

# A Convenient Vision-Based System for Automatic Detection of Parking Spaces in Indoor Parking Lots Using Wide-Angle Cameras

Shen-En Shih and Wen-Hsiang Tsai, *Senior Member, IEEE*

**Abstract**—A convenient indoor vision-based parking lot system using wide-angle fisheye-lens or catadioptric cameras is proposed, which is easy to set up by a user with no technical background. Easiness in the system setup mainly comes from the use of a new camera model that can be calibrated using only one space line without knowing its position and direction, as well as from the allowance of convenient changes in detected parking space boundaries. After camera calibration based on the new camera model is completed, parking space boundary lines are automatically extracted from input wide-angle images by a modified Hough transform with a new cell accumulation scheme, which can generate more accurate equal-width curves using the geometric relations of line positions and directions. In addition, the user may easily add or remove the boundary lines by single clicks on images, and parking spaces can be segmented out by region growing with the use of the boundary lines. Finally, vacant parking spaces can be detected by a background subtraction scheme. A real vision-based parking lot has been established and relevant experiments conducted. Good experimental results show the correctness, feasibility, and robustness of the proposed methods.

**Index Terms**—Hough transform, parking lot analysis, parking lot system, space line analysis, wide-angle cameras.

## I. INTRODUCTION

**N**OWADAYS, more and more cameras are equipped in our environment for various applications such as video surveillance, event recording, evidence collection, etc. In this paper, we investigate the use of cameras for parking lot management. Most cameras used in parking lot systems are perspective cameras; wide-angle cameras, such as fisheye-lens or catadioptric cameras, are not commonly adopted yet. No matter what type of camera systems is used, one problem in using them is the complicated system setup procedure, including camera calibration, environment learning, object modeling, etc., whose implementation usually requires the user to have a lot of tech-

nical knowledge. In addition, finding vacant parking spaces is often a trouble for people, particularly in large indoor parking lots, where one has to spend much time with concentration to drive through the entire parking area to visually search for a vacant space if the parking lot is nearly fully occupied. From these viewpoints, this study aims to develop an *intelligent* vision-based system using *wide-angle* cameras for parking lot management, which has the following merits: 1) the camera system can be set up *easily* by a common user with no technical background; 2) parking spaces can be detected *precisely*; and 3) vacant parking spaces can be identified *automatically* for convenient car parking.

On existing vision-based parking lot systems that use only perspective cameras, Blumer *et al.* [1] detected vacant parking spaces by a single perspective camera using edge and road color information and constructed the regions of the parking spaces manually. Huang and Wang [2] detected vacant parking spaces by perspective cameras and used a three-layer Bayesian hierarchical framework to tackle the problems of luminance variations, shadow effects, occlusions among vehicles, etc., after the 3-D models of the parking spaces were constructed in advance. Ichihashi *et al.* [3] detected vacant parking spaces using perspective cameras based on fuzzy *c*-means clustering and particle swarm optimization and marked the regions of parking spaces manually. Lixia and Dalin [4] determined the vacancy of a parking space based on image segmentation and local binary patterns, providing that the pixels of the parking space are known in advance. In these methods, the regions of the parking spaces were segmented out *manually*. On the contrary, the method proposed in this study aims to conduct this major step *automatically* to construct an easy-to-set-up system.

As to camera calibration, which also influences the convenience in system usage, the goal of this study is to provide a smart calibration process for a wide-angle camera that can be easily performed by a user with no technical knowledge. It is desired to utilize only a single image of the application environment and common features existing in it without knowing their directions, positions, etc. Many self-calibration methods have been proposed to make the calibration process easy. Deng *et al.* [5] proposed a method for calibrating a central (single-viewpoint) catadioptric camera by the use of a 1-D object with unknown motions by deriving the principal point and the image center first, followed by calculating the focal length, skew factor, and mirror parameters of the camera. Ramalingam *et al.* [6] calibrated the projection rays of a central camera purely from matches between images. Gasparini *et al.*

Manuscript received April 25, 2013; revised October 30, 2013; accepted December 13, 2013. Date of publication January 2, 2014; date of current version July 10, 2014. This work was supported in part by the National Science Council, Taiwan under Grant 102-2218-E-009-003 and in part by the Ministry of Education, Taiwan under the five-year project "Aiming for the Top University" (2011–2015). The review of this paper was coordinated by Dr. Y. Gao.

S.-E. Shih is with the Institute of Computer Science and Engineering, National Chiao Tung University, Hsinchu 30010, Taiwan (e-mail: peter159.cs98g@nctu.edu.tw).

W.-H. Tsai is with the Department of Computer Science, National Chiao Tung University, Hsinchu 30010, Taiwan, and also with the Department of Information Communication, Asia University, Taichung 41354, Taiwan (e-mail: whtsai@cis.nctu.edu.tw).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TVT.2013.2297331

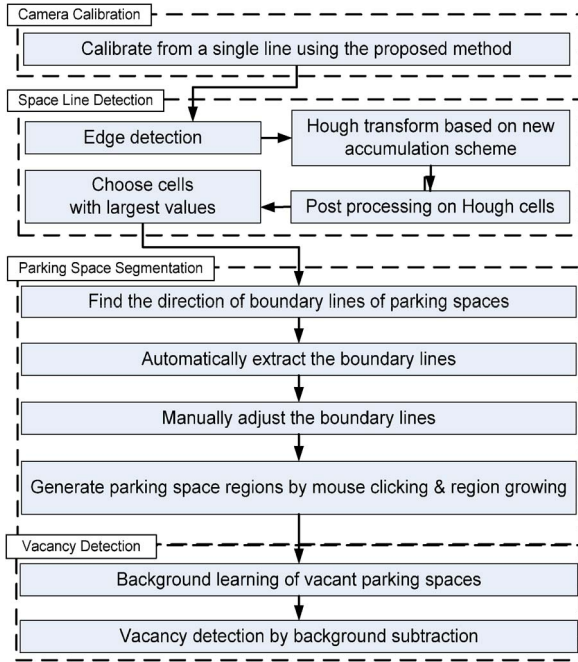


Fig. 1. Proposed parking lot system.

[7] showed that at least three homographies obtained from a grid-pattern board are required to calibrate a general central camera, and a single homography is enough to calibrate a parabolic catadioptric camera.

The indoor parking lot system proposed in this study utilizes multiple wide-angle cameras affixed on the ceiling and looking downward vertically. The system operations can be divided into four stages, as shown in Fig. 1: camera calibration, space line detection, parking space segmentation, and vacancy detection. In the first stage, a new camera model is proposed, leading to the use of *fewer* parameters than conventional models and a *single* space line for calibrating the model. This is why the proposed calibration process can be done by a user with no technical background. In the second stage, space lines appearing in the captured image are detected by an edge detection process, followed by a Hough transform based on a *new* cell accumulation scheme. *Equal-width* curves are so generated, leading to more precise space line detection results for use as the boundary lines of the parking spaces.

In the third stage, the detected boundary lines are analyzed and displayed automatically, and the user can then simply click on the image to add or remove the boundary lines. In addition, by clicking on any position within each parking space, the space region is generated by a region-growing process. Finally, after “learning” the background parameters of the vacant parking spaces for various environment conditions, vacant spaces can be found by background subtraction for use in parking management.

The proposed vision-based parking lot system has at least the following merits: 1) wide-angle cameras are used so that *less* cameras are needed to cover the area of a given parking lot; 2) the camera system can be calibrated using *only one line* in the environment, so that the system can be set up easily by a user with no technical background; 3) the proposed Hough transform with the new cell accumulation scheme

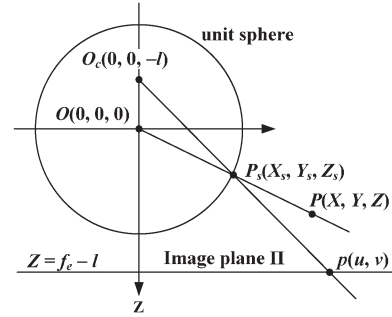


Fig. 2. Illustration of two-step spherical mapping.

generates equal-width curves, so that the proposed system is capable of directly dealing with distorted images captured by wide-angle cameras, and the space line detection results are more precise than those yielded by conventional methods; and 4) unlike many previous studies, which specify parking spaces *manually*, the proposed method detects them *automatically* for a convenient system setup as well.

In the remainder of this paper, the details of the proposed calibration scheme are described in Section II, the proposed space line detection method is described in Section III, and the proposed parking space segmentation and vacancy detection techniques are presented in Section IV. Experimental results and discussions are provided in Section V to show the correctness, feasibility, and robustness of the proposed method.

## II. PROPOSED CAMERA MODEL AND CALIBRATION METHOD

To design an easy-to-set-up vision-based parking lot system, the camera calibration process must be easy to carry out by users with no technical knowledge. In this paper, a simplified camera model is proposed for this aim. A calibration method is proposed accordingly, which makes use of *only one* space line in the environment without knowing its position and direction. In the following, an existing *unifying* camera model for wide-angle cameras is briefly reviewed first in Section II-A. Then, the proposed camera model is described in Section II-B, and the proposed calibration method is presented in Section II-C.

