

# Blind-Spot Vehicle Detection Using Motion and Static Features

Din-Chang Tseng, *Member, IACSIT*, Chang-Tao Hsu, and Wei-Shen Chen

**Abstract**—When driving a vehicle on a road, if a driver want to change lane, he must glance the rear and side mirrors of his vehicle and turn his head to scan the possible approaching vehicles on the side lanes. However, the view scope by the above behavior is limited; there is a blind spot area invisible. To avoid the possible traffic accident during lane change, we here propose a lane change assistance system to assist changing lane. Two cameras are mounted under side mirrors of the host vehicle to capture rear-side-view images for detecting approaching vehicles. The proposed system consists of four stages: estimation of weather-adaptive threshold values, optical flow detection, static feature detection, and detection decision. The proposed system can detect side vehicles with various approaching speed; moreover, the proposed system can also adapt variant weather conditions and environment situations. Experiment with 14 videos on eight different environments and weather conditions, the results reveal 96 % detection rate with less false alarm.

**Index Terms**—Advanced driver assistance system, blind spot detection, optical flow, underneath shadow features.

## I. INTRODUCTION

In these few decades, the rapidly-increasing vehicle number and factors of road situation, driving environment, and human attention make a large amount of traffic accidents and casualties. If there is a mechanism to help drivers to detect the road situation and driving environment, and then provide useful information to warn drivers in these situations, the danger is therefore avoided. The advanced driver assistance systems (ADAS) are actively developed to help drivers avoiding the possible dangers and assist drivers to drive in special environments [1]. Blind spot detection (BSD) system is a mechanism of ADAS for assisting driver to change lane. If there are near vehicles on the destination lane where the host vehicle want to change into, the system will alarm drivers to stop changing.

A beneficial BSD system should have the properties: accuracy, stable, high-speed execution, and low price. Two kinds of BSD systems have been developed to detect overtaking vehicles; one is radar-based systems and the other is vision-based systems. The radar-based systems are stable and have good performance, but the size is large, the detection function is limited, and the price is very expensive [1]-[3]. The vision-based systems capture a consequence images for

detection; moreover, both static and motion information can be used in the detection process. The static-feature detection methods extract features such as corner points, edge points, and underneath shadow of vehicles to detect possible approaching vehicles [4]-[9]. The motion detection methods used motion information such as optical flow to detect approaching vehicles on a side lane with various speeds [10]-[12].

The main challenge of the vision-based detection methods is to adapt various weather conditions and environment situations. The static methods may get false alarm to the static objects, build shadows, and traffic signals on ground. The optical-flow vectors provide the motion information for detecting relative moving vehicles; however, only using optical-flow features cannot detect the similar-speed side vehicles [13], [14]. Moreover, different-length optical flows may not always be detected in a long distance range of approaching vehicles on the side lane.

Steolo *et al.* [15] proposed a vehicle detection method using optical flow and Kalman filter. At first, they detected edge points; these edge points are then clustered into groups. Third, they calculated the optical-flow vectors of the edge points in each group. Fourth, the optical flows of groups were used to generate vehicle candidates. At last, a support vector machine (SVM) algorithm was used to recognize whether a vehicle candidate is a front part of a vehicle or not. The detection rate of the method is dependent on the training samples for the SVM. An approaching vehicle has different appearances in various distances; this fact influences the recognition rate of the SVM algorithm. Moreover, the similar-speed vehicle needs extra Kalman filter to decide; the process is time consuming and more complicated.

To benefit the performance of BSD systems, we here separately use static and motion features to detect approaching vehicles and then combine the detection results to improve the detection rate and reduce false alarm rate. The proposed blind-spot detection system is described in Fig. 1. The system consists of four stages: estimation of weather-adaptive threshold values, multi-resolution optical flow detection, static feature detection, and detection decision.

The proposed approach possesses the following properties. (i) We specially deal with the used threshold values such that the proposed system is able to adapt various weather conditions. (ii) The special-deal threshold values also make the proposed system being not influenced by building shadows and traffic signals on ground. (iii) We use multi-resolution optical flow to extract moving vehicles such that vehicles can be detected in far or near distance. (iv) We combine static and motion features to detect vehicles such that

Manuscript received May 12, 2014; revised June 28, 2014. This work was supported in part by the National Science Council, Taiwan under the grant of the research project NSC 100-2221-E-008-115-MY3.

Din-Chang Tseng, Chang-Tao Hsu, and Wei-Shen Chen are with the Institute of Computer Science and Information Engineering, National Central University, Jhongli, Taiwan 32001 (e-mail: tsengdc@ip.csie.ncu.edu.tw, justinhsuncu@gmail.com, easywine2@gmail.com).

the similar-speed side vehicles can also be detected. (v) The static and motion features are sequentially treated to detect moving vehicles and reduce the false alarm.

