

Vision-Based Parking Guidance and Obstacle Detection System

Din-Chang Tseng and Yu-Chi Lin

Abstract—In recent years, top-view monitoring systems are becoming a practical driving aid that help reducing collision hazards by eliminating blind spots. The U.S. Department of Transportation’s National Highway Traffic Safety Administration (NHTSA) issued a rule requiring rear visibility technology in all new vehicles under 10,000 pounds by May 2018. Many of such systems provide short range views surrounding the vehicle, limiting its application to parking and reversing. In this paper, we propose a practical system for creating images around the vehicle with the parking guidance line, and highlighting obstacles only relies on an embedded hardware and a wide-angle camera to capture images for analysis without sensors. By estimating the ego-motion of the vehicle using the input image sequence of the cameras, the proposed system is able to detect objects in the images by finding movements of features that do not correspond to ground motion relative to vehicle motion. Detected obstacles are highlighted in the multi-view imagery to warn the driver of potential hazards.

Index Terms—Advanced driver assistance systems, parking assistance, collision avoidance, motion analysis.

I. INTRODUCTION

Parking a vehicle into a parking space or garage is an essential skill for drivers; however, it is still hard for someone. Moreover, to maneuver a vehicle into a parking space with limited dimension or with other vehicles or obstacles around the parking space is difficult for most drivers. Even a familiar driver has an unpleasant experience to park a vehicle into a small parallel parking space.

In these few decades, cameras and related embedded system are more and more cheap, and have been used for vehicle driving assistance, such as lane departure warning, forward collision warning, blind spot detection, etc. [1]. The mentioned parking problem can be solved by an around view monitor system [2]-[4] or a back guiding monitor and detection system, which are just equipped cameras and an embedded system.

Nissan Motor Co., Ltd. [2] has developed a parking assistance system named as "Around View Monitor". The system uses four wide-angle cameras mounted on the front, sides, and rear of the vehicle to capture images of the surrounding area to generate a bird’s eye view of the vehicle with its surrounding scenery and a parking guiding view of the rear scenery as shown in Fig. 1.

Honda Motor Co., Ltd. [3] has provided a similar system

named “Multi-view Camera System” with extra functions for tight driving support as shown in Fig. 2. At the same time, Mitsubishi Motors Corp. [4] provided a similar system “Multi-around Monitor System”. After two years, Fujitsu Semiconductor America, Inc. announced the commercial availability of a breakthrough 3D imaging technology that provides a complete 360-degree view of a vehicle’s surrounding [5].



Fig. 1. Nissan Around view monitor system.

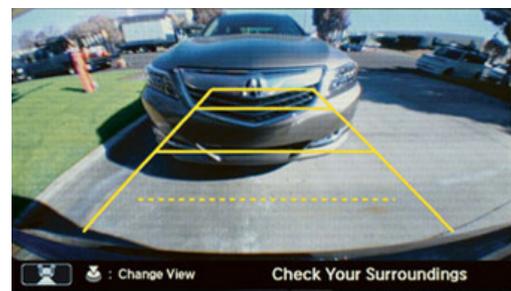


Fig. 2. Multi-view Camera system.

In this paper, we propose an image-based parking guidance (IPG) system to estimate the direction of front wheels without steering sensor. The hardware is just an ARM-based embedded system and a wide-angle camera. The camera is mounted on the rear of vehicle to capture sequential images of ground area just behind the vehicle. In the software development, off-line and on-line processes are sequentially constructed. The off-line process is used to calibrate 3D position of the camera to the ground coordinate system; then the on-line process is utilized to estimate the vehicle trajectory with respect to the ground coordinate system. At first, input images are first transformed into top-view images by a transformation matrix of homography. Then corner feature points on two continuous images are extracted to match each other. The feature-point pairs are further pruned by a least-square error metrics. The remained pairs are then used to estimate the motion parameters, where an isometric transformation model based on the Ackermann steering geometry [6] is proposed to describe the vehicle motion. A vehicle trajectory is then estimated based on the parameters.

Manuscript received January 15, 2019; revised May 7, 2019.

