Development of a Multimedia-Based Vehicle Lane Departure Warning, Forward Collision Warning and Event Video Recorder Systems

Lin, Che-Chung
MSL ITRI,Taiwan,R.O.C.
AlexLin@itri.org.tw

Hsieh, His-Wen
MSL ITRI,Taiwan,R.O.C.
ConnieHsieh@itri.org.tw

Huang, Dau-Chen,
MSL ITRI,Taiwan,R.O.C.
ChrisHuang@itri.org.tw

Lin, Chi-Wei
MSL ITRI,Taiwan,R.O.C.
Zack@itri.org.tw

Liao,Yung-Sheng
MSL ITRI,Taiwan,R.O.C.
AndersonLiao@itri.org.tw

Chen,Yung-Hsin
Asia Univerrsity,Taiwan,R.O.C.
thomaschen@asia.edu.tw

Abstract

This paper presents how an intelligent multimedia-based vehicle warning device is developed. The focal device incorporates the lane departure warning (LDW) function, the forward collision warning (FCW) function, and the event vide-recorder (EVR) function with a charged-coupled-diode (CCD) camera as the means to capture image. The LDW component uses a median filter, an edge-enhancement filter and the Hough Transform algorithm for lane recognition. The FCW component identifies vehicles with a feature-based approach while verifies the vehicle candidates by the appearance-based approach. Besides, we also propose a noble vehicle detecting scheme in which the task of vehicle detection depends not on the whole image within a frame but rather on the image's three constituent portions of different sizes, with a view to reduce the computing burden. The motion vector (MV) estimation is applied to track the detected vehicle in movement. This act helps that not all vehicles inside the image frame subject to detect in the vehicle detection stage. The EVR system is used to record the image captured in the event of a vehicle accident. The integration of LDW, FCW and EVR functions has successfully implemented in an ADI-BF561 600MHz dual core digital signal process (DSP).

1. Introduction

In the recent years, the annual casualty in traffic accidents amounted to more than 2,500 people on the average in Taiwan. It is attributable to drivers' distraction and fatigue in the course of movement. Therefore, developing an intelligent device in vehicle is one of the solutions to alleviate the concerns. The research literatures in the area of LDW and FCW grow in a rapid pace, but little discussion has been devoted to the synergy of combining them with EVR. Most of research outcomes concerning LDW and FCW take advantage of computing power in a powerful PC platform, however, are of little feasibility in the real world applications if consider the size and the cost. In this paper, we present an innovative approach to integrate the function of LDW, FCW and EVR into the hardware featuring a dual core DSP that will be apt for commercialization.

There are many ways to recognize the existence of lanes on the highway as described in [1] where the inverse perspective mapping (IPM) method is used to generate bird's-eye view images and remove the perspective effect; [2] where a rapidly adapting lateral position handler (RALPH) system is proposed to constitute an adaptive high-speed matching procedure to determine the lanes' curvature and its lateral offset; and [3] where the highway marks in the near vision field are fitted by the linear model and those in the far vision field are fitted by the parabolic model. Although these algorithms produce satisfactory results, they fail to be implemented in an embedded system for the following reasons: IPM consumes too much computing power for real time processing in an embedded system; RALPH approach trades computation cost for rapid response but sacrifices the requirement of precision. Its performance also deteriorates owing to the lack of sufficient parameters; the linear-parabolic model has to use floating point operation to fit the lane's curvature well and it is difficult to be implemented in the DSP as floating point unit is unavailable there.

For FCW studies in research, there are also no less than many: in [1] and [8], a 3-Dimensional approximation of the framed scene is reconstructed by a stereo system to detect vehicles by extracting features. In [11], [12], [13], a monocular camera is adopted to detect vehicles by extracting features too. However, stereovision is not appropriate for the embedded system because of its huge computing cost. Using specific features to distinguish vehicles and obstacles from image without considering their appearance cannot acquire high detection rate. Detecting vehicles by appearance-based method is another alternative. In

0-7695-3084-2/07 $25.00 © 2007 IEEE
DOI 10.1109/ISM.Workshops.2007.31

[5] and [6], principal component analysis (PCA) templates are used. In [4], a so called “multi-clustered modified quadratic discriminant function” (MC-MQDF) classification method is used. However, detecting vehicles by appearance-based method alone is not feasible for embedded system because the whole image must be split into many small sub-images of fixed size for the purpose of classification. This will result in unacceptable computing burden.