The remainder of this paper is organized as follows. The proposed method is presented in Section II. In Section III, experiment and their results are reported to demonstrate the performance of the proposed methods. At last, the conclusions are presented in Section IV.

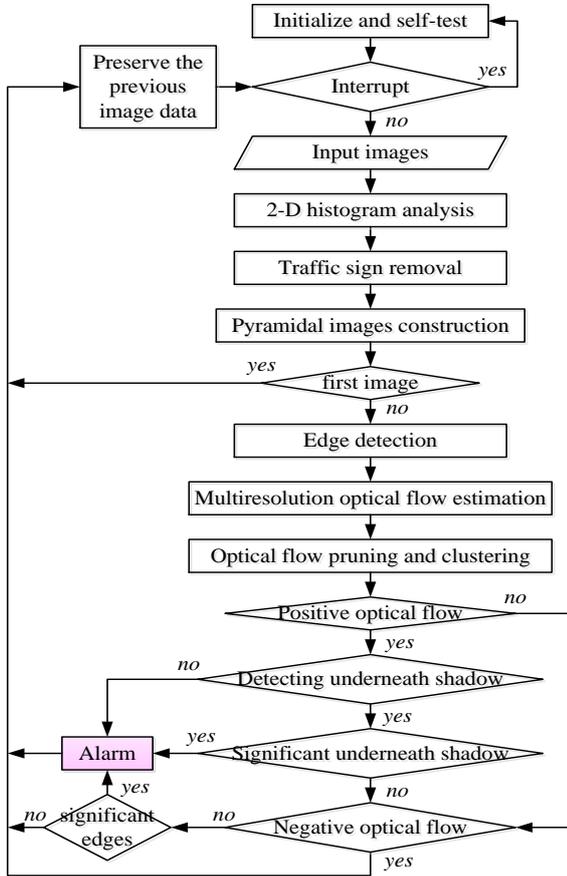


Fig. 1. The procedure of the proposed method.

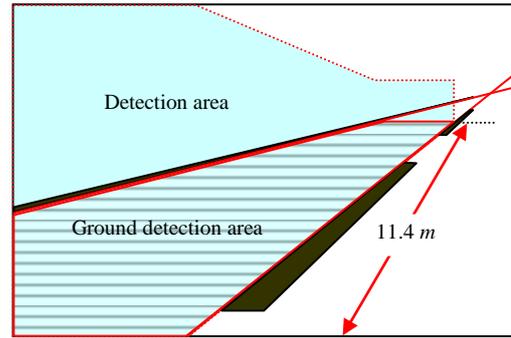
## II. THE PROPOSED METHOD

The procedure of the proposed system is sequentially presented as follows. Before vehicle detection, the detection area in images is defined as shown in Fig. 2(a) to avoid the interference of objects on the background. Moreover, to concentrate the local detection and to distinct different threaten degrees, the ground detection area is further divide into four far to near regions  $R_1$ ,  $R_2$ ,  $R_3$ , and  $R_4$  as shown in Fig. 2(b), where the ground detection area is included in the detection area to specify targets on the ground.

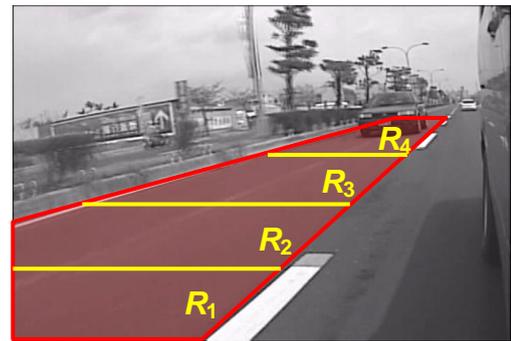
At different distances to the host vehicle, the side vehicles have different sizes on images; this is called the perspective effect in images. At long distance, the side vehicle appears small and is detected unstable. If we release the threshold value, many false alarms may then arise. That is, the perspective effect always influences the detection rate. To improve the performance, we define a distance weighting function to compensate the perspective effect. If  $j$  is the vertical coordinate of images, the distance weighting function  $w(j)$  is defined as

$$w(j) = 1/\text{length}(j), \quad (1)$$

where  $\text{length}(j)$  is the length of a line segment within the ground detection area at  $j$  vertical image coordinate as shown in Fig. 3, where partial  $\text{length}(j)$  is outside of image.



