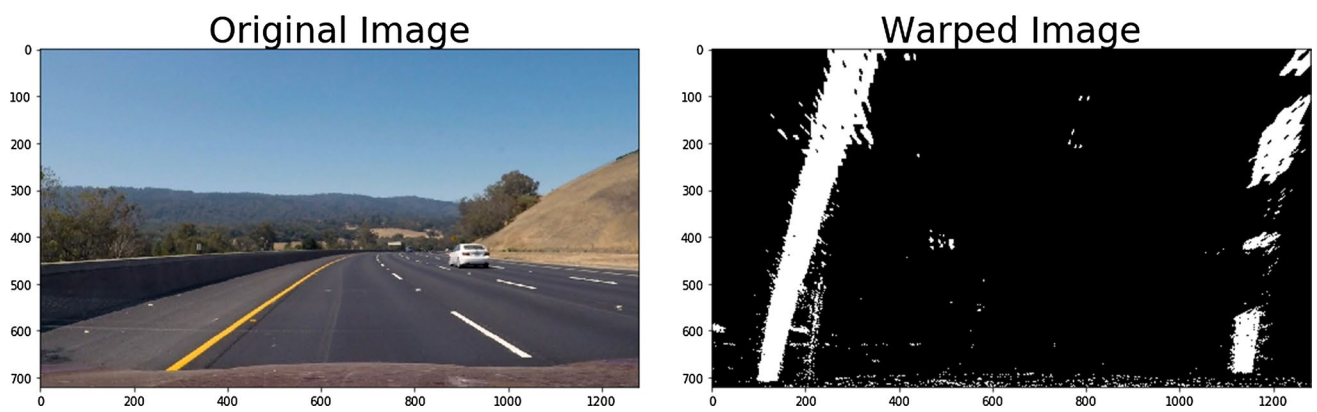


Fig. 22 The result of warped image

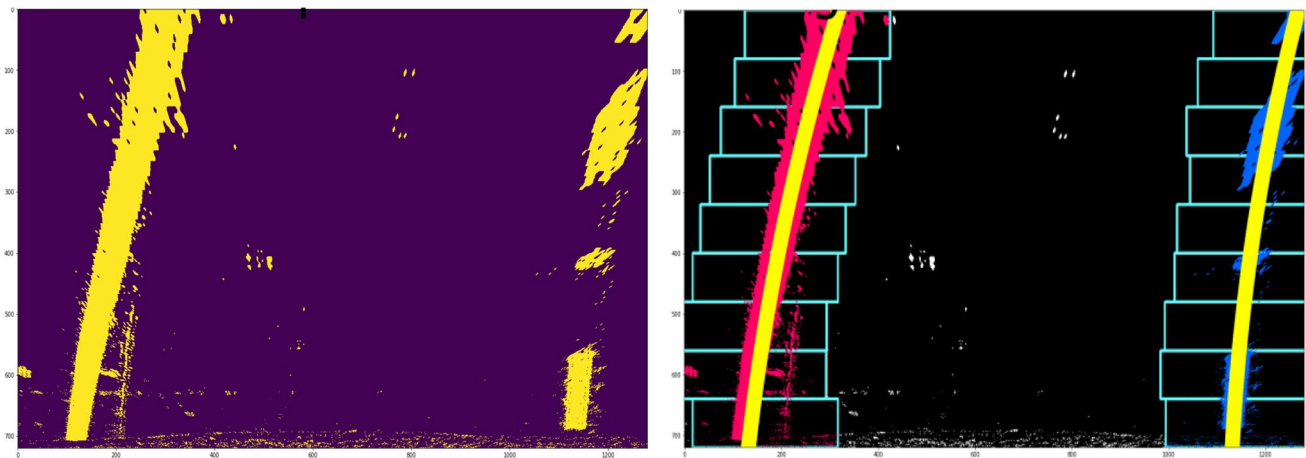


Fig. 23 Test results of perspective transform and sliding window search

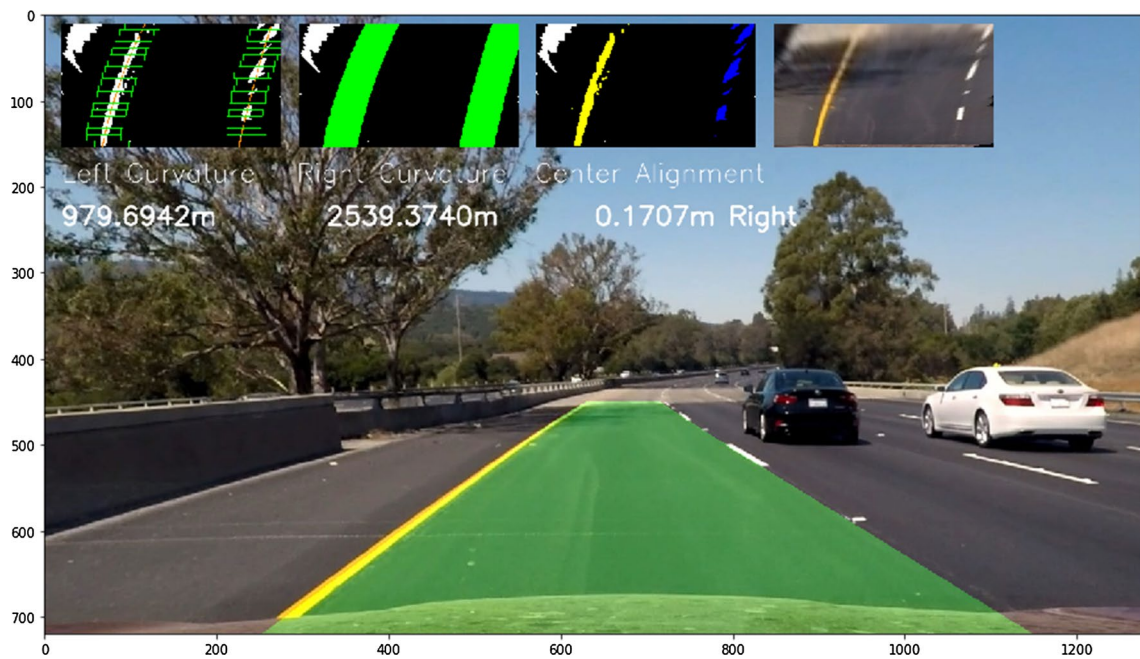


Fig. 24 The final result of detection of lane lines along with left and right curvature

proposes an algorithm using the Canny edge detection algorithm and the Hough transform technique. Finally, we have optimized the Hough transform to improve the lane detection accuracy rate. Subsequently, a curved lane detection algorithm using perspective transform and the Kalman filter has been proposed to achieve better accuracy. The proposed method for straight lane detection achieves a better detection accuracy and processing time are 16.7 FPS and 96.3%, respectively. The proposed method for curved lane detection achieves accuracy rate, precision, recall, and F1-score

are 97.74%, 98.15%, 97.35%, and 97.75%, respectively. Finally, the proposed methods have been simulated on TuSimple Datasets for straight and curved lane lines. And the results obtained confirm that the proposed lane detection methods achieve better performance than the existing ones, such as RANSAC + HSV, Gaussian filter + Hough transform, LD + Inverse Perspective, and MSER Algorithm HSV-ROI + Hough, Improved LD + Hough. Lane detection is proved to be the more effective and efficient for road boundaries detection.

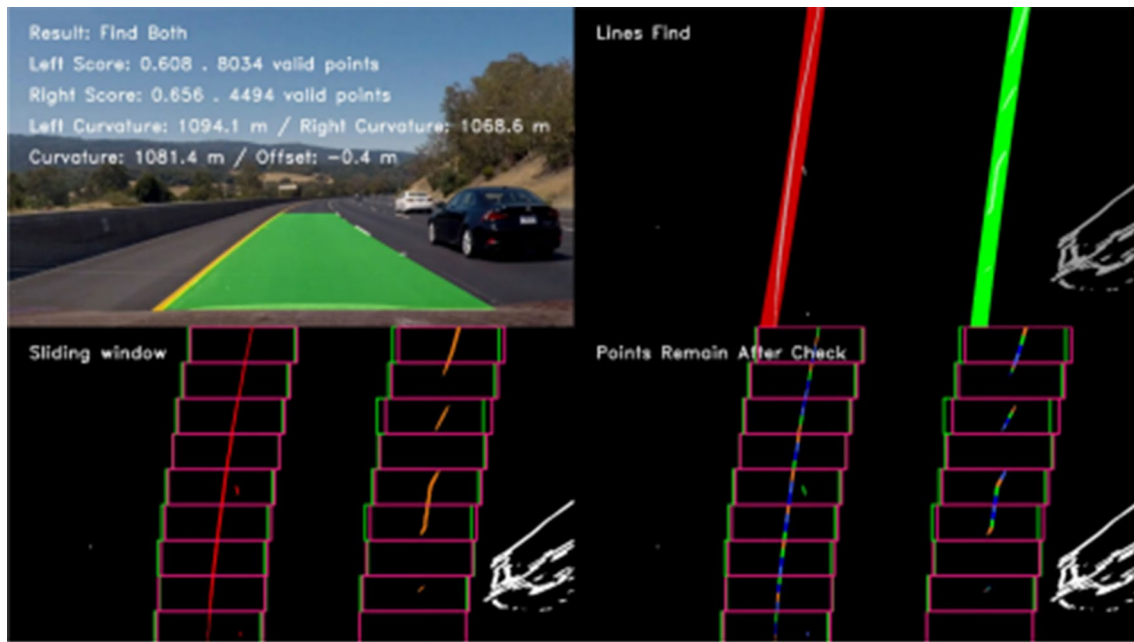


Fig. 25 Lane detection on highway with sliding window

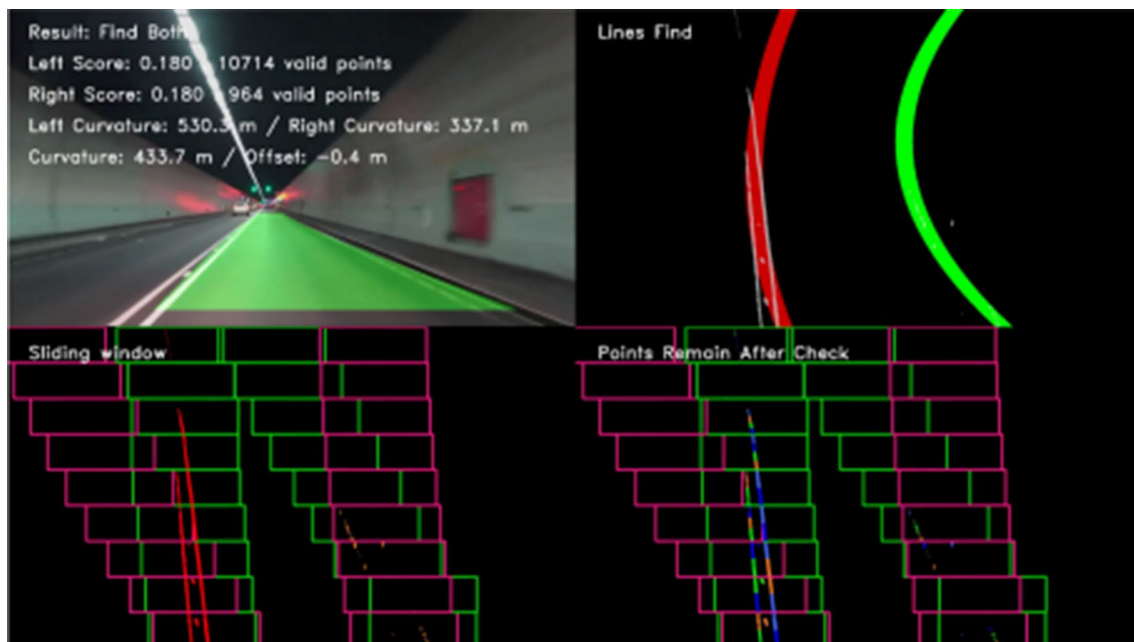


Fig. 26 The test result of lane detection under low light condition

- In our future work, lane detection to be enhanced for making it capable of dealing with the issues congested network for real time. The work can be extended for the lane detection of the roads which are zigzag in nature. In the lane detection system, accurate and effective systems are always the significant factors to support advanced driver assistance system (ADAS). Therefore, it is always a matter of concern to improve the accuracy of the systems.
- In future, this work can be further extended based on the deep learning methods and computer vision technology. Many of the existing lane detection algorithms fail