### A. Review of a Unifying Model for Wide-Angle Cameras

In the unifying model proposed by Geyer and Daniilidis [8], the relation between a space point  $P$  and its corresponding pixel  $p$  is described by a two-step mapping via a unit sphere, as shown in Fig. 2. Specifically, point  $P$  is first projected onto point  $P_s$  on the unit sphere as being viewed from the *effective viewpoint*  $O$  through the sphere. Then,  $P_s$  is perspective projected onto pixel  $p$  on image plane II as being viewed from the “pinhole” point  $O_c$  (usually the lens center). The two involved parameters  $l$  and  $f_e$ , as shown in Fig. 2, are defined in the same way as suggested by Ying and Hu [9].

The unifying model previously described has been proved to be equivalent to models for perspective, parabolic catadioptric, and hyperbolic catadioptric cameras [8]. Additionally, it was shown by Ying and Hu [10] to be suitable for modeling fisheye-lens cameras as well. In this paper, we simplify it to

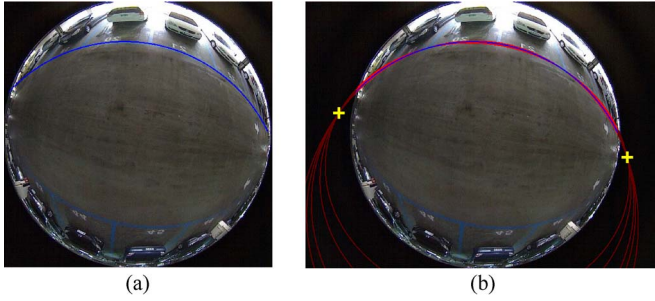


Fig. 3. Fitting a space line using different values of  $l$ . (a) Space line to be fit (marked as blue). (b) Fitting results using  $l = 2.0, 1.5, 1.0, 0.8,$  and  $0.5$  with larger ellipses corresponding to smaller values of  $l$ , and the two yellow crosses indicate the fixed vanishing points.

model fisheye-lens and hyperbolic catadioptric cameras, as will be described in the following section.

### B. Proposed Camera Model for Wide-Angle Cameras

The proposed simplified unifying camera model is based on the use of an optimal approximation value of parameter  $l$ , which is the distance from the effective viewpoint  $O$  to the pinhole point  $O_c$ , as shown in Fig. 2. The model has two merits: 1) it reserves important characteristics of space lines, as shown in this section; and 2) it can be easily calibrated by the use of a single space line, as described in Section II-C. These merits make the corresponding system setup process easy to conduct.

The rationale of finding a fixed optimal value of parameter  $l$  can be explained as follows. In Fig. 3(a), the image of a space line, called *line image* hereafter, is marked as a blue curve, and in Fig. 3(b), this line image is shown to be fit well enough by conic sections [i.e., by (2) derived later in this section] with different values of parameter  $l$ , whereas the two vanishing points are fixed (marked in yellow in the figure). This phenomenon leads to two conclusions: 1) parameter  $l$  cannot be well calibrated from line images; and 2) reversely, the value of parameter  $l$  did not affect the space line detection process. The first conclusion is consistent with some previous studies [9], [11]. Specifically, parameter  $l$  was fixed in the simulation experiments described by Ying and Hu [9]; hence, parameter  $l$  was not derived in the calibration process. Furthermore, as seen in the work of Deng *et al.* [11], parameter  $l$  is assumed to be known before the calibration. The second conclusion makes it possible to find an optimal value of parameter  $l$  to approximate that of any kind of wide-angle camera, without affecting the space line detection process.

To find the optimal value of  $l$ , we define first the range of parameter  $l$  of each kind of wide-angle camera. For parabolic catadioptric cameras, the value of  $l$  is known to be 1.0 [8]; for hyperbolic catadioptric cameras, the value of  $l$  is smaller than 1.0 and larger than 0.0 [8]; and for fisheye-lens cameras, the value of  $l$  is larger than 1.0 [10]. In this paper, we define the interesting range of parameter  $l$  to be  $0.5 < l < 2.0$ , which includes the commonly used values of  $l$ . For example, the values of  $l$  derived in [9] and [11] are 0.8, 0.966, and 1.0, the value derived in [12] is 0.9663, and the value derived in [7] is 1.07.

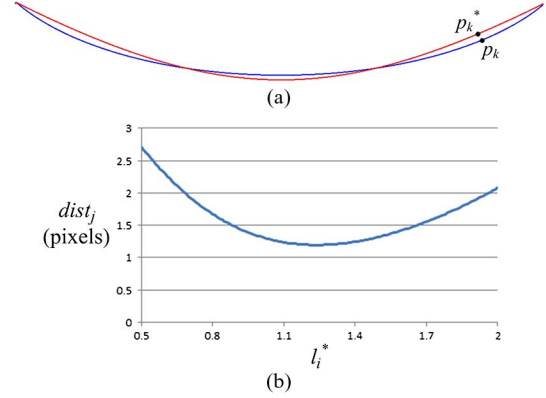


Fig. 4. Finding the optimal value  $l^*$ . (a) A line image  $I_i$  (marked in blue) and its best-fitting curve  $I_{ij}^*$  (marked in red). (b) The trend of the average  $dist_j$  of the distances between the best-fitting curve and the line images. The optimal value of  $l_j^*$  is 1.24, with  $dist_j = dist$  being about 1.1941 pixels.

Then, the optimal value  $l^*$  of  $l$  is found by a simulation process, as described in the following.

- 1) Generate simulated line images  $I_i$  with size  $1000 \times 1000$  for a set of sampled values of  $l$  in the interesting range  $0.5 < l < 2.0$  and for a set of sampled positions and directions of space lines.
- 2) For each sample value  $l_j^*$  in the range  $0.5 < l_j^* < 2.0$ , do steps 2a) through 2c) as follows:
  - a) find the best-fitting curve  $I_{ij}^*$  to each line image  $I_i$ , where  $I_{ij}^*$  is computed by (2) with  $l = l_j^*$  described in Section II-C and a Levenberg–Marquardt process;
  - b) compute distance  $dist_{ij}$  between  $I_i$  and  $I_{ij}^*$  as
$$dist_{ij} = \frac{1}{n} \sum_{k=1}^n \|p_k - p_k^*\| \quad (1)$$

where  $n$  denotes the number of pixels on line image  $I_i$ ,  $p_k$  is a pixel on  $I_i$ ,  $p_k^*$  is the nearest pixel on  $I_{ij}^*$  to  $p_k$ , and  $\|p_k - p_k^*\|$  is the distance between  $p_k$  and  $p_k^*$ ;
  - c) calculate the average distance  $dist_j$  of all  $dist_{ij}$  of all  $I_i$  as a measure of optimality of the sampled value  $l_j^*$ , with a smaller  $dist_j$  indicating a better fit of  $l_j^*$  to all  $I_i$ .
- 3) Choose as the desired optimal value  $l^*$  the  $l_j^*$  with the smallest  $dist_j$ , which is called the average fitting error and denoted as  $dist$  subsequently.

An experimental result of the given process is shown in Fig. 4, where Fig. 4(a) shows line image  $I_i$  marked in blue and a best-fitting curve  $I_{ij}^*$  marked in red, and Fig. 4(b) shows the trend of the value of  $dist_j$  for different  $l_j^*$  values, from which it can be seen that an optimal value of  $l_i^*$  does exist and is located at 1.24 for choice as  $l^*$  and that the line images can be well approximated by (2) with  $l = l^*$  resulting in an average fitting error  $dist \approx 1.1941$  pixels.

As a result of utilizing the aforementioned second conclusion, we propose a camera model that is identical to the unifying model proposed in [8] but with its parameter  $l$  fixed to be the optimal value 1.24 previously derived. One merit of this model is that it leads to the possibility of calibrating

wide-angle cameras using only one line image. This property is a great advantage over those of the conventional models [8], [13], by which the camera cannot be reliably calibrated from line images as previously proved in Section II-B. It also facilitates a nontechnical user being able to conduct the calibration process without difficulty as previously mentioned.

### C. Proposed Calibration Method Using Only One Space Line

Based on the proposed camera model using the fixed parameter  $l = 1.24$ , the idea of the proposed calibration process using a single line can be divided into three steps. First, a space line is chosen with its line image (in the shape of a conic-section curve) manually marked. Then, the best-fitting ellipse to this line image is computed, from which the unknown camera parameters and the space line are roughly estimated. Finally, a Levenberg–Marquardt algorithm is conducted to find the precise values of the camera parameters.