(a)



(b)

Fig. 2. Definition of detection area. (a) The definition of detection area. Four detection regions of the ground detection area.

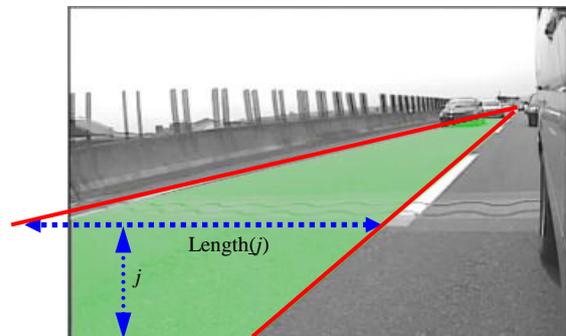


Fig. 3. The definition for distance weighting function.

### A. Estimation of Weather-Adaptive Threshold Values

In this study, we used three threshold values to judge edge pixels, lane marks, and underneath shadows of vehicles (USV); however, the threshold values are variant to the weather conditions. Here we provide a 2D histogram analytic method to adaptively extract the three threshold values.

An (1D) intensity histogram is a number distribution of intensity of pixels in an image. The proposed 2D histogram is similar to the traditional 1D histogram but with two independent variables; one is intensity and the other is gradient magnitude, as one example shown in Fig. 4 (b). The origin of the 2D histogram is the lower-left corner, the horizontal axis represents different intensities of pixels, and the vertical axis represents different gradient magnitudes of pixels. The brightness of a point in the 2D histogram represents the number of pixels with specific intensity and

gradient magnitude.

In general, only four materials: asphalt road surface, lane marks and traffic signals, underneath shadows of vehicles (*USV*), and vehicle bodies appear in the ground detection area. Moreover, the pixels of every material show similar gray level and then form a normal (Gaussian) distribution in brightness. Lane marks and traffic signals on the ground are brighter than asphalt road surface and have sharp edges. Underneath shadows of vehicles are always darker than road surface. Vehicle bodies may brighter or darker than road surface; they cannot be separated by brightness, but they have different gradient magnitudes. Based on the above facts, we want to find the threshold values to separate the four materials on the ground detection area.

The pixels on the ground detection area are represented by a multivariate Gaussian mixture model (*GMM*) to model their distribution in the 2D histogram. At first, 2D histogram is smoothed by a bilateral filter to eliminate noise but preserve the edge information. Second, the parameters of *GMM* are estimated by means of the Expectation-Maximization (*EM*) algorithm to maximize the posterior probability [16]. Assume that there are  $K$  Gaussian functions and the  $k$ -th Gaussian function is represented by  $G_k = G(\alpha_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ ,  $k = 1, 2, \dots, K$ , where  $\alpha_k$  is a mixing parameter for the  $k$ -th Gaussian function,  $0 \leq \alpha_k \leq 1$ , and  $\sum_{k=1}^K \alpha_k = 1$ .  $\boldsymbol{\mu}_k$  and  $\boldsymbol{\Sigma}_k$  are the 2D mean vector and covariance matrix of the  $k$ -th Gaussian function, respectively.

The *EM* algorithm is iterative procedure for modifying model parameters to a stable state. Assume that there are  $N$  pixels  $\mathbf{x}_i$ 's on the ground detection area;  $K$  Gaussian classes are considered. The model parameters  $\alpha_k$ ,  $\boldsymbol{\mu}_k$ , and  $\boldsymbol{\Sigma}_k$  for the next step are iteratively updated as

$$\begin{cases} \boldsymbol{\mu}_k^{t+1} = \frac{\sum_{i=1}^N p^t(k | \mathbf{x}_i) \mathbf{x}_i}{\sum_{i=1}^N p^t(k | \mathbf{x}_i)} \\ \boldsymbol{\Sigma}_k^{t+1} = \frac{\sum_{i=1}^N p^t(k | \mathbf{x}_i) \|\mathbf{x}_i - \boldsymbol{\mu}_k^t\|^2}{\sum_{i=1}^N p^t(k | \mathbf{x}_i)}, \\ \alpha_k^{t+1} = \frac{1}{n} \sum_{i=1}^N p^t(k | \mathbf{x}_i) \end{cases} \quad (2)$$

where

$$p^t(k | \mathbf{x}_i) = \frac{\alpha_k^t p^t(\mathbf{x}_i | k)}{p^t(\mathbf{x}_i)} = \frac{\alpha_k^t p^t(\mathbf{x}_i | k)}{\sum_{j=1}^K \alpha_j^t p^t(\mathbf{x}_i | j)} \quad (3)$$

is the posterior (conditional) probability for a given pixel  $\mathbf{x}_i$  belonging to the  $k$ -th model  $G_k$  at the  $t$ -th step,  $p^t(\mathbf{x}_i | k)$  is the prior (conditional) probability of  $\mathbf{x}_i$  while  $\mathbf{x}_i$  is selected from the  $k$ -th model.

