Dear Author

Please use this PDF proof to check the layout of your article. If you would like any changes to be made to the layout, you can leave instructions in the online proofing interface. Making your changes directly in the online proofing interface is the quickest, easiest way to correct and submit your proof. Please note that changes made to the article in the online proofing interface will be added to the article before publication, but are not reflected in this PDF proof.

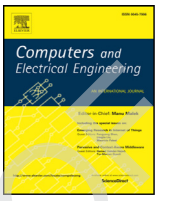
If you would prefer to submit your corrections by annotating the PDF proof, please download and submit an annotatable PDF proof by clicking here and you'll be redirected to our PDF Proofing system.

Computers and Electrical Engineering xxx (xxxx) xxx

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng



Lane detection technique based on perspective transformation and histogram analysis for self-driving cars ☆

Q1 Raja Muthalagu [a,*], Anudeepsekhar Bolimera [b], V. Kalaichelvi [b]

[a] Department of Computer Science, Birla Institute of Technology and Science Pilani, Dubai Campus, Dubai, United Arab Emirates
[b] Department of Electrical and Electronics Engineering, Birla Institute of Technology and Science Pilani, Dubai Campus, Dubai, United Arab Emirates

ARTICLE INFO

Article history:
Received 23 June 2019
Revised 16 March 2020
Accepted 2 April 2020
Available online xxx

Keywords:
Contemporary computer vision techniques
Perception algorithm
Self-driving cars
Lane detection
Perspective transformation

ABSTRACT

In this study, we present a perception algorithm that is based purely on vision or camera data. We focus on demonstrating a powerful end-to-end lane detection method using contemporary computer vision techniques for self-driving cars. We first present a minimalistic approach based on edge detection and polynomial regression which is the baseline approach for detecting only the straight lane lines. We then propose an improved lane detection technique based on perspective transformations and histogram analysis. In this latter technique, both straight and curved lane lines can be detected. To demonstrate the superiority of the proposed lane detection approach over the conventional approach, simulation results in different environments are presented.

1. Introduction

In today's world, whether we are transiting via public transport or walking on roads as pedestrians, a significant amount of time is being spent in our vehicles or interacting with other vehicles on the road. However, as the number of vehicles is increasing rapidly, fatalities from road accidents are also increasing. In such a world, road safety is of utmost importance. Fortunately, at the brink of the fourth industrial revolution as many scholars point out, transportation perhaps has seen the most significant advances in its technologies. Vehicles have moved from being just a mechanical locomotive to smart, fast wagons of luxury and comfort, and artificial intelligence is at the frontier of our most advanced technologies.

The safety and comfort of passengers have been the major driving forces for developing advanced driver assistance systems (ADAS) [1–3], which ultimately has led to the creation of fully autonomous self-driving cars. Self-driving car technology can be broken down into three major building blocks: perception, planning, and control. This study addresses developing and testing a perception algorithm that uses camera data and computer vision to help self-driving cars perceive the world around them.

Computer vision is at the core of perception algorithms, and cameras are the closest equivalent technology to how humans perceive the world around them. Though technologies such as Lidar [4] and radar [5] systems are used actively while developing perception technologies, cameras provide us with a powerful and cheaper way of extracting information about our environment. In this study, a robust lane detection algorithm that can determine the safe drivable region in front of a car is discussed in detail.

Q2
Q3 ☆ This paper is for regular issues of CAEE Reviews processed and recommended for publication to the Editor-in-Chief by Area Editor Dr. Huimin Lu.

* Corresponding author.
E-mail address: raja.sanjeeve@gmail.com (R. Muthalagu).

https://doi.org/10.1016/j.compeleceng.2020.106653
0045-7906/

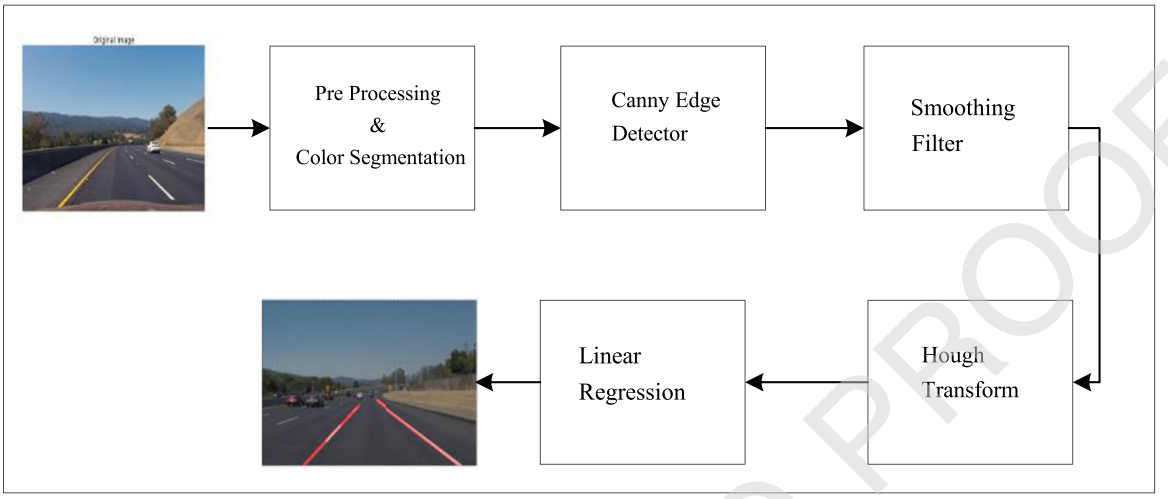


Fig. 1. Framework of the minimalistic lane detection approach.

The first building block of the vision perception module is lane detection. The task of navigation requires determining the position of the vehicle relative to the road. In the present work, a relatively primitive approach for detecting lanes and learning from its shortcomings has been analyzed to design a more robust lane detection algorithm that can both detect and predict the location of the vehicle with respect to the road itself. This approach emphasizes extracting relevant information from the environment using only a single red, green, and blue (RGB) camera fixed to the front of the vehicle as the source of data. A series of color-space explorations is used to isolate the lanes and proceed to detecting them using a Canny edge filter [6] and the Hough transform [7].