For analyzing vehicle accident, [17] introduces Event Data Recorder (EDR), which is capable of recording speed, turn signal, and the status of braking and steering. However, EDR only enables the system to store data at the occurrence moment of a car accident and this kind of data is not sufficient enough to reconstruct the cause of that accident.

The remaining part of this paper is structured as follows: in section 2, we will describe the operation of the LDW, FCW and EVR. Section 3 is to describe the algorithm of LDW, and Section 4 we discuss the algorithm of FCW, and in section 5 is to discuss the algorithm of EVR. The research outcomes implemented in an experiment car are shown in section 6. Finally we draw the conclusion and indicate the direction for further research.

2. System Overview

2.1Hardware Architecture

The architecture of hardware shown in Figure 1 contains a power subsystem that powers up the whole system, a UART interface for calibration, a video input subsystem that decodes NTSC format images (interlaced, 858x525 pixels), a dual core DSP to run the LDW, FCW and EVR algorithms, a video output subsystem for demonstrating the detection results, a vehicle interface for receiving vehicle signals, memories for booting and saving data, and JTAG for debugging and a buzzer for warning the driver.

2.2 Software Block Diagram

The software block diagram is shown in Figure 2. The DSP dual cores are denoted by A and B. Here, Core A is responsible for LDW and EVR, while core B is devoted to the FCW and its output will be transmitted to Core A for display the result of vehicle detection. The LDW has an efficient median filter [14], an edge-enhancement filter and Hough transform algorithm [15]. The FCW consists of vehicle detection and vehicle tracking where vehicle detection includes feature-based searching and appearance-based vehicle verification, vehicle tracking includes MV estimation. The EVR takes the responsibility of recording image into RAM before triggering, and compressing image to merge with recording compressed image into Flash memory after triggering.

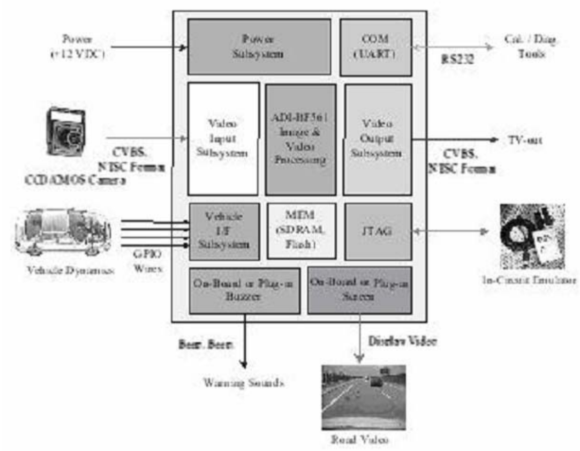


Figure 1 The Hardware Architecture.

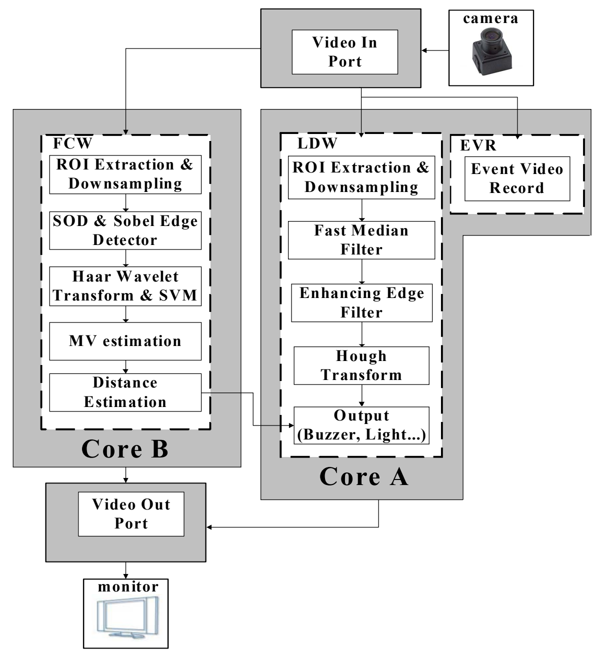


Figure 2 The Software Architecture.