The recursive refinement is executed until sum of the model error is less than a pre-defined threshold value. One 2D histogram is then modeled by a Gaussian mixture model and shown in Fig. 5.

Consequently, the largest group  $N_i$  (i.e., the Gaussian function whose  $\alpha_i$  is largest) is selected as the decision group (*DG*) which will be used to find the three threshold values. The threshold values are decided as shown in Fig. 6.

If  $\boldsymbol{\mu} = (\mu_x, \mu_y)$  and  $\boldsymbol{\Sigma} = (\sigma_x, \sigma_y)$  are the mean and standard deviation vectors of the largest 2D Gaussian model,

respectively, then the three threshold values  $T_{imin}$ ,  $T_{imax}$ , and  $T_e$  are determined as

$$\begin{aligned} T_{imin} &= \mu_x - 2\sigma_x, \\ T_{imax} &= \mu_x + 2\sigma_x, \text{ and} \\ T_e &= \mu_y + 2\sigma_y, \end{aligned} \quad (4)$$

where  $T_{imin}$  is used to separate the asphalt road surface and underneath shadows of vehicles (*USV*),  $T_{imax}$  is used to separate the asphalt road surface and the lane marks/traffic signals, and  $T_e$  is used to extract significant edge pixels which will be used to compute valuable optical flows. Before motion and static feature extraction, the pixels belonging to lane marks or traffic signals on ground are pre-eliminated to reduce false detection.

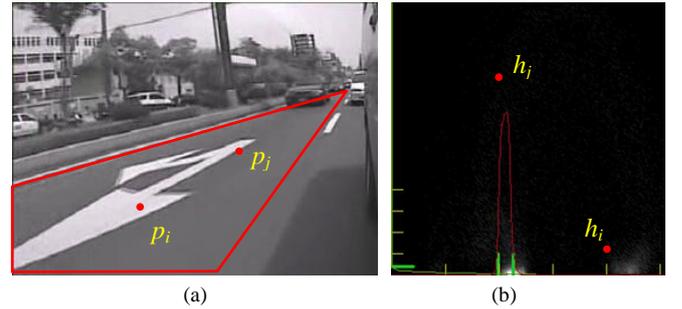


Fig. 4. The proposed 2D histogram. (a) A blind-spot image with pixels  $p_i$  and  $p_j$  in the ground detection area. (b) The corresponding 2D histogram of the ground detection area; where  $h_i$  and  $h_j$  are the corresponding point of  $p_i$  and  $p_j$  in the 2D histogram.

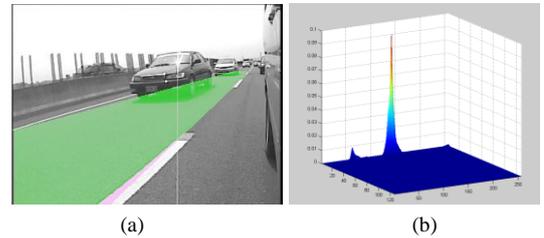


Fig. 5. The 2D histogram of a ground detection area. (a) An original image. (b) The *GMM* of 2-D histogram of (a).

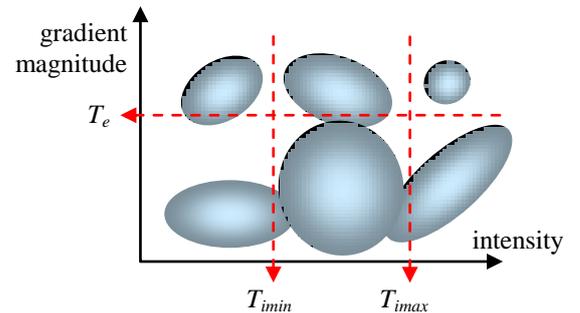


Fig. 6. Threshold values are defined from the largest group of pixels in a 2D histogram.

### B. Multi-Resolution Optical Flow Detection

Optical flow is the main feature for detecting approaching vehicles. Multi-resolution optical flows were estimated by the least-squared version of *L-K* differential optical flow estimation method [17]. The optical flows are estimated from significant edge points and the edge points are selected based on the pre-determined edge threshold value  $T_e$  described in the last section.

