2016 International Workshop on Emerging ICT

Vehicle Crash Avoidance based on Rear Lamp Detection Systems

Sung-hyun Lee, Soon-ho Shin, Seong-hun Son, and Ho-yel Jung
Department of Information and Communication Engineering,
Yeungnam University, Gyeongsan, Korea
{tjdgusdktk, soon6375}@gmail.com, fred_123@naver.com, hoyoul@yu.ac.kr

Abstract—**Nowadays, we are witnessing an unprecedented growth of automobiles on the roads. Therefore, the danger of crashes between the vehicles is increased severely. To this end, we propose a vehicle crash avoidance system that works on rear lamp detection and backside shade information of the vehicles. In this system, the distance between vehicles is autonomously estimated by measuring the width of front vehicle from the image. First, the candidates of lamps are designated by exploiting the color information such as Hue-Saturation-Value (HSV) and YCbCr. Later, the vehicles are detected through the classifier, which enables Haar-like feature and Adaboost algorithm, and extract the characteristics of vehicle's back luminance. If the distance to the front vehicle is less than the threshold, our system alerts the driver to avoid the impending crash. We demonstrate that our system works well in the actual city driving conditions.**

Keywords-vehicle detection; shade information; HSV; Ycbr; Haar-like feature; Adaboost algorithm;

I. INTRODUCTION

Currently, the advanced driver-assistance systems (ADAS) [1] [2] are undergone rapid growth due to increasingly developed accidents ratio on the roads. Moreover, the rate of road accidents at nighttime is significantly higher than that of day time in the developing world. However, in Korea there are almost one half of the traffic accidents that occur at nighttime [3]. It is therefore, the vehicle crash avoidance systems are of much interest. This kind of system can timely alert the drivers based on precisely calculated distance between the vehicles on runtime. Generally, these systems are equipped with the image processing techniques of rear-lamp detection, ultrasonic, radar sensors algorithm and video recorder in the black-box of vehicles. Moreover, stereoscopic cameras, three-dimensional shape measurement techniques are also used to avoid the uncertainty when the front vehicles are far away at nighttime and cloudy weathers.

In this paper, we proposed rear-lamp detection by Hue-Saturation-Value (HSV) and Ycbr to improve the efficiency and accuracy of the algorithm. This system significantly decreased the amount of operation of algorithm designating through candidate's interest region. For verifying vehicle stage, we use Haar-like feature which has luminance differences at the left and right bottom of the vehicle. Higher accuracy and more processing speed are made possible by organizing classifier and using Integral Image and Adaboost algorithm.

The rest of the paper is as follow. Section II describes the vehicle detection and complementation methods. Section III introduces measuring method between vehicles and hardware based vehicle crash avoidance system. Section IV demonstrates the various experiment results. Finally, we have drawn the conclusion along with the future dimensions in section V.

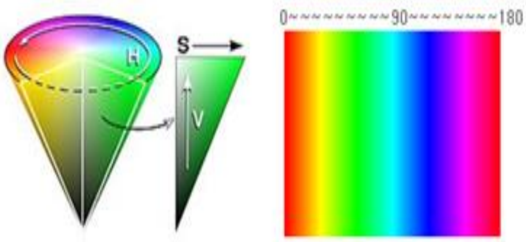


Figure 1. HSV color space

Figure 2. Binary image of rear-lamp candidate

II. WEHICLE DETECTION

In this section, we introduce vehicle detection and process detection by rear-lamp and final detection of each vehicle.

A. Detecting potential candidates

The most common approach makes use of red-color component in the vehicle rear-lamp. First, the inputted color image by camera is converted to HSV and VCbCr as seen in Figure 1. The red-color pixels are extracted by using H component of HSV value as

$$\text{HSV}_{\text{Red Value}} = 0 < H(Hue) < 30, 150 < H(Hue) < 180, \tag{1}$$

then red-color component are extract by Cb and Cr values as

$$\text{YCbCr}_{\text{Red Lamp Value}} = \begin{cases} 140 < Cb < 170 \\ 11 < Cr < 170 \end{cases} \tag{2}$$

After extracting the rear-lamp color information, we create a binary image as seen in Figure 2.