3. Lane Departure Warning System

In this device, LDW system consists of efficient median filter, edge-enhancement filter and Hough transform algorithm.

3.1 Efficient Median Filter

On image processing, to remove noise is one of the important issues, but it is usually a time-consuming process because most of filters have to perform convolution operation. Hereby, the efficient median filter for 3x3 arrays is introduced to this system and it only needs 19 comparisons in the worst case to get two neighboring medians.

3.2 Edge-enhancement Filter

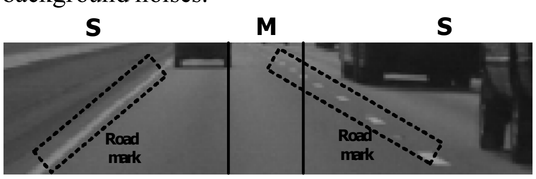
Two assumptions are postulated in the course of reducing the background noises. The first one is that background image noises such as vehicles, buildings and anything but road marks, appear much likely in two sides of ROI(Region Of Interest) of captured images. The other is that road marks in two sides of ROI are oblique. According to those assumptions, the ROI is divided into two parts: the middle part (denoted by M in fig. 3.a) and side part (S in fig. 3.a). The middle part is filtered by the general edge filter defined as:

$$ME = \begin{cases} 0 & \text{if } \left(|f_x| + |f_y|\right) < \theta \\ |f_x| + |f_y| & \text{if } \theta \le \left(|f_x| + |f_y|\right) < 255 \\ 255 & \text{if } \left(|f_x| + |f_y|\right) > 255 \end{cases} \tag{1}$$

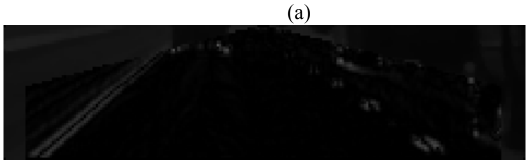
The side part is filtered by the oblique line-enhancing filter defined as:

$$SE = \begin{cases} 0 & \text{if } \left(\left||f_x + f_y| - |f_x - f_y|\right|\right) < \theta \\ \left||f_x + f_y| - |f_x - f_y|\right| & \text{if } \theta \le \left(\left||f_x + f_y| - |f_x - f_y|\right|\right) < 255 \\ 255 & \text{if } \left(\left||f_x + f_y| - |f_x - f_y|\right|\right) > 255 \end{cases} \tag{2}$$

where ME and SE are the filtered values of M and S respectively, f_x and f_y that come from Sobel edge operator are the vertical edge image and horizontal edge image respectively, and θ is a threshold value. Figure 3(b) is the filtered image of Figure 3(a), in which road marks become much distinguishable than background noises.



(a)



(b)

Figure 3 (a) The image example of ROI, which is divided into two parts: middle part (denoted by M) and side part (denoted by S), (b) The filtered image of (a) in which road marks are more obvious than background noises.

3.3 Hough Transform

After applying edge-enhancement filter, numerous pixels of lane edge are not exactly on the same line, but we can find the best-fitting one. Hough transform can be used to fit these pixels into straight line and represents straight line by mathematical equation.

4. Forward Collision Warning System

The process of FCW system is composed of three stages. The first stage is vehicle detection, the second stage is vehicle tracking, and the third stage is distance estimation and camera calibration

4.1 Vehicle Detection

The vehicle detection stage includes two steps. The first one is to extract vehicle candidates by feature-based approach. The second one is to verify these candidates by the appearance-based approach.

4.1.1 The Feature-Based Vehicle-Detection

The feature-based approach consists of

(a) ROI and Down-sampling, (b) SOD(Sum Of Difference) and Sobel edge detector.