In more detail, let  $L$  be a chosen space line,  $I_L$  be its line image, and  $E_L$  be the best-fitting ellipse to  $I_L$ . As derived in [9] and [13],  $I_L$  can be expressed as

$$F(u, v) = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}^T \begin{bmatrix} (l^2-1)n_x^2 + l^2n_z^2 & (l^2-1)n_xn_y & -f_en_zn_x \\ (l^2-1)n_xn_y & (l^2-1)n_y^2 + l^2n_z^2 & -f_en_zn_y \\ -f_en_zn_x & -f_en_zn_y & -f_e^2n_z^2 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = 0 \quad (2)$$

where  $(u, v)$  are coordinates of pixels on  $I_L$ ,  $(n_x, n_y, n_z)$  is the normal vector of plane  $\Pi$  formed by line  $L$  and the origin of the camera's lens center, and  $(l, f_e)$  are the parameters used in the unifying camera model, as described in Section II-A. In addition, let ellipse  $E_L$  be described by

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}^T \begin{bmatrix} a & b & d \\ b & c & e \\ d & e & f \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = 0. \quad (3)$$

Note that when  $n_z = 0$ , the line image is a straight line going through the image center so that parameter  $f_e$ , which is the *effective focal length* of the camera, cannot be calibrated [9], [14]. Ignoring this, we may rewrite (2), after dividing it by  $n_z^2 \neq 0$ , as

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}^T \begin{bmatrix} (l^2-1)G^2 + l^2 & (l^2-1)GH & -f_eG \\ (l^2-1)GH & (l^2-1)H^2 + l^2 & -f_eH \\ -f_eG & -f_eH & -f_e^2 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = 0 \quad (4)$$

where  $G = n_x/n_z$ , and  $H = n_y/n_z$ .

Now, the problem is to estimate *roughly* the values of parameters  $(l, f_e, G, H)$  such that (3) and (4) are close to each other. Since only rough estimation is needed, we first simplify (4) by assuming  $l = 1.0$ . Accordingly, the problem is reduced to finding parameters  $(f_e, G, H)$  that satisfy

$$\begin{bmatrix} 1 & 0 & -f_eG \\ 0 & 1 & -f_eH \\ -f_eG & -f_eH & -f_e^2 \end{bmatrix} \sim \begin{bmatrix} a & b & d \\ b & c & e \\ d & e & f \end{bmatrix} \quad (5)$$

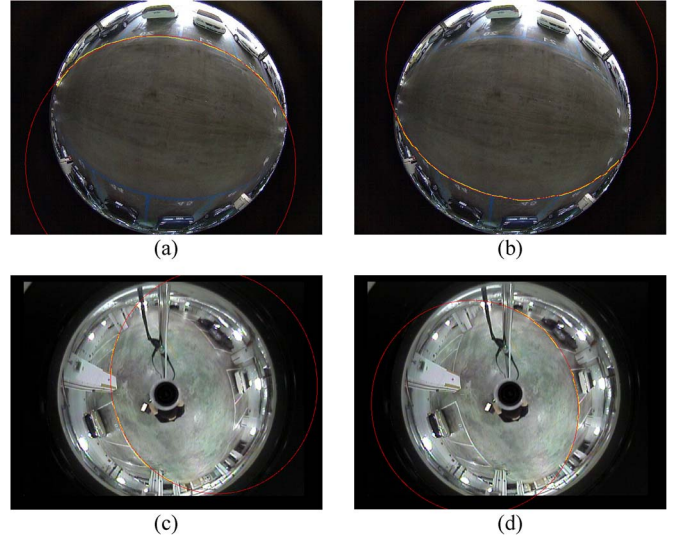


Fig. 5. Calibration results with yellow curves indicating manually marked pixels and red ellipses being the best-fitting results. (a) and (b) Results using a fisheye-lens camera with calibrated values of  $f_e$  being 319.90 and 319.57, respectively. (c) and (d) Results using a hyperbolic catadioptric camera with calibrated values of  $f_e$  being 266.73 and 269.53, respectively.

where “ $\sim$ ” means “equals up to a scale.” Let  $\lambda$  be the hidden unknown parameter for this scaling. Then, we have

$$\begin{bmatrix} \lambda & 0 & -\lambda f_e G \\ 0 & \lambda & -\lambda f_e H \\ -\lambda f_e G & -\lambda f_e H & -\lambda f_e^2 \end{bmatrix} = \begin{bmatrix} a & b & d \\ b & c & e \\ d & e & f \end{bmatrix}. \quad (6)$$

One solution to the given equation for use as rough estimates of the parameters is

$$\lambda = \frac{a+c}{2}; \quad f_e = \sqrt{-\frac{f}{\lambda}}; \quad G = \frac{-d}{\lambda f_e}; \quad \text{and} \quad H = \frac{-e}{\lambda f_e}. \quad (7)$$

A Levenberg–Marquardt process is finally conducted to derive the precise values of  $(f_e, G, H)$ , with the initial values being specified by (7) and the criterion being to minimize the value of

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}^T \begin{bmatrix} (l^2-1)G^2 + l^2 & (l^2-1)GH & -f_eG \\ (l^2-1)GH & (l^2-1)H^2 + l^2 & -f_eH \\ -f_eG & -f_eH & -f_e^2 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (8)$$

with respect to all the pixels  $(u, v)$  on line image  $I_L$ . After this optimization process is done, parameter  $f_e$  of the camera model is derived, completing the calibration process.

Four results of this calibration process are shown in Fig. 5. The calibrated values  $f_e$  in Fig. 5(a) and (b) are 319.90 and 319.57, respectively, and those in Fig. 5(c) and (d) are 266.73 and 269.53, respectively. The validity of the proposed calibration method can be shown by the good fitness of the best-fitting ellipse to the manually marked line image in each case and the closeness of the calibrated values  $f_e$  in the first two cases using a fisheye-lens camera and in the remaining two cases using a hyperbolic catadioptric camera.

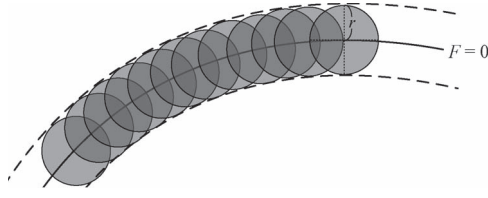


Fig. 6. Illustration of an equal-width curve along a curve  $F = 0$  with width  $r$ , defined to be the regions of all the gray circles (not all drawn) or, equivalently, the thick area bounded by the two dashed curves.

### III. PROPOSED SPACE LINE DETECTION METHOD

One of the important features in man-made environments is a straight line, and as formulated by (2), these space lines are projected by wide-angle cameras to form conic sections in the resulting image, called the *line image* so far. Here, the proposed method for detecting the conic-section curve is described, which, different from conventional methods, generates *equal-width regions* along a curve, as shown in Fig. 6. As a result, the proposed method can detect *thick curves* in images more reliably, overcome the noise produced by edge detection, and be utilized in accordance with the proposed camera model (described in Section II-B) for more accurate detection of parking space boundary lines.

#### A. Proposed Detection Method

The proposed space line detection method is based on the Hough transform with a new cell accumulation scheme to achieve the ability to detect equal-width thick curves. As described by (2), a line image, which appears to be a conic section in the acquired wide-angle image, can be parameterized by the normal vector  $(n_x, n_y, n_z)$  of plane  $\Pi$  formed by the corresponding space line and the origin of the camera. The normal vector can be expressed as

$$(n_x, n_y, n_z) = (A, B, \sqrt{1 - A^2 - B^2}) \quad (9)$$

where

$$A = \frac{n_x}{\sqrt{n_x^2 + n_y^2 + n_z^2}}, \quad B = \frac{n_y}{\sqrt{n_x^2 + n_y^2 + n_z^2}} \quad (10)$$

and  $0 \leq A, B \leq 1$ . Parameters  $(A, B)$  are quantized into  $n \times n$  values to form *Hough cells* in this study, with the line images (conic-section curves) corresponding to the cells being described by the equation  $F(u, v) = 0$  in (2), and the values  $(n_x, n_y, n_z)$  in (2) defined by (9) and (10).

The first step in the proposed detection method is to obtain a binary version  $I_b$  of the input wide-angle image by edge detection. Then, for each Hough cell at parameters  $(A, B)$  and for each “black” pixel  $p$  in  $B$  with coordinates  $(u, v)$ , if  $p$  is in the equal-width curve region of the line specified by parameters  $(A, B)$ , then the cell value at  $(A, B)$  is incremented by 1.

In more detail, by referring to Fig. 7, it can be figured out that pixel  $(u, v)$  is in the equal-width region with width  $r$  if and only if the curve  $F = 0$  intersects circle  $C$  centered at  $(u, v)$  with radius  $r$ . Furthermore, according to the intermediate value theorem [16], the curve  $F = 0$  intersects circle  $C$  if there exists a point  $(u', v') = (u + \Delta u, v + \Delta v)$  on  $C$  such that  $F(u', v')$  and  $F(u, v)$  are with opposite signs. Accordingly, if  $F(u, v)$  is

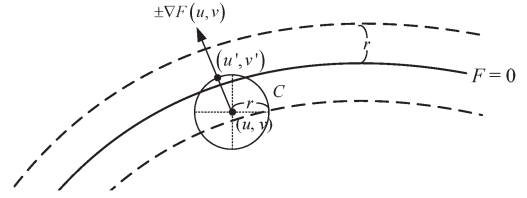


Fig. 7. Illustration for determining whether or not a pixel at  $(u, v)$  lies inside an equal-width curve.

