# Vehicle Detection Method using Haar-like Feature on Real Time System

Sungji Han, Youngjoon Han and Hernsoo Hahn

***Abstract*—**This paper presents a robust vehicle detection approach using Haar-like feature. It is possible to get a strong edge feature from this Haar-like feature. Therefore it is very effective to remove the shadow of a vehicle on the road. And we can detect the boundary of vehicles accurately. In the paper, the vehicle detection algorithm can be divided into two main steps. One is hypothesis generation, and the other is hypothesis verification. In the first step, it determines vehicle candidates using features such as a shadow, intensity, and vertical edge. And in the second step, it determines whether the candidate is a vehicle or not by using the symmetry of vehicle edge features. In this research, we can get the detection rate over 15 frames per second on our embedded system.

***Keywords*—**vehicle detection, haar-like feauture, single camera, real time

## I. Introduction

RECENTLY, the demand of automatic system is getting more and more. They want everything to be automatic. This is no exception for drivers. Most people want a vehicle that drives itself by just giving it orders. But it's still not easy to realize it. On these days, the research about vehicle detection has been progressing lively. This paper has researched a vision system for the intelligent vehicle too. The vision system captures images from a single camera in front part of a car window. Generally, a camera gives the vision system much information such as a pedestrian [1], a road and traffic sign [2], and a obstacle [3], etc. In this paper, the goal is to detect vehicles whose rear part are seen totally.

Until now a lot of researches have proposed vehicle detection algorithms. Most researches have been well arranged. Especially, the vehicle detection method in [4] is separated into two steps. The hypothesis generation step uses the knowledge, stereo vision and motion based methods. Knowledge-Based method uses many features (symmetry, color [5], shadow, corner, edge, and texture).

Sungji Han is with the Electronic Engineering Department, Soongsil University, Seoul, Korea. (corresponding author to provide phone: 82-2-821-2050; e-mail: hsj0224@ ssu.ac.kr).
Yoongjoon Han is with the Electronic Engineering Department, Soongsil University, Seoul, Korea. (corresponding author to provide phone: 82-2-821-2050; e-mail: young@ ssu.ac.kr).
Hernsoo Hahn is with the Electronic Engineering Department, Soongsil University, Seoul, Korea. (corresponding author to provide phone: 82-2-821-2050; e-mail: hahn@ ssu.ac.kr).

Our proposed method uses only the shadow and edge feature because they are consistent to the vehicle detection. And hypothesis verification step is separated into two methods: the template and appearance based method [6]. They are also based on the training or learning algorithm since the execution time of methods using databases tends too slow to be applied to an embedded system.

In order to use it on the real-time system, we use edge features of vehicles without using databases. To solve the problem, we propose a vehicle detection algorithm using Haar-like edge features [7]. By using this Haar-like feature we can get more accurate and faster result. This paper's overall flowchart is as follows.

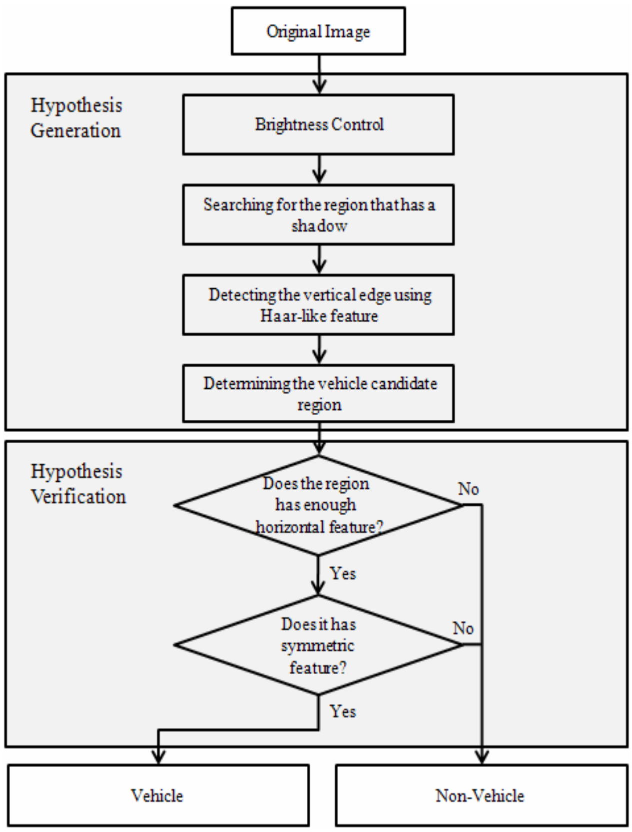



Fig. 1 Overall Flowchart

## II. Hypothesis Generation

Before hypothesis verification step, some candidate regions are determined whether it contains vehicle or not.

