Lane Departure Detection Using Geometrical and Intensity Patterns

Zahra Rahimi Afzal[1], Behzad Nazari[2], Saeid Sadri[3]

[1]Computer Science Department, Kansas State University, zrahimi@ksu.edu

[2]Electrical and Computer Engineering Department, Isfahan University of Technology, nazari@cc.iut.ac.ir

[3]Electrical and Computer Engineering Department, Isfahan University of Technology, sadri@cc.iut.ac.ir

Abstract- **A lane departure warning system (LDWS) is an essential part of an intelligent transportation system. This paper proposes a novel low-complexity LDWS that detects lane departures in video frames captured by smart phones with various lighting conditions and lane types and complicated road surfaces. The car used in the research was assumed to be traveling on a mostly straight road or highway signed with lane markings, and left and right lane markings were expected in fixed regions of frames. The Canny edge detector detected all the edges, allowing extraction of connected edge components. Left and right lane markings were selected from these components according to the position, orientation, and pixel intensity pattern. The presence or absence of lane markings in some consecutive frames was used to detect lane departure. This algorithm operated in real time and was successfully implemented on a tablet.**

Keywords— ***Lane departure detection, Lane departure warning system, Driver assistance, Connected component***

I. INTRODUCTION

Intelligent transportation systems, or driver assistance systems use lane detection to implement lane departure warnings, Lane Keeping Assist technology, lateral control, and collision warnings in order to decrease vehicle accidents and fatalities. A lane departure warning system (LDWS) warns a driver when driver inattention causes unintended lane departure. There have been many researches based on image processing in transportation systems like [1, 2, 3]. For example, [3] uses K-lane markings. However, feature-based methods are susceptible to existing noise or occlusion. The Generic Obstacle and Lane Detection (GOLD) system in [5] is a stereo vision-based system that recognizes lane structures. The perspective effect of the two stereo images was removed using Inverse Perspective Mapping (IPM), and the resulting image was used to detect lanes with morphological filtering. Region-based methods utilize classification methods to detect lanes in a road. These methods initially extract proper features and then classify image pixels as lane and non-lane. The research in [6] used color and texture features to segment images into road and non-road groups.

Model-based methods use a few parameters to represent a geometric model (straight or curved, etc.) for lane markings [7]. Although these models are less susceptible to noise and occlusion than feature-based methods, they contain calculation complexity and they utilize only special forms of road [5]. Wang et al. proposed a B-snake-based flexible lane model that Gabor filter to recognize lane markings.

means clustering and Kalman filtering to classify highway lanes. Our application utilizes image processing methods to detect lane departure using a mobile phone mounted behind the windshield of a car. The next following literature review is limited to our application.

Vision-based lane detection algorithms can be classified as feature-based, region-based, or model-based [4]. Feature-based methods detect lanes by merging low-level features, such as the edge or color of

detects complex lane structures, such as S-shape lane markings by the setting of control points [8]. A robust algorithm extracted a proper initial position needed for the geometric model. The suggested method is applicable to any type of marked and unmarked roads with shadows and lighting variations. In [9] a Catmull-Rom spline was proposed for flexible modeling of multilane detection with extended Kalman filter tracking and no assumption regarding the form and parallelism of lane markings. The research in [10] used the IPM algorithm and a bank of steerable filters to separate lane markings with various orientations, and then Random Sample Consensus (RANSAC) algorithm applied a parabolic model to fit the road. Lane model parameters were achieved via Kalman filtering. In [11] the algorithm initially applied selective oriented Gaussian filters on a top view of the road image, and then Bézier splines were used to model lane markings using the RANSAC algorithm. The research in [12] proposed use of a geometric model and