*positive*, we try to find point  $(u', v')$  on  $C$  with the *minimum* value of  $F(u', v')$  and then determine whether  $F(u', v')$  and  $F(u, v)$  are with opposite signs. Reversely, if  $F(u, v)$  is *negative*, we try to find point  $(u', v')$  on  $C$  with the *maximum* value of  $F(u', v')$  and then determine whether  $F(u', v')$  and  $F(u, v)$  are with opposite signs. For the first case with  $F(u, v) \geq 0$ , since  $F(u', v')$  should be the *minimum* value and since gradient  $\nabla F$  specifies the direction of *increasing* the function value  $F(u, v)$ , vector  $(\Delta u, \Delta v)$  should be in the *negated* direction of gradient  $\nabla F$ . Hence, under the constraint that the length of  $(\Delta u, \Delta v)$  is radius  $r$ , vector  $(\Delta u, \Delta v)$  may be expressed as  $[-\nabla F(u, v)/\|\nabla F(u, v)\|] \times r$ . For the second case with  $F(u, v) < 0$ , since  $F(u', v')$  should be the *maximum* value, it can be derived by similar reasoning that vector  $(\Delta u, \Delta v)$  should be in the *same* direction of gradient  $\nabla F$  so that it can be expressed as  $[\nabla F(u, v)/\|\nabla F(u, v)\|] \times r$ . As a summary, we have

$$\begin{aligned} \begin{bmatrix} u' \\ v' \end{bmatrix} &= \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} \\ \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} &= \begin{cases} \frac{-\nabla F(u, v)}{\|\nabla F(u, v)\|} r, & \text{if } F(u, v) \geq 0 \\ \frac{\nabla F(u, v)}{\|\nabla F(u, v)\|} r, & \text{if } F(u, v) < 0. \end{cases} \end{aligned} \quad (11)$$

Accordingly, the function value  $F(u', v')$  can now be derived by the second-order Taylor expansion to be

$$F(u', v') = F(u, v) + \nabla F(u, v)^T \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} + \frac{1}{2} \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix}^T HF(u, v) \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} \quad (12)$$

where  $HF(u, v)$  is the Hessian matrix of function  $F$  described by

$$HF = \begin{bmatrix} \frac{\partial^2 F}{\partial u^2} & \frac{\partial^2 F}{\partial u \partial v} \\ \frac{\partial^2 F}{\partial u \partial v} & \frac{\partial^2 F}{\partial v^2} \end{bmatrix} \quad (13)$$

whose details may be derived from (2) and are omitted here.

In conclusion, to detect an equal-width line in a binarized wide-angle image  $I_b$  by the proposed Hough transform, we conduct the following process.

- 1) For each pixel  $p$  with coordinates  $(u, v)$  in  $I_b$  and for each Hough cell with parameters  $(A, B)$ , calculate  $(\Delta u, \Delta v)$  and  $(u', v')$  by (11), and  $F(u, v)$  and  $F(u', v')$  by (2) and (12), respectively.
- 2) If  $F(u, v)$  and  $F(u', v')$  are with opposite signs, decide the pixel  $p$  at  $(u, v)$  to be in the equal-width region of the line specified by parameters  $(A, B)$  and increment the value of the Hough cell with  $(A, B)$  by 1.
- 3) Find out the parameters  $(A, B)$  of the desired line images (conic-section curves) from the resulting Hough space by the following steps.

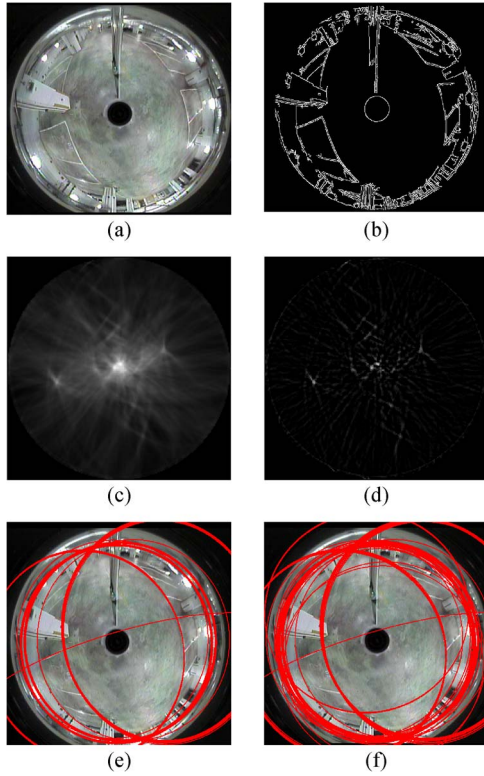


Fig. 8. Results of the proposed space line detection method. (a) Input image. (b) Canny edge result. (c) Hough space. (d) Hough space after applying filter (14). (e) and (f) Results of 30 and 50 lines, respectively.

- a) Enhance the local maxima of the Hough space by applying the following sharpening filter:

$$\frac{1}{25} \begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 24 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \end{bmatrix}. \quad (14)$$

- b) Pick out all the Hough cells with values larger than a preselected threshold  $\delta$  and take the cells' parameters  $(A, B)$  for use as those of the detected line images.
- 4) Use the derived parameters  $(A, B)$  to draw the conic sections of the detected line images on the input image as the output.

Moreover, the curve width  $r$  used in (11) is defined by considering the error introduced in the edge detection process as well as the error introduced by the approximation of parameter  $l$ , as mentioned in Section II-B. The former is approximated to be 1 pixel, and the latter is approximated to be 1.2 pixels, according to the data shown in Fig. 4. As a result, the width  $r$  used in this study is taken to be  $r = 1.0 + 1.2 = 2.2$  pixels.

A result of the given line image detection method is shown in Fig. 8. The input image captured by a hyperbolic catadioptric camera is shown in Fig. 8(a), with the Canny edge detection result obtained from it shown in Fig. 8(b). The generated Hough space is shown in Fig. 8(c), and the result of applying filter (14) is shown in Fig. 8(d). Finally, some cells with larger values were picked out, and the line images corresponding to the cells' parameters  $(A_i, B_i)$  were drawn, as shown in Fig. 14(e) and (f), where the numbers of drawn lines are 30 and 50, respectively.

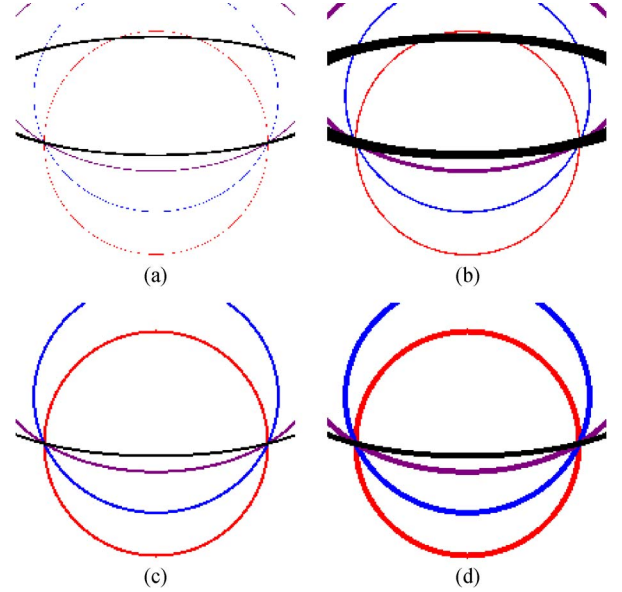


Fig. 9. Comparison of the conventional and proposed accumulation methods. (a) and (b) Non-equal-width results yielded by the conventional method with thresholds  $T = 40$  and  $150$ , respectively. (c) and (d) Equal-width results yielded by the proposed method with widths  $r = 1$  and  $2$ , respectively.

### B. Effectiveness of Proposed Cell Accumulation Scheme

As previously stated, each Hough cell is specified by two parameters  $A$  and  $B$  and corresponds to a conic-section curve  $F(u, v) = 0$  in the input image. Since parameters  $A$  and  $B$  and image coordinates  $u$  and  $v$  are all quantized to be discrete, the function value  $F(u, v)$  for a certain coordinate pair  $(u, v)$  may *not* be exactly zero. A conventional way to deal with this problem is to define threshold  $T$  and consider pixels with coordinates  $(u, v)$  satisfying the following inequality as being on the conic section described by  $F(u, v) = 0$ :

$$|F(u, v)| \leq T. \quad (15)$$

Two results yielded by this method are shown in Fig. 9(a) and (b) with different thresholds  $T = 40$  and  $150$ , respectively. As can be seen, the generated conic sections are *not* with equal widths. This phenomenon results from some undesirable operations that cause some edge pixels not belonging to the conic section to be accumulated into wrong Hough cells, and vice versa. In contrast, the proposed method described in Section III-A yields equal-width curves, as shown in Fig. 9(c) and (d), for different curve widths  $r = 1$  and  $2$  pixels, respectively. These results show the capability of the proposed method for detecting equal-width curves, which is superior to conventional methods and suitable for applications requiring higher line detection accuracy.