(a) ROI and Down-sampling

In order to achieve better system performance, one must only take into account the ROI. Hereby, we define three kinds of ROI for vehicle detection process. The three ROIs as shown in Figure 4 are long-range region detection, Middle-range region detection and short-range region detection. To further reduce the computing burden, middle-range detection region and short-range detection region are down-sampled for two-times and four-times separately.

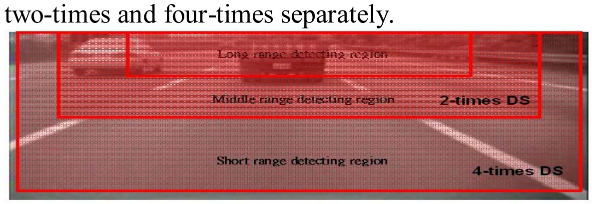


Figure 4 The Three Different Types of ROI.

Under FCW's operation, it is running in the sequence of short-range region detection, tracking, middle-range region detection, tracking, long-range region detection and tracking. By doing so, the computing power consumption of vehicle detection system can be then mitigated greatly.

(b) SOD and Sobel edge detector.

The location of preceding vehicle can be detected with the calculation of SOD and Sobel edge detector. The vehicle in the image has boundary distinctive to road image and has symmetric property. We can check if the enclosed image inside square frame has these two properties by SOD. The calculation of SOD is based on different default size of square frame corresponding to different row. The SOD(x,y) is defined as:

$$SOD(x,y) = \frac{\sum_{h_t=1}^{H_t(x,y)} \sum_{w_t=1}^{W_t(x,y)/2} \left|G(x - w_t, y - h_t) - G(x + w_t, y - h_t)\right|}{H_t(x,y) \times W_t(x,y)} \tag{3}$$

where G(x,y) is the gray-level value of points (x,y), while H_t and W_t denote the default height and width of the square frame of points (x,y). When combining the results from SOD and Sobel horizontal edge detector, vehicles can be located as shown in Figure 5.

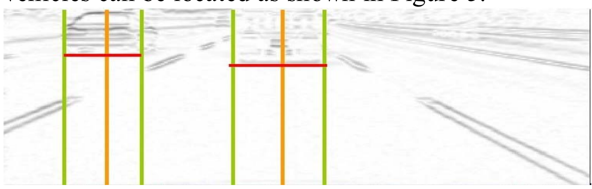


Figure 5 The Result of SOD and Sobel Edge Detector.

4.1.2 The Appearance-based Vehicle-verification

The appearance-based approach is firstly to train the SVM (Support Vector Machine) classifier off-line by feeding vehicle and non-vehicle images, which are transformed by Haar Wavelet Transform. Once the off-line training of SVM classifier is completed, it can be used on-line to verify vehicle candidates obtained from previous featured-based vehicle detection. The appearance-based vehicle verification consists of (a) Haar Wavelet Transform, (b) off-line SVM training and on-line SVM classifying.

(a) Haar Wavelet Transform.

Wavelet Transform is essentially a type of multi-resolution function approximation that allows for the hierarchical decomposition of a signal or image. Features based on Haar wavelet transform have yielded promising results in various applications including vehicle detection. In order to further simplify the Haar wavelet transformed image, the image will be quantized into –1, 0 and 1. This feature-simplifying scheme [7] is proved to be more efficient.

(b) Off-line SVM training and on-line SVM classifying.

SVM is primarily a two-class classifier, which is a systematic approach to learn linear or non-linear decision boundaries. In our case the dataset to be classified is the coefficients of the image after Haar Wavelet transforming and quantizing. Each dataset is defined as

$$\mathrm{X_i} = \{\mathrm{d_1, d_2, d_3, \ldots, d_n}\},\quad \mathrm{d_i} \in \{-1, 0, 1\} \tag{4}$$

and let y_i be the class label of X_i .while the decision boundary should classify all points to satisfy

$$y_i(w^T x_i + b) \geq 1, \quad \forall \mathrm{i} \tag{5}$$

where w is the normal vector of the hyper-plane, b is bias. The goal of classification is to maximize the margin

$$m = \frac{1}{2}\|w\|^2 \tag{6}$$

This is a constrained optimization problem, and it is solved by software called LOQO [19], which is developed by Princeton University. Once the training process is completed, we can obtain w^T and b that will be later implemented in embedded system for on-line classifying.