The authors are with the Department of Computer Science and Information Engineering at National Central University, Zhongli, Taiwan (e-mail: tsengdc@ip.csie.ncu.edu.tw, devil5417@gmail.com).

At last, a top view or rear view of parking guidance-lined images are provided to assist the driver. The proposed system is shown in Fig. 3.

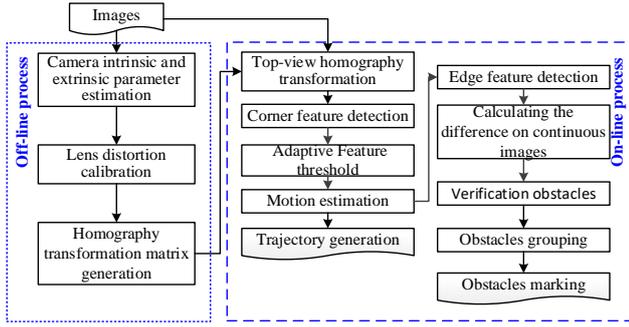


Fig. 3. The proposed image-based parking guiding system.

II. TOP-VIEW MONITORING

A. Camera's Parameter Calibration

The purpose of camera calibration is to obtain the corresponding relation between three-dimensional world coordinate system and the two-dimensional image coordinate system. Before the cameras are installed, we can calibrate the intrinsic parameters for getting the transformation between image coordinates and camera coordinates. After mounting the cameras on the vehicle, we calibrate the extrinsic parameters which relate world coordinates to each of cameras coordinates.

Zhang [7] proposed a flexible, robust and low cost method for camera calibration. We print the pattern and attach to a planar surface, and then only utilize the camera to observe a planar pattern shown in two or more different orientations. Either the camera or the planar pattern can be freely moved. The motion need not be known.

We utilize the algorithm and camera model to calculate the constraints on the camera's intrinsic and extrinsic parameters by estimating the homography between a planar model containing the calibration target and several images of this target.

In the Zhang's model [7], if the coordinates of a 3D point p_w is $[x_w \ y_w \ z_w]^T$ and its 2D image point q is $[u \ v \ 1]^T$, then

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A[\mathbf{R} \ \mathbf{t}] \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \quad (1)$$

where s is a non-zero scale factor; \mathbf{R} and \mathbf{t} are 3×3 rotation matrix and 3×1 translation vector, respectively; A is camera intrinsic parameters matrix,

$$A = \begin{bmatrix} \alpha & \gamma & u_o \\ 0 & \beta & v_o \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

where (u_o, v_o) is the camera optical axis center on the image coordinate system; α and β are the focus lengths in image u and v axes, respectively; γ is the skew parameter between photosensitive element array (CCD) and lens structure.

There are totally twelve extrinsic parameters and five

intrinsic parameters. Zhang has proposed a sequence computation to initially estimate the intrinsic parameters and then refine the extrinsic parameters by solving the nonlinear minimization problem with the Levenberg-Marquardt algorithm [7].

B. Wide-Angle Lens Distortion Correction

Based on the characteristic of wide-angle camera, we can achieve the surrounding monitor system with a few cameras; however, the wide-angle lens bright distorted images. The goal of distortion calibration is just to correct the lens distortion to get the mapping between the actual image plane and the perspective camera model. Many distortion models have been proposed such as the classical polynomial model [8], the division model [9], the rational model [10], stereographic projection [11], and the unified catadioptric model [12].

We here utilized the FOV model proposed by Deverney and Faugeras [13] to calibrate the lens distortion. The larger incident angle between 3-D point and optical axis is, the larger distance between image point and image center is. In the FOV model, the change of incident angle is proportional to the change of distance,

$$r_d = \frac{1}{\omega} \tan^{-1}(2r_u \tan \frac{\omega}{2}) \quad (3)$$

where ω is the distortion parameter associated to field of view, and the inverse function is

$$r_u = \frac{\tan(r_d \omega)}{2 \tan \frac{\omega}{2}} \quad (4)$$

One example result of lens distortion correction is shown in Fig. 4.