Some experiments were also conducted to compare the space line detection results yielded by the conventional Hough cell accumulation scheme and those by the proposed method. An input image and the corresponding Canny edge detection result are shown in Fig. 10(a) and (b), respectively. The 10 and 30 detected space line images yielded by the conventional scheme are drawn in Fig. 10(c) and (d), respectively, and those yielded by the proposed method are shown in Fig. 10(e) and (f),

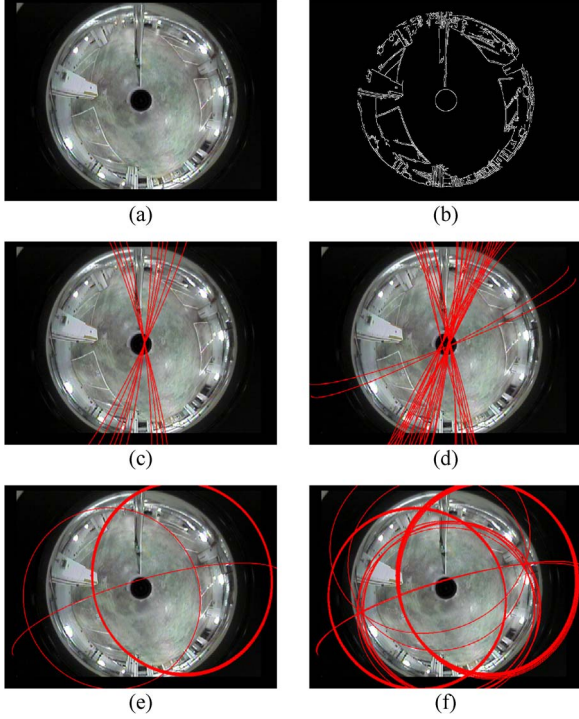


Fig. 10. Comparisons of space line detection results yielded by the conventional and proposed methods. (a) Input image. (b) Edge detection result. (c) and (d) The 10 and 30 detected curves using the conventional method. (e) and (f) The 10 and 30 detected curves using the proposed method.

respectively. As shown in the figure, the curves detected by the conventional scheme mostly go through the image center, which are not good results. This undesirable phenomenon may be explained from the simulation results shown in Fig. 9. Specifically, as shown in Fig. 9(b), the conventional scheme generates a thicker curve when the real curve appears nearer the image center, and vice versa. This means that each Hough cell corresponding to a curve nearer the image center will collect “votes” from a *thicker line band* in the input image (i.e., *more edge pixels* will be “accumulated” into such a cell), and vice versa. Therefore, when using the conventional scheme to detect curves by picking up Hough cells with *larger votes*, those curves nearer to image center are *more likely* to be chosen first, yielding the results shown in Fig. 10(c) and (d). In contrast, since the proposed method generates equal-width curves *everywhere*, this phenomenon does not appear in the detection results yielded by the proposed method shown in Fig. 10(e) and (f).

#### IV. DETECTION OF PARKING SPACE VACANCY

Each parking space is usually marked by some colored boundary lines. Here, some properties of such boundary lines are first derived in Section IV-A. Then, a method to detect such boundary lines is proposed in Section IV-B. Finally, in Section IV-C, a method to mark the regions of parking spaces using boundary lines and a simple vacancy detection method based on *background subtraction* are proposed, by which whether a car is parked in a parking space can be decided.

##### A. Properties of the Boundary Lines Around Parking Spaces

Three properties of the boundary lines of parking spaces can be identified: 1) lying on the ground; 2) being either parallel or perpendicular to one another; and 3) being ignorable if too far away from the camera. These properties are discussed in more detail as follows.

As stated in Section I, the wide-angle cameras used in the proposed parking lot system are affixed on the ceiling to look right downward; hence, the optical axis of the camera, namely, the  $Z$ -axis of the camera coordinate system as shown in Fig. 2, is vertical with respect to the ground specified as the  $XY$ -plane. Given a boundary line  $L_1$  on the ground, its directional vector  $(d_{1x}, d_{1y}, d_{1z})$  is parallel to the  $XY$ -plane; hence, we have

$$d_{1z} = 0. \quad (16)$$

Let  $(A_1, B_1)$  be the Hough parameters of line  $L_1$  found by the space line detection method proposed in Section III and  $(n_{1x}, n_{1y}, n_{1z})$  be the normal vector of the plane formed by  $L_1$  and the camera origin. Then, from (9), we have

$$(n_{1x}, n_{1y}, n_{1z}) = \left( A_1, B_1, \sqrt{1 - A_1^2 - B_1^2} \right). \quad (17)$$

In addition, since  $(n_{1x}, n_{1y}, n_{1z})$  is perpendicular to the directional vector  $(d_{1x}, d_{1y}, d_{1z})$  of  $L_1$ , we have

$$(n_{1x}, n_{1y}, n_{1z}) \cdot (d_{1x}, d_{1y}, d_{1z}) = 0 \quad (18)$$

which, when combined with (16) and (17), leads to

$$A_1 d_{1x} + B_1 d_{1y} = 0 \quad (19)$$

so that the directional vector  $(d_{1x}, d_{1y}, d_{1z})$  can be expressed in terms of  $A_1$  and  $B_1$  as

$$(d_{1x}, d_{1y}, d_{1z}) = (B_1, -A_1, 0). \quad (20)$$

Let  $(A_2, B_2)$  be the parameters of another boundary line  $L_2$ . Then, the directional vector  $(d_{2x}, d_{2y}, d_{2z})$  of  $L_2$  can be similarly derived to be

$$(d_{2x}, d_{2y}, d_{2z}) = (B_2, -A_2, 0). \quad (21)$$

Now, since the two boundary lines  $L_1$  and  $L_2$  are either parallel or perpendicular to each other, two cases can be identified. For the first case where the two boundary lines are parallel, their directional vectors satisfy

$$(d_{1x}, d_{1y}, d_{1z}) = \lambda(d_{2x}, d_{2y}, d_{2z}) \quad (22)$$

which, when combined with (20) and (21), leads to

$$(A_1, B_1) = \lambda(A_2, B_2) \quad (23)$$

where  $\lambda$  is a scalar value. For the other case where the two lines are perpendicular, the inner product of their directional vectors satisfy

$$(d_{1x}, d_{1y}, d_{1z}) \cdot (d_{2x}, d_{2y}, d_{2z}) = 0 \quad (24)$$

which, with the use of (20) and (21), leads to

$$(A_1, B_1) \cdot (A_2, B_2) = 0. \quad (25)$$

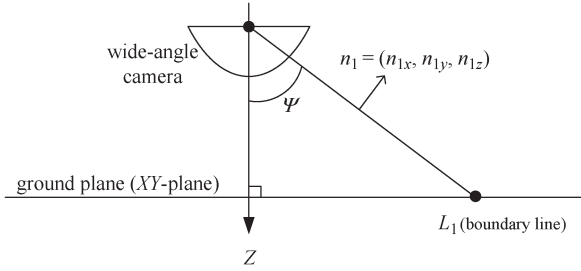


Fig. 11. Definition of the elevation angle  $\Psi$  of a boundary line  $L_1$ .

Moreover, to ignore boundary lines that are too far away, we check the *elevation angle*  $\psi$  of each boundary line, which is defined in a way as shown in Fig. 11 such that the larger the angle  $\Psi$  is, the farther the boundary line is from the camera. According to our experimental experience, if the camera is mounted at 2.5 m high, the farthest usable boundary line is at a distance about 10.0 m from the camera. As a result, the maximum value of  $\Psi$  is  $\tan^{-1}(10.0/2.5) = 75.96^\circ$ , which is equivalent to  $n_{1z} \leq \sin(\Psi) = 0.970$ . Using this result with (17), we can derive another constraint on parameters  $(A, B)$  as  $n_{1z} = \sqrt{1 - A_1^2 - B_1^2} \leq 0.970$  or, equivalently, as

$$A_1^2 + B_1^2 \geq 0.059. \quad (26)$$

### B. Automatic Detection of Boundary Lines

As shown in Fig. 8(e) and (f), the results of the line image detection method proposed in Section III are too noisy to be used directly. Here, a method is proposed to make use of the geometric properties previously mentioned in Section IV-A to generate a better segmentation result.

As previously defined, let  $L_1$  and  $L_2$  be two boundary lines with parameter pairs  $(A_1, B_1)$  and  $(A_2, B_2)$ , respectively. If  $(A_1, B_1)$  and  $(A_2, B_2)$  are treated as two vectors, then if the two lines are parallel to each other, (23) can be derived. Similarly, if the two lines are perpendicular to each other, (25) can be derived. Recalling that the boundary lines are either mutually parallel or perpendicular, we define the *direction*  $\theta$  of the boundary lines in such a way that the parameters  $(A, B)$  of each boundary line are either “parallel” or “perpendicular” to  $(\cos \theta, \sin \theta)$ , i.e., are equal, up to a scalar, to either  $(\cos \theta, \sin \theta)$  or  $(\cos(90^\circ + \theta), \sin(90^\circ + \theta))$ .