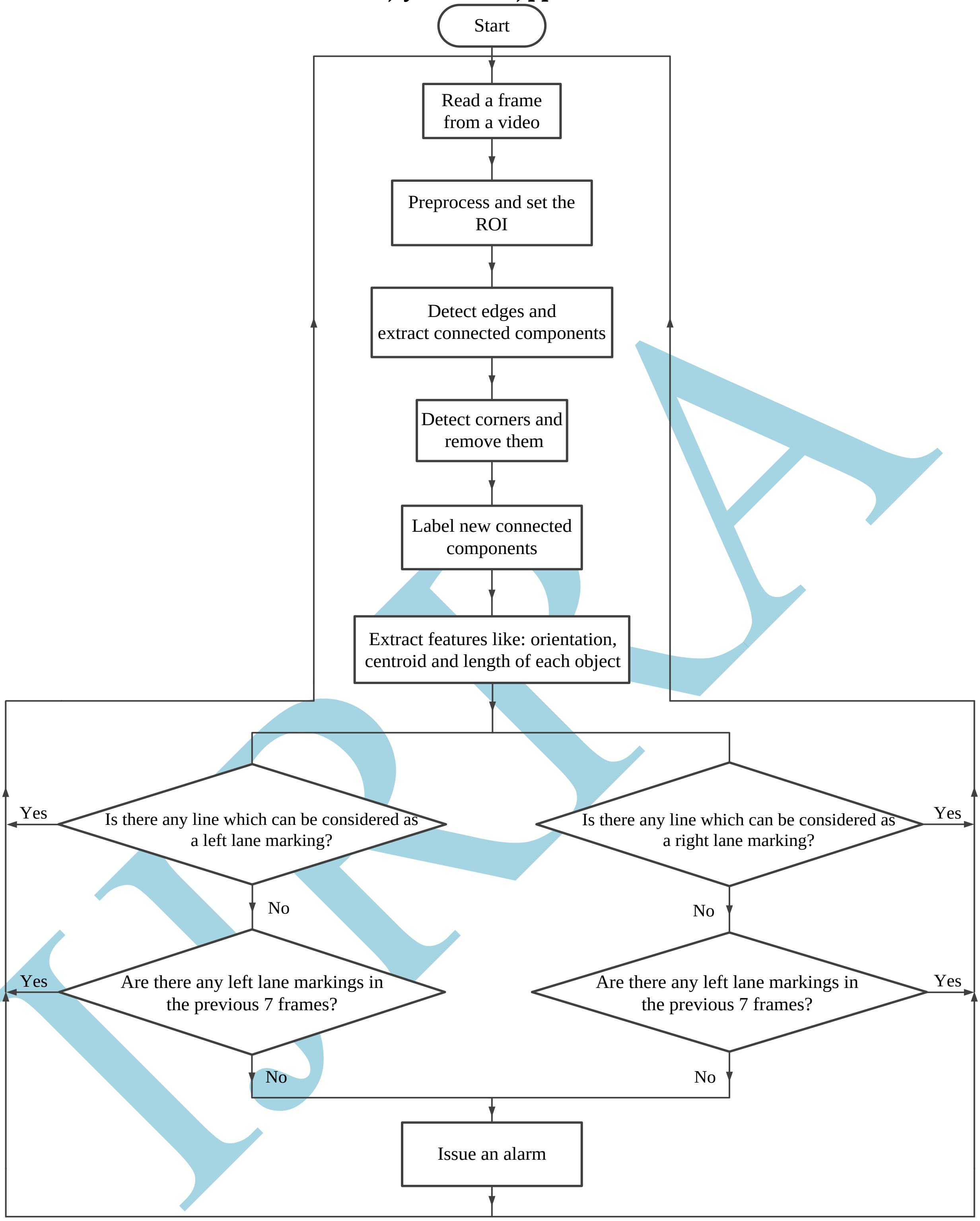


Fig. 1 Flowchart of proposed lane markings detection algorithm.

This method is robust against problems like shadows on the road. An extended edge-linking algorithm was used in [13] to select lane-mark candidates with features like lane-mark orientation and width; color in YUV space was examined to verify these candidates, and a Bayesian probability model applied for lane continuity using the lane-mark color and edge-link length ratio. In [14] a typical weak lane model and particle filtering of lane boundary points were applied to robustly detect lanes. The study described in [15] applied a modified Hough algorithm to detect lanes in real time. Some articles implemented algorithms on embedded processors with limited computational capacity such as smartphones. One study used an iPhone to conduct a simple Hough transform in order to detect lanes [16]. The algorithm in [17], which was implemented on a smartphone, used color transform, segmentation, edge detection, and a Hough transform to detect lane markings. In [18] features such as the intensity value of lanes were used to detect lanes within a parabolic polynomial lane model. This Windows-based algorithm was implemented on a smart phone. The algorithm in [19] utilized color transform, color filtering, and edge filters to detect lanes. This algorithm was designed based on parallel programming in order to reduce execution speed, and the algorithm was implemented on a quad-core mobile phone.

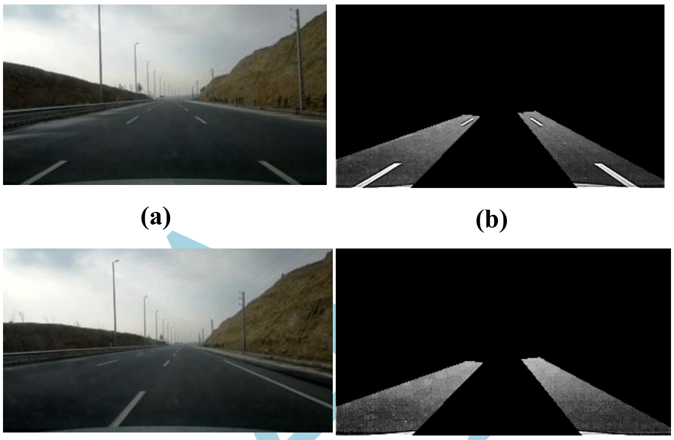
This paper presents a novel lane detection departure system for use on embedded processors, such as smart phones, thereby reducing computational cost while presenting an algorithm with decreased susceptibility to noise. Drowsiness mostly occurs in straight roads or highways. Lane markings are often available nowadays. Therefore, the target of our research is to detect lane departure in such roads. Video frames were captured by mobile cameras mounted behind the windshield of a car. The proposed method was tested on various roads scenes with dashed or continuous white or yellow lane markings. The lanes were curved or straight and flat or non-flat. The algorithm exhibited robustness and accuracy when run on real conditions.