As we detect some features that only appear the region of vehicles, we can determine the vehicle candidate regions. In this paper, some specific features are used to determine vehicle part. The first specific feature is shadow. The shadow saliently appears under the vehicles. And the second feature is a vertical edge. The vertical edge mostly appears to the right and left part of the vehicles.

### *A. Detecting the distant region using a shadow feature*

Generally, a feature of shadow provides the significant information for finding any objects. It can be used to remove the useless region too. But it is not easy to get the object region with shadow because shadow tends to depend on the brightness of image. In order to compensate for this variation, the brightness of image needs to be normalized. By the brightness normalization make sure that a vehicle candidate region is reliably detected.

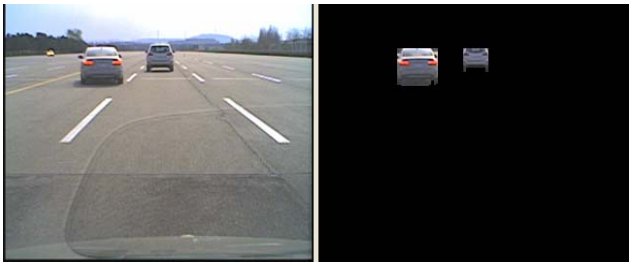
(a) Original Image (b) Dark Image (c) Result Image

(d) Original Image (e) Bright Image (f) Result Image

Fig. 2 Auto intensity control considering the image brightness

Fig.2 (a) shows the case that brightness of original image is too dark. So it is difficult to separate the shadow part on the image. But Fig.2(c), normalized the brightness, is brighter than Fig. 2(b). The shadow part of the vehicle is clearly separated from the road. And the second case is when the brightness of original image is too bright (Fig.2(d)). So the shadow that appears in under the vehicles is also removed. For this reason it is also hard to separate. But after normalizing the brightness, it can be clearly separated too. Fig.2(e) shows that the shadow of vehicle is diminished in order not to be normalized. So by optimizing brightness it is important to get each shadow point when the image is too bright or dark.

(a) Original Image (b) The image with noise removed



(c) Original Image (d) The image with noise removed

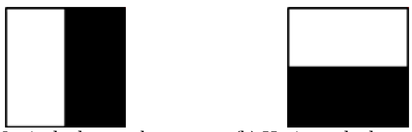
Fig. 3 The image expressed above shadow point

After the normalization of the image brightness, the region that contains shadow can be found. After determining the shadow point in each height, this paper obtains just the part above the shadow point as shown in Fig. 3. By using this method, the distinct regions have high possibility being detected as vehicles.

### *B. Detecting vertical edge using Haar-like feature*

The second salient feature that appears at the vehicle is the vertical edge. The vertical edge feature mostly appears at the vehicle's right and left parts. So the vertical edge provides the important information for vehicle detection.

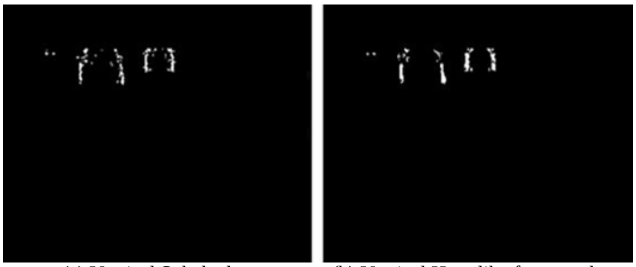
In order to get stronger and clearer edge, this paper uses Haar-like feature. The vertical edge obtained by Haar-like feature is more robust and clearer than any other edge features(Sobel edge feature, Prewitt edge feature, etc)[7]. In this paper, 6 * 6 Haar-like mask is used to get the edge image. The integral image method that was proposed by Paul Viola and Michael Jones is used for this paper. our proposed algorithm is very fast.