To find the correct value of the direction  $\theta$ , we first generate the Hough space and apply the filter, as stated in Section III. From this Hough space, about 0.3% of the Hough cells with the largest values and satisfying (26) are chosen and put into set  $S$ . In addition, we create 90 bins in the range  $0^\circ \sim 90^\circ$  for different values of  $\theta$ . For each cell with parameters  $(A_i, B_i)$  in set  $S$ , we find the *closest* bin  $b_i$  with value  $\theta_i$  such that  $(A_i, B_i)$  is either parallel or perpendicular to  $(\cos \theta_i, \sin \theta_i)$  and increment bin  $b_i$  by the value of the Hough cell with  $(A_i, B_i)$ . After applying a Gaussian filter with standard deviation  $\sigma = 1$ , the bin with the largest accumulation value is chosen to be the direction of the boundary lines  $\theta$ .

An experimental result of the given process is described in the following. An image captured by a hyperbolic catadioptric camera for use as the input is shown in Fig. 8(a). By following

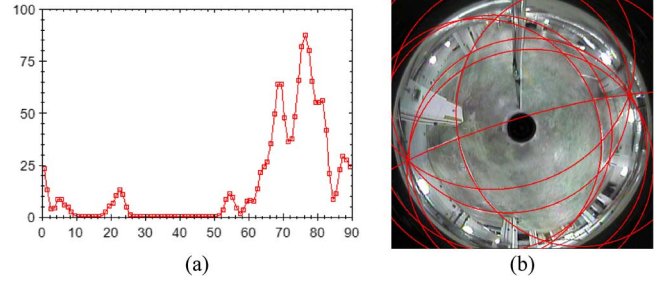


Fig. 12. Example of parking space detection results with input image shown in Fig. 8(a). (a) Cell accumulation result of the direction  $\theta$  of boundary lines. (b) Results of boundary lines found by the proposed method, showing improvement over line detection results shown in Fig. 8(e) and (f).

the line image detection method proposed in Section III, the 2-D Hough space, after applying the filter, becomes as that shown in Fig. 8(d). The results of the accumulated bin values of  $\theta$  are shown in Fig. 12(a), where the largest bin is with value  $\theta = 76.5^\circ$ , which is taken finally as the direction of the boundary lines.

After the direction  $\theta$  of the boundary lines is derived, the boundary lines that do not satisfy the direction  $\theta$  are removed as follows. For each Hough cell in set  $S$  with parameters  $(A_i, B_i)$ , if there exists a real number  $\theta_{\text{offset}}$  within the range  $-10^\circ \leq \theta_{\text{offset}} \leq 10^\circ$  such that the two vectors  $(A_i, B_i)$  and  $(\cos(\theta + \theta_{\text{offset}}), \sin(\theta + \theta_{\text{offset}}))$  are parallel or perpendicular to each other, then the cell is considered to satisfy the direction  $\theta$  and so kept in set  $S$ ; otherwise, it is removed. Afterward, the connected components  $S_i$  in set  $S$  are found according to the criterion that two cells with parameter values  $(A_1, B_1)$  and  $(A_2, B_2)$  are connected if

$$|A_1 - A_2| \leq 0.05 \quad \text{and} \quad |B_1 - B_2| \leq 0.05. \quad (27)$$

Finally, the center of each connected component  $S_i$  is calculated, and their parameters  $(A_i, B_i)$  are then used to specify the boundary lines. By this process, the result corresponding to Fig. 12(a) is shown in Fig. 12(b), in which one can see that all the boundary lines of the parking spaces have been detected.

### C. Parking Space Segmentation and Vacancy Detection

Two more steps are proposed here to segment out the regions of the parking spaces in the acquired wide-angle image. Then, a vacancy detection scheme based on background subtraction is adopted. These three steps are designed to be easily performed by a user with no technical background, so that the vision-based parking lot system can be set up and utilized conveniently as claimed.

The first step is to adjust the boundary lines manually by clicking on the image to add and remove the boundary line, i.e., one line at a time. The boundary lines are first divided into two “mutually” perpendicular groups, as shown in Fig. 13(a), and distinguished by colors. The user may select either group and then simply click on an existing boundary line to remove it or click on any other position to add a new boundary line going through the mouse position. It is important to note that, with the direction  $\theta$  previously found, the boundary line can now be uniquely determined with only *one* point marked by the



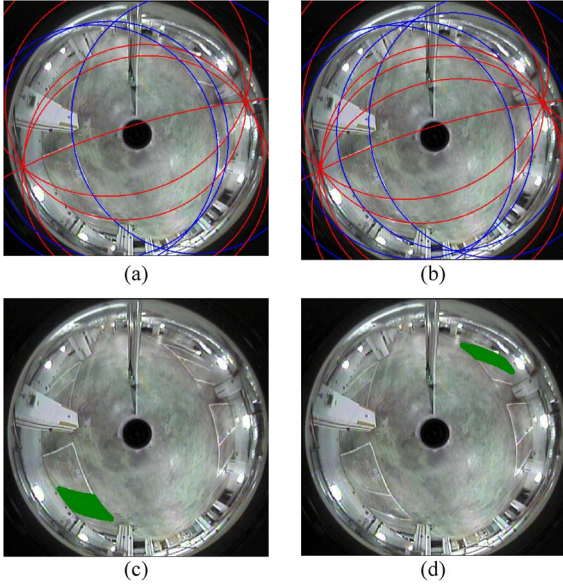


Fig. 13. Parking space segmentation. (a) Boundary lines detected by the proposed method are divided into two groups marked by red and blue, respectively. (b) Boundary lines after being manually adjusted. (c) and (d) Regions of two green-marked parking spaces automatically found by the proposed scheme.

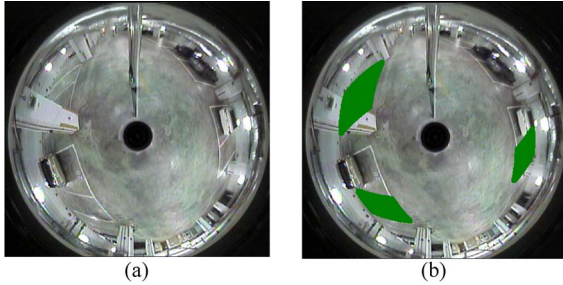


Fig. 14. Parking space vacancy detection. (a) Input image. (b) Vacant parking spaces marked as green regions.

user; without  $\theta$ , there will be an infinite number of lines going through the point marked by the user. For example, from the detection results shown in Fig. 13(a), the result after two lines are added and one line is removed manually is shown in Fig. 13(b).

The second step is to find parking space regions. In this step, the user may simply click on any position of a parking space in the image, and a region-growing algorithm is immediately performed to find the region of the parking space using the boundary lines. Two results are shown in Fig. 13(c) and (d).

The final step is to find vacant parking spaces. To implement this, the user has to capture beforehand an image of each parking space when it is vacant and consider the image as the *background* of the parking space. Subsequently, to determine the vacancy of the parking space, each pixel in the acquired wide-angle image is considered a *foreground pixel* if the absolute difference between the pixel's value and that of the background pixel is larger than a threshold value in at least one of the R, G, and B channels. Then, if the number of foreground pixels is larger than 20% of that of the pixels in the parking space region, the parking space is considered *occupied*; otherwise, it is considered *vacant*. An experimental result is shown in Fig. 14. A real indoor vision-based parking lot system implemented in this study is described later in Section V-D.

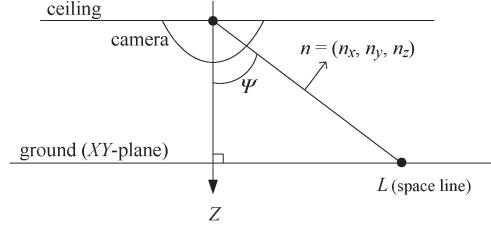


Fig. 15. Definition of the elevation angle  $\Psi$  of a space line  $L$  on the ground.

## V. EXPERIMENTAL RESULTS AND DISCUSSIONS

Experiments have been conducted to demonstrate the correctness, feasibility, and robustness of the proposed method. Some results and discussions are presented here.