This paper is organized as follows: Section 2 describes the minimalistic approach for lane detection. Section 3 describes the advanced lane detection approach and preprocessing. Section 4 presents a novel improved technique based on perspective transformation and histogram analysis and also discusses proposed sanity check techniques. Section 5 gives experimental results and analysis of the proposed algorithm. A conclusion and future scope of work are given in Section 6.

2. Minimalistic approach for lane detection

2.1. Color-space exploration

The framework of the minimalistic approach for lane detection is shown in Fig. 1. In this study, color segmentation is used to detect objects or parts of the image that are of a particular color. At the end of the process, the algorithm processes only sections of the image that have the specified colors. This approach is particularly useful for extracting information about lane markings, because, around the world, lane markings are dominantly painted in either yellow or white. Fig. 2 illustrates the preprocessing performed on the image. The process involves loading the image, then converting it from its original RGB color space to a hue, saturation, and lightness (HLS) color space, and further splitting it into its color bands of hue, lightness, and saturation.

The HLS color space is an alternative representation of the RGB color space. It is conceptualized and organized based upon human visual perception, as shown in Fig. 3. The HLS color model is particularly useful, being more intuitive to use for color segmentation. Conversion from RGB to HLS color space is fast, with the computation occurring in real time. The RGB-to-HLS color-space conversion model can be described as follows:

$$V_{max} \leftarrow \max(R, G, B), \tag{1}$$

$$V_{min} \leftarrow \min(R, G, B), \tag{2}$$

$$L \leftarrow \frac{V_{max} + V_{min}}{2}, \tag{3}$$

$$S \leftarrow \begin{cases} \frac{V_{max} - V_{min}}{V_{max} + V_{min}} \ if \ (L < 0.5), \\ \frac{V_{max} - V_{min}}{2 - (V_{max} + V_{min})} \ if \ (L \geq 0.5), \end{cases} \tag{4}$$

$$H \leftarrow \begin{cases} (G - B)/(V_{max} - V_{min}) \ if \ (V_{max} = R), \\ 120 + 60(B - R)/(V_{max} - V_{min}) \ if \ (V_{max} = G), \\ 240 + 60(R - G)/(V_{max} - V_{min}) \ if \ (V_{max} = B). \end{cases} \tag{5}$$

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

Fig. 2. Example of preprocessing.

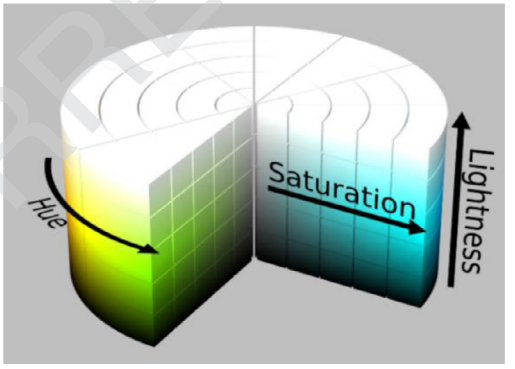


Fig. 3. HLS color model.

Table 1
Threshold values for color segmentation.

Yellow				White			
Channel	H	L	S	*Channel*	H	L	S
Upper	40	255	255	*Upper*	255	255	255
Lower	10	0	100	*Lower*	0	200	0

If $H < 0$ then $H \leftarrow H + 360$ on output $0 \leq L \leq 1$, $0 \leq S \leq i$, $0 \leq H \leq 360$, where L is lightness, S is saturation, and H is hue. To create an eight-bit image, the HLS values are scaled as follows:

$$V \leftarrow 255 \cdot V, \; S \leftarrow 255 \cdot S, \; H \leftarrow H/2. \tag{6}$$

Once the image is split into individual bands, the lane marking can be isolated by applying different threshold masking for the two colors, namely, yellow and white, individually. The thresholding applied is given in Table 1. Once the yellow and white lane markings are individually isolated, they can be combined to make the color segmentation more robust. The

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

Fig. 4. Original images.

Fig. 5. Images after color segmentation.

Fig. 6. Grayscale conversion.

combined masks filter the lane markings very clearly. Fig. 4 shows the original image that is loaded, and Fig. 5 shows the image after color segmentation is applied.

2.2. Edge detection, noise reduction, and line detection

One of the features of road lane markings is the sharp contrast between the painted lines and the road or any non-pavement surface. This contrast is represented by a large gradient in pixel intensities. The first step in detecting this gradient is converting the color-segmented image into grayscale, which is nothing but the representation of pixel intensities, and then using an edge detector to find these high gradients. In this algorithm, the Canny edge detector [8] was used based on its performance and ability to produce the best edge image. Fig. 6 shows some experimental results of grayscale-converted images. Fig. 7 illustrates the experimental results of detecting high gradients by using the Canny edge detector.

When taking into consideration any real-world system, noise is an inherent problem that has to be addressed. The processing algorithm must be designed to be noise-tolerant or the noise has to be eliminated by filtering. The noise present in

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

Fig. 7. Result of edge detection.

Fig. 8. Result of noise reduction.

our system will hinder the detection of the edges for the lane markings, and hence the removal of noise becomes a prerequisite to our edge detection. The areas where there is utmost certainty that the lanes will not be found can be treated as noise, and we can defined a region of interest (ROI) where the probability of finding the lanes is maximum and mask the reminder of the scene. This prevents the algorithm from detecting unnecessary features: hence, improving its performance. Another type of noise that hampers the performance of edge detection is the presence of sharp edges, which causes the algorithm to detect even minute objects that are of least significance but may cause the algorithm to diverge. To address this issue, a smoothing filter is used to smoothen the gradient, and only the prevailing gradients are detected in our edge detection algorithm. In Fig. 8, it can be seen that the search space is restricted to reduce noise as not all edges represent lane markings, and it is observed that most of the noise is effectively removed.

Once the edges are detected, the lines can be detected from these edges. For this, the Hough transform [9] with a restricted search space is used. The Hough transform converts the lines into a parameter space as a point, making it possible to associate a line in the image as a single point in the parameter space, and hence the lines can be detected, as shown in Fig. 9.

2.3. False positive and outlier detection

One of the key considerations while developing this algorithm was outlier removal and the detection of false positives. Though the reduction of the Hough search space tries to address the former issue, further processing can be performed by filtering out the outliers based on their slopes. This method is sufficient because it is fairly trivial that a horizontal line would not be an indicator of a lane marking. Hence, the detected lines are filtered based on their slopes and then separately grouped as left and right lane lines.

The next step in the algorithm is to find the dominant lines that have the maximum probability of being a part of the road lane markings. One of the two metrics used to filter out these lines is their length. The length of a line gives us a good measure of the certainty that the line belongs to the road lane markings because the lanes always appear to be straight and continuous nearer the vehicle. Hence, it is observed that long continuous straight lines have the maximum probability of marking the lane lines. The second metric is the slope, which was used in outlier filtration. After the most dominant lines

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

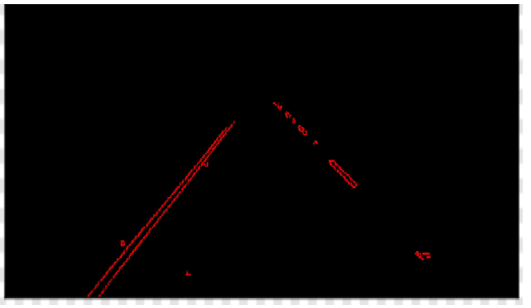


Fig. 9. Lines detected by hough transform.

Fig. 10. Result of minimalistic lane detection.

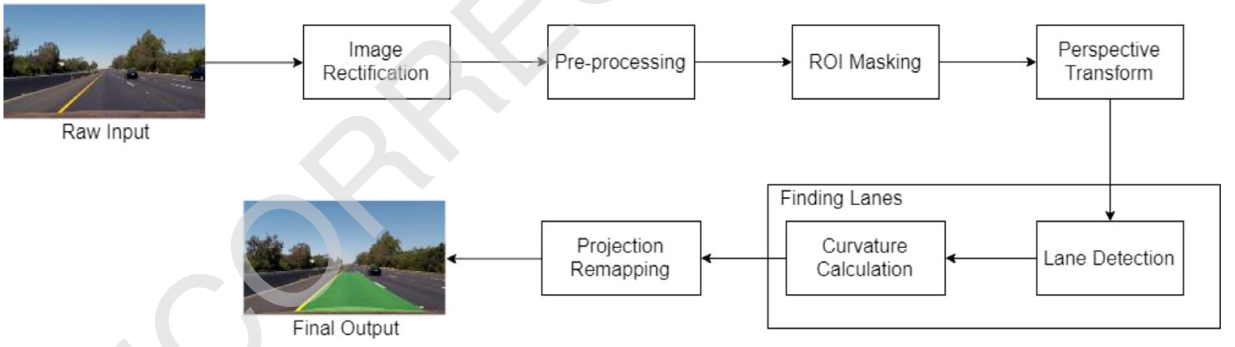


Fig. 11. Framework of the advanced approach for lane detection.

are determined, the slope for the lanes can be determined by using linear regression, which is our final goal as it helps us describe the lanes fully in the form $y = mx + c$. This approach can be used to define the detected lane markings.

As shown in Fig. 10, the algorithm was successful because the video images clearly show that the lane lines are detected properly and that lines are very smoothly handled. The algorithm only detects straight lane lines. Handling curved lanes (or the curvature of lanes) is an advanced topic, requiring the use of perspective transformation and fitting to curved lane lines rather than to straight lines. However, the lanes near the car are mostly straight in the images. The curvature appears only at further distance unless it is a steep curve. Therefore, this basic lane-finding technique is still very useful. Note that the algorithm will not work for steep (up or down) roads because the ROI mask is assumed from the center of the image. For use on steep roads, detection of the horizontal line (between the sky and the earth) is required so that we can tell how far the lines should extend.

3. Pre-processing for advanced lane detection

The framework of the advanced approach for lane detection is shown in Fig. 11. From the previously described minimalistic approach, the failure points were identified for the proposed algorithm. To improve upon these failure points, a more robust lane-finding algorithm has been proposed. The algorithm is optimized to detect the lanes and the region of safety for

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

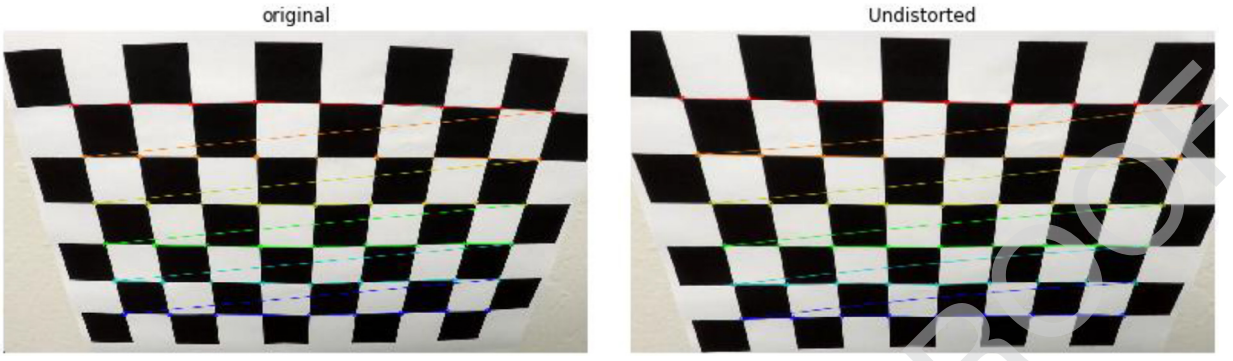


Fig. 12. Demonstration of camera calibration.

the autonomous vehicle. It also calculates the relative position of the ego vehicle with respect to the road lane markings. The improved robustness of the algorithm can be attributed to the following improvements.

3.1. Camera distortion correction

Lens distortion and occlusions are inherent properties of every camera and are errors that need to be addressed while developing a robust computer vision algorithm. In the present work, a camera calibration method described in [10] is used to calibrate our cameras before proceeding with further steps of the algorithm. This ensures consistency of the algorithm. Fig. 12 illustrates the difference between the original image and the undistorted image.