(a) Vertical edge mask (b) Horizontal edge mask

Fig. 4 Haar-like mask to detect vertical and horizontal edges

In order to search the vertical edge we used Haar-like feature like "Fig. 4(a)". By using this method we can get very strong edge feature.



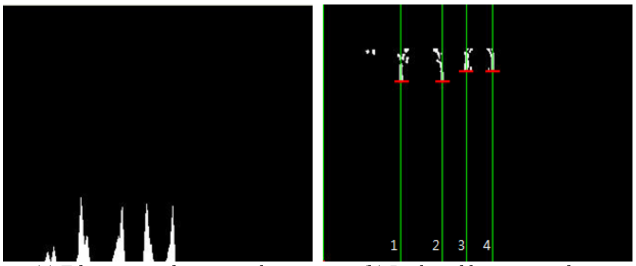
(a) Vertical Sobel edge (b) Vertical Haar-like feature edge

Fig. 5 Comparison of Sobel edge and Haar-like feaure edge

Fig.5 compare the result (Fig. 5(a)) of the vertical Sobel operator with the one(Fig.5(b)) of the vertical Haar-like feature. We can recognize that Haar-like feature is superior to Sobel operator.

### *C. Determining the candidate regions*

By using above two features, candidate regions of vehicles are determined. First, distinct regions with the shadow region are found. Second, the vertical edge image is obtained to detect the vehicle candidate region. The next step is to determine the candidate region of vehicle using the edge accumulation graph.




(a) Edge accumulation graph (b) Peak and bottom result

Fig. 6 Edge accumulation graph & Peak and bottom result

Fig. 6(a) shows the accumulation graph of edge image. By using this graph, a valid point that expresses the bottom point of the vehicle is obtained. In Fig. 6(a), we call a peak point as a valid point. In Fig. 6(b), the valid point is expressed as green line. After finding valid points, the proposed algorithm searches the pair of the valid points for vehicle candidates. Before determining a pair of valid points, bottom points of vehicle are found in Haar-like edge image. It's not difficult to find the bottom points because of just finding last point of peak point. The red line in Fig. 6(b) expresses bottom points.

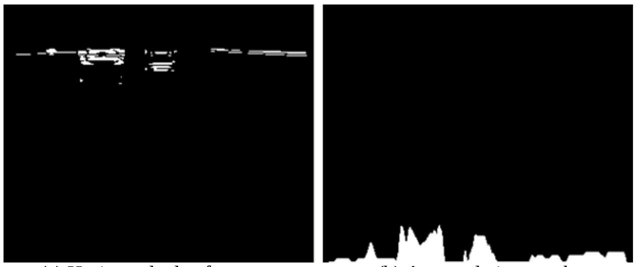
In order to obtain more reliable bottom points, the proposed algorithm compares them with shadow points. After finishing this work of determining bottom points, it is possible to find a pair of valid points. If any pair of valid points have the similar height of bottom points, it could be the one of the same vehicle. For example in Fig. 6(b), the pair of valid points (1 and 2) can belong to one of vehicles. And 3, 4 also can be pair. Only if the vehicle is detected, vehicle's top point is not so important information. So this paper empirically determines the height of the vehicle candidate to 2/3 of its width.

## III. Hypothesis Verification

After vehicle candidates are determined, we must verify whether the candidates are vehicle or not. In this hypothesis verification step of determining whether the vehicle is real or not, several methods are considered in this paper. The first method of verification uses the complexity of horizontal edge feature obtained by horizontal Haar-like feature. And the second method uses the symmetric properties of the vehicle. Most rear parts of vehicles have symmetric feature. It's one of features that appear continuously without any changes. So the symmetric feature of the vehicle is most important feature for detecting vehicles.

### *A. Verification using complexity of vehicles*

In order to detect the vehicles with high probability the more times this verification process is repeated, the better it is. Most important thing is to achieve maximum success rate using minimum algorithms.



