

A study on obstacle detection method of the frontal view using a camera on highway

Van-Quang Nguyen¹, Jeonghyeon Park², Changjun Seo², Heungseob Kim¹ and Kwangsuck Boo^{1,*}

¹Department of Mechanical and Automotive Engineering, Inje University, Gimhae, Gyeongnam, Korea.

²Department of Electronic Engineering, Inje University, Gimhae, Gyeongnam, Korea.

*Email: mechboo@inje.ac.kr

Abstract. In this work, we introduce an approach to detect vehicles for driver assistance, or warning system. For driver assistance system, it must detect both lanes (left and right side lane), and discover vehicles ahead of the test vehicle. Therefore, in this study, we use a camera, it is installed on the windscreen of the test vehicle. Images from the camera are used to detect three lanes, and detect multiple vehicles. In lane detection, line detection and vanishing point estimation are used. For the vehicle detection, we combine the horizontal and vertical edge detection, the horizontal edge is used to detect the vehicle candidates, and then the vertical edge detection is used to verify the vehicle candidates. The proposed algorithm works with of 480×640 image frame resolution. The system was tested on the highway in Korea.

1. Introduction

According to the WHO, each year lives of approximately 1.25 million people are cut short as a result of road traffic crash. Between 20 and 50 million people suffer from non-fatal injuries, which sometimes incur disabilities [1]. Road traffic injuries bring considerable economic losses to victims, their families, and nations as a whole. Therefore, in 2016 many firms or corporation has declared that they were and will participate in the development of the automatic vehicle. Volvo Corporation has promised that by 2020, no one will be killed or seriously injured by one of its new cars by using driving assistance system and warning [2].

Many researchers have studied and proposed the different strategies and track vehicle and lane, but most of the works have been done separately. For the lane detection, there are many main features that can extract the lane mark pixel from an image, consisting of edge feature [3], color feature [4], Haar-like feature [5], and hough transform [6]. Among the vehicle detection approaches, there are also many approach methods based on features, such as Haar-like Feature [7], Haar-like feature and artificial neuron networks [8], Histogram of oriented gradients (HOG) and support vector machine (SVM) classifier [9, 10] proposed. These proposed algorithms, which are powerful and robust, operate in real-time with high accuracy. However, only road detection or vehicle detection is not sufficient to support driving assistance system, warning system, or lane change system.

In this paper, the key idea is to integrate the relationship between lanes and vehicles to support the warning system, and a driver assistance system. In figure 1a, when drivers want to change lane on the right-side, they have to monitor the frontal lane, and right-side lane. For instance, there are vehicles on these lanes and near the driver's vehicle, so drivers could not change lane, inversely there are not



vehicles on these lane, drivers could change lane. This is one example for the purpose of my paper. This idea consists of two main steps, the information lane will be detected in the first step. And then, vehicle will be detected inside area among the detected lanes by vehicle features. Figure 1b shows the block diagram of proposed system. In the block diagram, we have a notions as a region of interest (ROI). ROI is defined as an area close to the test vehicle. The full image is used to detect vehicles. There are two phases: 1) lane detection on ROI, and 2) vehicles detection based on the detected lanes. This paper is divided as follows. Section 2 describes steps for the lane detection. The method of vehicle detection and tracking is shown in section 3. Section 4 presents the experimental results.

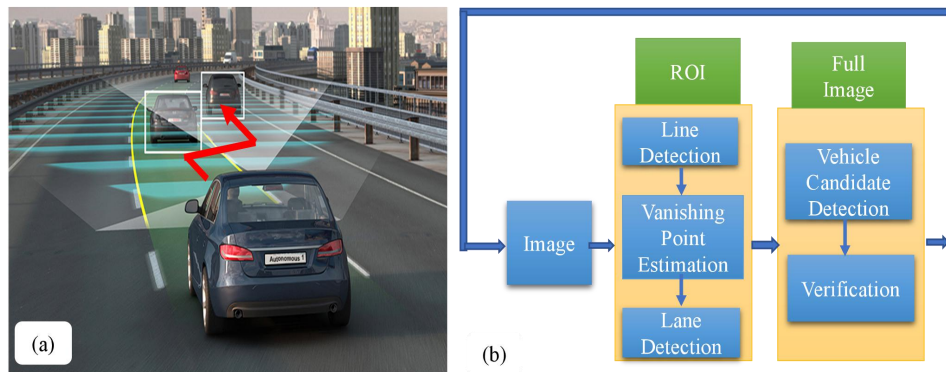


Figure 1. (a) Vehicles monitoring for lane change, (b) The proposed method.