4.2 Vehicle Tracking and Motion Vector Estimation

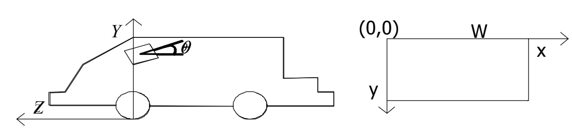
Vehicle tracking refers to the prediction of the future position of a previously detected vehicle and decide if this vehicle exists close enough to the predicted position. If the previously detected vehicle does exist close enough to the predicted position, its position will be updated. If not, its position will be compensated by the predicted one. Fortunate enough, Motion Vector estimation [16] can be applied to implement vehicle tracking. This explains why detecting vehicles with the whole image in each frame is unnecessary and ROI can be defined with three different sizes in vehicle detection stage.

4.3 Distance Estimation

In order to estimate the distance between the ego-vehicle and the preceding vehicle, we must define two coordinate systems in advance, namely, the vehicle coordinate system and the image coordinate system. The vehicle coordinate system is defined based on the flat ground plane as shown in Figure 6(a). The image coordinate system is defined with respect to the output image as shown in Figure 6(b). Once the bottom (shadow) position of the preceding vehicle is detected, the distance between the ego-vehicle and the preceding vehicle can be estimated by

$$\mathrm{Z_v} = \frac{h(\mathrm{fS_y}\cos\theta - (y_v - \frac{\mathrm{H}}{2})\sin\theta)}{\mathrm{fS_y}\sin\theta + (y_v - \frac{\mathrm{H}}{2})\cos\theta} \tag{7}$$

where f is the focal length of the camera, h is the camera height from the ground plane, and. H and W are the height and width of the image respectively y_v is the location of the preceding vehicle in the image coordinate. Z_v is the distance between ego-vehicle and preceding vehicle. S_y is the scaling factor in y-axis. θ is the tilt angle as shown in Figure 6.



(a) (b)

Figure 6 (a) Vehicle coordinates, (b) Image coordinates

5. Event Video Recorder

The EVR is used to record video at the occurrence moment of a car accident. EVR records 20 seconds duration of image data where 15 seconds image is prior to the occurrence moment of accident and 5 seconds is posterior to the occurrence moment of accident. By detecting the rapid speed variation in speedometer or large acceleration in G sensor, EVR can be triggered. The image data of 15 seconds duration before triggering and the image data of 5 seconds duration after triggering are stored in RAM. After the recording process, the entire image data of 20 seconds will be compressed and stored in flash memory. The stored image can be then downloaded into PC via RS-232 port for reconstructing the cause of the accident.

There are many video or image compression algorithms and each of it has its specific application. We choose Discrete Cosine Transformation (DCT) to implement EVR in our embedded system while the original complicated Huffman entropy coding is replaced with our proposed easy entropy coding. Figure 7 shows the DCT compression flow chart.

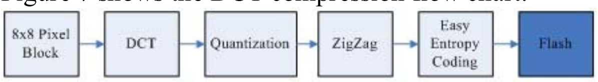


Figure 7 The Compression Flow Chart

5.1. Discrete Cosine Transformation

DCT is commonly used in signal and image processing, especially for lossy data compression. In our system the input video is from external B/W CCD camera. Before applying DCT, we must split image into sub-image with fixed size (8x8) in advance. The DCT is defined as

$$F(u,v)=\frac{1}{4}C(u)C(v)\left[\sum_{i=0}^{7}\sum_{j=0}^{7}f(i,j)\cos\frac{(2i+1)u\pi}{16}\cos\frac{(2j+1)v\pi}{16}\right] \qquad (8)$$

$$C(u),C(v)=1/\sqrt{2},\text{while}\, u,v=0$$

$$C(u),C(v)=1,\text{otherwise}$$

Where f(i,j) is the intensity of the pixel in row i and column j, F(u,v) is the DCT coefficient in row u and column v of the DCT matrix. The terms C(u),C(v) are the coefficients which are used to make the transform matrix orthogonal.