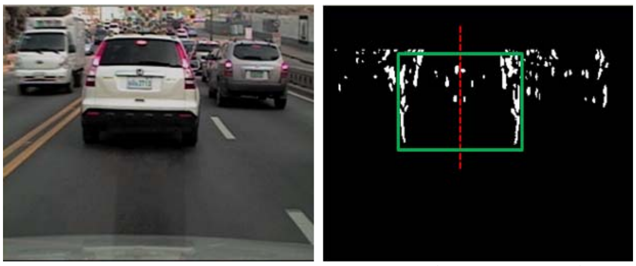
(a) Horizontal edge feature (b) Accumulation graph

Fig. 7 Horizontal edge feature using Horizontal Haar-like edge mask

This horizontal edge feature is also obtained by using horizontal Haar-like feature. As shown in Fig. 7(a), the Horizontal edge feature is salient on vehicle region. The candidates that are not vehicle are removed by the horizontal edge feature. Sometimes, the candidate can be overlapped when it is placed near another vehicle. For example, if bottom lines of vehicle are placed on the similar height such as Fig. 6(b), (1, 2), (2, 3), (3, 4) can be pair respectively. But by inspecting the complexity of horizontal edge, such as a pair of the valid points (2, 3) can be removed. But a problem occurs when some objects have horizontal edge feature. To solve the above problem, this paper uses the symmetric feature of vehicles.

### *B. Final verification using vehicle's symmetric feature*

Because most vehicles have symmetric feature, edge image of vehicle tends to be symmetric. Therefore the proposed method uses this symmetric feature to finally verify whether the candidate is vehicle or non-vehicle. A lot of features may be compared for detecting real vehicles. There are features such as an edge, intensity, and color. In this paper we consider only the symmetry feature of the vertical edge. This feature appears saliently when the vehicle is getting closer to camera.



(a) Original Image (b) Example of edge symmetry feature

Fig. 8 Verification using symmetry feature

In Fig. 8(b), the green rectangular box represents the candidate region. And the red line represents center line of candidate region. When this algorithm is used, the center line on candidate region is firstly calculated. And the matching rate is as follows:

$$M(r) = \frac{\sum_{n=1}^{x}\sum_{m=1}^{y} W(n, m)}{\sum_{n=1}^{x}\sum_{m=1}^{y} T(n, m)} * 100 \qquad (1)$$

where T represents the number of total pixel in candidate region, and W is the number of white pixel in candidate region.

## IV. Experimental Result & Consideration

Before applying the proposed algorithm to an embedded system, the execution time has been considered. The following Table I shows the time comparison of each edge detection methods(Sobel, Prewitt, and Haar operator).

TABLE I. Time comparison of each edge detecion methods

| Detection Methods | *Sobel* | *Prewitt* | *Haar* |
|---|---|---|---|
| **Time(ms)** | 2 | 2 | 3 |

As shown in Table I, Haar-like method is so useful that it saves the execution time within 3ms. This result is calculated on desktop computer of Intel Core 2 Duo CPU 3.0Ghz, 2GB RAM.