2. Lane detection

2.1. Line analysis

The first step of the method is to discover the line segments from the image inputs. For line detection, EDLines are used [11], EDlines algorithm only takes from 10ms to 20ms with an Intel 2.2 GHz CPU. Moreover, EDLines algorithm runs even faster when it is applied to ROI. For any image in this work, we determine a region interest (ROI) for lane detection, ROI are defined as the rectangle inside the red lines, as shown in figure 2a. Figure 2b shows the line segment detected by EDlines.

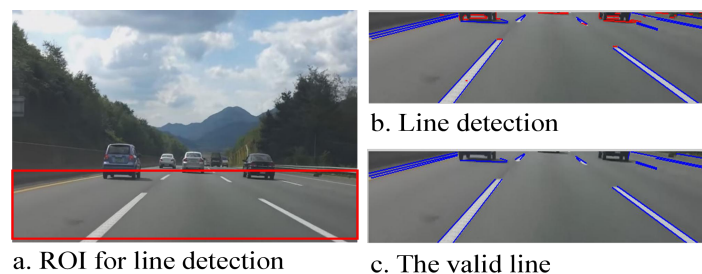


Figure 2. Line detection and analysis.

After line segments are extracted from the image, two eliminative steps are performed to remove the irrelevant line segments. Firstly, all of the vertical line segments, horizontal line segments, and very short line segments are removed and these are the red line segments in figure 2b. The remaining line segments are shown in figure 2c. Next, we have to reject the irrelevant remaining line segments and keep the relevant ones. The relevant line segments are the lane markings, and the lines intersecting the lane markings at the vanishing point. The line segments are divided into two subsets, left and right candidate sets. The left segments are indicated by blue and the right segments are indicated by green. The angles between all of these segments and the horizontal axis are calculated as shown in figure 3a, in which θ_{left} is the angle between left segments and horizontal axis and θ_{right} is the angle between right segments and horizontal axis. We have set range limits for each angle, 30° to 85° for θ_{left} and

120° to 175° for θ_{right} . It means that any segment that does not belong to the ranges will be removed. In figure 3b, the red segment will be removed and the relevant segments of the two sides will be kept for the next step.

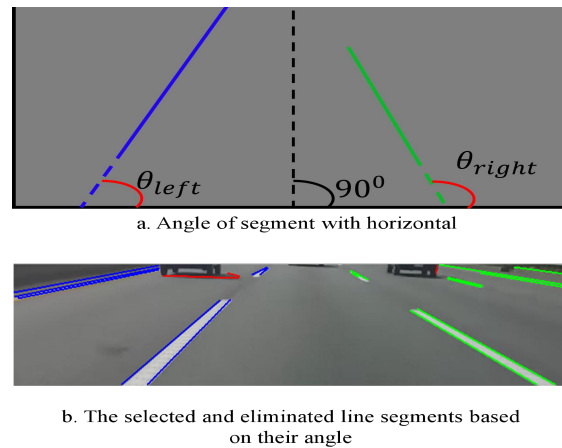


Figure 3. Line segment selection based on their angle with horizontal axis.

2.2. Vanishing point estimation

As previously mentioned, we have two clusters of the relevant segments, the left and the right. The vanishing point is the intersection where the lane markings or the road boundaries cross. Therefore, all lines on each cluster are extended as in figure 4a and we have found the intersection points where each line of the left cluster meets the right cluster ones as shown in figure 4b (red points). In order to define the vanishing point, we create a grid formed by square cells of size equal to the permitted error in estimating the vanishing point position. Finally the cell that has the maximum number of the intersection points, is selected and its centre weight is returned as an approximation of the vanishing point position. In figure 4c, we present the test image in which the vanishing point is clearly visible with a red.