5.2. Quantization

After DCT, the image was transformed into frequency domain. In fact, human eyes are more sensitive to low frequency than high frequency signal. For most images, greater portion of the image energy was gathered at low frequency. In order to extract important information at low frequency and ignore information at high frequency, quantization table can be applied. This will result information at low frequency more obvious than at high frequency.

5.3. ZigZag

After quantizing, the ZigZag scan will be applied. The ZigZag scan will place coefficients, which are at high frequency and closer to zero, in the end and this helps the later easy entropy coding.

5.4. Easy Entropy Coding Method

An example of coefficients after applying ZigZag scan is: 70,12,0,0,2,0,5,1,0,0,0…0. The size of it remains 64 but the values after the eighth coefficient are all zero. The easy entropy coding is proposed to deal with the coefficients sequence, which contains many zero numbers. In easy entropy coding, each non-zero numbers remains but zero number or numbers will be encoded as two numbers. i.e. zero and the amount of zero numbers. An exception is for the case that when from certain number to the end of the sequence is all zero, it will be encoded as two zero. The above-mentioned coefficients after applying easy entropy coding will be 70,12,0,2,2,0,1,5,1,0,0. The original size (64 bytes) remains 11 bytes after encoding. Typically, the average compression efficiency will be 15% to 18%.

6. Experimental Results

The process time of LDW, FCW and EVR is depicted in Figure 8. The whole system containing LDW, FCW and EVR needs 90 ms per frame.

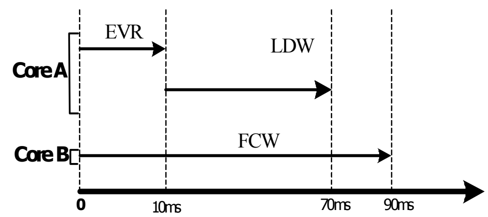
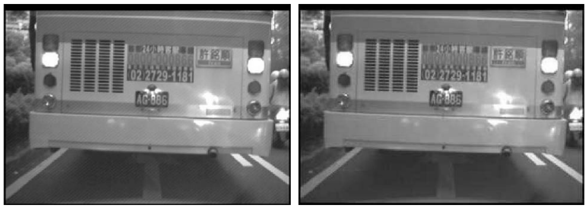


Figure 8 The processing time

According to the experimental results, the LDW can adapt to dry lane (Figure 9 (a)), wet lane (Figure 9(b)). However, the FCW is more sensitive to the environmental condition than the LDW. In general, the FCW can detect most of vehicles if their appearance and feature are obvious as shown Figure 9. The image after compression is almost identical to the image before compression and is still adequate for reconstructing the cause of the accident as shown in Figure 10.

(a) (b)

Figure 9 Detection Results.



(a) (b)

Figure 10 (a) Image before Compression, (b) Image after Compression.

7. Conclusions and Future Work

We have developed a set of algorithms for LDW and EVR functions in core A and FCW function in core B. In regard to FCW, the vehicles can be accurately detected successfully using our proposed algorithms to identify vehicles by feature-based approach and to verify the vehicle candidates by appearance-based approach. Execution time for LDW, FCW and EVR is 90ms with our dual core system. However, the performance of the dual core DSP is not exerted fully yet because Core A must wait for Core B to finish the calculation of FCW after EVR and LDW completes calculation at each frame. The future work of our research will focus on how to share the computing load equally between Core A and Core B. Besides, in order to install the system with ease, we will develop a new calibration tool to facilitate the measurement of CCD height and tilt angle without using parallel lanes to detect vanishing point [18].

References

[1] M. Bertozzi and A. Broggi, “GOLD: a parallel real-time stereo vision system for generic obstacle and lane detection,” IEEE Transactions on Image Processing, vol. 7, no. 1, pp. 62-81, Jan. 1998.

[2] D. PomerLeau, “RALPH: Rapidly Adapting Lateral Position Handler,” Proceedings of IEEE Intelligent Vehicles Symposium, pp. 506-511, 1995.

[3] C. R. Jung and C. R. Kelber, ” A robust linear-parabolic model for lane following,” Proceedings of XVII Brazilian Symposium on Computer Graphics and Image Processing, pp. 72-79, Oct. 2004.

[4] Kato. T, Ninomiya, Y. Masaki. I, “Preceding vehicle recognition based on learning from sample images,” IEEE Transactions on Intelligent Transportation Systems, Volume 3, Issue 4, pp. 252 - 260, Dec. 2002

[5] M. Betke, E. Haritaglu and L. Davis, “Multiple vehicle detection and tracking in hard real time,” Proceedings of IEEE Intelligent Vehicles Symposium, pp. 351-356, 1996.

[6] J. Ferryman, A. Worrall, G. Sullivan, and K. Baker, “A generic deformable model for vehicle recognition,” Proceedings of British Machine Vision Conference, pp. 127-136, 1995.

[7] Zehang Sun, George Bebis and Ronald Miller, “Quantized wavelet features and support vector machines for on-road vehicle detection,” Control, Automation, Robotics and Vision International Conference, vol. 3, pp. 1641-1646, Dec. 2002.

[8] A. Bensrhair, M. Bertozzi, A. Broggi, A. Fascioli, S. Mousset, and G. Toulminet, “Stereo Vision-based Feature Extraction,” Proceedings of IEEE Intelligent Vehicles Symposium, vol. 2, pp. 465–470, June 2002.

[9] H. Li, S. Bouaziz, and F. Devos, “An Embedded System for Autonomous Collision Avoidance and Line Tracking Using Artificial CMOS Retina Sensor,” Proceedings of IEEE Intelligent Vehicles Symposium, pp. 273–278, Oct. 2000.

[10] S. Heinrich, “Fast Obstacle Detection using Flow/Depth Constraint,” Proceedings of IEEE Intelligent Vehicles Symposium, pp.658-665, June 2002.

[11] Zielke T., Brauckmann M., and Von Seelen W., “Intensity and Edge-Based Symmetry Detection with application to Car-Following,” computer vision, Graphics and Image Processing: Image Understanding, vol. 58, No. 2, pp. 177-190, 1993.,

[12] Hao-Yuan Chang; Chih-Ming Fu; Chung-Lin Huang, “Real-time vision-based preceding vehicle tracking and recognition,” Proceedings of IEEE Intelligent Vehicles Symposium, pp.514-519, June. 2005

[13] S. S. Huang, C. J. Chen, P. Y. Hsiao and L. C. Fu, “On-Board Vision System for Lane Recognition and Front-Vehicle Detection to Enhance Driver’s Awareness,” Proceedings of IEEE Robotics and Automation, vol. 3, pp. 2456 – 2461, May 2004.

[14] Manfred Kopp and Werner Purgathofer, “efficient 3x3 median filter”, Available: http://citeseer.ist.psu.edu/67259.html

[15] Robert M. Haralick, Linda G. Shapiro Computer and Robot Vision Vol.1, Addison-Wesley Publishing Company, Inc. 1992.

[16] B. Furth, J. Greenberg, R. Westwater, “Motion estimation algorithms for video compression,” Kluwer academic publishers, 1997.

[17] Hampton C. Gabler, Douglas J. Gabauer, Heidi L. Newell, Michael E. O’Neill, “Use of Event Data Recorder (EDR) Technology for Highway Crash Data Analysis,” NCHRP Project, pp.17-24, December.2004

[18] E. K. Bas and J. D. Crisman, “An Easy to Install Camera Calibration for Traffic Monitoring,” IEEE Conference on Intelligent Transportation System, 9-12, Nov.1997.
System

[19] Robert J. Vanderbei “LOQO”, Available: http://www.princeton.edu/~rvdb/