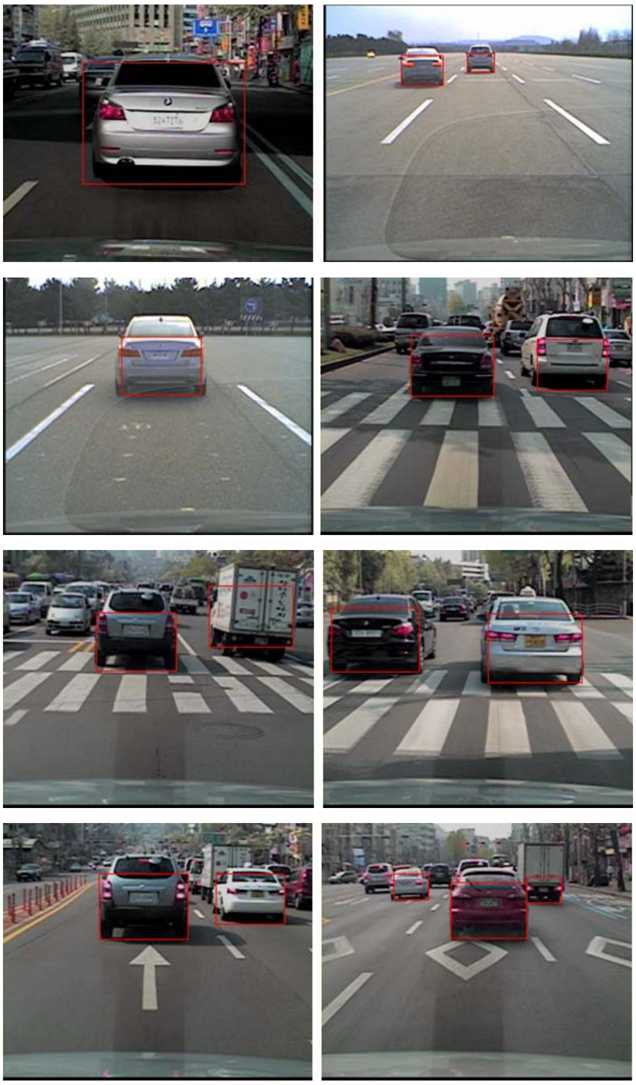


Fig. 9 Result Images

In the Fig. 9, red rectangular boxes express detected vehicles. It normally takes within 0.03s, about 33 fps, to execute on a desktop computer. And it can process over 15fps on the embedded system, or real-time system.

This method is robustly applied to a road image with complex mark. Because it considers only the vehicle with the shadow, it has the good result even though the road image contains the complex road mark as shown in Fig.8. The table II shows detection rate on the two situations.

TABLE II. Detecting Rate

| Position | *Same Lane* | *Side Lane* |
|---|---|---|
| **Detecting Rate(%)** | 91.2 | 76.2 |

These detecting rate are calculated by the image that obtained a highway and an expressway. Because the symmetry of vehicle is used, the vehicle only with the whole rear part can be detected. So the vehicle on the same lane is detected better than the vehicle on the side lane. Some error results are shown below. Most errors are resulted from image brightness. It depends on working environment.


(a) The case of too bright (b) Complex & Too dark case

Fig. 10 Error Results

Fig. 10(a) is the case that the image is too bright. Because the circumstance is too bright, the shadow under the vehicle is also too bright. And Fig. 10(b) is the case that the image is too dark. In both two cases the shadow detection is failed. So vehicle is not detected. Moreover in Fig. 10(b) there are many obstacles. Sometime, non-vehicle object may be detected as vehicle. And when the vehicle changes the lane, the side and rear parts of a vehicle are shown at the same time. In that situation, the edge feature is getting weak, and sometimes a vehicle couldn't be detected.

## V. Conclusion

This paper proposed a method using Haar-like feature and the symmetric feature in order to detect vehicles. Haar-like feature is used to obtain stronger feature. And we obtained the more reliable detection result by applying the symmetric feature to verify the vehicle candidate. In fact, the proposed algorithm was aimed to detect the vehicle on real time system. The execution time of the proposed algorithm was very short. So, it is proper to apply it to the real-time system. But it has a limit to detect vehicles using single camera. And an error was sometime generated according to illumination changes of road environment. So we have to make the proposed algorithm more robust to work on embedded system.

## Acknowledgment

"This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD). This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute for Information Technology Advancement)" (IITA-2009-(C1090-0902-0007))

## References

[1] F. Moutarde, B. Stanciulescu, A. Breheret, "Real time visual detection of vehicles and pedestrians with new efficient adaboost features," IEEE International Conference on Intelligent Robots Systems (IROS 2008), September 2008.

[2] H. Fleyeh, "Color detection and Segmentation for road and Traffice signs", Proceedings of the 2004 IEEE Cybernetics and Intelligent Systems, vol. 2, pp. 809-814, December 2004

[3] R. Labayrade, D. Aubert, JP. Tarel, "Real Time Obstacle detection in stereovision on Non flat road geometry through 'V-disparity' Representation", Proceedings of IEEE Intelligent Vehicle Symposium, vol. 2, pp. 646-651, 2002

[4] R. Miller, Z. Sun, G. Bebis, "On Road vehicle Detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, pp. 694-711, May 2004

[5] J. Rojas, J. Crisman, "Vehicle Detection in Color Images", in Proc. IEEE Conference Intelligent Transportation, pp. 403-408, November 1998

[6] G. D. Sullivan, K. D. Baker, A. D. Worrall, "Model based vehicle detection and classification using orthographic approximations", British Machine Vision Conference, vol. 15,Issue 8, pp. 649-654, August 1997

[7] P. Viola, M. jones, "Rapid Object Detection using a Boosted Cascade of Simple Features"In Proc. IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 511-518, 2001