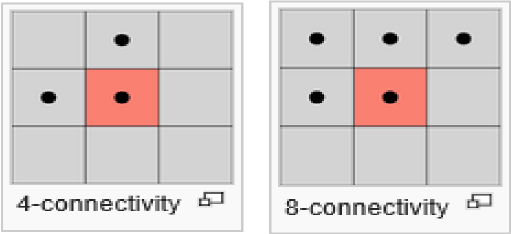


Figure 3. Two Pass Algorithm – Labeling

Figure 4. Rear-lamp candidate labeling

We use the Progress Blob-Labeling to specify the rear-lamp object and to remove the nuisance factor from the detected pixel. In this process, 8-connected labeling is used by Two-Pass algorithm as shown in Figure 3. Moreover, the pixels that have size less than a threshold are simply eliminated to avoid the disturbance factor. Figure 4 demonstrates the result of rear-lamp candidate objects been labeling process.

The progress pairing by image cross-correlation, average and distribution are used for specific verification of detected rear-lamp candidate, symmetry analysis, and for comparing objects size and location. We remove error detected candidate base on the position of two adjacent candidates. Then for the first round of verification, we compare number of pixels and width and calculate differences between candidates. Then, the candidates under the threshold value are selected. Finally, vehicle candidates over the threshold are chosen by the cross correlation as

$$r = \sum_{x,y} \frac{(F_{left}(x,y) - \overline{F_{left}})(F_{right}(x,y) - \overline{F_{right}})}{\sigma_{left}\sigma_{right}} \tag{3}$$

Figure 5 demonstrates the final detected vehicle rear-lamp.

B. Vehicle Inspection

We inspect the vehicles using Haar-like feature and Adabost algorithm for candidates that exist in non-vehicle area from the inputted images. As seen in Figure 6, vehicle has regular patterns of back wheels, shade of bottom part and light intensity of left and right side, so it is useful to use as a feature. In fact, Harr-like feature is the calculation of pixels inside of adjacent rectangular region. It has advantages of simple and easy calculations. Harr-like feature use weak classifier that has feature form as in Figure 7. We used basic feature in integral image to reduce the operation of algorithm.

Figure 5. Final detected vehicle rear-lamp

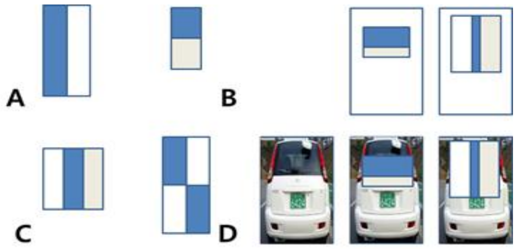


Figure 6. Harr feautre at back of vehicle

Similarly, integral image is used as the quick and effective way to calculate summations over the image sub-regions. As mentioned, we utilize Harr-like feature by Adaboost algorithm to generate vehicle detection classifier. The Adaboost algorithm makes high suspended selection criteria (strong classifier) by sum of easy selection criteria (weak classifier). It is boost algorithm that uses a simple combination of learning-machine and can reduce the whole of errors including training errors as shown in Figure 8. This is so that it generates the strong-classifier by combination of many weak-classifiers.

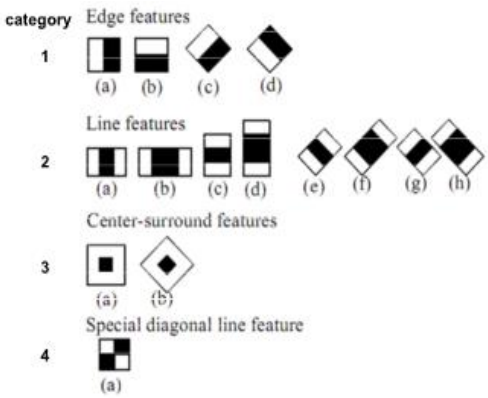


Figure 7. Haar-like feature

Moreover, the Adaboost algorithm makes use of detection and verification of many objects and high accuracy along with Support Vector Machine (SVM) by setting different weighted values on each stage. We process learning by total 14 stages and use four Haar-like feature mask 2000 of positive images and 4000 of negative images as shown in Figure 9. In return, we could confirm excellent performance at 1:2 ratio of positive and negative images.