### A. Experimental Results of Proposed Calibration Method

The proposed camera calibration method described in Section II uses the image of only a space line, without knowing its position or direction, to derive the parameters of the proposed general model for fisheye-lens, parabolic catadioptric, and hyperbolic catadioptric cameras. By assuming that the image center has been already calibrated by the use of the circular boundary of the omni-image as done in [11] and [15], the effective focal length value  $f_e$  is the only remaining parameter to be calibrated. For this, a series of experiments was conducted using space lines lying on the ground at different positions, with the camera affixed on the ceiling and looking downward. Specifically, as shown in Fig. 15, one of the parameters of space line  $L$  is its elevation angle  $\psi$ ; hence, several simulated space lines with different elevation angles  $\psi$  in the range of  $0^\circ \sim 90^\circ$  were generated for use in the experiments. The parameter  $f_e$  of these simulated lines was set to be  $f_e^* = 500$ , and their line images generated to be of the size  $800 \times 800$  identically. In addition, the coordinates of the pixels on each line image were disturbed by Gaussian noise with zero mean and varying standard deviations  $\sigma$ . Then, the proposed calibration method was applied to derive parameter  $f_e$  using these simulated line images. This process was repeated 100 times, and the mean and variances of the derived values of  $f_e$  are plotted in Fig. 16(a)–(c) for different noise levels  $\sigma = 1.0, 3.0$ , and  $5.0$  pixels, respectively. Recalling that the actual value of parameter  $f_e$  is  $f_e^* = 500$ , one can see in the figure that parameter  $f_e$  can be robustly calibrated under a large noise level when the elevation angle  $\Psi > 15^\circ$ . Note also that a space line with elevation angle  $\Psi$  near  $0^\circ$  will appear to be a short and nearly *straight* line segment going through the image center, so that parameter  $f_e$  cannot be reliably calibrated using it, as expected according to the theory of the proposed calibration method described in Section II-C.

### B. Calibration of Camera Parameters

The main objective of the proposed methods is the ease for a nontechnical user to set up the system. Here, achievement of this goal is demonstrated, i.e., the user only has to select a single space line with its position and direction unknown. Images such as those shown in Fig. 17 were captured by a hyperbolic catadioptric camera with MapCam MRC530N manufactured by EeRise, Inc. Four experimental results are shown in Fig. 17(a)–(d). In each experiment, the user only has

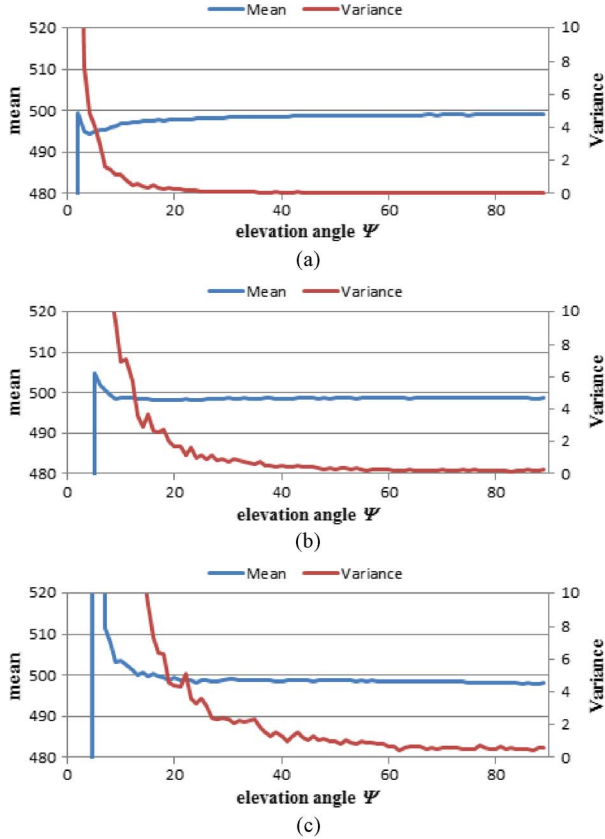


Fig. 16. Computed mean and variances of calibrated values  $f_e$  for simulated space lines with different elevation angles  $\Psi$ , where the standard deviation of Gaussian noise is 1.0 pixel for panel (a), 3.0 pixels for panel (b), and 5.0 pixels for panel (c).

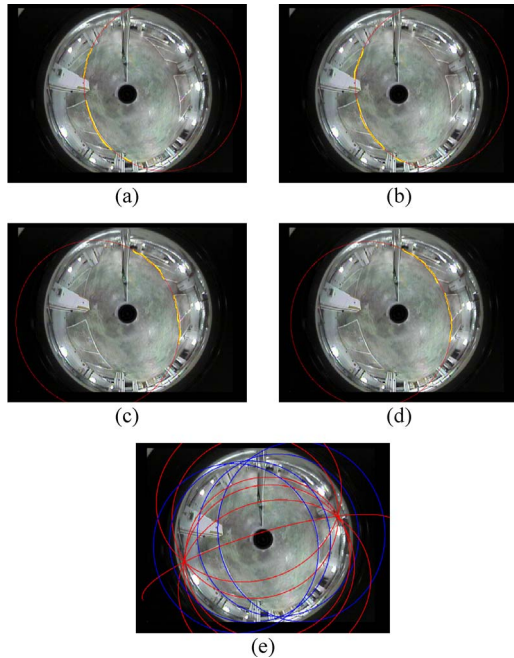


Fig. 17. Applying the proposed calibration method to derive the parameter  $f_e$  of a hyperbolic catadioptric camera. (a)–(d) Four results with calibrated values  $f_e = 265.34, 258.14, 276.84,$  and  $272.36,$  respectively. (e) Boundary lines found by the proposed detection method with  $f_e = 265.34.$

to arbitrarily choose an obvious space line in the captured image and mark *roughly* the pixels on the line as yellow points by mouse dragging. Then, the proposed calibration method was

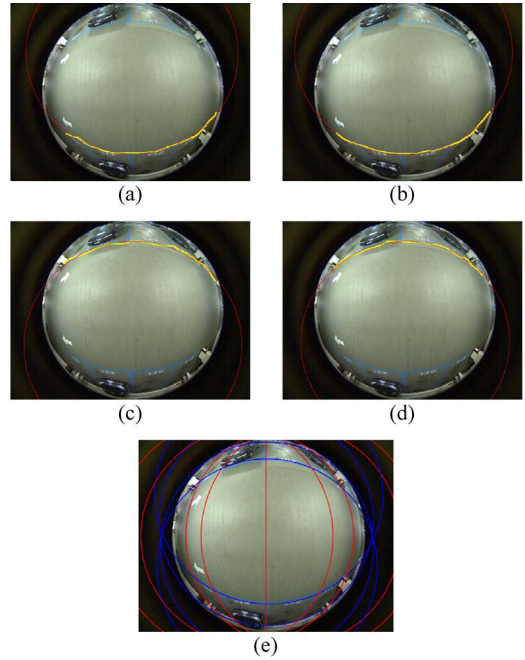


Fig. 18. Use of the proposed calibration method to calibrate a fisheye-lens camera. (a)–(d) Four results with calibrated values  $f_e = 323.74, 331.12, 339.74,$  and  $328.57,$  respectively. (e) Boundary lines found by the proposed detection method with  $f_e = 331.12.$

applied to derive the value of the effective focal length  $f_e$  using the yellow points. The values of  $f_e$  so derived for the cases shown in Fig. 17(a)–(d) are 265.34, 258.14, 276.84, and 272.36, respectively, and the best-fitting ellipses to the marked space lines are shown in red. With parameter  $f_e = 265.34,$  the boundary lines of the parking spaces were finally found and drawn in Fig. 17(e) using the proposed detection method. In addition, conducted were similar experiments with a fisheye-lens camera manufactured by Hunt Electronic, Inc. with model No. HLC-1NAD, and the results are shown in Fig. 18. The results of both experimental series show the feasibility and robustness of the proposed calibration method by the fitness of the drawn red ellipses to the marked yellow points, the closeness of the calibrated values  $f_e$  to the real value, and the nearly perfect overlapping of the drawn boundary lines on the real ones.

### C. Experimental Results of Boundary Line Detection

In the boundary line detection method proposed in Section IV-B, the direction of the boundary lines, denoted by  $\theta,$  is computed first, and the boundary lines are derived accordingly. Some experiments were conducted to test the capability of this scheme of finding  $\theta$  as described next. First, an image of an indoor parking lot was captured by a hyperbolic catadioptric camera, as shown in Fig. 19(a). Then, this image was rotated through angles  $30^\circ, 60^\circ,$  and  $90^\circ,$  as shown in Fig. 19(b)–(d), respectively. The boundary line directions  $\theta$  in these four images were first derived. Then, the boundary lines were automatically detected and superimposed on the original images, as shown in Fig. 19(b)–(d), respectively. In the figure, one can see that the boundary lines fit well to the real ones appearing in the captured images, which means that the

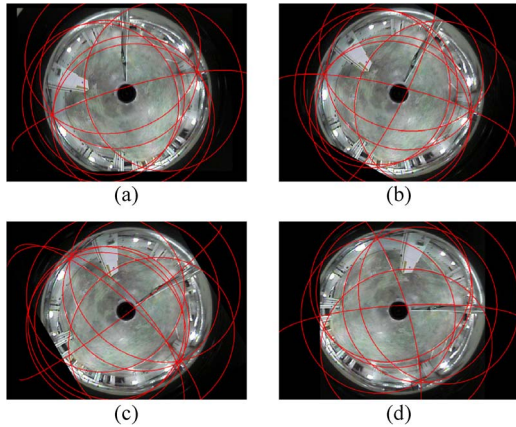


Fig. 19. Automatic detection results of boundary lines. Rotation angles of these images are (a)  $0^\circ$ , (b)  $30^\circ$ , (c)  $60^\circ$ , and (d)  $90^\circ$ , respectively. The detected boundary lines are drawn and superimposed on the images.