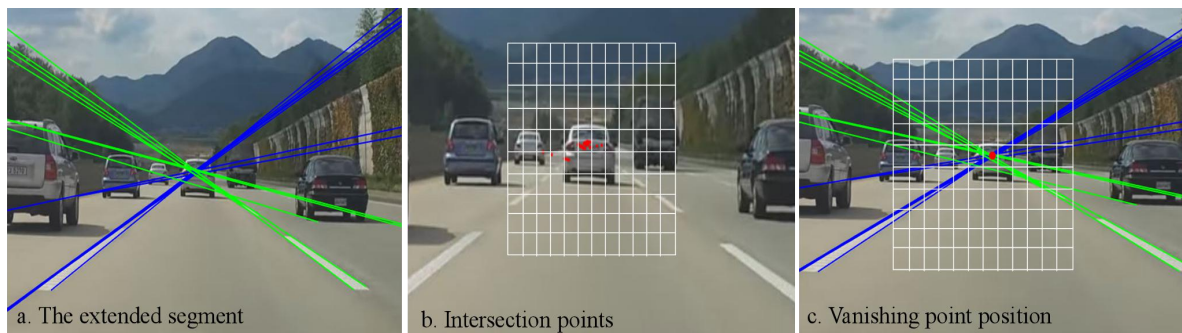


Figure 4. Vanishing point estimation.

2.3. Lane detection

To detect lane markings, the horizontal straight line is created, they intersect with the extended segment at red points, these points are in the red circles. Based on the distance among these points, the points are close which are clustered a group. As in figure 5a there are 5 groups. Visually, we can define that group 2 and group 3 belong to the lane marking of the middle lane. Distance between group 2 and group 3 is d , because distance between group 1 and group 2 closely equals with d , distance between group 3 and group 5 closely equals with d . So, group 1 and group 2 belong to the lane marking of the left lane, and group 3 and group 5 are on the lane marking of the right lane. Finally, the lane markings are shown as in figure 5b.

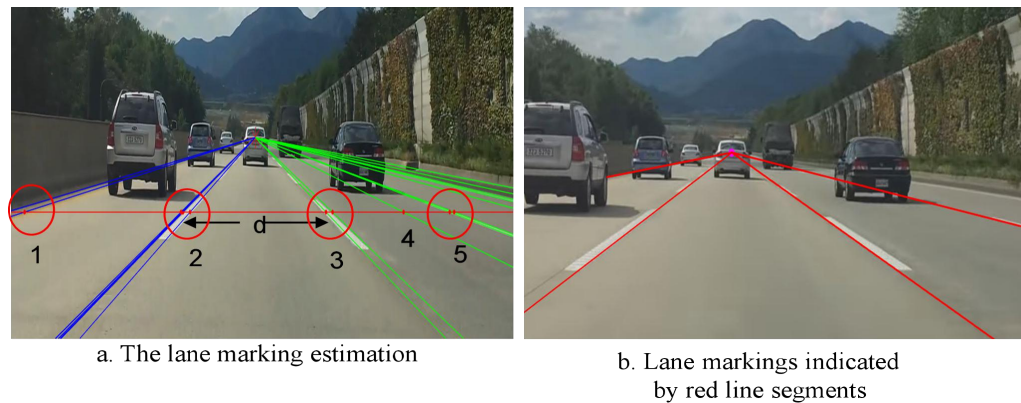


Figure 5. The lane marking estimation.

3. Vehicle detection and tracking

3.1. Vehicle candidate detection

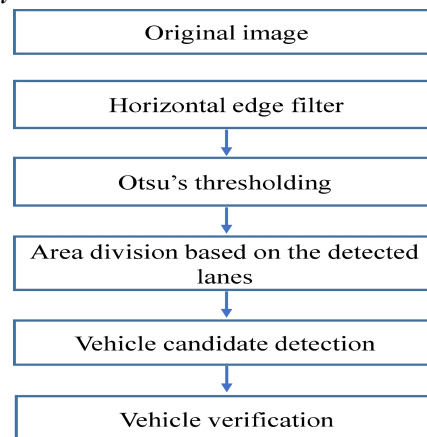


Figure 6. Algorithm for vehicle detection.