II. PROPOSED METHODOLOGY

Our lane detection and tracking system consists of several stages, as illustrated in Fig. 1, and each frame is processed separately. Due to fixed camera installation, left and right lane markings are located in fixed regions of each frame. Therefore, a mask is used to segment regions of interest (ROIs) in order to restrict calculations. Edges are extracted by the Canny edge detector, and then connected edge components are labeled. Right and left lane markings are chosen based on features such as orientation, location, and surrounding pixel intensity pattern. The presence or absence of left or right lane markings in frames is examined in order to detect lane departure. Our method has the advantages of simplicity, robustness against noise, and applicability in real time.

A. PREPROCESSING AND SETTING THE ROI

Two cameras, both in RGB format with resolutions of 720✕1280 and 1080✕1920, were used in this research. The images were converted to grayscale and down-sampled by 4 to reduce computational complexity. As shown in Fig. 1, the first step of our algorithm was to set the ROI, consequently limiting all calculations to this region and increasing the processing speed. Fig. 2a illustrates a state in which a car moves in just one lane, and Fig. 2b shows ROIs as two non-black areas with lane markings within the ROIs. Because the ROI mask was manually adjusted for each camera installation, a series of video frames was initially captured and the lanes in each frame were segmented, as described in Section III.B. Then the ROI mask was generated by fusing the detected lanes into a separate frame. Fig. 2c shows a car departing from its lane, and Fig. 2d reveals no lane marking in the ROIs.

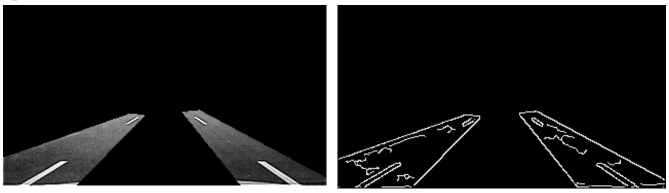


(a) (b) (c) (d)

Fig. 2 Car positions at normal state and lane departure. (a) A car moving in a specific lane, (b) Lane markings within ROIs, (c) A car departing its lane, and (d) No lane markings detected in ROIs.

B. EDGE DETECTION

Our research utilized the Canny edge detector to detect lane edges, in which parameters of the detector were set experimentally (Variance of Gaussian = 2, High Threshold = 0.05, Low Threshold = 0.02; for gray levels scaled between 0 and 1). Fig. 3 shows results from a sample image. Although lane marking edges are evident in the figure, additional detected edges were removed in later stages.



(a) (b)

(a) Original image in ROIs, (b) Edges by the Canny edge detector.

C. EXTRACTING CONNECTED COMPONENTS

Following application of the Canny edge detector, all connected components were extracted and labeled in the resulting binary image using an 8-connectivity neighborhood, as shown in Fig. 4.

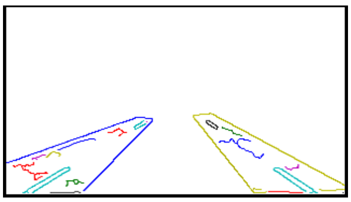


Fig. 3 Extracted connected components.

D. EXTRACTING INTERNAL LANE MARKINGS

The goal of internal lane marking extraction is to determine the internal lines of lane markings. Therefore, because road lane markings are typically in a polygon form, a convex polygon was circumscribed on each connected component, and apexes of the polygon were extracted using the algorithm from [20], allowing detection of the corners of the connected component. In addition, the sides of lane markings were separated by removing the corners and a small neighborhood of corners. Fig. 5 illustrates application of this procedure to a left-side synthetic lane marking.

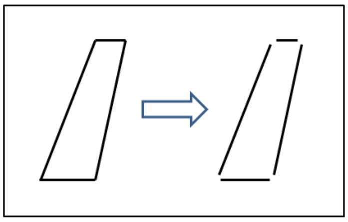
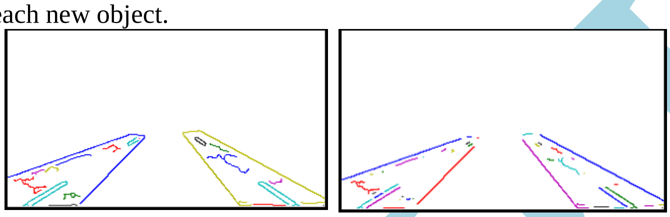


Fig. 4 Removal of corners in each polygon.