The procedure of optical-flow detection consists of three main stages: estimation, filtering, and clustering. In the estimation stage, the ground detection area is divided into three subareas as shown in Fig. 7 to calculate three levels of optical-flow vectors. The three subareas are named  $O_1$ ,  $O_2$ , and  $O_3$  from near to far on the ground detection area with decision boundaries at  $x = d_1$ ,  $d_2$ , and  $d_3$ . Three levels of optical flows are estimated for subarea  $O_1$ , two levels of optical flows are estimated for  $O_2$ , and only one level of optical flows is estimated for  $O_3$ . Multi-level optical flows are estimated by a coarse-to-fine strategy and the coarser optical flows have been propagated to the finer level to calculate finer optical flows such that a long range of optical flows can all be extracted.

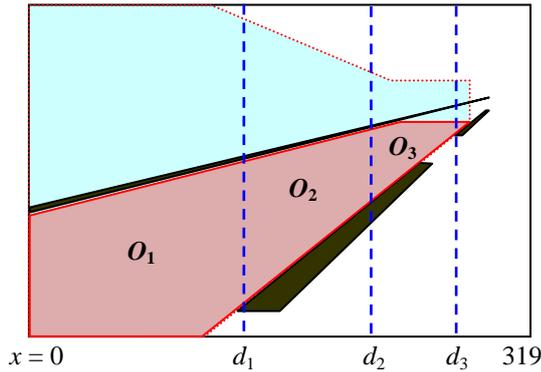


Fig. 7. Three subareas of ground detection area are defined for estimating multi-level optical flows.

In the filtering stage, optical-flow vectors are filtered by their lengths and directions. Different-distance approaching vehicles have different-length optical-flow vectors even they are in the same speed; thus the optical flows are separated into three classes to filter. The classification is the same as the optical-flow subareas  $O_1$ ,  $O_2$ , and  $O_3$  shown in Fig.7. Three subareas use different threshold values to filter optical flows. The decision rules for length are defined as

$$\mathbf{u} \text{ is remained, if } \begin{cases} \mathbf{u} \in O_1 \text{ and } |\mathbf{u}| > 2.00 \text{ pixels,} \\ \mathbf{u} \in O_2 \text{ and } |\mathbf{u}| > 1.00 \text{ pixels,} \\ \mathbf{u} \in O_3 \text{ and } |\mathbf{u}| > 0.25 \text{ pixels,} \end{cases} \quad (5)$$

where  $\mathbf{u}$  is an optical-flow vector.

The direction of an optical-flow vector is useful for judging a feature point approaching or leaving the host vehicle. In this study, optical-flow vectors are classified into positive and negative optical flows. A positive optical flow is an approaching motion vector and a negative optical flow is a leaving motion vector relative to the host vehicle. The decision rules for positive and negative optical flows are defined as

$$\mathbf{u} \text{ is a positive optical flow,} \\ \text{if } \{ \mathbf{u} \in O_1 \text{ and } -20^\circ < \theta < 20^\circ \} \text{ or} \\ \{ \mathbf{u} \in O_2 \text{ and } -15^\circ < \theta < 15^\circ \} \text{ or} \\ \{ \mathbf{u} \in O_3 \text{ and } -10^\circ < \theta < 10^\circ \} \quad (6)$$

$$\mathbf{u} \text{ is a negative optical flow,} \\ \text{if } \{ \mathbf{u} \in O_1 \text{ and } 160^\circ < \theta < 200^\circ \} \text{ or} \\ \{ \mathbf{u} \in O_2 \text{ and } 165^\circ < \theta < 195^\circ \} \text{ or} \\ \{ \mathbf{u} \in O_3 \text{ and } 170^\circ < \theta < 190^\circ \} \quad (7)$$

where  $\theta$  is the phase angle of  $\mathbf{u}$ .

At last, all remained positive and negative optical flows are clustered into groups by a simple clustering method. The results of positive and negative optical-flow groups are shown in Fig. 8. In the figure, the red blocks are positive groups and blue blocks are negative groups.

To concentrate the local detection and to distinct different threaten degrees, a detection mechanism is proposed based on the extracted positive optical flows.

In the detection area, we accumulate the numbers of positive optical flows in every detection regions  $R_i$ ,  $i = 1, \dots, 4$ . If  $R_i$  has more than  $m$  optical flows, we say that there is a vehicle candidate in  $R_i$  and then increase  $a_i$  by one. Initially,  $a_i$  is zero. As the time is progressive, the captured images are sequentially detected one by one. If a vehicle candidate appears in  $R_i$ ,  $a_i$  is increased by one until  $a_i$  equals six; if there is no vehicle candidate in  $R_i$ ,  $a_i$  is decreased by one until  $a_i$  equals zero. At any minute, if condition

$$\begin{cases} 5 < a_i + a_{i+1} < 12, i = 1, 2, 3 \text{ or} \\ 3 < a_i < 6, i = 1, 2, 3, 4 \end{cases} \quad (8)$$

is met, one approaching is said to be detected based on the motion features.