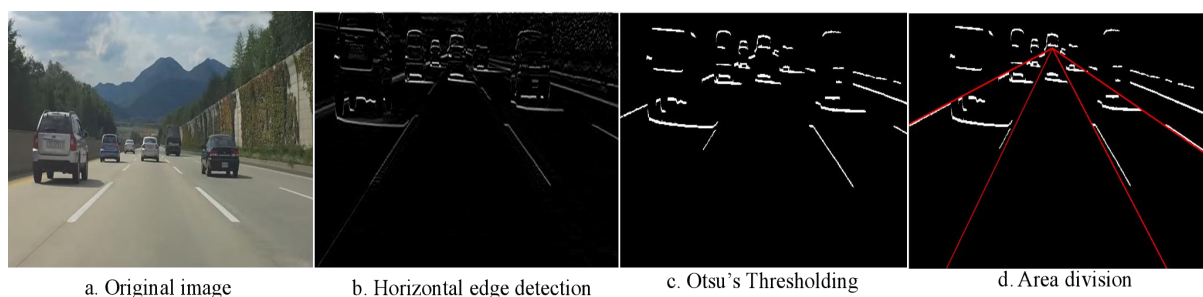


Figure 7. Demonstration for image processing algorithm.

Vehicle detection is the most important problem in this paper. Hence, the proposed method is shown in figure 6, and which is based on the detected lane markings from the previous section. In order to approach this feature, the original image in figure 7a is processed through horizontal edge filter shown in figure 7b. Then, we convert the received image from grayscale to binary as in figure 7c using otsu's thresholding. After having the binary image, we divide image into lane areas based on the detected lane markings. Information of the lane areas as in figure 7d is used to detect vehicle.

Figure 8a is the binary image of the left-side lane, Figure 8b presents the sum of the horizontal pixel following the vertical axis. From the graph in figure 8b, we can detect the candidate information.

Equivalently, candidates can be defined on the middle lane and right-side lane as shown in figure 8c and 8d.

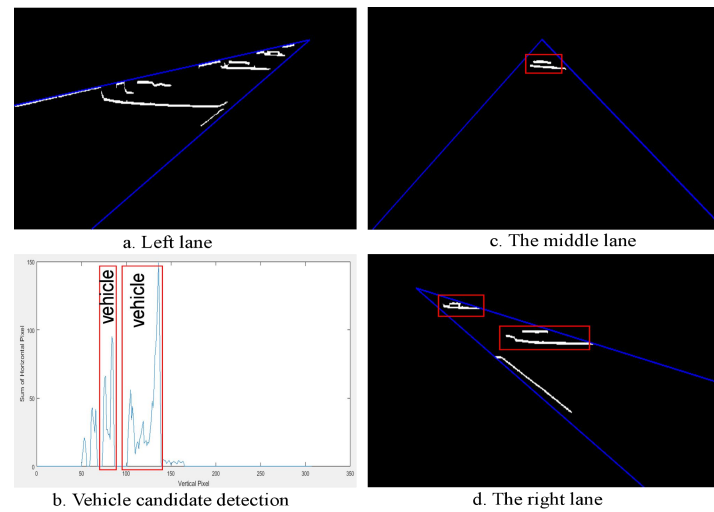


Figure 8. Vehicle candidate estimation.

3.2. Vehicle verification

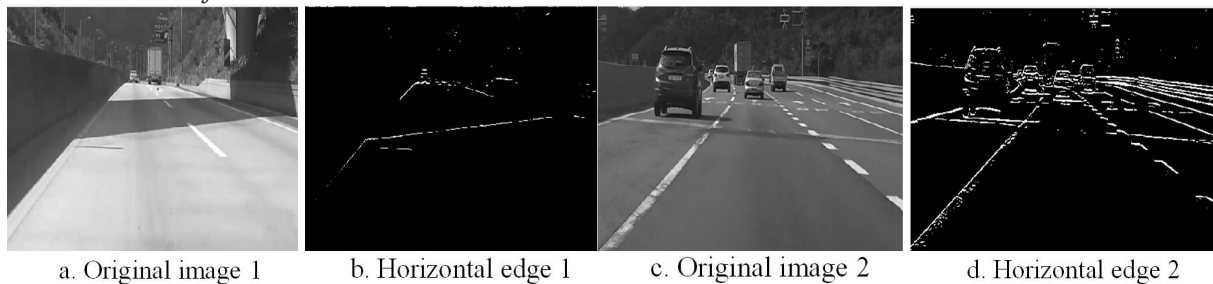


Figure 9. Noise on highway.