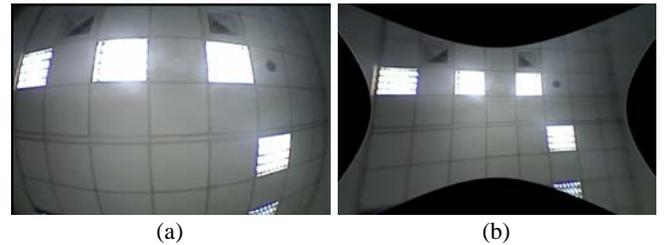


Fig. 4. The result of distortion correction. (a) Original image. (b) After distortion correction.

C. Top-View Transformation

The top-view transformation is to back project images onto the ground surface. The task can be achieved by a homographic matrix. If $q_i = [u_i \ v_i]^T$ and $q_j = [u_j \ v_j]^T$ are projected from one 3D point on planes I_i and I_j , respectively. The point-to-point relation can be expressed by

$$s \mathbf{q}_j = \mathbf{H} \mathbf{q}_i \quad (5)$$

where \mathbf{H} is a homography matrix.

We can use the intrinsic and extrinsic parameters of the cameras to construct the homography matrix. Assume the ground is the plane of WCS which means $Z = 0$. The

top-view image is generated from a virtual camera with image plane being parallel to the ground. Then the relation between a 3D point $[X Y 0]^T$ on the ground and the image point $[u v]^T$ is just

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A[R|t] \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} = H \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (6)$$

The projection is a one-to-one transformation; thus

$$s \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = H^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (7)$$

Assume the relation between the virtual image plane and the WCS is

$$s \begin{bmatrix} u_v \\ v_v \\ 1 \end{bmatrix} = H_v \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (8)$$

where (u_v, v_v) is the point of the virtual image plane. The transformation from the image point $[u v]^T$ to a point on the virtual image is

$$s \begin{bmatrix} u_v \\ v_v \\ 1 \end{bmatrix} = H_v H^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (9)$$

The 3×3 homography matrix $H_v H^{-1}$ can also be estimated by a least-squares estimation method by taking

$$s \begin{bmatrix} u_v \\ v_v \\ 1 \end{bmatrix} = H \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (10)$$

One example through the distortion calibration and top-view transformation is shown in Fig. 5.

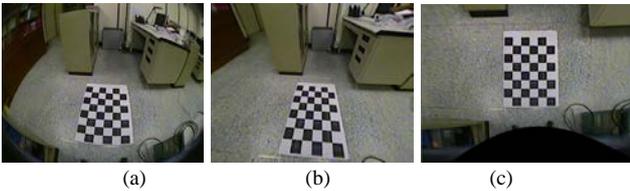


Fig. 5. Top-view transformation. (a) A wide-view image. (b) Distortion-calibrated image. (c) Top-view transformed image.

III. TRAJECTORY GENERATION

A. Feature Matching

Matching two continuous images to acquire the motion parameters of the host vehicle, we need to extract feature points in advance. Rosten-Drummond *FAST* corner detector [14] was utilized to acquire corner feature points for matching.

Lowe [15] has proposed the scale-invariant feature transform (*SIFT*) for feature extraction and matching. *SIFT* is

an excellent feature and method for application; however, it is very time consuming. In general, the vehicle speed is slow during parking and then the difference between two continuous images is small; thus a *SAD*-based matching method was alternatively used for feature-point matching.

For corner point (u, v) on image I_t matching a corner point (u'_k, v'_k) on image I_{t-1} , the sum of absolute difference (*SAD*) criterion is defined as

$$\min_{k=1,2,\dots,K} SAD[(u, v), (u'_k, v'_k)] = \min_{k=1,2,\dots,K} \sum_{i,j=-N/2}^{N/2} |f(u+i, v+j) - f(u'_k+i, v'_k+j)| \quad (11)$$

where $f(u, v)$ is the gray level of pixel (u, v) , K is the number of corner points on I_{t-1} and N is the radius of the window for computing a *SAD*. One example of corner-matching result is shown in Fig. 6.