3.2. Expanded color segmentation

In contrast to the previously described approach of using a single color space (as explained in Section 2.1), a combination of two different color spaces is used in the present work: HLS and LAB (with L for lightness and A and B for the color dimensions). The two color spaces adversely detect yellow and white lines. Similar to the previous work, the present work is also focused on retaining the technique of binary thresholding and masking after each of the color spaces. Mathematical modeling of the RGB-to-LAB color-space conversion is illustrated as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \leftarrow \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \tag{7}$$

$$X \leftarrow X/X_n, \text{ where } X_n = 0.950456, \tag{8}$$

$$Z \leftarrow Z/Z_n, \text{ where } Z_n = 1.088754, \tag{9}$$

$$L \leftarrow \begin{cases} 116Y^{1/3} - 16 \ for \ (Y > 0.008856), \\ 903.3Y \ for \ (Y \le 0.008856), \end{cases} \tag{10}$$

$$a \leftarrow 500(f(X) - f(Y)) + \delta, \tag{11}$$

$$b \leftarrow 200(f(Y) - f(Z)) + \delta, \tag{12}$$

where

$$f(t) = \begin{cases} t^{1/3} \ for \ (t > 0.008856), \\ 7.787t + 16/116 \ for \ (t \le 0.008856) \end{cases} \tag{13}$$

and

$$\delta = \begin{cases} 128 & \text{for eight} - \text{bit images}, \\ 0 & \text{for floating} - \text{point images}, \end{cases} \tag{14}$$

The output ranges are as follows: $0 \le L \le 100, -127 \le a \le 127$, and $-127 \le b \le 127$, where L is the lightness from black (0) to white (100), a is from green (−) to red (+), and b is from blue (−) to yellow (+). With the new combined color-segmentation approach, the proposed algorithm is less susceptible to external environmental changes. Fig. 13 depicts the obtained results for the image expanded for color segmentation.

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

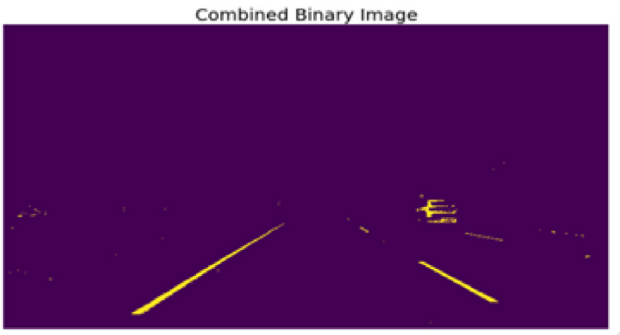


Fig. 13. Expanded color segmentation.

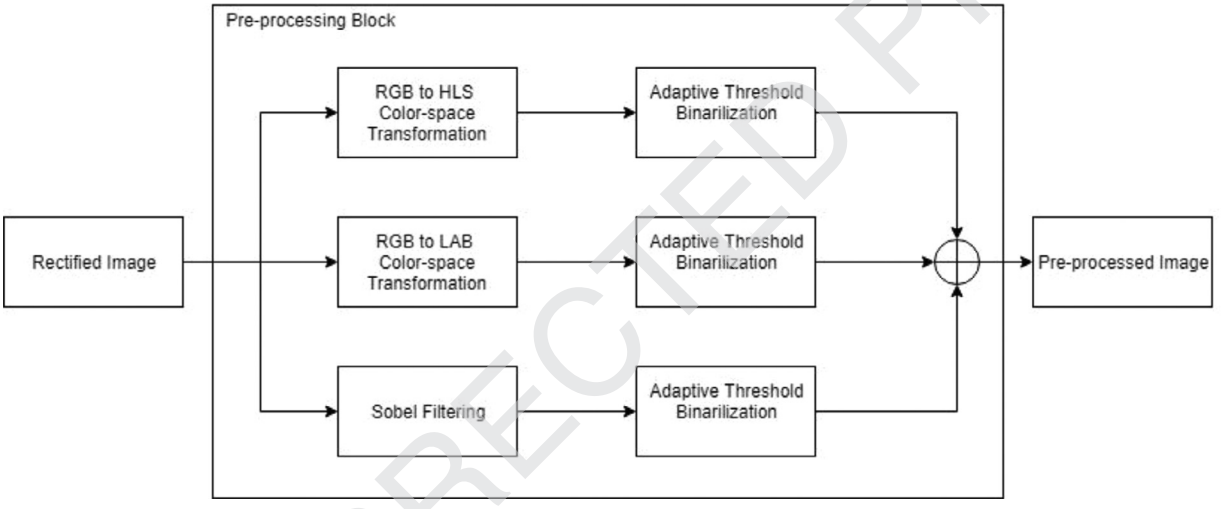


Fig. 14. Block diagram for the preprocessing of the proposed algorithm.

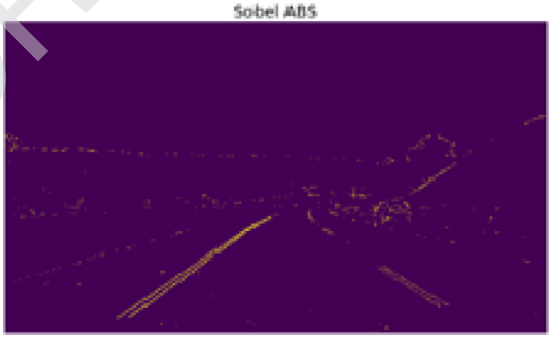


Fig. 15. Simple edge detection.

3.3. Combined edge- and color-based thresholding

Scenarios in which the lane markings are faded may occur frequently in rural or suburban areas. In such cases, simple color segmentation will not be sufficient for determining the lanes. As humans perceive road markings by the slightest of impressions of the faded lane markings, it is possible to highlight this feature by using edge detection. Fig. 14 illustrates the preprocessing framework used in the proposed algorithm. The algorithm uses a series of Sobel operators to detect the edges. By applying a smoothing filter to smoothen the gradient, only the prevailing gradients are detected in our edge detection algorithm, as shown in Fig. 15. A combination of the edge image with color-based thresholding gives us a more robust mechanism for detecting the lanes. Fig. 16. depicts the improved result of combining edge- and color-based thresholding methods. Finally, an ROI, where the probability of finding the lanes is maximum, is applied to mask the remainder of the scene and reduce the noise, as shown in Fig. 17.

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

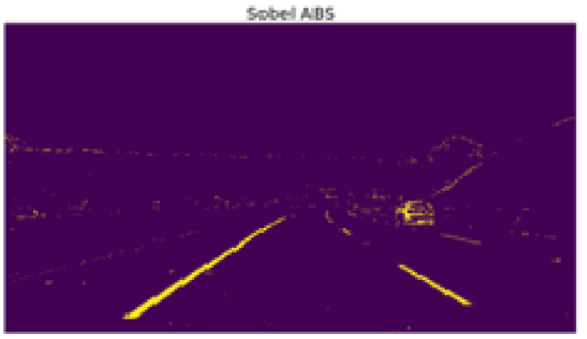


Fig. 16. Combined edge and color thresholding.

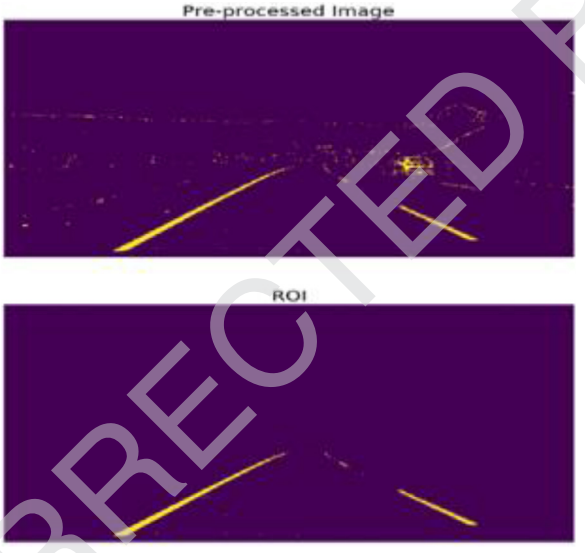


Fig. 17. Improved robustness by removing noise.

Table 2
Mapping coordinates for perspective transformation.

Mapping Coordinates	
Source Image	Destination Image
[190, 700]	[380, 720]
[580, 450]	[380, 10]
[730, 450]	[950, 10]
[1160, 700]	[950, 720]

3.4. Perspective transformation

The algorithm discussed in Section 2 failed to detect curved lanes, and that is a significant flaw in its performance. Here a method for lane detection is proposed that uses a perspective transformation to improve lane detection. Certain features can be detected with higher accuracy when looked upon from a different perspective. The curvature of the road lane markings is such a feature. The proposed technique applies a perspective transform that transforms the undistorted image to a "birds-eye view" of the road, which focuses only on the lane lines and displays them in such a way that they appear to be relatively parallel to each other. This will make it easier, later on, to fit polynomials to the lane lines and measure the curvature.

Table 2 lists the mapping coordinates for the perspective transformation. Fig. 18 shows the bird's-eye view of the original image after the perspective transformation [11]. We calculate the matrix of a perspective transform such that

$$\begin{bmatrix} t_i x_i' \\ t_i y_i' \\ t_i \end{bmatrix} = M \cdot \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}, \tag{15}$$

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

Fig. 18. Perspective transformation of the original image.

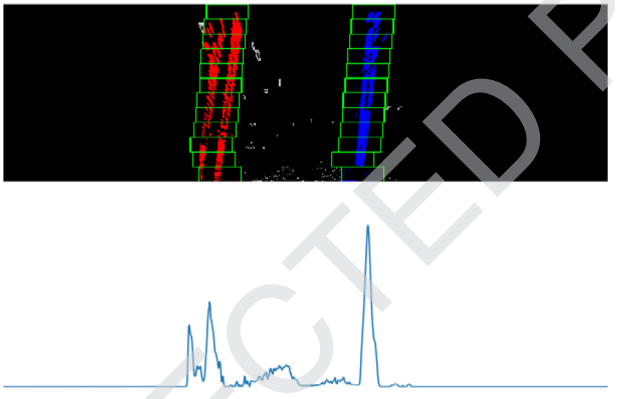


Fig. 19. Visualization of sliding window search method.

where $(x'_i, y'_i), i = 0, 1, 2, 3$ are the coordinates of the corresponding quadrangle vertices in the destination image and $(x_i, y_i), i = 0, 1, 2, 3$ are the coordinates of quadrangle vertices in the source image. We then use the calculated [3 × 3] matrix M to transform the image as follows:

$$dst(x,\ y) = src\left(\frac{M_{11}x + M_{12}y + M_{13}}{M_{31}x + M_{32}y + M_{33}},\ \frac{M_{21}x + M_{22}y + M_{23}}{M_{31}x + M_{32}y + M_{33}}\right), \tag{16}$$

where $dst(x,\ y)$ are the coordinates of pixels in the transformed image and $src(x,y)$ are the coordinates of pixels in the source image.

4. Improved technique based on perspective transformation and histogram analysis

4.1. Sliding window search methods

The first step in accurately detecting the lane marking is to identify the maximum probability region of its existence. The maximum probability region can be determined by observing the histogram of the bird's-eye view, which produces two distinct peaks, one for the left lane and the other for the right. Now the lanes can be identified by using the sliding window search approach [12–14] to identify the regions of the frame with the highest density of nonzero pixels, as shown in Fig. 19. Once these regions are identified, a third-order spline is fitted over these identified points of maxima. The spline marks the detected lane. The sliding window search algorithm slides the windows through the entire frame each time, which is ineffective and makes the algorithms slow and computationally heavy but provides the best chance of accurately detecting the lane markings. A slightly modified version of the sliding window search is used to handle the issue of repeated computation. Because the sliding window search already identifies the first lane, the lane in consecutive frames is likely to lie in the region close to the initial spline, and it is sufficient to search this region alone to detect the lane. This approach significantly reduced the search time and improved the algorithm's performance, as shown in Fig. 20.

4.2. Proposed sanity check techniques based on multilevel false positive detection mechanism

As with any computer vision algorithm developed, there is always a case of false positives adversely affecting the performance of the algorithm. This calls for an intensive false positive detection mechanism to ensure the highest levels of ac-

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

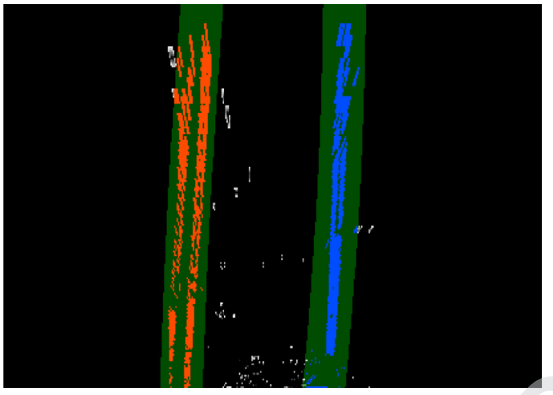


Fig. 20. Visualization of neighboring search method.

Fig. 21. Result of the proposed lane-finding algorithm.

curacy and performance. A multilevel false positive detection mechanism that checks for the sanity of each detected spline and then determines the best-fitting spline has been proposed.

At the highest level, the algorithm checks for any detection of the spline, and, based on the result of the algorithm, the search mechanism is selected. As discussed earlier, this improves the speed and accuracy of the algorithm. Then it performs a high-level sanity check as follows:

1. At a point closest to the car in the observable frame, lane markings maintain a consistent distance between them. Therefore, we check whether the intercepts of the detected splines are separated by approximately the correct distance horizontally. If so, the splines are detected as the lane marking. Any set of detected splines not satisfying this condition is not considered as lane markings, and the detected splines are discarded.
2. A secondary sanity check is conducted to ensure consistency of the detection. The curve of the most recent detected spline is compared to the best-fitting spline, and, if it is within acceptable limits, the best-fitting spline is updated. The best fit is calculated by averaging the five most recent detection updates and represents the lane detected.

Once lanes are detected and passed through the sanity checks, the radius of curvature and position of the vehicle are calculated. The detected lane boundaries are transformed back onto the original image and the visual display of the lane boundaries is output, thus completing numerical estimation of lane curvature and vehicle position. The final result of the proposed perspective transformation and histogram-based lane-finding algorithm is shown in Fig. 21.

5. Experimental results and analysis

5.1. Performance comparison of proposed and conventional algorithms

The result of comparing the proposed lane-finding algorithm with the conventional algorithm [15] is shown in Fig. 22. The proposed algorithm is able to detect lanes successfully even when the road surface changes, and we can observe that the minimalistic lane-finding algorithm fails in scenarios where there is a change in the road surface, as shown in Fig. 22. The above example demonstrates the robustness of the proposed algorithm to changes in lighting conditions and road surface. The combination of proposed sanity checks and filtering techniques prove to be more robust when compared to the existing

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

Fig. 22. Result of the proposed lane-finding algorithm compared with that of the conventional algorithm.

Table 3
Processing time analysis of the proposed system (in Milliseconds).

Image Rectification	11.4
Preprocessing	29.91
ROI Masking	1.33
Perspective Transformation	4.41
Finding Lanes	12.12
Projection Remapping	4.48
Total	63.65

minimalistic approach. The improved perspective-transform-based approach improves the robustness of the algorithm so it can detect curved roads with greater accuracy than the previously described minimalistic approach.

5.2. Process time analysis of the proposed algorithm

The processing time of our proposed system is given in Table 3. The system was tested on a sample video of 1280×720 pixel (HD quality) resolution on a system with an Intel Core i5 (6th generation) CPU with 8 GB of RAM and no GPU acceleration. The algorithm time analysis data demonstrate that the proposed algorithm is effective and only takes 63.65 ms on average. That averages to 15.7 fps in real-world situations. Stepwise time analysis data of the proposed algorithm over the number of frames and time distribution are illustrated in Fig. 23.

5.3. Performance of the proposed algorithm in different environmental conditions

To further experiment and establish the robustness and effectiveness of the algorithm, the proposed algorithm for lane detection was tested on data from two different datasets [16] with varying environmental conditions, as illustrated in Fig. 24. It can be observed from the simulation that the proposed algorithm was successfully able to detect the lane and the safe drivable area for the ego vehicle successfully.

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

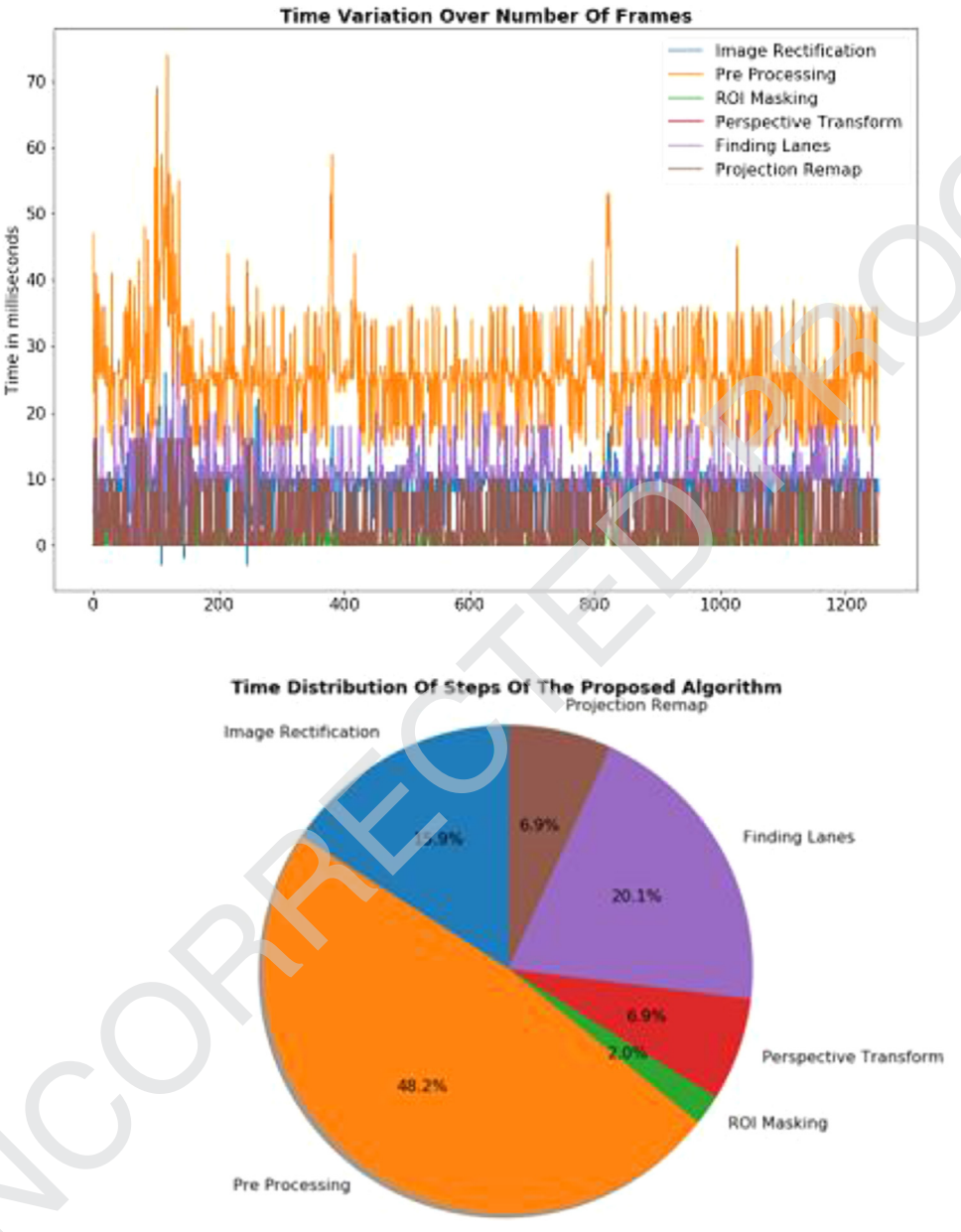


Fig. 23. Stepwise time analysis data from the proposed algorithm.

5.4. Accuracy of the proposed algorithm

Fig. 25 shows a comparison between the experimental results of the proposed algorithm and the ground truth labels, where results of the algorithm were tested on a large dataset. The root mean square error (RMSE), given by

$$\text{RMSE} = \sqrt{\frac{\sum_{i=0}^{N} (dist(p_i - g_i))^2}{N}}, \tag{17}$$

where p_i is the target value, g_i is the predicted value, and N is number of samples, was used to determine the variation of the lane determined using the proposed algorithm from the ground truth values. Fig. 26 gives the comparison between the RMSE in left and right lanes detected by using the proposed algorithm from the ground truth frame-wise. The simulation results gave a mean RMSE value of 5.5126 with an accuracy of 95.75%.

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

Fig. 24. Results of the proposed lane-finding algorithm compared in different environmental conditions.

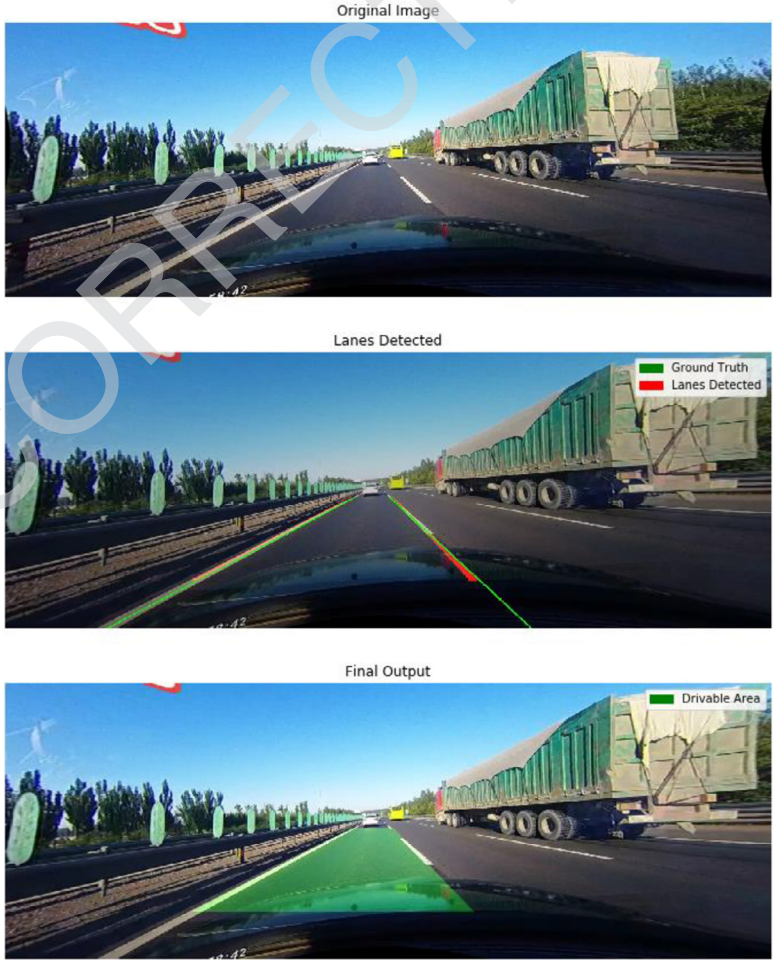


Fig. 25. Comparison of experimental results from the proposed algorithm to ground truth labels.

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

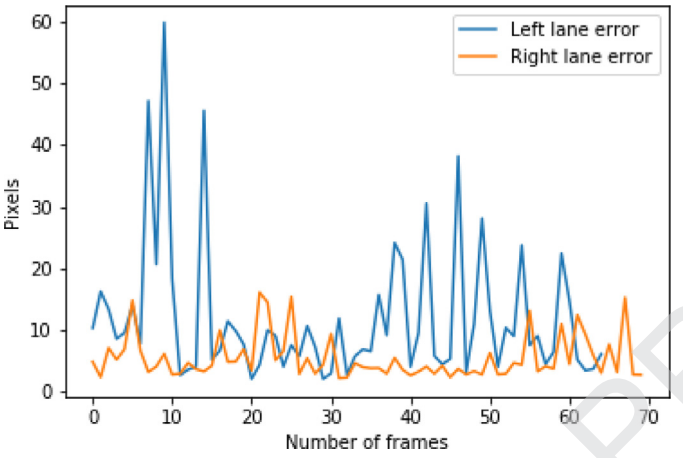


Fig. 26. RMSE result of lanes detected vs ground truth.

5.5. Limitations and shortcomings

Although the proposed system detects the lanes with high accuracy and is robust when compared to minimalistic approaches, there are certain limitations of this algorithm that need to be addressed. The algorithm heavily depends on preprocessing for achieving a reasonable detection rate, as explained in the earlier sections, and it is the most computationally expensive step of the proposed algorithm. This provides us with a scope to improve the efficiency and robustness of the algorithm significantly by reducing its dependency on heavily preprocessed images. In some cases, such as when there are very steep curves in the foreground, the algorithm may fail to detect the lanes because they disappear from the defined ROI, which is another limiting factor of the algorithm. However, in 95% of the cases, the lanes would always appear in the ROI and the algorithm can detect them effectively.

6. Conclusion and future work

Although the minimalistic lane-finding algorithm was successful in detecting straight lane markings, it cannot be used for curved and steep lane markings. Consequently, we have proposed the use of perspective transformation and a histogram-based search to detect curved or steep lane markings. The shortcomings of the minimalistic approach were overcome with the advanced lane-finding approach, which is more robust and less susceptible to different environmental conditions. By observing the simulation results, we can conclude that the proposed advanced lane detection approach is better than the conventional techniques. This work can be extended for big data and cloud environments, which requires intelligent techniques using neural networks and deep learning.

CRediT authorship contribution statement

Raja Muthalagu: Conceptualization, Methodology, Writing - original draft, Software, Validation, Formal analysis, Writing - review & editing. **Anudeepsekhar Bolimera:** Data curation, Methodology, Software, Validation, Formal analysis, Writing - review & editing. **V. Kalaichelvi:** Visualization, Investigation, Software, Validation, Writing - review & editing.

References

[1] Lu M, Wevers K, Heijden RVD. Technical feasibility of Advanced Driver Assistance Systems (ADAS) for road traffic safety. Transp Plan Technol June. 2005;28(3):167–87.
[2] Bengler K, Dietmayer K, Farber B, Maurer M, Stiller C, Winner H. Three Decades of Driver Assistance Systems: Review and Future Perspectives. IEEE Intell Transp Syst Mag 2014;6(4):6–22.
[3] Abraham H, Reimer B, Mehler B. Advanced Driver Assistance Systems (ADAS): a consideration of driver perceptions on training, usage & implementation. In: Proceedings of the human factors and ergonomics society annual meeting, 61; 2017. p. 1954–8.
[4] Domínguez R, Onieva E, Alonso J, Villagra J, González C. LIDAR based perception solution for autonomous vehicles. In: Proceedings of the eleventh international conference on intelligent systems design and applications; 2011. p. 790–5.
[5] Rasshofer H, Gresser K. Automotive radar and lidar systems for next generation driver assistance functions. Adv Radio Sci May 2005;3:205–9.
[6] Yan X, Li Y. A method of lane edge detection based on Canny algorithm. In: Proceedings of the Chinese automation congress (CAC); 2017. p. 2120–4.
[7] Talib ML, Rui X, Ghazali KH, Mohd Zainudin N, Ramli S. Comparison of Edge Detection Technique for Lane Analysis by Improved Hough Transform. In: Lecture notes in computer science, 8237. Cham: Springer; 2013. p. 176–83.
[8] Yan X, Li Y. A method of lane edge detection based on Canny algorithm.. In: Proceedings of the Chinese automation congress (CAC); 2017. p. 2120–4.
[9] Amaradi P, Sriramoju N, Dang Li, Tewolde GS, Kwon J. Lane following and obstacle detection techniques in autonomous driving vehicles. In: Proceedings of the IEEE international conference on electro information technology (EIT); 2016. p. 0674–9.
[10] Assidiq AA, Khalifa OO, Islam MR, Khan S. Real time lane detection for autonomous vehicles. In: Proceedings of the international conference on computer and communication engineering; 2008. p. 82–8.

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653

[11] Huang Y, Li Y, Hu X, Ci W. Lane detection based on inverse perspective transformation and Kalman filter. KSII Trans Internet Inf Syst 2018;12(2):643–61.
[12] Wang Z, Li X, Jiang Y, et al. swDMR: a sliding window approach to identify differentially methylated regions based on whole genome bisulfite sequencing. PLoS One 2015;10(7).
[13] Yi SC, Chen YC, Chang CH. A lane detection approach based on Intelligent Vision. Comput Electr Eng 2015;42:23–9.
[14] Wu BF, Huang HY, Chen CJ, Chen YH, Chang CW, Chen YL. A vision-based blind spot warning system for daytime and nighttime driver assistance. Comput Electr Eng 2013;39(3):846–62.
[15] Assidiq AA, Khalifa OO, Islam MR, Khan S. Real time lane detection for autonomous vehicles. In: Proceedings of the international conference on computer and communication engineering; 2008. p. 82–8.
[16] Pan X, Shi J, Luo P, Wang X, Tang X. Spatial as deep: Spatial CNN for traffic scene understanding. In: Proceedings of the thirty-second AAAI conference on artificial intelligence; 2018.

Raja Muthalagu received his Ph.D. in Wireless Communication from the National Institute of Technology (NIT), Tiruchirappalli, India, in 2014. He joined the Birla Institute of Technology Pilani, Dubai campus, United Arab Emirates, in 2015, where he is currently working as an Assistant Professor at the Department of Computer Science. His research interests include orthogonal frequency division multiplexing, multiple-input multiple-output systems, network security, and computer vision.

Anudeepsekhar Bolimera received his B.E. in Electronics and Communication Engineering from the Birla Institute of Technology and Science, Dubai campus, United Arab Emirates. He is a budding entrepreneur, a self-driving car engineer, and an independent researcher in the field of computer vision and robotics. His research interests include deep learning, perception for self-driving cars, autonomous robotic systems, and computer vision.

V.Kalaichelvi received her Ph.D. in Instrumentation Engineering from Annamalai University, India, in 2007. She joined the Birla Institute of Technology and Science Pilani, Dubai Campus, United Arab Emirates, in 2008, and she is currently working as an Associate Professor in the Electrical and Electronics Engineering department. Her research interests include intelligent techniques applied to robotics, computer vision, mechatronics, and process control.

Please cite this article as: R. Muthalagu, A. Bolimera and V. Kalaichelvi, Lane detection technique based on perspective transformation and histogram analysis for self-driving cars, Computers and Electrical Engineering, https://doi.org/10.1016/j.compeleceng.2020.106653