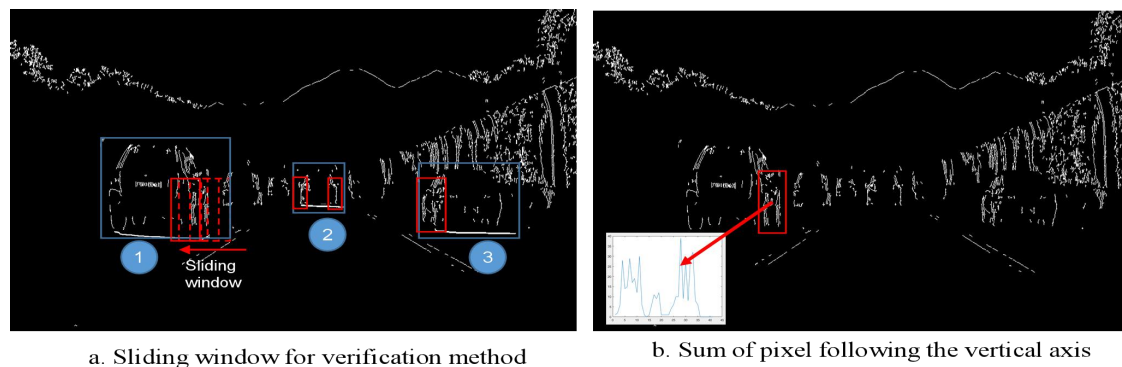


Figure 10. Verification method of candidates.

By using horizontal edge to detect vehicles, some wrong cases happened in test processing. In the detail, many noise appears from long highway such as shadow of bridges, shadow of the tension poles, or markings of repaired road in shown in figure 9. They affect our method.

In order to eliminate noise effect, because the back of vehicles has straight edge structure, we compare the detected candidates with this feature of vehicles. Figure 10a shows three objects which were identified from previous section, one of them is on the left lane, another is on the middle lane, and the left one is on the right lane. To verify them, the sliding window are created to check the

vertical edge. For candidates on the left lane, the sliding window is used to check the vertical edge on the right side of candidates, for candidates on the right lane, the sliding window is used to check the vertical edge on the left side of candidates, and with objects on the middle lane, the vertical edge is checked on the both side of candidates as shown in figure 10a. If sum of the pixel of the sliding window following vertical axis is enough large as shown in figure 10b, we can conclude that these candidate are vehicle, on the contrary, they are not vehicle.

4. Experiment and Discussion

4.1. Lane detection

Before performing the experiments, we test the line detection algorithm and vehicle detection algorithm with various conditions such as rainy or foggy weather, and inside of a tunnel. The EDLines algorithm is used for testing in the three cases of conditions stated in figure 11. The line segments are detected clearly and do not affect by various conditions, line segments on the road markings are big and not separate. Therefore, we can use these segments to detect lanes by using the proposed method.



Figure 11. EDLines algorithm with the various conditions.

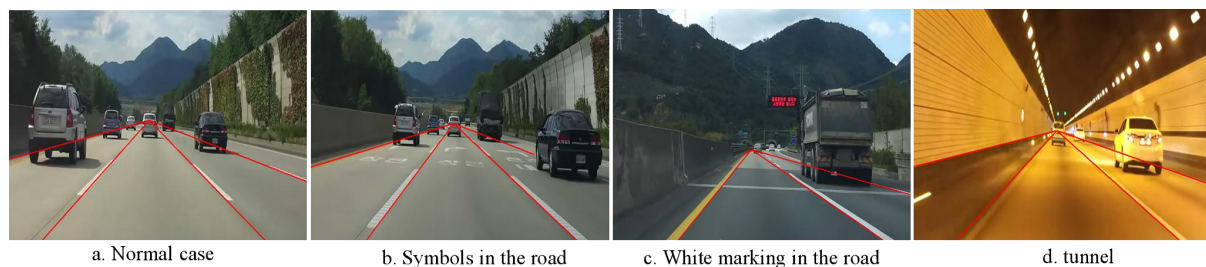


Figure 12. Result for lane detection.