Fig. 8. Examples of optical flow clustering: (a) Positive optical-flow groups. (b) Negative optical-flow groups.

### C. Static Feature Detection

To avoid the false detection of similar-speed vehicles from motion-feature detection, we continuously uses two static features: underneath shadow of vehicles and sum of edge magnitudes to extract all possible vehicle candidates on the ground detection area. Based on the defined threshold value  $T_{min}$  described in Section II-A, the pixels whose gray levels are less than  $T_{min}$  are classified as members of underneath shadow of vehicles. These pixels are then clustered. If any detection region  $R_i$  has a significant cluster of underneath shadow pixels, a vehicle is then detected.

The underneath shadow is unclear in dark hours; underneath shadow cannot be used to detect approaching vehicles. However, there are head light of approaching vehicles in the dark hours. Thus, the edge clusters can be used to detect approaching vehicles in the dark hours. Based on the defined threshold value  $T_e$  described in Section II-A, the pixels whose gradient magnitudes are greater than  $T_e$  are classified as significant edge points. If any detection region  $R_i$  has a significant number of edge points, a vehicle is then detected.

### D. Detection Decision

To compensate the perspective effect, the gray levels and edge magnitudes in all above detection criteria are need to be

adjusted by Eq. (1). Other than the above decision of positive optical flows, the negative optical flows are directly used to reject the possible of approaching vehicles. The usage of motion features and static features for blind-spot vehicle detection is special treated as above description and shown in the last part in Fig. 1.

### III. EXPERIMENTS

The proposed approach was conducted to demonstrate the performance of blind-spot vehicle detection. All experimental RGB color images were captured from a waterproof digital camera; they are all in size of 320×240 pixels. The digital camera was mounted below the right-side wing mirror. The proposed methods were implemented by C language with standard ANSI C library on a general PC, Intel® Core 2 Duo® P8700 2.53 GHz CPU with Microsoft® Windows 7 operation system. The source code was also compiled to execute on an embedded system TI DaVinci DM6737 to evaluate the execution performance.

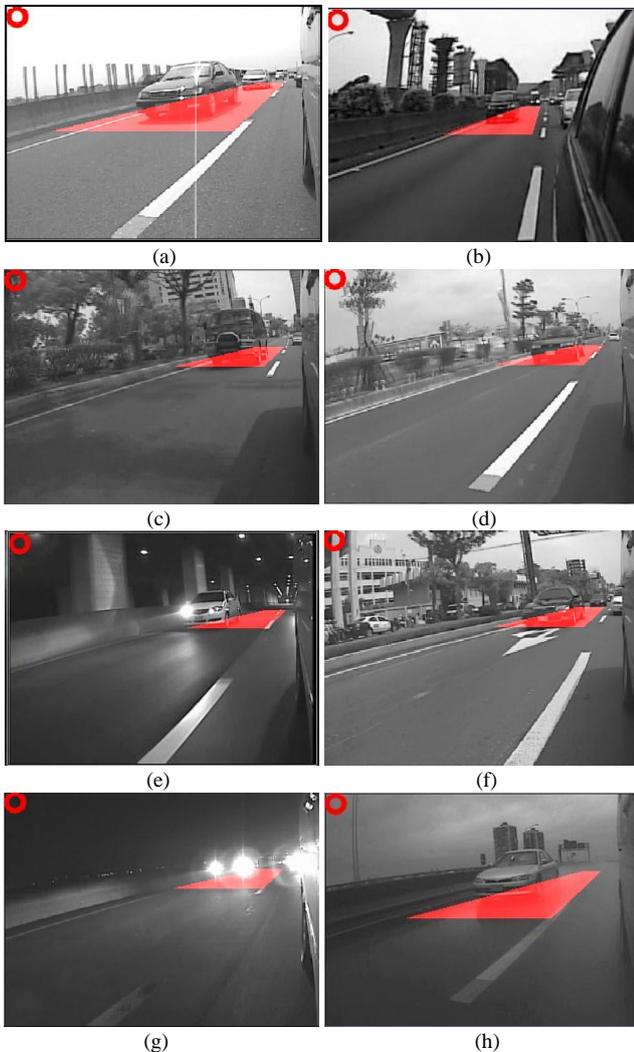


Fig. 9. Examples of approaching vehicle detection in different weather conditions and environment situations. (a) Sunny day. (b) cloudy day. (c) Tree shadow influence. (d) High-speed approaching vehicle. (e) In tunnel. (f) Traffic sign on ground. (g) At night hour. (h) Heavily rainy day.

There are totally 4,268 images in eight weather conditions or environment situations for examination. These conditions

and situations are Sunny day, cloudy day, tree shadow influence, high-speed approaching vehicle, in tunnel, traffic sign on ground, at night hour, and heavily rainy day. Samples of detection vehicles in the eight conditions and situations are illuminated in Fig. 9.

The detection performance is quantitatively evaluated by theory of statistical hypothesis testing. Four relations between detection results and the actual situation: true positive (*TP*), true negative (*TN*), false positive (*FP*), and false negative (*FN*) are represented in Table I. From the definition, the accuracy of detection is given as

$$accuracy (\%) = 100 \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

The experimental results are summarized in Table II. The overall accuracy is 95.67 %. The accuracy of the sunny days and cloudy days are higher than that in night and heavily rainy day. The sunny days and cloudy days have the highest accuracy in four weather conditions: Sunny day, cloudy day, at night hour, and heavily rainy day. The road surface is dusky and unclear in the night and heavily rainy days; all features: optical flow, edge, and underneath shadow, are unstable to be detected. Thus the accuracy is not so good.

From the experimental results, we can find that all four environment situations: tree shadow, high-speed approaching vehicle, in tunnel, and traffic sign on ground, do not influence the detection rate of our blind-spot detection.

The proposed system was executed on the mentioned personal computer. We got the average execution time for one frame is about 28 milliseconds. The numbers of feature points (edge points) in sunny and cloudy days are greater than that in night and heavily rainy days; thus the execution time in sunny and cloudy days is slightly greater than that in night and heavily rainy days.

TABLE I: FOUR RELATIONS BETWEEN DETECTION RESULTS AND THE ACTUAL SITUATION

Detection Action	Vehicle exist	No vehicle
Vehicle detect	True positive ( <i>TP</i> )	False positive ( <i>FP</i> ) (Type I error)
Don't detect vehicle	False negative ( <i>FN</i> ) (Type II error)	True negative ( <i>TN</i> )

TABLE II: THE STATISTIC RESULTS IN EACH WEATHER CONDITIONS

Weather conditions	Total frames	Vehicle number	# of detected vehicles	Type I/Type II error #	Accuracy
Sunny	717	546	542	12/4	97.77%
Cloudy	274	162	157	8/5	95.26%
Tree shadow	350	220	218	4/2	98.29%
Quick vehicle	209	57	55	3/2	97.61%
In tunnel	694	317	307	21/10	95.53%
Traffic sign	476	321	317	10/4	97.06%
Night hour	874	432	409	39/23	92.91%
Heavily rainy	674	537	512	13/25	94.36%

### IV. CONCLUSION

A blind-spot detection system was proposed in this study.

The proposed blind-spot detection system consists of four stages: estimation of weather-adaptive threshold values, multi-resolution optical flow detection, static feature detection, and detection decision.

The proposed approach has the following properties: (i) The approach is adaptive to various weather conditions. (ii) The detection is not influenced by building shadows and traffic signals on ground. (iii) The detection is not influenced by complicated background due to the pre-defined detection area. (iv) Multi-resolution optical flow is used to detect vehicles in far or near distance. (v) Multi-resolution optical flow is able to detect various-speed approaching vehicles. (vi) Static and motion features were proposed to detect vehicles such that the similar-speed side vehicles can also be detected. (vii) The static and motion features were sequentially treated to improve the detection rate and reduce the false alarm.

#### REFERENCES

- [1] R. Sosa and G. Velazquez, "Obstacles detection and collision avoidance system developed with virtual models," in *Proc. IEEE Int. Conf. on Vehicular Electronics and Safety*, Beijing, China, Dec. 13-15, 2007, pp. 1-8.
- [2] T. Mondal, R. Ghatak, and S. R. B. Chaudhuri, "Design and analysis of a 5.88GHz microstrip phased array antenna for intelligent transport systems," in *Proc. Int. Symp. on Antennas and Propagation*, Toronto, Ontario, Canada, July 11-17, 2010, pp. 1-4.
- [3] J. Teizer, B. S. Allread, and U. Mantripragada, "Automating the blind spot measurement of construction equipment," *Automation in Construction*, vol. 19, pp. 491-501, 2010.
- [4] O. Achler and M. M. Trivedi, "Vehicle wheel detector using 2D filter banks," in *Proc. IEEE Intelligent Vehicles Symp.*, Parma, Italy, Jun.14-17, 2004, pp. 25-30.
- [5] N. Blanc, B. Steux, and T. Hinz, "LaRASideCam - a fast and robust vision-based blindspot detection system," in *Proc. IEEE Intelligent Vehicles Symp.*, Istanbul, Turkey, June 13-15, 2007, pp. 480-485.
- [6] E. Y. Chung, H. C. Jung, E. Chang, and I. S. Lee, "Vision based for lane change decision aid system," in *Proc. 1st Int. Forum on Strategic Technology*, Ulsan, Korea, Oct. 18-20, 2006, pp. 10-13.
- [7] M. Krips, J. Velten, A. Kummert, and A. Teuner, "AdTM tracking for blind spot collision avoidance," in *Proc. IEEE Intelligent Vehicles Symp.*, Parma, Italy, June 14-17, 2004, pp. 544-548.
- [8] B.-F. Wu, W.-H. Chen, C.-W. Chang, C.-J. Chen, and M.-W. Chung, "A new vehicle detection with distance estimation for lane change warning systems," in *Proc. IEEE Intelligent Vehicles Symp.*, Istanbul, Turkey, June 13-15, 2007, pp. 698-703.
- [9] J. Zhou, D. Gao, and D. Zhang, "Moving vehicle detection for automatic traffic monitoring," *IEEE Trans. on Vehicular Technology*, vol. 56, no. 1, pp. 51-59, 2007.
- [10] S. Mota, E. Ros, E. M. Ortigosa, and F. J. Pelayo, "Bio-inspired motion detection for a blind spot overtaking monitor," *Int. Journal of Robotics and Automation*, vol. 19, no. 4, pp. 190-196, 2004.
- [11] S. Mota, E. Ros, J. D áz, G. Botella, F. Vargas-Martin, and A. Prieto, "Motion driven segmentation scheme for car overtaking sequences," in

- Proc. 10th Int. Conf. on Vision in Vehicles*, Granada, Spain, Sep. 7-10, 2003, pp.1-6.
- [12] N. Ohta and K. Nijjima, "Detection of approaching cars via artificial insect vision," *Electronics and Communications in Japan*, vol. 88, no.10, pp. 57-65, 2005.
- [13] J. D. Alonso, E. R. Vidal, A. Rotter, and M. Mühlenberg, "Lane-change decision aid system based on motion-driven vehicle tracking," *IEEE Trans. on Vehicular Technology*, vol. 57, no. 5, pp. 2736-2746, 2008.
- [14] J. Wang, G. Bebis, and R. Miller, "Overtaking vehicle detection using dynamic and quasi-static background modeling," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Reno, Nevada, June 20-26, 2005, pp. 64-71.
- [15] M. A. Sotelo, J. Barriga, D. Fernández, I. Parra, J. E. Naranjo, M. Marrón, S. Alvarez, and M. Gavilán, "Vision-based blind spot detection using optical flow," in *Proc. 11th Int. Conf. on Computer Aided Systems Theory*, Las Palmas, Spain, Feb. 12-16, 2007, pp. 1113-1118.
- [16] M.-H. Yang and N. Ahuja, "Gaussian mixture model for human skin color and its applications in image and video databases," in *Proc. IS&T/SPIE Conf. on Storage and Retrieval for Image and Video Databases VII*, San Jose, CA, Jan. 23, 1999, pp. 458-466.
- [17] J. Y. Bouguet, *Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the algorithm*, Intel Microprocessor Research Labs., 2007.



**Din-Chang Tseng** received his Ph.D. degree in information engineering from National Chiao Tung University, Hsinchu, Taiwan, in June 1988. He has been a professor in the Department of Computer Science and Information Engineering at National Central University, Zhongli, Taiwan since 1996. He is a member of the IEEE. His current research interests include computer vision, image processing, and virtual reality; especially in the topics: computer vision system for advanced safety vehicle, computer vision techniques for human computer interaction, and view-dependent multi-resolution terrain modeling.



**Chang-Tao Hsu** received his B.S. and M.S. degrees in the Department of Biomedical Engineering from Chung Yuan Christian University, Taiwan, in 1996 and 1998, respectively. He is currently working for his Ph.D. degree in the Department of Computer Science and Information Engineering at National Central University, Zhongli, Taiwan. His research interest is real-time image processing in intelligent transportation systems.



**Wei-Shen Chen** received his B.S. degree in the Department of Computer Science and Information Engineering from Yuan Ze University, Jhongli, Taiwan, in 2010, and M.S. degree in the Institute of Computer Science and Information Engineering from National Central University, Zhongli, Taiwan, in 2012. He is now working in the VIA Technologies, Co. as a software engineer.