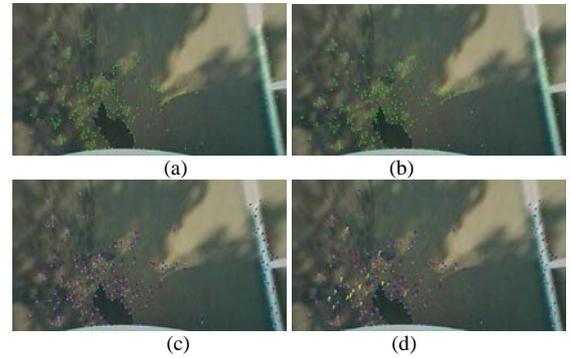


Fig. 6. Feature point matching. (a) The corner points on image I_t . (b) The corner points on image I_{t-1} . (c) The extracted point pairs. (d) The yellow lines are the credible point pairs.

B. Generation of Vehicle Trajectory

Ackermann geometry [6] described the motion trajectory of a four-wheel vehicle as shown in Fig. 7. In moving, all wheels have their axles arranged as radii of a circle with a common center point p . As the rear wheels are fixed, this center point must be on a line extended from the rear axle. Thus the center point can be found as the intersection of the rear-wheel and front-wheel axles.

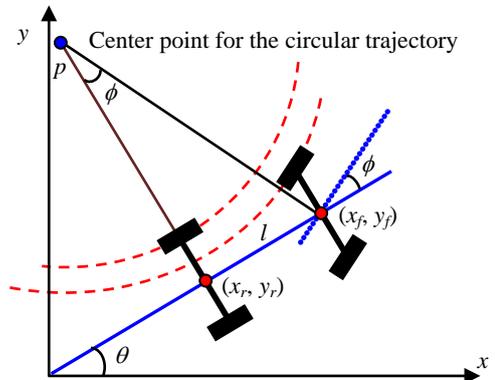


Fig. 7. Ackermann steering geometry and vehicular trajectory.

We have no steering sensor, so we cannot get the directions of the front wheels directly. Here we use the movement of the vehicle to obtain the wheel direction based on the feature points on images.

We analyze the geometric transformation between two continuous images based on the extracted feature point pairs as shown in Fig. 8. The isometric coordinate transformation model is used to simulate the Ackermann steering geometry to describe the vehicle motion. According to the Ackermann steering geometry, the vehicle is rotated with respect to a center point $p = (a, b)$. We use Eq.(12) to describe the vehicle motion from $t-1$ coordinate system (u', v') rotating θ angles to t coordinate system (u, v) .

$$\begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \left(\begin{bmatrix} u' \\ v' \end{bmatrix} + \begin{bmatrix} a \\ b \end{bmatrix} \right) \quad (12)$$

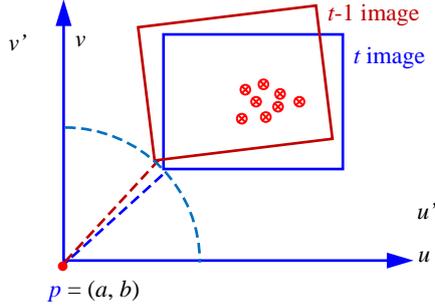


Fig. 8. The coordinate transformation model.

If we set $c = \cos\theta$ and $d = \sin\theta$, then Eq.(12) can be rewritten as two linear equations

$$\begin{cases} u = c(u'+a) - d(v'+b) - a \\ v = d(u'+a) + c(v'+b) - b \end{cases} \quad (13)$$

and then becomes

$$\begin{cases} u = c u' - d v' - a + c a - d b \\ v = d u' + c v' - b + d a + c b \end{cases} \quad (14)$$

Let

$$\begin{cases} e = (c-1)a - d b \\ f = d a + (c-1)b \end{cases} \quad (15)$$

then Eq.(14) becomes two linear equations of four unknowns $c, d, e,$ and $f,$

$$\begin{cases} u = c u' - d v' + e \\ v = d u' + c v' + f \end{cases} \quad (16)$$

If there are n corresponding points on the two continuous images, we can get $2n$ linear equations. Integrating these equations to form a linear system

$$\begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ \vdots \\ u_n \\ v_n \end{bmatrix}_{2n \times 1} = \begin{bmatrix} u'_1 & -v'_1 & 1 & 0 \\ v'_1 & u'_1 & 0 & 1 \\ u'_2 & -v'_2 & 1 & 0 \\ v'_2 & u'_2 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ u'_n & -v'_n & 1 & 0 \\ v'_n & u'_n & 0 & 1 \end{bmatrix}_{2n \times 4} \begin{bmatrix} c \\ d \\ e \\ f \end{bmatrix}_{4 \times 1} \quad (17)$$

The equation is simply written as

$$\beta = A \xi \quad (18)$$

then the unknowns (c, d, e, f) can be estimated by the

least-square estimation method,

$$\xi = (A^T A)^{-1} A^T \beta \quad (19)$$

At last, the center point (a, b) can be found by Eq.(15)

$$\begin{bmatrix} c-1 & -d \\ d & c-1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} e \\ f \end{bmatrix} \quad (20)$$

$$\begin{cases} a = \frac{e(c-1) + d f}{(c-1)^2 + d^2} \\ b = \frac{(c-1)f - d e}{(c-1)^2 + d^2} \end{cases} \quad (21)$$

Since $\cos^2\theta + \sin^2\theta = 1$; that is, $c^2 + d^2 = 1$; thus

$$\begin{cases} a = \frac{e(c-1) - d f}{2-2c} \\ b = \frac{(c-1)f + d e}{2-2c} \end{cases} \quad (22)$$

and $\theta = \tan^{-1}(d/c)$.

Some point pairs may involve more error; thus these point pairs should be pruned out. We utilize the square error to prune the point pairs which have more error. Substituting a point pair $[(u_i, v_i) (u'_i, v'_i)]$ into Eq.(16) to compute the square error

$$\varepsilon_i = [u_i - c u'_i + d v'_i - e]^2 + [v_i - d u'_i - c v'_i - f]^2 \quad (23)$$

If the error is greater than a pre-defined threshold value, the point pair is discarded. After that, all remained point pairs are used to re-compute the least-square estimation of transformation parameters again. At last, the parameters $a, b,$ and θ are ego-motion displacement vectors to generate the vehicular trajectory and obstacle detection.

With the coordinate transformation method, the vehicular trajectory is generated and then the parking guiding lines are drawn on top-view and original images as examples shown in Fig.9.



Fig. 9. The parking guiding lines. (a) drawn on the top-view image. (b) drawn on the original image.

IV. OBSTACLE DETECTION

The detection is done by comparing feature points in previous frame and in current frame displaced by ego-motion. This section describes the steps in detecting obstacles and eliminated ground-surface features such as lane markings that are not obstacles. By estimating ego-motion accurately, above-ground objects with detectable features will be detected, surface features will be rejected.

All features on ground surface, such as lane markings and manholes, will always have a rigid motion on hard ground. The ground's motion is the opposite of ego-motion, so transform a ground feature F'_t in current frame by ego-motion, we get its position that feature F'_{t-1} in previous frame. As shown in Fig 10, the feature F'_t in WCS is calculated from F_t in ICS by inverses of camera transformation and distortion function and F_{t-1} is calculated by projecting F'_{t-1} back to ICS.

We define a threshold T_g that F_t and F_{t-1} are the same surface features in current and previous frame if the sum of absolute difference is less than T_g , otherwise F_t is part of an obstacle. As shown in Fig. 11, the detected feature points have been divided into two groups, gray points are the removed ground features, and white points are part of obstacle.

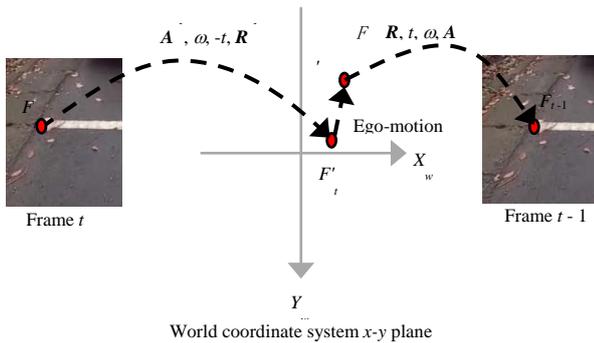


Fig. 10. Transform feature F_t in current frame to position in previous frame.

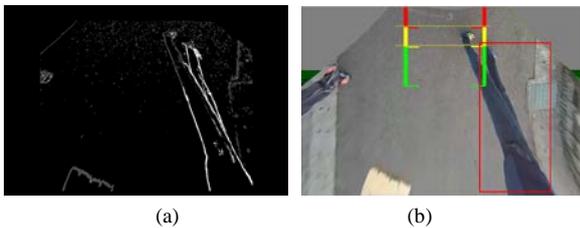


Fig. 11. The detected feature points have been divided into two groups. (a) Gray points are the removed ground features. (b) white points are part of obstacle.

V. EXPERIMENTS

The proposed methods were implemented in C++ programming language and Microsoft Foundation Class (MFC) Library, and all experiment were excepted on a general PC with Intel® Pentium® Core2 Duo 2.66GHz and 1.99GB RAM, Microsoft® Windows 7 operation system. A low-cost wide-angle camera with a view field of 136° angle in horizontal and 115° angle in vertical was used to capture images.

The performance of the proposed system is dependent on the speed of on-line processes: geometric transformation of images, feature detection, motion estimation, and obstacle detection. Currently, we can generate the guidance line images with 15 - 20 frames per second, and obstacle detection with 10 - 15 frames per second. Both can be switch between top-view and original images by driver. Two experiments with sequences of synchronous images are presented as shown in Fig. 12 and Fig. 13. Fig. 12 are a sequence images of parking guiding. Fig. 13 are a sequence images of obstacle detection.

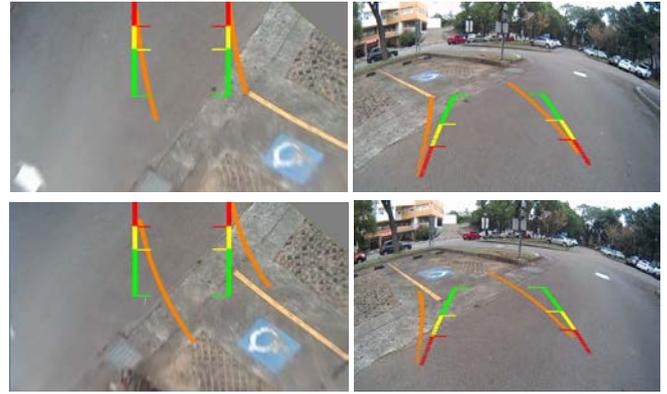


Fig. 12. A sequence images of parking guiding.

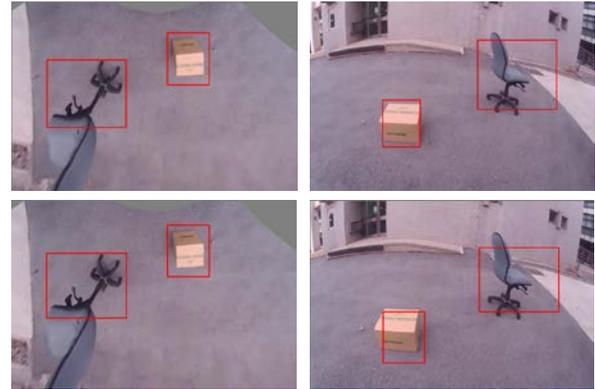


Fig. 13. A sequence images of obstacle detection.

VI. CONCLUSION

In this paper, an imaged-based parking guidance and obstacle detection system is proposed to help drivers parking their cars into parking space. The proposed system only uses a wide-angle camera to capture images for vehicle trajectory generation; the proposed system needs no steering sensor; it is a money-saved technique; moreover, it is suitable for used cars and after-market usage.