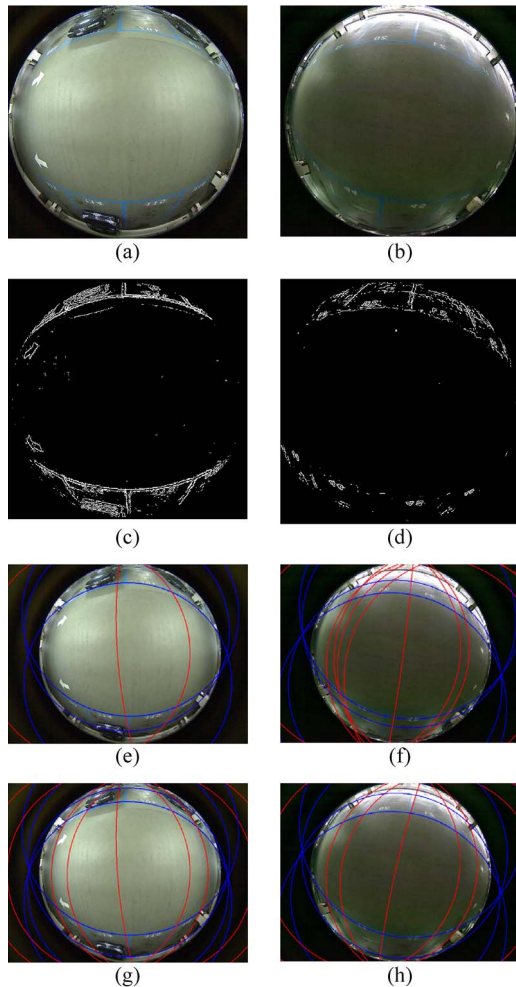


Fig. 20. Results of two experiments under bad conditions, with those of the first experiment shown in panels (a), (c), (e), and (g) and those of the second experiment shown in panels (b), (d), (f), and (h). (a) and (b) Input images. (c) and (d) Edge detection results. (e) and (f) Automatic detection results of boundary lines. (g) and (h) Adjusted boundary lines.

directions of the boundary lines were correctly found by the proposed method.

To demonstrate that the direction  $\theta$  of the boundary lines can be robustly found even under bad conditions, two more exper-

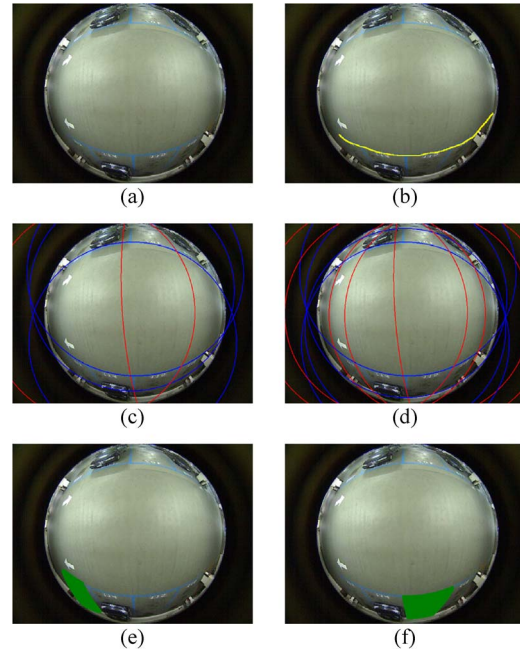


Fig. 21. Steps in setting up the proposed system. (a) Image captured at midnight. (b) Manually marked pixels (in yellow) of a space line for the calibration task. (c) Results of automatic detection of boundary lines. (d) Result of boundary line adjustment. (e) and (f) Regions of two found parking spaces (marked in green).

iments were conducted with input images shown in Fig. 20(a) and (b), in which noise was present, cars were parked in the spaces, and the lighting situations were poor. The results of the first experiment are shown in Fig. 20(c), (e), and (g), and those of the second experiment are shown in Fig. 20(d), (f), and (h). Specifically, the edge detection results are shown in Fig. 20(c) and (d), respectively, in which one can see that the boundary lines were poorly detected; moreover, some of the boundary lines are missing, as shown in Fig. 20(e) and (f). However, the directions of the detected and drawn boundary lines are correct, which means that the direction  $\theta$  can be precisely derived even under bad conditions. With this correct value of  $\theta$ , the boundary lines can be easily adjusted by the proposed method, as demonstrated by the results shown in Fig. 20(g) and (h).

#### D. Setup of a Real Vision-Based Indoor Parking Lot System

A real indoor vision-based parking lot system was established, and its use for this study is introduced here. A fisheye-lens camera manufactured by Hunt Electronic, Inc. with model No. HLC-1NAD was affixed on the ceiling of the parking lot. An image, as shown in Fig. 21(a), was first acquired at midnight and then used both for calibration and for detecting the boundary lines of the parking spaces. Specifically, a space line in the image was arbitrarily chosen according to a rule of thumb that the line should be *long enough and not straight*. The line pixels were traced and marked in yellow, as shown in Fig. 21(b), by which parameter  $f_e$  was calibrated to be 331.1.

Afterward, the direction  $\theta$  of the boundary lines was derived, and the boundary lines were automatically detected, with the results drawn in Fig. 21(c). Subsequently, boundary lines were added or removed by *single clicks* on the image, as previously

described. A result of such adjustments is shown in Fig. 21(d). Finally, the parking spaces were segmented out by clicking on any point within the region of each space and applying a region-growing algorithm. The regions of two parking spaces so found and drawn in green are shown in Fig. 21(e) and (f), respectively. After using the captured image shown in Fig. 21(a) as background, the system setup process was completed, and parking space vacancy detection was started. In one of the experiments we conducted, this system was maintained to run for 24 hours to detect vacant parking spaces every minute. The resulting detection accuracy was 99.67%, which is good enough for real applications.

## VI. CONCLUSION AND FUTURE WORKS

A convenient indoor vision-based parking lot system has been proposed in this study, which is easy to set up by a typical user with no technical background and can detect vacant parking spaces automatically. The system uses wide-angle cameras, such as fisheye-lens or catadioptric cameras, and analyzes parking space boundary lines based on a new camera model proposed in this study. This model approximates optimally one of the parameters used in a conventional model (the distance  $l$  from the effective viewpoint  $O$  to the pinhole point  $O_c$ ) while reserving some important characteristics of line images, including the shape of the curve and the locations of the vanishing points. A new line-based calibration method has been also proposed to calibrate the camera model using only one space line without knowing its location and direction, so that the calibration process can be easily done by a user without any technical background. A new Hough transform has been proposed as well to detect space lines, in which a skillful cell accumulation scheme is used to generate equal-width curves, yielding a more robust and accurate detection result of parking space boundary lines. A convenient adjustment method has been also developed such that a user can add or remove boundary lines by simple clicks on input images.

Currently, the background images should be learned for various lighting conditions to conduct parking space vacancy detection based on background subtraction. More intelligent methods may be developed to remove this weakness in the future study.

## REFERENCES

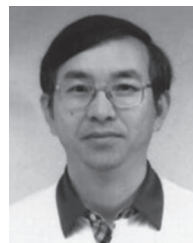
- [1] K. Blumer, H. R. Halaseh, M. U. Arsan, H. Dong, and N. Mavridis, "Cost-effective single-camera multi-car parking monitoring and vacancy detection towards real-world parking statistics and real-time reporting," in *Proc. Neural Inf. Process.*, 2012, pp. 506–515.
- [2] C. C. Huang and S. J. Wang, "A hierarchical Bayesian generation framework for vacant parking space detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1770–1785, Dec. 2010.
- [3] H. Ichihashi, T. Katada, M. Fujiyoshi, A. Notsu, and K. Honda, "Improvement in the performance of camera based vehicle detector for parking lot," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, 2010, pp. 1950–1956.
- [4] W. Lixia and J. Dalin, "A method of parking space detection based on image segmentation and LBP," in *Proc. 4th IEEE Int. Conf. Multimedia Inf. Netw.*, 2012, pp. 229–232.
- [5] X. Deng, F. Wu, Y. Wu, L. Chang, W. Liu, and H. Wang, "Calibration of central catadioptric camera with one-dimensional object undertaking general motions," in *Proc. 18th IEEE Int. Conf. Image Process.*, 2011, pp. 637–640.

- [6] S. Ramalingam, P. Sturm, and S. K. Lodha, "Generic self-calibration of central cameras," *Comput. Vis. Image Understanding*, vol. 114, no. 2, pp. 210–219, