



Lane detection under artificial colored light in tunnels and on highways: an IoT-based framework for smart city infrastructure

Safwan Ghanem¹ · Priyadarshi Kanungo² · Ganapati Panda² · Suresh Chandra Satapathy³ · Rohit Sharma⁴ 

Received: 5 September 2020 / Accepted: 13 April 2021
© The Author(s) 2021

Abstract

Lane detection (LD) under different illumination conditions is a vital part of lane departure warning system and vehicle localization which are current trends in the future smart cities. Recently, vision-based methods are proposed to detect lane markers in different road situations including abnormal marker cases. However, an inclusive framework for driverless cars has not been introduced yet. In this work, a novel LD and tracking method is proposed for the autonomous vehicle in the IoT-based framework (IBF). The IBF consists of three modules which are vehicle board (VB), cloud module (CM), and the vehicle remote controller. The LD and tracking are carried out initially by the VB, and then, in case of any failure, the whole set of data is passed to CM to be processed and the results are sent to the VB to perform the appropriate action. If the CM detects a lane departure, then the autonomous vehicle is driven remotely and the VB would be restarted. In addition to the proposed framework, an illumination invariance method is presented to detect lane markers under different light conditions. The simulation results with real-life data demonstrate lane-keeping rates of 95.3% and 95.2% in tunnels and on highways, respectively. The approximate processing time of the proposed method is 31 ms/frame which fulfills the real-time requirements.

Keywords Vehicle localization · IoT-based framework · Illumination invariance · Cloud module · Vehicle board

✉ Rohit Sharma
rohitapece@gmail.com
Safwan Ghanem
safwanghanem@gmail.com
Priyadarshi Kanungo
pkanungo@gmail.com
Ganapati Panda
ganapati.panda@gmail.com
Suresh Chandra Satapathy
suresh.satapathyfcs@kiit.ac.in

- ¹ Department of Computer Science and Engineering, C. V. Raman Global University, Bhubaneswar, Odisha 752054, India
- ² Department of Electronics and Telecommunication Engineering, C. V. Raman Global University, Bhubaneswar, Odisha 752054, India
- ³ School of Computer Engineering, KIIT Deemed to be University, Bhubaneswar, Odisha, India
- ⁴ Department of Electronics and Communication Engineering, Faculty of Engineering and Technology, SRM Institute of Science and Technology, NCR Campus, Delhi-NCR Campus, Delhi-Meerut Road, Modinagar, Ghaziabad, UP, India

Introduction and related work

The stumbling of drivers on the road is a vital risk factor in road safety. Therefore, driver assistance systems (DAS) are developed, implemented, and adopted by many manufacturers. The LD is a essential subsystem in DAS. The main component in vehicle localization is lane detection (LD) which is employed in the lane departure warning (LDW) system for DAS in autonomous vehicles. Many research papers have reported LD related problem statements; however, few of them have addressed the deployment of LD for the industrial purpose in an integrated framework. Using IoT is increasing recently which yields better performance and cheaper solutions for complicated real-life problems. Nevertheless, relying on one module to achieve road safety is not enough in practice. Instead, an alternative solution should be ready in place of device failure. To achieve this, a cloud computing environment is used to enhance the model robustness and to make the faster decisions. Cloud computing uses remote resources which saves the cost of servers and other equipment. In addition, hardware failures do not lead to data loss because of networked backups. The architecture of the

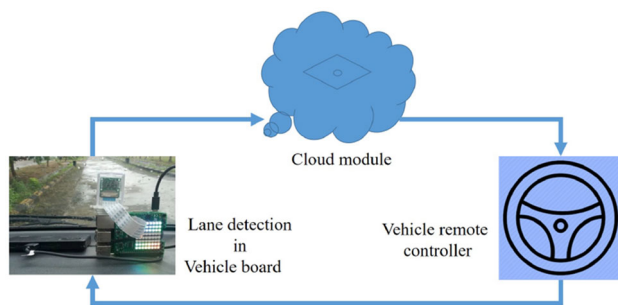


Fig. 1 Block diagram of the IBF

IoT-based framework (IBF) for LD and tracking is presented in Fig. 1. Simplified vanishing point detection method is employed in [1], and a scan-line method is applied to detect lane ridge features. Multi-LD algorithm that is robust to the challenging road conditions has been proposed in [2]. An adaptive threshold has been applied to extract strong lane features from images with obstacles and barely visible lanes. Then, an improved Random Sample Consensus algorithm has been introduced using the feedback from lane edge angles and the curvature of lane history to prevent false LD. Dynamic ROI extraction, edge detection, and Hough straight-line detection have been applied to extract the lane line in [3]. The model predictive control has been applied to track the extracted lane line and the front wheel steering angle has been corrected by the fuzzy controller based on the yaw angle and the yaw rate. Some reported articles have addressed the issue of LD and tracking using vision-based techniques without or with little knowledge of road geometry [4,5]. These techniques mostly depend on the color threshold and work well on highways and urban avenues under daylight or white lighting conditions. YCbCr color model has been used to focus on the most important visual information contained in the Y component, and to reduce time complexity [6]. In another method, the HSV color space is employed to achieve an 86.21% accuracy in detection rate [7]. Under daylight or white light conditions, the white lane markers and yellow lane markers almost preserve individual true colors. Therefore, global thresholding on different color plane in YCbCr or HSV color model efficiently segments the color planes to extract the lane markers. However, in the tunnel because of the color light, the color of the lane markers does not preserve their true colors. Hence, the global thresholding methods yield poor results in the segmentation of lane markers. Therefore, in this paper, an efficient LD method is developed to detect the lane markers efficiently in the tunnel either with artificial colored light as well as on highway under the daylight.

The general LD procedure starts with preprocessing the input frame to remove the perspective noisy information from the image. The preprocessing stage is followed by ROI iso-

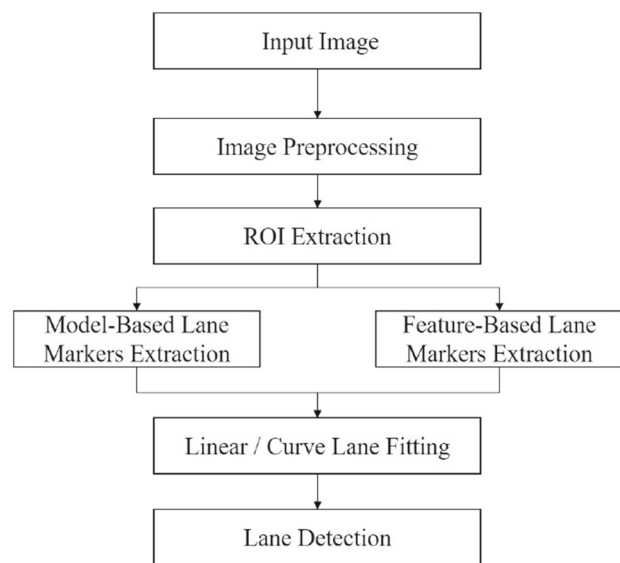


Fig. 2 Generalized flowchart for LD scheme

lation where lane markers are likely to be presented. Within ROI, extraction of features of lane marker is carried out. The lane marker detection stage is divided into two categories, feature-based and model-based. The feature-based LD methods and lane marker candidates are identified by some features like color, shape, and orientation. Whereas, in the model-based LD methods, search for lane markers systematically according to either the geometric measurements perceived in the road scene or the persistent information between successive frames. The candidate lane markers extracted from the previous stage are then validated to reduce the false-positive rate. Either linear or curved lane fitting is applied to validate lane markers as the output of the LD process. This output is used for many purposes including lane departure warning, automatic lane centering, and adaptive cruise control systems. In this paper, a model-based lane marker detection is proposed to address the problem arises due to the color light in tunnel. Figure 2 shows the generalized flowchart of LD. Image preprocessing is carried out to reduce the distortion and noise present in the captured image. Distortion is caused by the perspective effect when acquiring the image using normal cameras. To reduce the distortion, inverse perspective mapping (IPM) is employed to convert the input image into the bird's view image [2,8–11]. The IPM is also used in calculation of lane width between left- and right-lane boundaries [12]. Despite its efficiency, the IPM is sometimes avoided because of its significant processing time of around 35.3 ms/frame [13]. The LD potentiality is affected by noise due to different surrounding illumination sources, on-lateral-road objects, and weather conditions. A shadow invariant method is introduced [14] using the maximally stable extremal region (MSER)-based approach in the blue color channel as well as using Hough transform. Employing the

averaging of the pixel values of preceeding frames approach is introduced in [2] to improve the low-quality lane markers. Noise smoothing Gaussian filter to remove the noise of the mounted camera is dealt in [15]. The image pyramid approach is adopted in [16] to diminish the details and to present the high-frequency data. A four-level Gaussian pyramid model is employed to reduce the image dimensions and to make the edge drawing lines algorithm works effectively for the lowest resolution image at the top level of pyramids. In [17], the ROI containing lane markers is considered as the bottom third of the input image. The extraction of vanishing point (VP) is used in [2,8,12,14,18–22,26]. The VP is considered to be the point in which the most extracted lines of image are intersected. Adaptive ROI based on the longitudinal velocity changes of the vehicle is introduced [3] in which the upper boundary line moves down and up according to autonomous vehicle speed. The ROI extraction based on the minimum safe distance between the ego vehicle and the vehicle in front of it is proposed in [23]. In this paper, authors have reported that 150-pixel height is enough to cover 35 meters ahead when the vehicle speeds at 110 km/h. Generally, lane markers within the ROI are extracted using either predefined model or features. In feature-based LD scheme, lane marker candidates are initially defined by some features such as the orientations to x -axis [8], the width–height ratio [14], length, angles, and y -intercept in Hough transform [11]. The model-based LD techniques search for lane markers in a systematic method according to either the geometric measurements perceived in the road scene or the persistent information between successive frames. The temporal–spatial information matching method is suggested in [24]. In this paper, the top-view binary image is searched linearly for lane markers and, thereafter, the extracted markers are fitted using a cubic B-spline method. An improved version of the random sample consensus (RANSAC) algorithm is dealt in [10]. In this algorithm, two lane fittings, straight-line fitting model in the near field to the vehicle, and third-degree curve fitting for the far-field models are presented. The RANSAC is also employed in [12] to reduce the false-positive rate by removing the outliers from the lane candidates pool. A sliding window-based approach is used to extract the left- and right-lane markers using the highest peaks from the sliding window histogram for a horizontal scan line. The extracted lane markers are then connected using a polynomial fitting method and the final output is validated using a multi-sensor fusion method. A parallel constraint is applied [25] to an open snake model to detect broken lane markers. The lateral curvatures of the road are estimated in [15], and subsequently, the control points are extracted using vector fuzzy connectedness (VFC) technique. Road boundaries are built using a non-uniform B-spline interpolation method.

In few articles, the authors report the test results using their LD methods in tunnels [19,26]. A real-time illumination

invariant LDA (IILDA) using a third-order polynomial function of the longitudinal distance between vehicle and lane is proposed in [26]. In [26], it is reported that the average detection rate in the tunnel under daylight and in the night time is 91.17% in approximately 34.3 ms/frame. The lowest detection rate for this method observed when entering the tunnel is 87.4% because of sudden changing in illumination. A two-stage feature extraction (TSFE) method is proposed in [19] to detect two boundaries of lanes. To enhance robustness, lane boundary is taken as collection of small line segments. Thereafter, a modified Hough transform is applied to extract small line segments of the lane contour, which are then divided into clusters using the density based spatial clustering of applications with noise clustering algorithm. Then, the lanes are identified by curve fitting. False-negative test cases are reported in [19] when exiting from tunnel to the high way. A field-programmable gate array (FPGA)-based dual-stage lane detection (DSLDD) algorithm [31] to cope with real-world challenges such as cast shadows, occlusion of lane markers, brightness variations, wear, etc. In the first stage, Sobel operator and adaptive threshold are used to extract lane edges, followed by Hough transform to extract the road markers is proposed. The second stage of the algorithm operates on the original grayscale image and identifies stripe features near several candidate points with highest probabilities to find the landmarks. These extracted features are then used to detect the lane boundaries with high accuracy.

The review of the literature reveals that the color-based solution alone is not enough for correct LD. Under daylight or white light, the white and yellow lane markers almost preserve their true colors. Therefore, global thresholding on different color plane in YCbCr or HSV color model can efficiently segment the color planes to extract the lane markers. However, in the tunnel because of the color light condition, the color of the lane markers does not preserve their true colors. Hence, the global thresholding methods produce poor results in the segmentation of lane markers. To alleviate this problem, in this paper, an efficient LD method is proposed to detect the lane markers efficiently in the tunnel with artificial colored light as well as highway under daylight condition. The review indicates that none of the previous articles addressed this issue of development of a suitable LD framework using IoT and cloud computing techniques under artificial colored light in tunnels and on highways. It may be noted that in many real-life situations, artificial colored light in tunnel and highway are experienced by drivers. Therefore, the main objective of this research is to develop a concrete model using vision-based LD and tracking method and cloud computing which can be used for driverless tasks in smart cities. The proposed method is shown to be efficient when testing on tunnel traffic images captured by a vehicular onboard camera. Time complexity of the proposed method is found to be less than that of other reported methods.

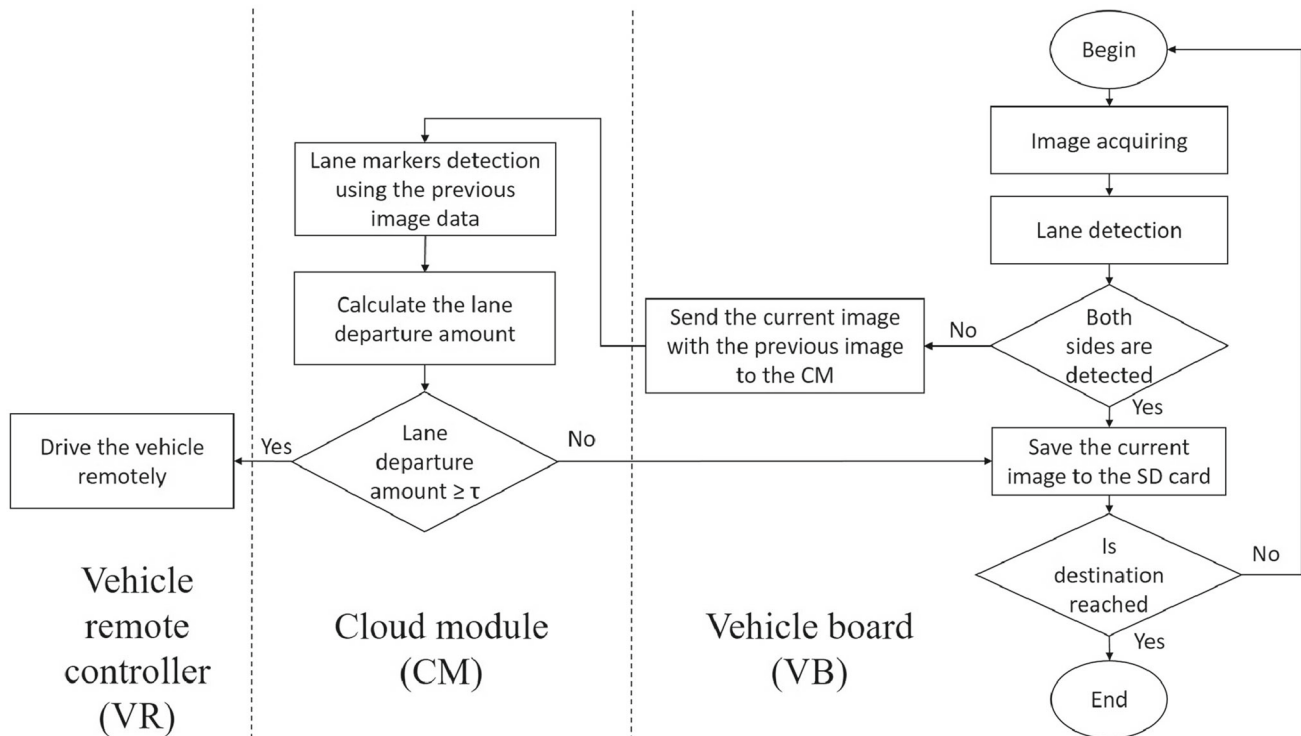


Fig. 3 Flowchart of the proposed IBF scheme

Therefore, the proposed method fulfills one of the real-time application requirements.

This section has dealt with the current state of vehicle localization. It also presents the motivation and the key objective of the current investigation. Sect. 2 details the proposed framework and its modules in detail. Sect. 3 reports the findings and contribution of the proposed framework as well as comparison with the results obtained by existing work. Conclusion and future extension of this work are presented in Sect. 4.

The proposed lane detection and tracking module on IoT framework

The architecture of the IBF for LD and tracking is illustrated in Fig. 1. Three modules connect each other using a 5G network to sustain safety while autonomous driving. The corresponding flowchart is shown in Fig. 3. The 5G mobile networks require an end-to-end latency of within 1 ms, (including the wireless section; the required one-way latency of the wired section, particularly in 5G mobile networks, is about 100 μ s). The camera in the VB module acquires the road image and passes it to the lane detection algorithm (LDA) which detects both sides of the lane and the captured image along with the localized lane markers are stored in the SD card for further processing. If LDA fails to

detect the equations for lane lines, then the current image accompanying the last stored image is sent to the CM where the current lane markers are detected using information from the last image. Thereafter, the amount of lane departure is measured, and if this value is bigger than a certain threshold, τ , then the overall control is delivered to man-powered remote which controls the vehicle in the vehicle remote controller (VR) module.

Vehicle board module

The IoT device to be employed in the implementation of the proposed scheme has the following specifications: Raspberry Pi 4 Model B, 1.5 GHz 64-bit quad-core Arm Cortex-A72 CPU, has three RAM options (2 GB, 4 GB, 8 GB), gigabit Ethernet, integrated 802.11ac/n wireless LAN, and Bluetooth 5.0. That enables the IoT module to execute the computational processes fast and to give a real-time decision. Raspberry Pi 4 Model B is installed in the vehicle, and it is chosen, because it decodes the video using the H265 standard and gives 4K P60 quality. Moreover, it can work at a high temperature up to 50° which enables installing it outside the vehicle compartment. The Raspberry Pi Camera Module v2 is attached with Raspberry Pi to enable video recording of 8MP for the traffic scene and to store them on the SD card to be processed by the LDA. It is the major part of the VB module and it does the LD tasks in tunnels and on the high ways.

The performance of the proposed algorithm is robust and does not get affected when illumination conditions change, like when colored artificial light presents.

The color-based thresholding is observed to have poor detection potentiality of lane markers under the colored light condition of the tunnel. However, the structural features are invariant to day light as well as colored light. Therefore, the knowledge of lane structure is considered to extract the structural features of the lanes in the proposed LD approach. The vanishing point (VP)-based method is used in LDA to identify the region of interest (ROI). Thereafter, lane markers are extracted using textural features that are extracted using the standard deviation filter. Lane markers are segmented using associated geometric characteristics, and then, these are clustered and fitted to find the formula of lateral lines for the lane. The output of the LDA is represented in two equations, one equation for each side of the lane, and if this output cannot be met, then the currently captured image along with the previous image is sent to the CM to be processed using LD using the cloud algorithm. The flowchart of the proposed structural feature-based LD algorithm is placed in Fig. 4. The details of LDA are explained in the following subsections.

The region of interest (ROI) extraction

The bottom half of the road image contains most lane segments, while the top half shows other objects not related to lane markers. The VP-based ROI extraction is a standard method in LD algorithms in which the ROI is considered to be the region under the VP. The VP is defined as the point in which the most extracted lines are intersected [26]. In this work, first, edges are extracted from the gray image by applying the Canny method because of its robustness against noise [27]. Second, the line segments are extracted using Hough transform. The 2-D accumulation array with the same size as the inputted raw image is used to get the VP coordinates. Each cell of the accumulative array is increased by 1 whenever it satisfies an extracted line equation. Finally, the cell with the maximum value is marked, and its indices are used to identify the VP coordinates. Figures 5 and 6 demonstrate the aforementioned steps.

Standard deviation filter

The standard deviation (SD) filter is a textural filter that provides information on the local intensity variation. The response of SD filter is smaller when the texture is smoother, and hence, the SD filter is used in this paper to act as an indicator of the degree of variability of pixel values in a region. This SD filter calculates the SD for the neighbor pixels to the pixel of interest. The SD at each pixel over a 3×3 neighborhood on each RGB color plane of the ROI is evaluated. The

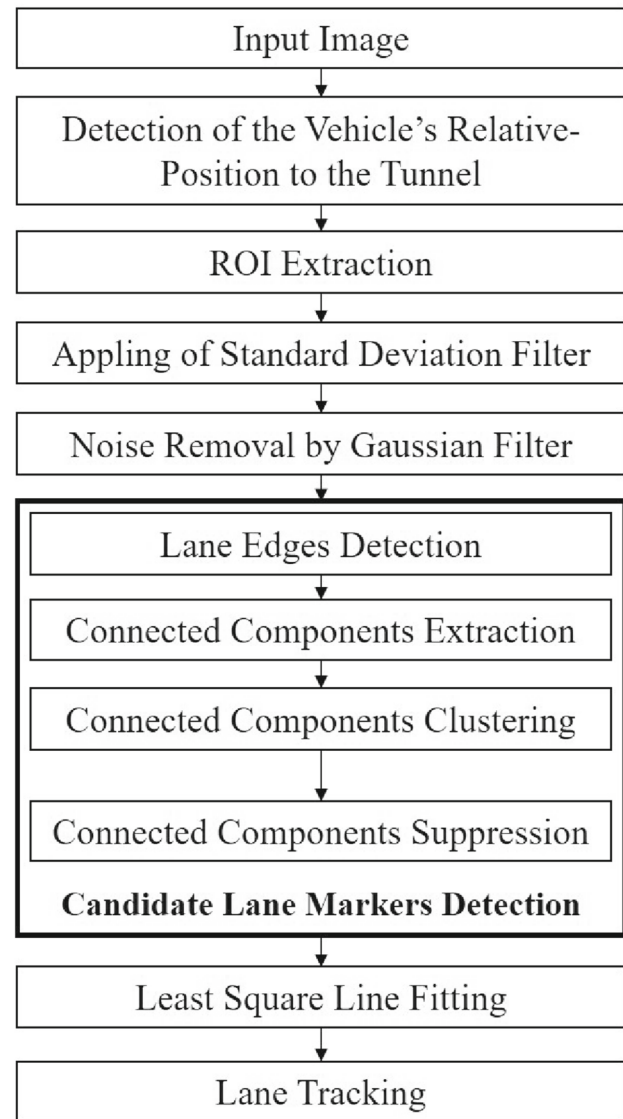


Fig. 4 Flowchart of the proposed LD algorithm

response of the SD filter at each pixel in a particular color plan is obtained using (1):

$$s_k(i, j) = \sqrt{\frac{\sum_{m=-1}^1 \sum_{n=-1}^1 [x_k(i+m, j+n) - \mu_k(i, j)]^2}{[(r \times c) - 1]}}, \quad (1)$$

where $k = 1, 2$, and 3 for color planes R, G, and B, respectively. The mean μ_k value is defined as given in Eq. (2)

$$\mu_k(i, j) = \frac{1}{9} \sum_{m=-1}^1 \sum_{n=-1}^1 x_k(i+m, j+n). \quad (2)$$

The symbol $x_k(i, j)$ represents the pixel at (i, j) position in the k th color plane. Based on the response of SD filter in each color plane, a monochromatic SD plane S_q is generated as in [28] using Eq. (3):

$$S_q = 0.2989S_1 + 0.5870S_2 + 0.11405S_3. \quad (3)$$

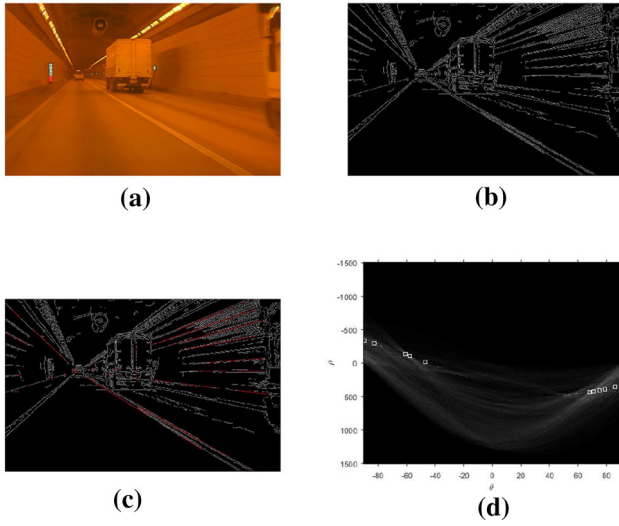


Fig. 5 Linear segments extraction using Hough transform. **a** Original tunnel image. **b** Canny edges. **c** Hough lines. **d** Hough transform

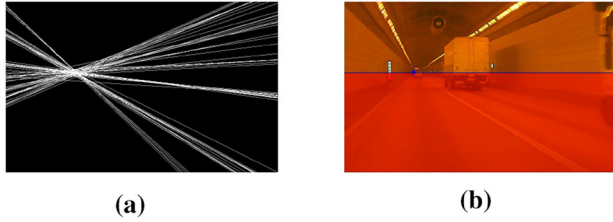


Fig. 6 Vanishing point extraction. **a** Accumulative map. **b** Extracted vanishing point

The S_q plane for highway ROI and tunnel ROI is shown in Fig. 7c, d. From these two figures, it is observed that the SD filter has a very significant response in smooth areas. This variation is suppressed using the Gaussian smoothing kernel of size 3×3 for $\sigma = 8$. The Gaussian mask and the associated weights for $\sigma = 8$ are given in Eqs. (4) and (5), respectively:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (4)$$

$$G_{(3 \times 3, 8)} = \begin{bmatrix} 0.1105 & 0.1114 & 0.1105 \\ 0.1114 & 0.1123 & 0.1114 \\ 0.1105 & 0.1114 & 0.1105 \end{bmatrix}. \quad (5)$$

The smoothed ROI is defined by Eq. (6) and is shown in Fig. 8:

$$S_{ROI} = G_{(3 \times 3, 8)} * S_q. \quad (6)$$

Lane edge detection

The lane markers appear generally in four directions, vertical, horizontal, primary, and secondary diagonal [24]. The

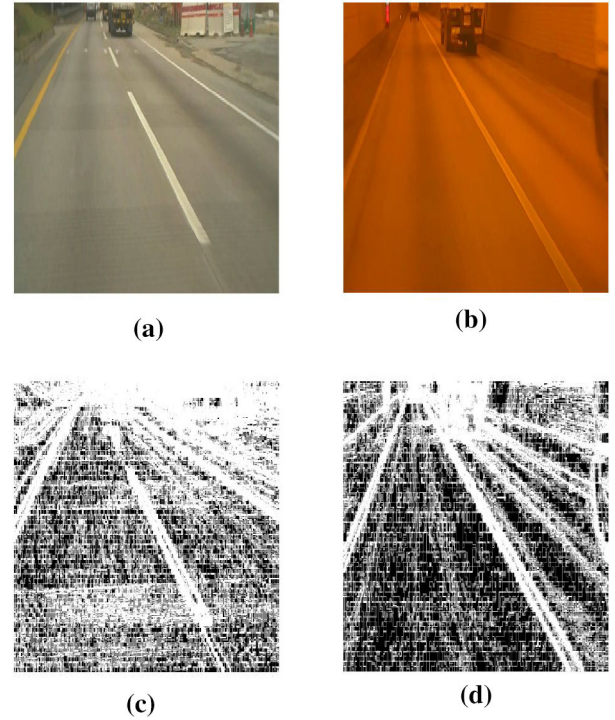


Fig. 7 The S_q plane for highway ROI and tunnel ROI. **(a)** ROI image on the highway. **(b)** ROI image in the tunnel. **(c)** Standard deviation filter results for **(a)**. **(d)** Standard deviation filter results for **(b)**

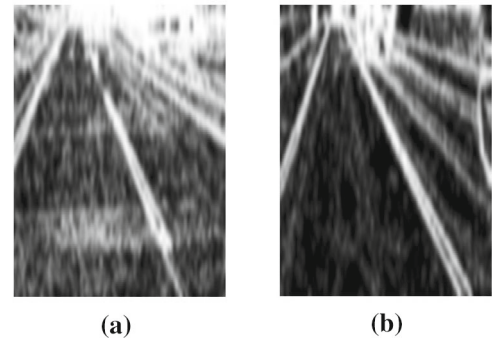


Fig. 8 The smoothed ROI. **(a)** The smoothed ROI on the highway. **(b)** The smoothed ROI in the tunnel

left and right lines for a lane are parallel in real world. Moreover, the perspective projection of these lines does not appear in horizontal or vertical direction in the image plane [29]. Instead, they tend to give high responses for 45° and 135° filter kernels [30,31]. Some other proposed methods search within an angle range for a better LD template matching [18,20]. Equation (7) shows the Sobel kernels A_{45° and A_{135° used to detect the lane markers

$$A_{45^\circ} = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}; \quad A_{135^\circ} = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix}. \quad (7)$$



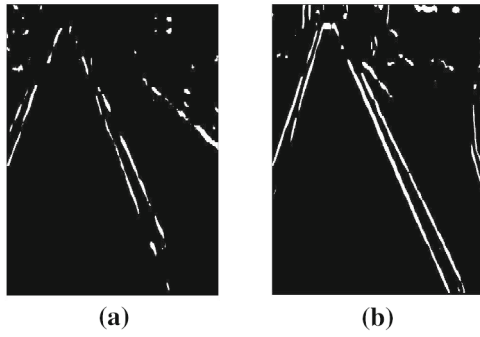


Fig. 9 The extracted 45° and 135° edges for the ROIs. (a) On the highway. (b) In the tunnel

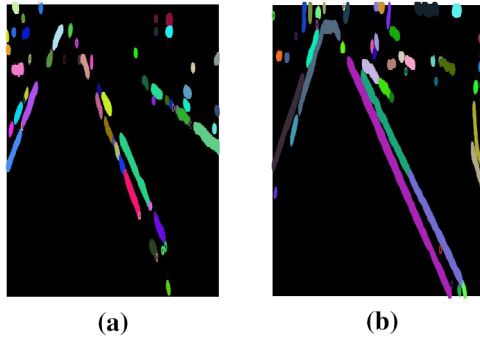


Fig. 10 The binary connected components for ROI images. (a) On the highway. (b) In the tunnel

The responses of these masks in the smoothed SD image are as in Eq. (8)

$$\begin{aligned}\omega_{45^\circ} &= A_{45^\circ} * S_{ROI} \\ \omega_{135^\circ} &= A_{135^\circ} * S_{ROI}.\end{aligned}\quad (8)$$

The resultant of ω_{45° and ω_{135° is given in Eq. (9):

$$R = \sqrt{\omega_{45^\circ}^2 + \omega_{135^\circ}^2}.\quad (9)$$

A threshold, α , is used to binarize the final image R using the condition given in Eq. (10):

$$M_{ij} = \begin{cases} 1, & \text{if } R(i, j) \geq \alpha \\ 0, & \text{otherwise.} \end{cases}\quad (10)$$

The value α in Eq. (10) is set as 70 which is obtained by trial and error basis. Figure 9 shows the left and right edges of lane along with some redundant edge points not belonging to lane edges of a highway and tunnel ROIs. These redundant edges are further reduced through the connected component clustering approach.

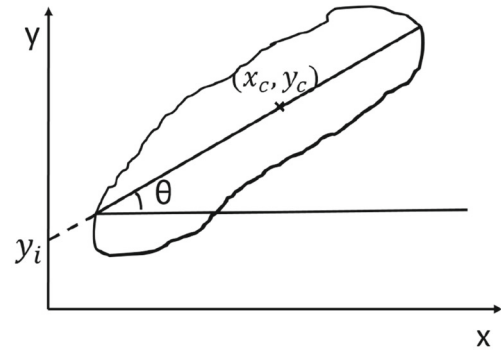


Fig. 11 θ and y_i properties of a component

Connected components detection and clustering

Binary connected components are detected using the 8-neighbor pixel connectivity. Figure 10a, b shows the different regions based on binary pixels in Fig. 9a, b. It is observed from these figures that the connected regions over the lane are discontinued. Moreover, other connected regions that do not belong to the lane would reduce the accuracy of the lane fitting stage. Consequently, these outliers would generate false lane markers which would reduce the safety level in autonomous vehicles. Therefore, there is a need to cluster the connected components to detect the candidate lane markers and minimize the outliers. Two parameters of each region, component angle θ and y-axis intercept y_i , are considered for the clustering the regions in Fig. 11. To calculate θ of a region, first, the center of the corresponding region (x_c, y_c) is calculated from Eq. (11), and then, the slope m of the longest diagonal line which passes through (x_c, y_c) is evaluated to measure θ as given in Eq. (13)

$$\begin{aligned}x_c &= \frac{\sum_{i=1}^n x_i}{n} \\ y_c &= \frac{\sum_{i=1}^n y_i}{n},\end{aligned}\quad (11)$$

where (x_i, y_i) refers to the i th pixel location in a region in line segment in Fig. 11, and n is the number of connected components of respective region. Let m be the slope of the largest diagonal line passing through (x_c, y_c) of a region, as shown in Fig. 11. This largest diagonal line can be defined as in Eq. (12):

$$y_i = y_c - x_c(\tan \theta).\quad (12)$$

Then, the parameter θ is given by (13):

$$\theta = \arctan(m).\quad (13)$$

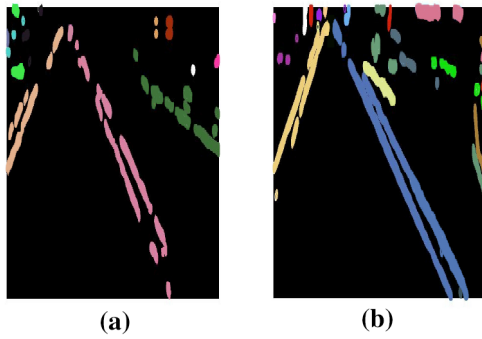


Fig. 12 Connected components clustering for ROI images. (a) On the highway. (b) In the tunnel

Regions satisfying both conditions of Eq. (14) are marked with same label:

$$\begin{aligned} |\theta_i - \theta_j| &\leq \varepsilon_1 \\ |y_i - y_j| &\leq \varepsilon_2, \end{aligned} \quad (14)$$

where ε_1 and ε_2 equal 0.035 and 0.016, respectively. It is observed from the Fig. 12 that all the clusters belonging to a particular lane mark have similar label. Outlier clusters not belonging to lane mark increase the false detection of lane mark. To suppress these outliers, the regions with the nearest orientation to 135° and 45° are considered as the candidate lane marker regions. Therefore, any region having an orientation between $(30^\circ$ and $60^\circ)$ or $(120^\circ-150^\circ)$ is considered as belonging to lane mark. Any region does not satisfy these two conditions is deleted. If none of the lane mark regions in Fig. 12 falls in the above orientation criteria, then information from the previous frame is compared with the under-processing frame.

Least-square line fitting

The least-square method finds the coefficient in such a way that the cost function (sum of the square of the deviations) between the data and the estimated ones are minimized. The lane markers generally take straight-line shapes within the ROI. Because it contains only a few portions of the road ahead of the vehicle [21]. Therefore, a polynomial fitting of first degree has been considered. Accordingly, the least-square approach has chosen to achieve the instantaneous curve line fitting. The polynomial fitting of the first degree considered in this work may cause double lines in each side of the lane. This problem cannot be avoided by minimizing ε_1 and ε_2 in Eq. (14) which may lead to increase of the number of clusters and, consequently, to a wrong LD. Therefore, the fitted lines with the nearest slope angle of 45° and 135° are considered as lane borders and the other fitted lines, if any, are discarded. Figure 13 shows the output of the line fitting algorithm and the final result after suppression.

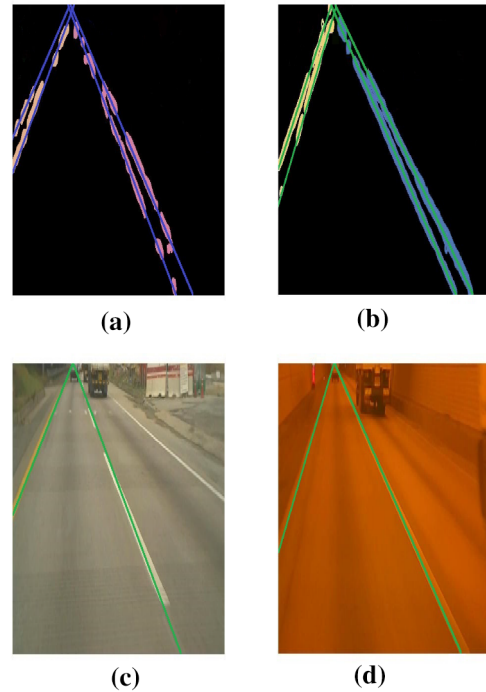


Fig. 13 The line fitting results and the final result after suppression . (a) Line fitting for ROI image on highway. (b) Line fitting for ROI image in tunnel. (c) Suppression of final lines for ROI image on highway. (d) Suppression of final lines for ROI image in tunnel

The proposed approach employs structural-based features instead of the colored light feature-based thresholding for the lane marker extraction process. The use knowledge of the lane geometry is more potential features compared to the color features in capturing the lane marker points. Therefore, the proposed structure-based features efficiently detect the lanes under colored as well as in day light conditions. Consequently, the road image is segmented well which leads to correct detection.

Lane tracking

Tracking helps in case of inaccurate detection or occlusion caused by imperfect lane markers or by a vehicle which makes lane departure at the time of image acquiring. The Kalman filter is employed in this work to perform the lane tracking process as it helps to converge to real values faster than other methods. It is observed from Eq. (15) that Kalman gain k_g becomes small if inaccurate detection happens which leads to large error, e_m , in measurement. Consequently, the estimated value for t th frame becomes approximately the same as it in the previous $(t - 1)$ th frame. That signifies that detection relies more on information from the previous frame compared to that of the current ones:

$$\begin{aligned}
 k_g &= \frac{e_s}{e_s + e_m} \\
 s_t &= s_{(t-1)} + k_g[m - s_{(t-1)}] \\
 e_{s(t)} &= (1 - k_g)(e_{s(t-1)}),
 \end{aligned} \tag{15}$$

where k_g is Kalman gain, e_s is the error in estimation, e_m is the error in measurement, and s_t is the estimated value in the current frame. $s_{(t-1)}$ is the estimated value for lane marker position in the previous frame, and the m is the measured position of lane marker in the current frame.

Cloud module and vehicle remote controller

The cloud module (CM) provides more reliability to the IBF, because the possibility of hardware failure is less than it in the VB. The main role of the CM is to differentiate between temporal failure and crucial failure. The crucial failure occurs when the vehicle has departed the lane which causes a non-safety problem for the ego vehicle and the other vehicles, as well. In such a case, the CM decides that VR should take over the driving remotely until the destination is reached. In case of VB failure, the CM receives two successive frames, f_1 and f_2 , and applies the LDA on f_2 . If the lane markers are not correctly detected, then the LDA is applied on the f_1 and the amount of departure, α , is calculated. The value of α is compared to a safe threshold, $\tau = 10$ cm; if $\alpha \leq \tau$, then the failure is classified as a crucial and the full control for driving the vehicle is immediately deliver to the VR. The VR controls the driving of the autonomous vehicle using a secure 5G connectivity.

Experimental results

Based on the visual perception, the proposed framework efficiently localizes the vehicle within the lane markers in all the images of Figs. 14 and 15. It is also noticed that the proposed LDA does not detect the lane markers perfectly in case of blurred vision, small-radius curvature, and if there is another object which has the same features as lane markers such as footpath. In Fig. 16a, the left-lane markers are not properly detected, which signifies that the method relies on the information of previous frames. According to Fig. 16b, it is observed that if lane markers are deeply curved, it produces a wrong detection in the far field from the ego vehicle. The cross-walk boundaries in Fig. 16c act like lane markers and hence lead to false detection. The performance of proposed LDA is tested using Caltech [32] and DIML [26] datasets. Caltech contains road images for traffic scene with 640×480 resolution. The DIML contains videos with 1280×800 resolution and 15 frames per second. Three tunnel scenarios are

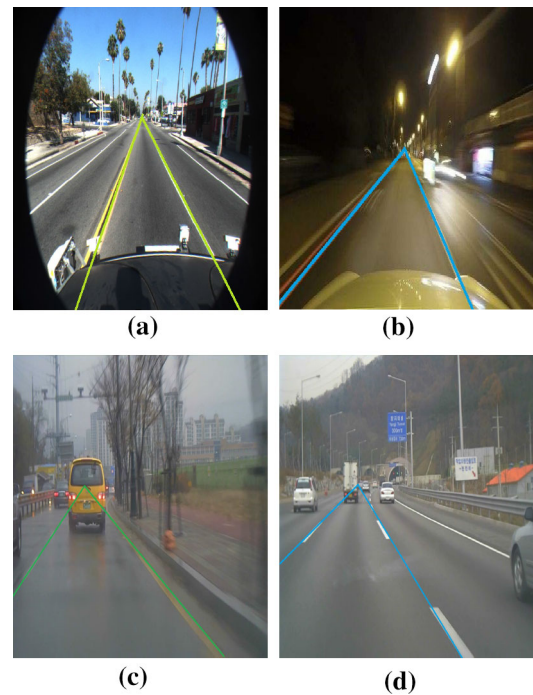


Fig. 14 LD results in different conditions. (a) Daylight. (b) Night time. (c) Rainy weather. (d) Occlusion

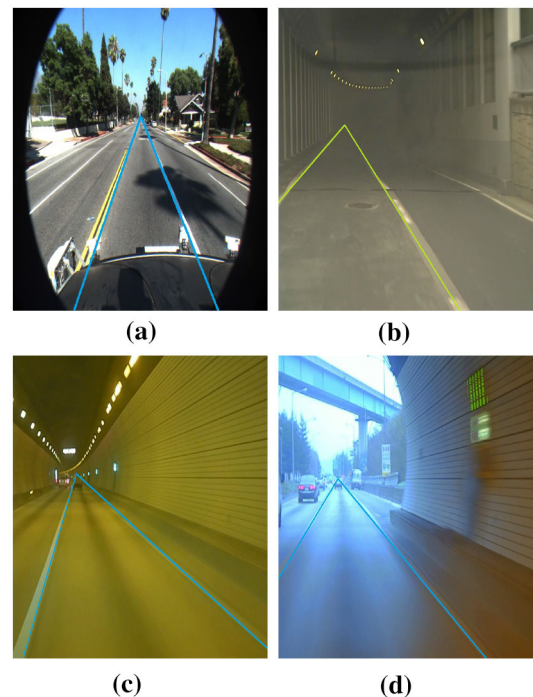


Fig. 15 LD results in different conditions. (a) Tree shadow. (b) Tunnel entering. (c) Inside tunnel. (d) Exiting from tunnel

considered under daylight and in the night time, entering, inside, and exiting of tunnel. The results of different tunnel scenarios of the proposed LDA and a comparison with

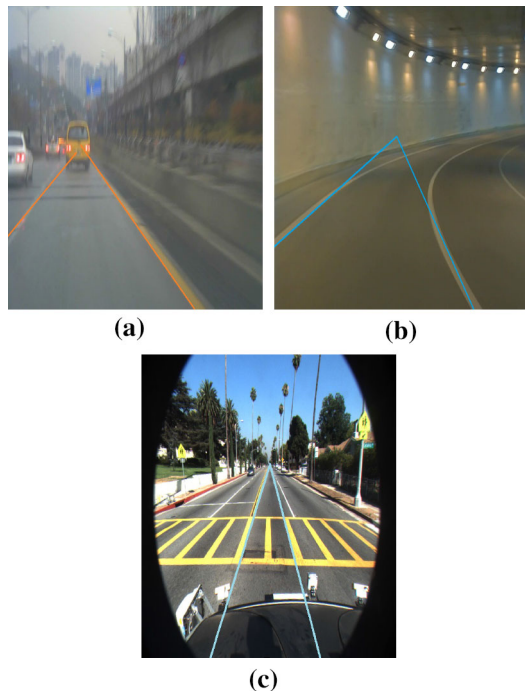


Fig. 16 Wrong detection cases . (a) Blurred vision. (b) Small-radius curved lane. (c) Cross-walk ahead the vehicle

Table 1 Comparison between the percentage detection rate in the proposed method and IILD [26] in tunnel scenes, DIML

Condition	Tunnel scenario	Proposed method	IILD [26]
Day light (1000 frames)	Entering	94.4	83.5
	Inside	96.7	93.2
	Exiting	96.1	86.7
Night time (790 frames)	Entering	92.1	91.3
	Inside	97.7	98.1
	Exiting	94.5	94.2
Average detection rate		95.3	91.2

Bold values indicate the best values between the compared values in one row

method reported in [26] are shown in Table 1. The correct detection rate is obtained, and the average detection time in millisecond per frame is calculated. The detection rate, R , is calculated using Eq. (16):

$$R = \frac{C}{N} \times 100\%, \quad (16)$$

where C is the number of true positive samples out of total N samples. The maximum detection time of the proposed method is 54 ms. Whereas, the average detection time of the proposed framework is found to be 31 ms/frame which fulfills the requirement of real-time application. The detection time depends on the number of redundant responses in (10)

Table 2 Comparison of detection rate of different methods

Condition	Proposed method	IILD [26]	DSLID [31]	TSFE [19]
Daylight	95.7	94	92.1	91.5
Night time	95.8	94.3	89.6	95.9
Rain	94.3	91	–	94.8
Tunnel	95.3	91.2	97.6	92
Caltech	94.9	93.6	93.8	96.5
Average	95.2	92.8	93.3	94.1

Bold values indicate the best values between the compared values in one row

due to noise or similar structure(s) in the ROI. If the number of redundant points increases, then the time required to extract the connected components and connected component clustering also increases. Furthermore, when the detection process fails it relies on the previous lane marking data. This information is provided by the detected lane markers from the previous frame. The time of normal LD is added to the time of tracking which makes the detection time longer. The detected lines have matched with the ground truth based on θ and y_i parameters. The findings from the proposed LDA are compared with those reported methods in [19,26,31]. Table 2 shows experimental results in different weather and lighting conditions. In the Caltech dataset, the scene is visualized through a circular window. Commonly, the camera captures the scene in a rectangular window. The proposed method when uses the circular vision of this dataset of the error in ROI extraction stage becomes high. However, the proposed method still produces improved LD results.

Conclusion and future work

A novel vehicle localization framework is proposed in this paper along with LD and tracking scheme. The framework consists of three modules. The VB module is the main module that runs the LDA using Raspberry Pi 4 Model B equipped with a compatible camera. The second module on cloud CM ensures the robustness of the VB and detects the lane markers in case of any failure in VB. Moreover, the CM measures the amount of lane departure which has been made by the vehicle due to the failure and provides this information to the VR stage, which finally controls the vehicle to save lives and properties. A novel and illumination invariance lane detection algorithm is also proposed. The vanishing point-based ROI extraction is employed to reduce the time complexity and to enhance detection accuracy. The input image is filtered to extract structural features using the standard deviation filter, followed by the Gaussian filter to reduce noise. The candidate lane markers are detected using 45° and 135° filter kernels for edge detection. The connected components are

clustered according to the cluster slope and y-intercept. The least-square lane fitting approach is used to form the left and right line equations for the lane. Finally, the lane is tracked using the Kalman filter. The experimental results demonstrate the robustness of the proposed LDA in tunnel scenarios and on highways, as well. The proposed framework is shown to be more efficient than other reported works. The average detection time fulfills real-time application requirements. But still, limitation and shortcomings exist in the proposed scheme. The fitting algorithm used does not work well in case of high curvature lane. The differentiation between left and right lines of the lane is required in DAS. Therefore, the proposed LDA needs further improvement to overcome this shortcoming. Enhancement of blurred vision which could happen because of shaking or under rainy weather conditions needs to be addressed and solved for achieving a robust and efficient vehicle localization framework. Furthermore, for ROI extraction and efficient detection of curved lane, Deep Learning methods such as RCNN, LSTM, and GAN can also be gainfully employed which can be take up as a future extension of the paper.

Declarations

Funding 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.

Conflict of interest The authors declare no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Jiao X, Yang D et al (2019) Real-time lane detection and tracking for autonomous vehicle applications. *Proc Inst Mech Eng Part D J Automob Eng* 233(9):2301–2311
- Son Y, Lee ES et al (2019) Robust multi-lane detection and tracking using adaptive threshold and lane classification. *Mach Vis Appl* 30(1):111–124
- Hu J, Xiong S et al (2020) Lane detection and trajectory tracking control of autonomous vehicle based on model predictive control. *Int J Automot Technol* 21(2):285–295
- Chen Y, Chen W, Wang X et al (2019) Learning-based method for lane detection using regionlet representation. *IET Intell Transp Syst* 13(12):1745–1753
- Cualain D, Glavin M, Jones E et al (2012) Multiple-camera lane departure warning system for the automotive environment. *IET Intell Transp Syst* 6(3):223–234
- Narote S, Bhujbal P, Narote A et al (2018) A review of recent advances in lane detection and departure warning system. *Pattern Recognit* 73:216–234
- Kim K, Yoo H, Song D (2017) Real time road lane detection with ransac and HSC color transformation. *J Inf Commun Converg Eng* 15(3):187–192
- Piao J, Shin H (2017) Robust hypothesis generation method using binary blob analysis for multi-lane detection. *IET Image Process* 11(12):1210–1218
- Wang J, Kong B, Mei T et al (2019) Lane detection algorithm based on temporal-spatial information matching and fusion. *CAAI Trans Intell Technol* 2(4):154–165
- Ding Y, Xu Z, Zhang Y et al (2017) Fast lane detection based on bird's eye view and improved random sample consensus algorithm. *Multimed Tools Appl* 76(21):22979–22998
- Zheng F, Luo S, Song K et al (2018) Improved lane line detection algorithm based on hough transform. *Pattern Recognit Image Anal* 28(2):254–260
- Goa J, Murphey Y, Zhu H (2019) Personalized detection of lane changing behavior using multisensor data fusion. *Computing* 101(12):1837–1860
- Li W, Qu F, Wang Y et al (2019) A robust lane detection method based on hyperbolic model. *Soft Comput* 23(19):9161–9174
- Kucukmanisa A, Tarim G, Urhan O (2017) Real-time illumination and shadow invariant lane detection on mobile platform. *J Real-Time Image Proc* 16:1–14
- Fang L, Wang X (2017) Lane boundary detection algorithm based on vector fuzzy connectedness. *Cogn Comput* 9(5):634–645
- Gamal I, Badawy A, Al-Habal A et al (2019) A robust, real-time and calibration-free lane departure warning system. *Microprocess Microsyst* 71:102874
- Wu P, Chang C, Lin C (2014) Lane-mark extraction for automobiles under complex conditions. *Pattern Recognit* 47(8):2756–2767
- Nguyen V, Kim H, Jun S et al (2018) A study on real-time detection method of lane and vehicle for lane change assistant system using vision system on highway. *Int J Eng Sci Technol* 21(5):822–833
- Niu J, Lu J, Xu M et al (2016) Robust lane detection using two-stage feature extraction with curve fitting. *Pattern Recognit* 59:225–233
- 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
- Wang H, Wang Y, Zhao X et al (2019) Lane detection of curving road for structural highway with straight-curve model on vision. *IEEE Trans Veh Technol* 68(6):5321–5330
- Lee C, Moon J (2018) Robust lane detection and tracking for real-time applications. *IEEE Trans Intell Transp Syst* 19(12):4043–4048
- Andrade D, Bueno F, Franco F et al (2018) A novel strategy for road lane detection and tracking based on a vehicles forward monocular camera. *IEEE Trans Intell Transp Syst* 20(4):1497–1507
- Wang Z, Wang W (2018) The research on edge detection algorithm of lane. *Eurasip J Image Vide* 2018(1):98
- Li X, Fang X, Wang C et al (2015) Lane detection and tracking using a parallel-snake approach. *J Intell Robot Syst* 77:597–609
- Son J, Yoo H, Kim S et al (2015) Real-time illumination invariant lane detection for lane departure warning system. *Expert Syst Appl* 42(4):1816–1824
- He T, Li X, Jiang Y (2014) Improved HT object detection algorithm based on canny edge operator. *J Multimed* 9(9):1089
- Lu J, Plataniotis KN (2009) On conversion from color to gray-scale images for face detection. In: *IEEE computer society conference on computer vision and pattern recognition workshops*. Miami, pp 114–119

29. Hajjouji I, Mars S, Asrih Z et al (2019) A novel FPGA implementation of hough transform for straight lane detection. *Int J Eng Sci Technol Technol* 23:274–280
30. Xiao J, Li S, Sun B (2016) A real-time system for lane detection based on FPGA and DSP. *Sens Imaging* 17(1):6
31. Malmir S, Shalchian M (2019) Design and FPGA implementation of dual-stage lane detection, based on hough transform and localized stripe features. *Microprocess Microsyst* 64:12–22
32. Aly M (2008) Real time detection of lane markers in urban streets. In: 2008 IEEE intelligent vehicles symposium. IEEE, pp 7–12

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.