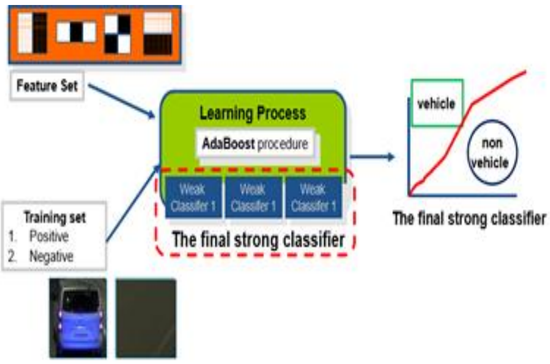


Figure 8. Operation of Adaboost algorithm

Figure 9. Image Training (Positive)

C. Shade Information

We use shade information as a complement to detect vehicles more precisely under night time and dark conditions. The process of detecting shadow of vehicle consists of four intermediate steps as shown in Figure 10. First, the color image is converted into the gray image. Second, the histogram equalization method is used to get better details. Third, binary image changed and image threshold is processed. Finally, erosion and dilation function is used to figure out the shadow of vehicle.

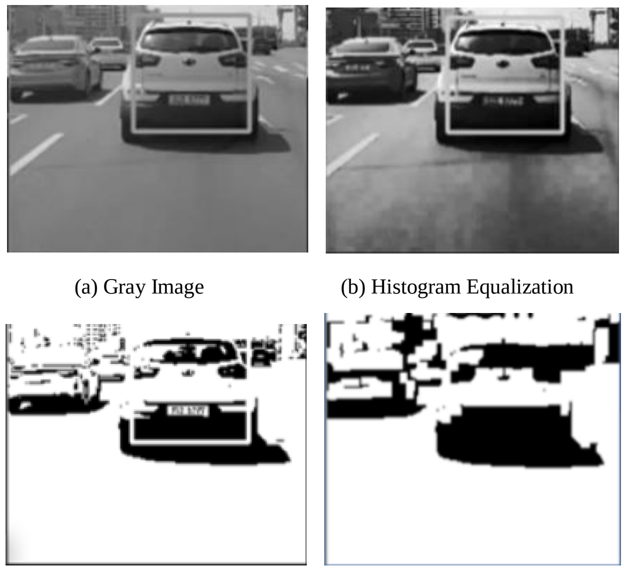
III. Distance Computation and Hrdware

A. Distance Compuation

In single image, we figure out distance between the vehicles by one half of the width of vehicles through the camera sensor. The utilize properties of camera that vehicle is projected on image plane are shown in Figure 12. We use W for the width of vehicle, D for the distance between the vehicle and camera, w for the width of vehicle when image is projected on the image plane, f for the focal distance to convert the pixel. Finally, we find $W:D = w:f$. Hence, the distance between vehicles can simply be calculated as

$$D = \frac{f \times W}{w} \tag{4}$$

We execute experiment by general car to get experimental data of actual width of vehicle and the actual distance between vehicle and camera. Since every vehicle has different width,



(a) Gray Image (b) Histogram Equalization

(c) Binary Image (b) Erosion and Dilation

Figure 10. Process of detecting shadow of vehicle

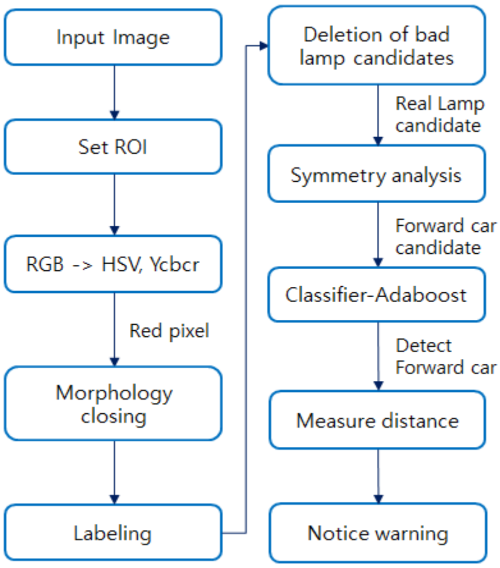


Figure 11 Image procesing block diagram

therefore, we used a general car design with the constant value by assuming a marginal error with the actual distance. Figure 13 shows the computation of distance experiment.

The detailed process is explained through the block diagram as shown in Figure 11.