E. EXTRACTING THE LENGTH AND ORIENTATION OF EACH OBJECT

Four new connected components were created after the corner points and neighborhood pixels were removed from previous connected components. The new components, which were essentially straight lines, were labeled, and the number of pixels in each labeled component represented its length. A line was fitted through the component pixels; the angle between this line and the x-axis represented its orientation. Components with lengths less than a threshold were removed. Fig. 6b shows new connected components after corners in Fig. 6a were removed. Then two features, length, and orientation, were extracted for each new object.



(a) **(b)**

Fig. 5 (a) Initial connected components, (b) Subsequent connected components after corner removal.

F. DETECTING LEFT OR RIGHT INNER LANE MARKINGS

In order to detect inner left and inner right lane markings, a base point was initially assumed in the middle of the ROIs, located at the centroid of the triangle created by the mask (Fig. 7a) and is shown by a red point. Fig. 7b shows a base point on the road image, and section III.C describes how to select this point.

(a) **(b)**

Fig. 6 (a) Base point on the mask image, (b) Base point on the road image.

The perpendicular distance from the base point to each object was used as the feature to determine internal lane markings, as shown in Fig. 8 and computed as

$$ShortestDistance(i) = \frac{|y_1 - m_i x_1 - b_i|}{\sqrt{{m_i}^2 + 1}} \qquad (1)$$

where (x_1, y_1) is the base point coordinate and m_i and b_i are parameters that represent the line $y_1 = m_i x_1 + b_i$ passing through pixels in i-th component.

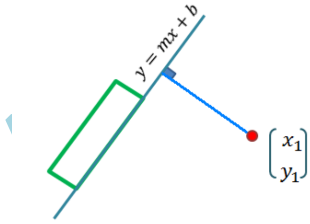


Fig. 7 Distance between the base point and the line passing through each object.

Objects were then partitioned into two groups. Objects with centroid coordinates in the left half of the frame were considered candidates for left lane marking; all other objects were categorized as right lane marking. The object nearest to the base point in each group was assumed to be the left or right inner marking according to three conditions. If any of these conditions was not satisfied, the next nearest object in each group was considered.

The first condition verified orientation validity. As shown in the video frames, the left and right lanes were expected to have positive or negative orientation, respectively. Therefore, if the slope of the inner lane marking did not match its left or right group, it was not considered to be an inner lane marking and was removed from the list. Lane marking orientations had to range from 25 to 90 degrees with respect to the horizontal direction. Objects out of this range were not considered to be inner lane markings.

The second condition verified the horizontal intensity pattern around the object. The Canny edge detector, however, could potentially detect shadows, road cracks, or skid marks, as shown in Fig. 9, which may be nearest to the base point. If a horizontal profile of these objects was considered, a bright-dark-bright profile was created on each horizontal line of the frame; in contrast, a profile for a lane marking was dark-bright-dark. This pattern was considered to be a feature in order to remove the listed, non-desired objects.

(a) **(b)**

(c)

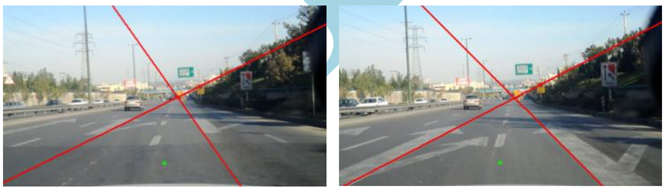
Fig. 8 (a) Cracks on road surface, (b) Skid marks on road surface, and (c) Shadows on road surface.

While examining the intensity pattern for component i (C_i), the following formula was used to estimate the intensity difference around this component:

$$AvgDif(C_i) = \sum_{(x,y)\in C_i} \sum_{j=1}^{N} \frac{[I(x,y+j) - I(x,y-j)]}{MN} \quad (2)$$

where M is the number of pixels in this component, I(x, y) is the edge image, and N is the window width. If AvgDif(C_i) was greater than a positive threshold (set to 0.07), this component was considered to be a right lane marking; if AvgDif(C_i) was less than a negative threshold (set to -0.07), the component was considered to be a left lane marking. If the above conditions were not satisfied, the component was determined to be a road crack or skid mark instead of a lane marking. The threshold values were set experimentally using various video frames. The value of N (set to 5 in the calibration process) depended on camera resolution and lens magnification. Therefore, the maximum number of pixels that represented the width of lane markings was determined, and N was chosen a bit above it. Shadows did not alter this process, however, because intensities of the lane marking and its surrounding area on the road decreased identically in shadowy regions.

The third condition removed arrow markings since these markings can erroneously be detected as lane markings (Fig. 10). As shown in Fig. 10, arrow markings are generally much wider than lane markings and can be identified using a threshold on the marking width. Because of perspective effects, the width of both arrows and lane markings decreased when transitioning from the bottom of Fig. 10b to the top. If the width of the lane marking nearest to the camera was above the threshold (threshold = 7), the lane marking was determined to be an arrow.



(a) **(b)**

Fig. 9 (a) Incorrect detection of lane departure due to arrow markings on the road far from the camera, (b) Incorrect detection of lane departure due to arrow markings on the road near to the camera.

G. LANE DEPARTURE IDENTIFICATION SYSTEM

A lane departure identification system consists of lane marking detection and proper criterion for lane departure detection. Lane markings previously were detected in single video frames: If a lane marking was found, the driver was assumed to be driving within the lane; if no lane marking was found, the driver was assumed to have experienced a lane departure. However, use of results from a single frame often results in an inflated number of false alarms because some frames do not contain lane markings in the specific ROI. Because our system must be able to operate for extended periods of time, the rate of false alarms must be minimal. Therefore, a lane departure was declared only if no lane marking was found on some reasonable consecutive frames. This number of previous consecutive frames was referred to as K_{sum}.

Fig. 11 illustrates a lane departure between lane markings, where D is width of the road, V is speed of the car, and θ is the angle of car departure from its path. As shown, departure of the car decreased the distance between the car wheels and left lane marking at a speed of Vsin(θ). Therefore, K_{sum} depends on vehicle speed and the departure angle from the driver's path, as detailed in Section III.A.

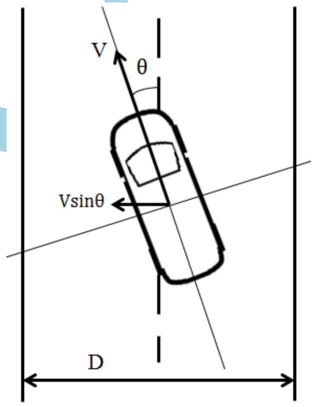


Fig. 10 Geomtery of a car in lane departure.

III. EXPERIMENTAL RESULTS

The proposed departure identification system was evaluated with images captured by mobile phone cameras mounted behind the windshield of a car. Apple iPhone 4s and Nokia N8 cell phones were used with 8 and 12-megapixel resolution cameras, respectively. The resolution of each video frame in the iPhone 4s was 1080×1920 and 720×1280 pixels in the Nokia N8. Both cameras were in RGB format with frame frequencies of 30 frames per second. In order to run the algorithm in real time, only 4 out of 30 frames were passed to the algorithm and the captured frames were down-sampled by a factor of 4. As a result, 4 frames were processed per second, allowing the algorithm to run in real time. For the sake of generality, the proposed algorithm was tested using six road video clips that included many city roads with various lighting conditions and lane types and complicated road surfaces, such as surfaces containing cracks and shadows. A total of 7828 frames were processed.

The red lines in all subfigures of Fig. 12 illustrate results of lane marking detection on challenging frames. Although arrow lane marking signs are evident in Fig. 12a, the lane markings are properly detected. In addition, lane markings are properly detected with road curvature in Fig. 12b and accurately detected in shadowy frames such as Figs. 12c, d, e, and f. Also Fig. 12g shows a patch lane marking, and Fig. 12h identifies lane markings adjacent to light traffic.

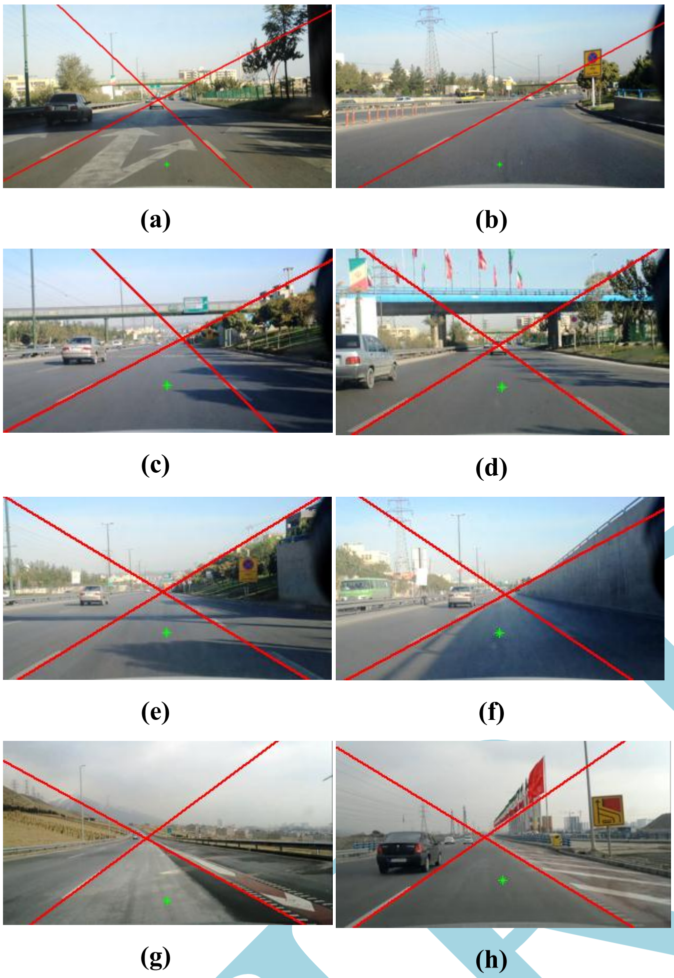


Fig. 11 Results of lane marking detection in challenging frames. (a) Road with arrow lane marking signs, (b) Road with curvature, (c,d) Roads with simple shadows, (e,f) Roads with complex shadows, (g) Road with patch lane marking, and (h) Presence of nearby light traffic.

Results of our algorithm on various videos are shown in Table 1, where K_{sum} was set to 7. In this research, a false alarm referred to activation of an alarm when no lane departure was evident. False alarms occurred due to failure to see lane markings as a result of conditions such as improper lighting while the car was correctly moving on its path.

Table 1 Results of lane departure detection algorithm on different videos.

Video	Number frames	Number of lane departure	Number of detected lane departures	False alarms
1	820	3	3	0
2	2750	5	6	1
3	480	2	2	0
4	1729	8	8	0
5	1636	7	7	0
6	413	3	3	0
Total	7828	28	29	1

A. THE EFFECT OF K_{sum} ON FALSE ALARMS

If a car deviated from the center of the lane at a speed of Vsin(θ), then K_{sum} depended on vehicle speed and departure angle from the driver's path. Six videos were used to evaluate the effect of this parameter, and the detection accuracy and number of false alarms were obtained for each video by changing the number of K_{sum} from 2 to 9 frames, as shown in Fig. 13. Car speed ranged between 70 and 100 kilometers per hour, and the deviation angle of the car steering wheel was approximately 20-30 degrees. According to evaluation results, setting K_{sum} as greater than 6 decreased false alarms, while making K_{sum} equal to 7 resulted in correct detection of all lane departures. Increasing K_{sum} was proven to decrease false alarms.

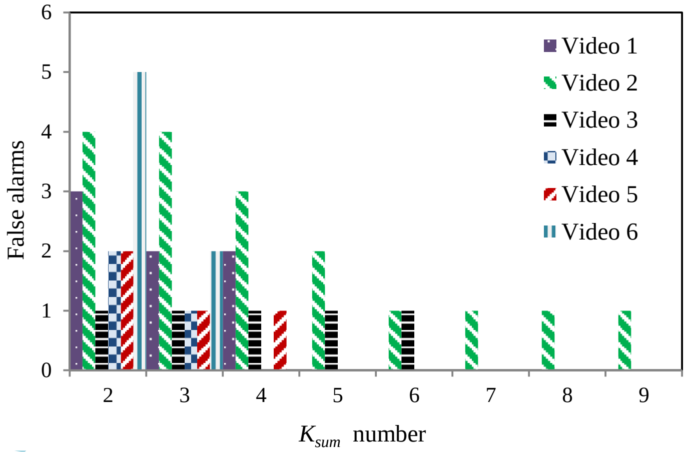


Fig. 12 Results of the lane departure false alarms versus K_{sum} for six videos.

As mentioned in Section II.G, car speed and deviation angle of the car steering wheel were necessary in order to select the number of previous frames for detecting lane departures with minimum false alarms. According to experimental limitations, detailed effects of these parameters cannot be investigated, but as discussed in this section, the system is robust to speed variations. However, if car speed or deviation angle increase, the system must be able to quickly warn the driver. Therefore, in order to simulate various car speeds, frames used in the algorithm may be reduced. For example, if only every other frame is used, the speed would be considered twice as fast. Consequently, six previous videos were used, but only odd frames were applied in order to double the speed.

Fig. 14 shows the number of false alarms in new simulation. As expected, the results in Fig. 14 are similar to the first experiment. The number of false alarms reduced to zero even when K_{sum} was equal to 7. If the time duration for a departure was 2 seconds (corresponding to 60 frames) when considering speed and departure angle, then lane departures were correctly detected for K_sum equal to 7. Therefore, the system is robust against variations of car speed (by a factor of 2), and K_{sum} equal to 7 warns drivers in the least amount of time.

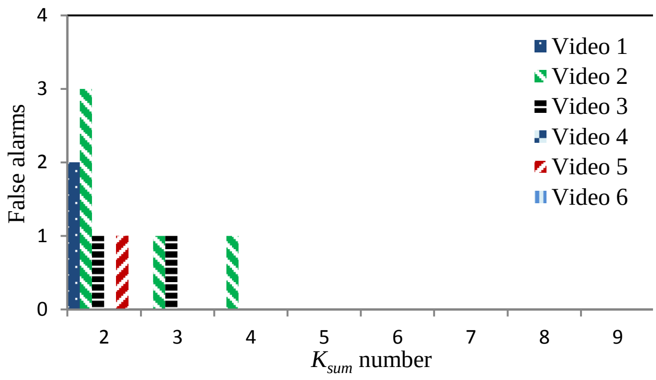


Fig. 13 Results of the lane departure false alarms versus K_{sum} with double speed.

B. CAMERA CALIBRATION

Camera calibration is essential in many image and video processing systems, so the calibration procedure must be as simple as possible in consumer applications. In the proposed algorithm, the mobile camera was positioned on a mobile holder underneath the front mirror. At the beginning of camera calibration, the car was assumed to move in correct lane position in order to provide calibration frames. In addition, the ROI mask and base point were determined in this phase. Appropriate frames were selected in order to create the ROI mask. Then the lane markings of these frames were overlaid onto one frame, and the ROI mask was chosen as a polygon surrounding all overlaid lane markings, as shown in Fig. 15. If the camera installation did not change, the above ROI mask was valid for the algorithm. The base point was introduced in Section II.F. Selection of this point did not significantly affect algorithm accuracy, as discussed in the next section.

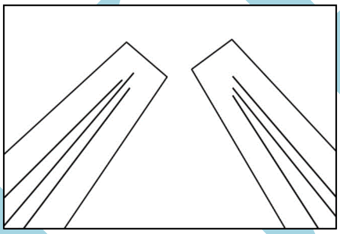


Fig. 14 Overlaid lane markings to create the ROI mask.

C. INVESTIGATING DISPLACEMENT OF BASE POINT IN ROIS

This section investigates the robustness of the algorithm to base point position, in which the base point of Fig. 16a was partially displaced in horizontal and vertical directions, as shown in Figs. 16b, c, d, and e.

(a)

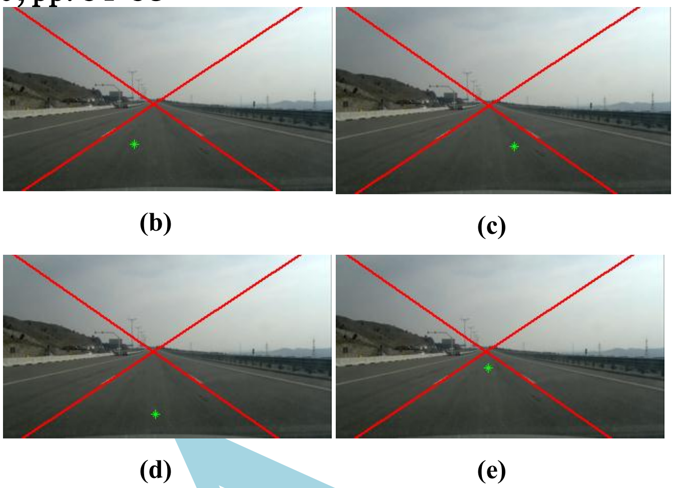


Fig. 15 Base point position. (a) Original position, (b,c) Horizontal displacement, and (d,e) Vertical displacement.

Results of base point displacement showed that base point position did not significantly affect the results. In order to investigate the effect of base point position, even with K_sum less than 7 (resulting in more false alarms), K_{sum} was chosen to be 3, and variations of the base point were evaluated. Fig. 17 shows the number of false alarms on six videos with various base point positions and K_{sum} equal to 3.

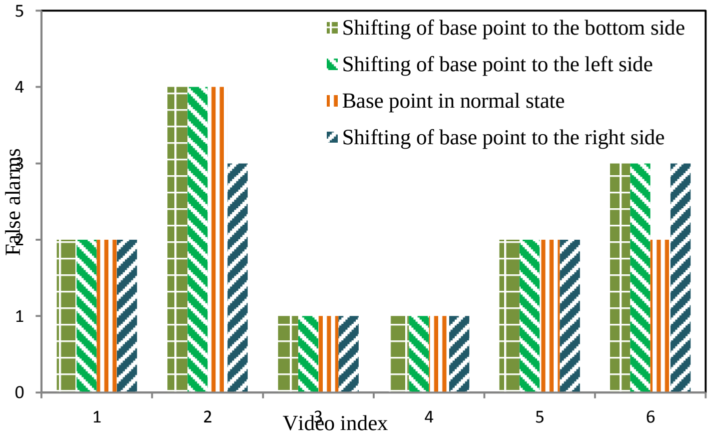


Fig. 16 Number of false alarms with different base point positions and K_{sum}=3.

As shown in Fig. 17, variation of base point did not severely alter false alarms. Results also showed that all departures in these positions were detected. In fact, because camera displacement can be considered base point displacement, the algorithm can also be considered to be robust against partial displacement of the camera.

IV. CONCLUSION

This paper proposed a robust, novel, real-time lane departure detection algorithm that was tested on real, challenging video sequences with shadows, road cracks, and skid marks. The algorithm was implemented in real-time on an android tablet, P7500 Galaxy Tab 10.1 3G with Dual-core 1 GHz Cortex-A9 Processor CPU and 1 GB RAM. Results indicated detection of all lane departures under the mentioned conditions, as shown in Table 1, and an acceptable false alarm rate. Results also showed that the algorithm was robust against partial camera position and displacement.

ACKNOWLEDGEMENTS

We would like to give our special thanks to Ms. Negar Golestani, Ph.D. student from University of Southern California, for her assistance in simulating this algorithm on the tablet, Mr. Nader Zad, Ph.D. student from Kansas State University, and Ms. Sarah Hernandez, M.A., from University of Dayton, for comments that greatly improved the manuscript.

REFERENCES

[1] Díaz, J., Ros, E., Mota, S., Botella, G., Cañas, A., Sabatini, S.: Optical flow for cars overtaking monitor: the rear mirror blind spot problem. Ecovision (European research project) (2003)

[2] Sabzevari, R., Scaramuzza, D.: Multi-body Motion Estimation from Monocular Vehicle-Mounted Cameras. IEEE Transactions on Robotics, vol. 32, pp. 638-651 (2016)

[3] Melo, J., Naftel, A., Bernardino, A., Santos-Victor, J.: Detection and classification of highway lanes using vehicle motion trajectories. IEEE Transactions on intelligent transportation systems, vol. 7, pp. 188-200 (2006)

[4] Samadzadegan, F., Sarafaz, A., Tabibi, M.: Automatic Lane Detection in Image Sequences for Vision-Based Navigation Purposes," ISPRS Image Eng. Vis. Metrol. (2006)

[5] Bertozzi M., Broggi, A.: GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection," Image Process. IEEE Trans., vol. 7, no. 1, pp. 62–81 (1998)

[6] Thorpe, C., Hebert, M. H., Kanade, T., Shafer, S. A.: Vision and navigation for the Carnegie-Mellon Navlab. Pattern Anal. Mach. Intell. IEEE Trans., vol. 10, no. 3, pp. 362–373 (1988)

[7] Shin, B., Xu, Z., Klette, R.: Visual lane analysis and higher-order tasks: a concise review. Mach. Vis. Appl., vol. 25, no. 6, pp. 1519–1547 (2014)

[8] Wang, Y., Teoh, E. K., Shen, D.: Lane detection and tracking using B-Snake. Image Vis. Comput., vol. 22, no. 4, pp. 269–280 (2004)

[9] Zhao, K., Meuter, M., Nunn, C., Müller, D., Müller-Schneiders, S., Pauli, J.: A novel multi-lane detection and tracking system. In: IEEE Intelligent Vehicles Symposium, Proceedings, pp. 1084–1089 (2012)

[10] Sivaraman, S., Member, S., Trivedi, M. M.: Integrated Lane and Vehicle Detection, Localization, and Tracking: A Synergistic Approach. Intell. Transp. Syst. IEEE Trans., vol. 14, no. 2, pp. 906–917 (2013)

[11] Aly, M.: Real-time Detection of Lane Markers in Urban Streets. Intelligent Vehicles Symposium, 2008 IEEE. pp. 7–12 (2008)

[12] Zhou, S., Jiang, Y., Xi, J., Gong, J., Xiong, G., Chen, H.: A novel lane detection based on geometrical model and gabor filter. Intell. Veh. Symp. (IV), 2010 IEEE, pp. 59–64 (2010)

[13] Lin, Q., Han, Y., Hahn, H.: Real-Time Lane Departure Detection Based on Extended Edge-Linking Algorithm. In: Second Int. Conf. Comput. Res. Dev., pp. 725–730 (2010)

[14] Ruyi, J., Reinhard, K., Tobi, V., Shigang, W.: Lane detection and tracking using a new lane model and distance transform. Mach. Vis. Appl., vol. 22, no. 4, pp. 721–737 (2011)

[15] Yi, S.-C., Chen, Y.-C., Chang, C.-H.: A lane detection approach based on intelligent vision. Comput. Electr. Eng., vol. 42, no. 2, pp. 23–29 (2015)

[16] Ren, F., Huang, J., Terauchi, M., Jiang, R., Klette, R.: Lane detection on the iPhone. In: Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, vol. 30 LNICST, pp. 198–205 (2010)

[17] Lan, M., Rofouei, M., Soatto, S., Sarrafzadeh, M.: SmartLDWS: A robust and scalable lane departure warning system for the smartphones. In:12th Int. IEEE Conf. Intell. Transp. Syst., pp. 1–6, Oct (2009)

[18] Chen, C.-J., Wu, B.-F., Lin, W.-H., Kao, C.-C.: Mobile lane departure warning systems. In: IEEE 13th Int. Symp. Consum. Electron., no. 1, pp. 90–93 (2009)

[19] Abdullah, D., Al-hafidh, M. M.: Developing Parallel Application on Multi-core Mobile Phone. Int. J. Adv. Comput. Sci. Appl., vol. 4, no. 11, pp. 89–93 (2013)

[20] Haralick, R. M., Shapiro, L. G.: Computer and Robot Vision, vol. 1. Addison-Wesley (1991)