The proposed system only relies on image analysis to monitor the rear traffic situation for driver to avoid the possible collision. Compared to current rear anti-collision systems, our system gives the driver immediate view of vehicle's rear environment and the freedom to change view point. The visual detection system provides additional hint of above-ground objects in the environment, allow faster focus on potential obstacles.

The system is useful for driving in the narrow street or parking. The top-view helps the driver focus on understanding the parking size and the collision distance from the vehicle, it is especially useful in slower speed. It is sufficient in time to avoid possible collision around the host vehicle. The result of the top-view monitoring system is affected by the camera view angle and the image resolution. We are able to improve the quality of synthesis images by employ the wider-angle camera and higher resolution video devices. Furthermore, the system is just migrated into an embedded system.

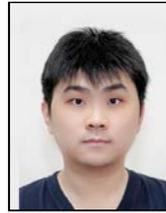
REFERENCES

- [1] R. Bishop, *Intelligent Vehicle Technology and Trends*, Boston: Artech House, 2005.

- [2] Nissan. *Around View Monitor*. [Online]. Available: <http://www.nissan-global.com/en/technology/overview/avm.html>
- [3] Honda. *Multi-View Camera System*. [Online]. Available: <http://world.honda.com/news/2008/4080918Multi-View-Camera-System/>
- [4] K. Ueminami, K. Hayase, and E. Sato, "Multi-around monitor system," *Mitsubishi Motors Technical Review*, no. 19, pp. 55-58, 2007.
- [5] Fujitsu. *360-Degree Wrap-around Video Imaging Technology*. [Online]. Available: http://www.fujitsu.com/us/news/pr/fma_20101019-02.html
- [6] D. King-Hele, "Erasmus Darwin's improved design for steering carriages-and cars," *Notes and Records of the Royal Society of London*, vol. 56, no. 1, pp. 41-62, Jan. 2002.
- [7] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. on Pattern Analysis and Machine Intell.*, vol. 22, no. 11, pp. 1330-1334, 2000.
- [8] C. C. Slama, Edit., *Manual of Photogrammetry*, 4th edition, American Society of Photogrammetry and Remote Sensing, Falls Church, Virginia, 1980.
- [9] A. W. Fitzgibbon, "Simultaneous linear estimation of multiple view geometry and lens distortion," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Kauai, Hawaii, Dec. 11-13, 2001, vol. 1, pp. 125-132.
- [10] D. Claus and A. W. Fitzgibbon, "A rational function lens distortion model for general cameras," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, San Diego, CA, 2005, vol. 1, pp. 213-219.
- [11] M. M. Fleck, "Perspective projection: The wrong imaging model," Technical Report TR 95-01, Computer Science, University of Iowa, 1995.
- [12] C. Geyer and K. Daniilidis, "Catadioptric projective geometry," *International Journal of Computer Vision*, vol. 45, no. 3, pp. 223-243, 2001.
- [13] F. Devernay and O. Faugeras, "Straight lines have to be straight," *Machine Vision and Application*, vol. 13, no. 1, pp. 14-24, 2001.
- [14] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," *Lecture Notes in Computer Science (ECCV 2006)*, vol. 3951, pp. 430-443, 2006.
- [15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91-110, Nov. 2004.



Din-Chang Tseng received his Ph.D. degree in computer science and 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 IEEE. His current research interests include computer vision, image processing, and virtual reality, especially in the topic: computer vision system for advanced safety vehicle, computer vision techniques for human computer interaction, and view-dependent multi-resolution terrain modeling



Yu-Chi Lin received his B.S. degree in Information Engineering and Computer Science from Feng Chia University, Taichung, Taiwan and M.S. degree in Information and Computer Engineering from Chung Yuan Christian University, Zhongli, Taiwan, in 2007 and 2009 respectively. He is pursuing the Ph.D. degree in the Department of Computer Science and Information Engineering at National Central University, Zhongli, Taiwan. His research interests include image processing and pattern recognition