Figure 12 shows some experiments for lane detection. Normal cases are represented in figure 12a in which lane is detected in good weather. In figure 12b, figure 12c, there are some noise which consists of cross-lines, white/yellow lines, direction indicating lines on the road, and objects which are on the road. Noise directly affects accuracy of the algorithm, making the system unstable. However, result of the proposed algorithm has turned out to be really good. Figure 12d is lane detection inside the tunnel.

4.2. Vehicle detection

In this paper, we show three scenes for vehicle detection. There are three vehicles in front of the host vehicle as shown in figure 13a and 13b. If the host vehicle wants to change lane to either left, or right side, it has to detect vehicles on the front and on the left lane or the right. The result of detection is shown in these figure. Figure 13c shows the detected vehicles on the road which has two lanes, and figure 13d shows the detected vehicle in tunnel.

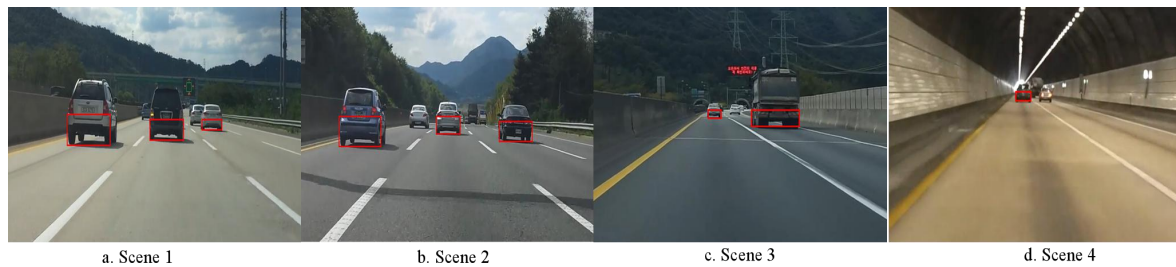


Figure 13. Result for vehicle detection.

5. Conclusion

In this paper, we proposed the approach to detect lanes, detect the multi-vehicles for the driver assistance system. For lane detection, to detect lane in real-time, we use EDLines algorithm which can detect line segments between 10 ms and 20 ms. Our algorithm detects three lane areas of the frontal view. Based on these areas combined with horizontal edge feature of vehicles, vehicles are detected in each lane. From the outcome of experiments, our method obtains the goal in driver assistance system or warning system.

Acknowledgments

This work was supported by the Development Program for Economic Cooperative industry through the Ministry of Trade, Industry and Energy and the Korea Institute for the Advancement of Technology (R0003972).

This work was supported by the Human Resource Training Program for Regional Innovation and Creativity through the Ministry of Education and National Research Foundation of Korea (2015H1C1A1035919).

References

- [1] <http://www.who.int/mediacentre/factsheets/fs358/en/>
- [2] <http://money.cnn.com/2016/01/20/luxury/volvo-no-death-crash-cars-2020/index.html>
- [3] G Dezhi, L Wei, D Jianmin and Z Banggui 2009 In. Conf. on Au. and Log. 980-85.
- [4] C Rotaru, T Graf, and . Zhang 2008 J. of Real-Time I. Pro. 3, 311–22.
- [5] J Heechul, M Junggon, and Kim Junmo 2013 IEEE In. Ve. Sym (IV).
- [6] M He and X MEI 2012 In. Conf. on Wa. Ac. Me. Tech. and In. Pro.
- [7] H Sungji, H Youngjoon and H Hernsoo 2009 W. a. of s. en. and tech 59, 455-59.
- [8] M Atibi, I Atouf, M Boussaa and A Bennis 2015 In. Conf. on Ad. Wi. In. and Com. Tech 73, 24-31.
- [9] N Mohamed, M Ahmad and S Swamy 2014 12th In. N. Cir. and Sys. Conf.
- [10] P MinWoo, P Jungpil and J SoonKi 2013 Pro. of the 2013 ACM Re. in A. and Con. Sys, 171-76.
- [11] C Akinlar and C Topal 2011 In. Conf. on I. Pro.