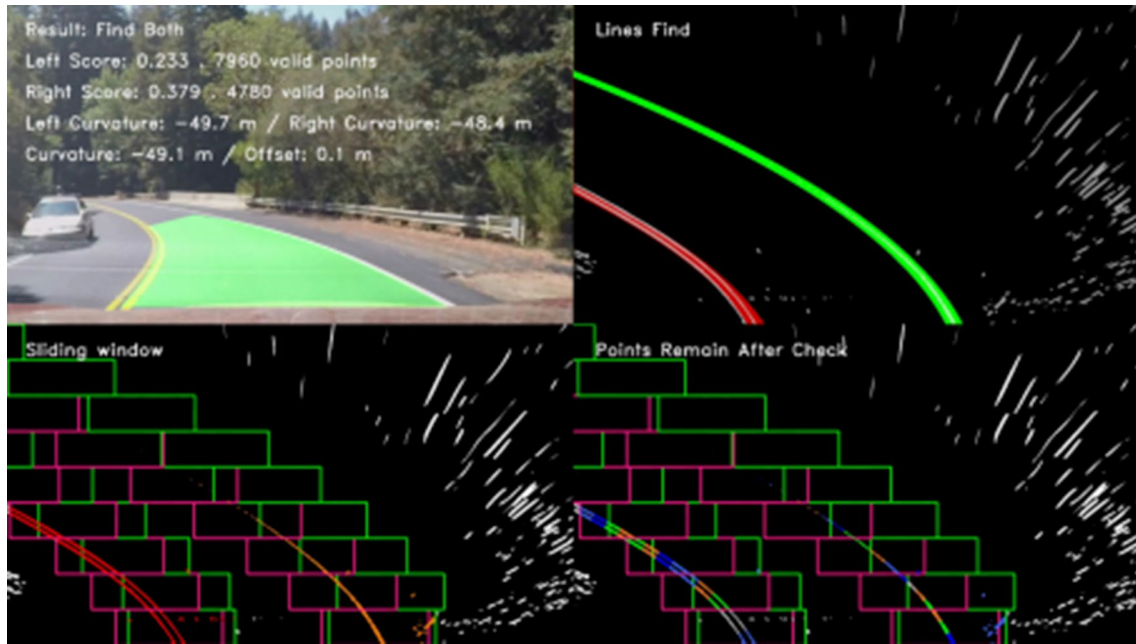


Fig. 27 The lane detection result on curvy road

Table 3 The performance of the proposed algorithm for lane detection under different conditions

SNo	Different environment	Total number of frames	Number of detected frames	Missed and incorrect frame	Detection rate	Average processing time per frame
1	Highway road	1533	1499	34	97.74%	25.6
2	Tunnel road	1761	1665	96	94.54%	29.8
3	Curvy road	1468	1416	52	96.46%	20.5

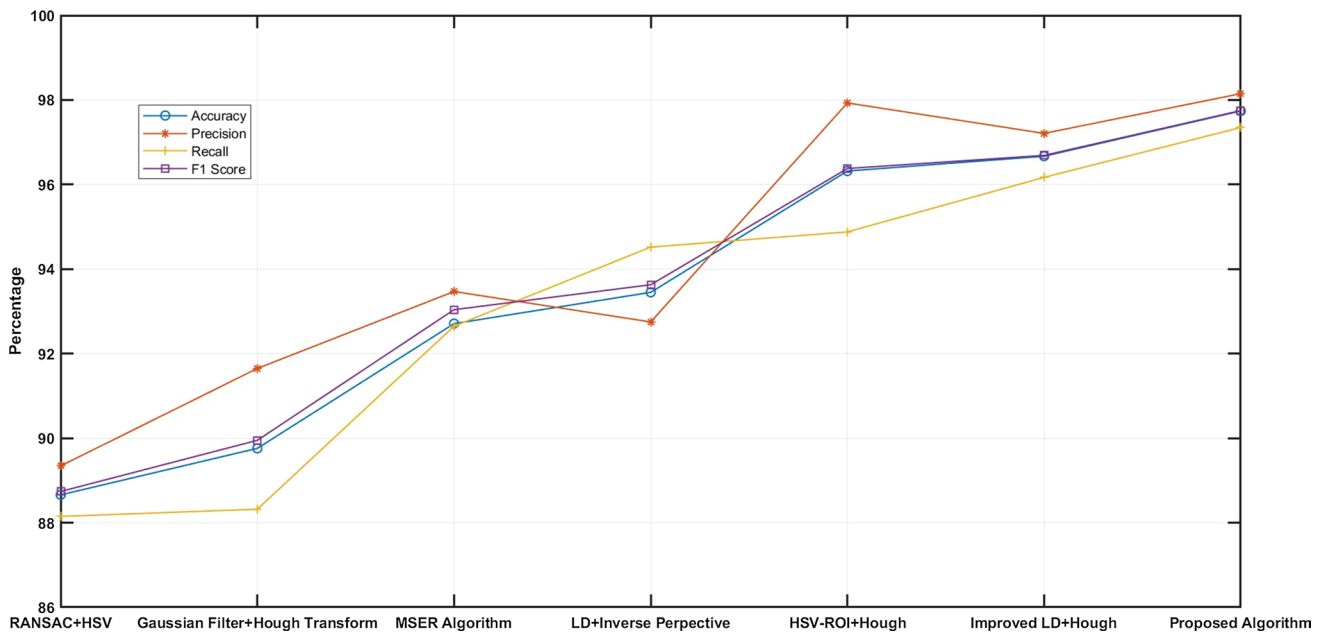


Fig. 28 Comparative analysis of proposed algorithm with existing algorithms

Table 4 Comparison of performance parameters used for curved lane detection

Algorithm	Accuracy %	Precision	Recall	F1-Scores
RANSAC+ HSV [56]	88.66	89.34	88.15	88.74
Gaussian filter+hough transform [38]	89.76	91.65	88.32	89.95
MSER algorithm [57]	92.71	93.47	92.65	93.04
LD+Inverse perspective [58]	93.45	92.75	94.52	93.63
HSV-ROI+Hough [51]	96.32	97.93	94.88	96.38
Improved LD+Hough [52]	96.67	97.21	96.17	96.69
Proposed Algorithm	97.74	98.15	97.35	97.75

to detect lane lines under complex traffic and low light conditions. We will propose an effective lane detection method to overcome these shortcomings in our future work. The effective lane detection method will employ the vertical and contextual spatial features to obtain a high detection rate for crowded scenes and occluded lane lines. In addition, more contextual information can be obtained by feature merging block and information exchange block.

Declarations

Conflict of interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The authors declare the following financial interests/personal relationships which may be considered as potential competing interests.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed Consent For this type of study, formal consent is not required.

References

- Dinakaran K, Sagayaraj AS, Kabilesh SK, Mani T, Anandkumar A, Chandrasekaran G (2020) Advanced lane detection technique for structural highway based on computer vision algorithm. *Mater Today: Proceed* 45:2073–2081
- Nguyen V, Kim H, Jun S, Boo K (2018) A study on real-time detection method of lane and vehicle for lane change assistant system using vision system on highway. *Eng Sci Technol Int j* 21(5):822–833
- Dorj B, Hossain S, Lee DJ (2020) Highly curved lane detection algorithms based on kalman filter. *Appl Sci* 10(7):2372
- Gayko JE (2012) Lane departure and lane keeping. *Handbook of Intelligent Vehicles*. Springer, London, pp 689–708
- Visvikis C et al (2008) Study on lane departure warning and lane change assistant systems. *Trans Res Lab Proj Rpt PPR* 374:1–13
- Muthalagu R, Bolimera A, Kalaichelvi V (2021) Vehicle lane markings segmentation and keypoint determination using deep convolutional neural networks. *Multimed Tools Appl* 80(7):11201–11215
- Sugawara T, Altmannshofer H, Kakegawa S (2017) Applications of road edge information for advanced driver assistance systems and autonomous driving. *Advanced microsystems for automotive applications*. Springer, London, pp 71–86
- Ghanem S, Kanungo P, Panda G, Parwekar P (2021) An improved and low-complexity neural network model for curved lane detection of autonomous driving system. *Soft Computing*. <https://doi.org/10.1007/s00500-021-05815-0>
- Amaradi P, Sriramoju N, Li Dang GS, Tewolde, Kwon J (2016) Lane following and obstacle detection techniques in autonomous driving vehicles. In: *IEEE International Conference on Electro Information Technology (EIT)*, Grand Forks, ND, pp 0674–0679
- Ghanem S, Kanungo P, Panda G, Satapathy SC, Sharma R (2021) Lane detection under artificial colored light in tunnels and on highways: an IoT-based framework for smart city infrastructure. *Complex Intelligent Systems*. <https://doi.org/10.1007/s40747-021-00381-2>
- Talib ML, Rui X, Ghazali KH, Mohd. Zainudin N, Ramli S, (2013) Comparison of edge detection technique for lane analysis by improved hough transform, vol 8237. Springer, Cham, pp 176–183
- Yan X, Li Y (2017) A method of lane edge detection based on Canny algorithm. In: *2017 Chinese Automation Congress (CAC)*. pp 2120–2124
- Canny J (1986) A computational approach to edge detection [J]. *IEEE Trans Pattern Anal Mach Intell* 8(6):679–698
- Richard OD, Peter EH (1972) Use of the Hough transformation to detect lines and curves in picture. *Graphics Image Process* 15(1):11–15
- Ye YY, Hao XL, Chen HJ (2018) Lane detection method based on lane structural analysis and CNNs. *IET Intel Transport Syst* 12(6):513–520
- Kuk JG, An JH, Ki H, Cho (2010) Fast lane detection and tracking based on Hough transform with reduced memory requirement. In *IEEE conference on intelligent transportation systems*. 1344–1349
- Hsiao PY, Yeh CW (2006) A portable real-time lane departure warning system based on embedded calculating technique. In: *2006 IE 63rd Vehicular Technology Conference*. 6:2982–2986
- Dinakaran K, Sagayaraj AS, Kabilesh SK, Mani T, Anandkumar A, Chandrasekaran G (2021) Advanced lane detection technique for structural highway based on computer vision algorithm. *Mater Today: Proceed* 45:2073–2081
- Ding Y, Xu Z, Zhang Y, Sun K (2017) Fast lane detection based on bird's eye view and improved random sample consensus algorithm. *Multimed Tools Appl* 76(21):22979–22998
- Collado, Hilario C, Escalera A, Armingol JM (2005) Detection and classification of road lanes with a frequency analysis. *IEEE Intelligent Vehicles Symposium*. Nevada, USA, 7883
- Wang Y, Teoh E, Shen D (2004) Lane detection and tracking using b-snake. *Image Vis Comput* 22(4):269–280
- King HL, Kah PS, Li-Minn A (2009) Lane detection and kalman-based linear-parabolic lane tracking. In: *Proceedings of IEEE international conference on intelligent human-machine systems and cybernetics*. IEEE, Hangzhou, p 351–354

23. Lu WN, Zheng YC, Ma YQ, et al. (2008) An integrated approach to recognition of lane marking and road boundary. In: Proceedings of International Workshop on Knowledge Discovery and Data Mining. University of Adelaide, Australia, p 649–653
24. Chen Q, Wang H (2006) A real-time lane detection algorithm based on a hyperbola-pair model. IEEE Intelligent Vehicles Symposium. IEEE, Tokyo, p 510–515
25. Teo TY, Sutopo R, Lim JMY, Wong K (2021) Innovative lane detection method to increase the accuracy of lane departure warning system. *Multimed Tools Appl* 80(2):2063–2080
26. Dorj B, Lee DJ (2016) A precise lane detection algorithm based on top view image transformation and least-square approaches. *J Sens* 6:2016
27. Kim J, Kim W (2022) Direction-aware feedback network for robust lane detection. *Multimed Tools Appl*. <https://doi.org/10.1007/s11042-022-12541-8>
28. Chen C, Tang L, Wang Y, Qian Q (2019) Study of the lane recognition in haze based on kalman filter. In: 2019 International conference on artificial intelligence and advanced manufacturing (AIAM). pp. 479–483
29. TuSimple (2017) TuSimple Velocity Estimation Challenge in <http://github.com/TuSimple/tusimple-benchmark/tree>
30. Pan X, Shi J, Luo P, Wang X, Tang X (2018) Spatial cnn for traffic scene understanding. In: 32nd AAAI Conf Artif Intell. 1–8
31. Yu F, Chen H, Wang X, Xian W, Chen Y, Liu F, Madhavan V, Darrell T (2020) BDD100K: a diverse driving dataset for heterogeneous multitask learning. In: IEEE Conf Int'l Conf Comput Vis Pattern Recognit. 2636–2645
32. Xing Y, Lv C, Chen L, Wang H, Wang H, Cao D, Velenis E, Wang FY (2018) Advances in vision-based lane detection: Algorithms, integration, assessment, and perspectives on ACP-based parallel vision. *IEEE/CAA J Autom Sin* 5:645–661
33. Ko Y, Jun J, Ko D, Jeon M, Key points estimation and point instance segmentation approach for lane detection. arXiv 2020, [arXiv:2002.06604](https://arxiv.org/abs/2002.06604)
34. Liang D, Guo YC, Zhang SK, Mu TJ, Huang X (2020) Lane Detection: a survey with new results. *J Comput Sci Technol* 35:493–505
35. Yang W, Zhang X, Lei Q, Shen D, Huang Y (2020) lane position detection based on long short-term memory (LSTM). *Sensors* 20:3115
36. Zou Q, Jiang H, Dai Q, Yue Y, Chen L, Wang Q (2020) Robust lane detection from continuous driving scenes using deep neural networks. *IEEE Trans Veh Technol* 69:41–54
37. Jiang L, Li J, Ai W (2019) Lane line detection optimization algorithm based on improved Hough transform and r-least squares with dual removal. In 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC) Vol. 1, pp. 186–190. IEEE
38. Dubey A, Bhurchandi KM (2015) Robust and real time detection of curvy lanes (curves) with desired slopes for driving assistance and autonomous vehicles. arXiv preprint [arXiv:1501.03124](https://arxiv.org/abs/1501.03124)
39. Ma C, Mao L, Zhang Y, Xie M (2010, July) Lane detection using heuristic search methods based on color clustering. In 2010 international conference on communications, circuits and systems (ICCCAS) pp. 368–372. IEEE
40. Srivastava S, Lumb M, Singal R (2014) Improved lane detection using hybrid median filter and modified Hough transform. *Int J Adv Res Comput Sci Softw Eng* 4(1):30–37
41. Wang J, Ma H, Zhang X, Liu X (2018) Detection of lane lines on both sides of road based on monocular camera. In 2018 IEEE International Conference on Mechatronics and Automation (ICMA) (pp. 1134–1139). IEEE
42. Li Y, Zhang W, Ji X, Ren C, Wu J (2019) Research on lane a compensation method based on multi-sensor fusion. *Sensors* 19(7):1584
43. Haselhoff A, Kummert A (2009) 2D line filters for vision-based lane detection and tracking. In 2009 International Workshop on Multidimensional (nD) Systems pp. 1–5. IEEE
44. Piao J, Shin H (2017) Robust hypothesis generation method using binary blob analysis for multi-lane detection. *IET Image Proc* 11(12):1210–1218
45. Salarpour A, Salarpour A, Fathi M, Dezfoulian M (2011) Vehicle tracking using Kalman filter and features. *Sign Image Process* 2(2):1–3
46. Wu Y, Chen Z (2016) A detection method of road traffic sign based on inverse perspective transform. In 2016 IEEE International Conference of Online Analysis and Computing Science (ICOACS). pp. 293–296. IEEE
47. Jung, C. R., & Kelber, C. R. (2004, June). A lane departure warning system based on a linear-parabolic lane model. In IEEE Intelligent Vehicles Symposium, 2004. pp. 891–895. IEEE
48. Lee, D. H., & Liu, J. L. (2021). End-to-end deep learning of lane detection and path prediction for real-time autonomous driving. arXiv preprint [arXiv:2102.04738](https://arxiv.org/abs/2102.04738)
49. Lee, M., Lee, J., Lee, D., Kim, W., Hwang, S., & Lee, S. (2021). Robust lane detection via expanded self-attention. arXiv preprint [arXiv:2102.07037](https://arxiv.org/abs/2102.07037)
50. Muthalagu R, Bolimera A, Kalaichelvi V (2020) Lane detection technique based on perspective transformation and histogram analysis for self-driving cars. *Comput Electr Eng* 85:106653
51. Li M, Li Y, Jiang M (2018) Lane detection based on connection of various feature extraction methods. *Advances in Multimedia*, 2018
52. Zheng F, Luo S, Song K, Yan CW, Wang MC (2018) Improved lane line detection algorithm based on Hough transform. *Pattern Recognit Image Anal* 28(2):254–260
53. Wang JG, Lin CJ, Chen SM (2010) Applying fuzzy method to vision-based lane detection and departure warning system. *Expert Syst Appl* 37(1):113–126
54. Son J, Yoo H, Kim S, Sohn K (2015) Real-time illumination invariant lane detection for lane departure warning system. *Expert Syst Appl* 42(4):1816–1824
55. Zhang X, Zhu X (2019) Autonomous path tracking control of intelligent electric vehicles based on lane detection and optimal preview method. *Expert Syst Appl* 121:38–48
56. Kim KB, Song DH (2017) Real time road lane detection with RANSAC and HSV Color transformation. *J Inform Comm Convergence Eng* 15(3):187–192
57. Mammeri A, Boukerche A, Lu G (2014) Lane detection and tracking system based on the MSER algorithm, hough transform and kalman filter. In Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems. pp. 259–266
58. Huang Y, Li Y, Hu X, Ci W (2018) Lane detection based on inverse perspective transformation and Kalman filter. *KSII Trans Internet Inform Syst (TIIS)* 12(2):643–661
59. Yan X, Li Y (2017) A method of lane edge detection based on Canny algorithm. In 2017 Chinese Automation Congress (CAC). pp. 2120–2124. IEEE
60. Hoang TM, Hong HG, Vokhidov H, Park KR (2016) Road lane detection by discriminating dashed and solid road lanes using a visible light camera sensor. *Sensors* 16(8):1313
61. Talib ML, Rui X, Ghazali KH, Zainudin NM, Ramli S (2013) Comparison of Edge Detection Technique for Lane Analysis by Improved Hough Transform. In: International Visual Informatics Conference. pp. 176–183. Springer: Cham

62. Lee JW, Yi UK (2005) A lane-departure identification based on LBPE, Hough transform, and linear regression. *Comput Vis Image Underst* 99(3):359–383
63. Olson CF (1999) Constrained Hough transforms for curve detection. *Comput Vis Image Underst* 73(3):329–345
64. Cao J, Song C, Song S, Xiao F, Peng S (2019) Lane detection algorithm for intelligent vehicles in complex road conditions and dynamic environments. *Sensors* 19(14):3166
65. Lim KH, Seng KP, Ang LM, Chin SW (2009) Lane detection and Kalman-based linear-parabolic lane tracking. In 2009 International Conference on Intelligent Human-Machine Systems and Cybernetics. Vol. 2, pp. 351–354. IEEE
66. Wang Z, Li X, Jiang Y, Shao Q, Liu Q, Chen B, Huang D (2015) swDMR: a sliding window approach to identify differentially methylated regions based on whole genome bisulfite sequencing. *PLoS ONE* 10(7):e0132866
67. Yi SC, Chen YC, Chang CH (2015) A lane detection approach based on intelligent vision. *Comput Electr Eng* 42:23–29
68. Wu BF, Huang HY, Chen CJ, Chen YH, Chang CW, Chen YL (2013) A vision-based blind spot warning system for daytime and nighttime driver assistance. *Comput Electr Eng* 39(3):846–862