B. Rear-end Accident Notifying Hardware

To implement detecting vehicle and distance estimation algorithm, we use Raspberry Pi 3 embedded board based on Linux and design circuit as shown in Figure 14. This circuit gives prompt alert to the drivers. We used embedded board conditions for input image by variable camera on real-time. Later, we measure the risk of rear-end accident by image-processing and transmit variable situational notifications to drivers as Dot Matrix and Text LCD of Arduino board as shown in Figure 15 . Moreover, we used the auditory alert by speaker before the risk of accident appears.

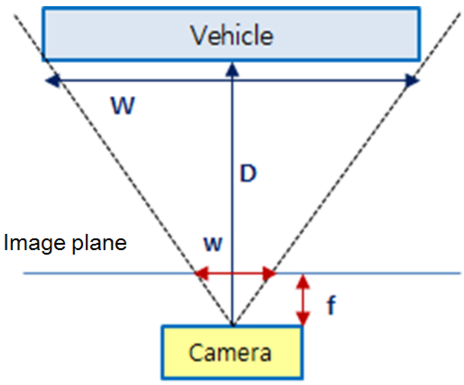


Figure 12. The result images of detected vehicle

Figure 13. Computation distance of detected vehicle

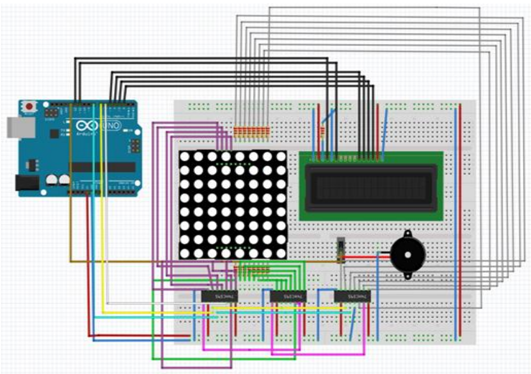


Figure 14. Circuit of output device

C. *Experiemntal result*

The experiment was processed by the front image which is extracted from the actual vehicle camera and applied on city-drive image under clear and cloudy day conditions. We used C920 camera produced by Logitech and rasberry cam module. The processing speed at PC is 25~29 Fps and at embedded conditions is 10~15 Fps. The calculated detection rate is the results of the experiment that comprises of almost 1000 pages of image. Our results showed about 95% of the detection rate.

D. *Analysis result*

We compare the experimental results with the analytical results of detection algorithm using the shadow information.

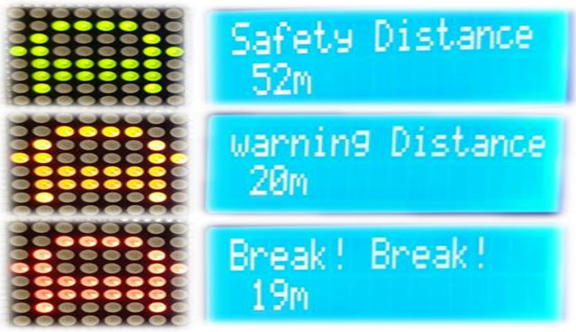


Figure 15.Accident Notification output device

We got 90% of accuracy due to error detection problems and disturbances such as lights.

I. CONCLUSION

In this paper, we have proposed a new vehicle crash avoidance system. We detect the vehicles from the characteristics of rear lamps and classify the vehicles using the Haar-like feature and Adaboost algorithm. Finally, we compute the distance by the width of the front vehicle. For the realtime processing, we set the Region of Interest (ROI) and use the integral image. Our system can be affected by the camera properties since we depend on the color information. To deal with this problem, we could use the edge information in addition to the color, and improve the robustness of the system. As a future research, we will apply the Kalman filter or the mean shift tracking method to enhance the detection performance. Furthermore, we will use the ultrasonic sensor to figure out precise distance between vehicles while utilizing the relative velocity of vehicles from the acceleration sensor.

II. REFERENCES

[1] Ye Li, Qingming Yao, "Rear Lamp Base Vehicle Detection and Tracking for Complex Traffic Conditions", Beijing Engineering Research Center.

[2] Ronan O'Malley, Edward Jones, Real Lamp Vehicle Detection and Tracking in Low-Exposure Color Video for Night Conditions, IEEE Transactions On Intelligent Trasportation Systems, 11(2), 2010.

[3] http://www.data.go.kr/comm/file/download.do?atchFileId=FILE_000000001292615&fileDetailSn=1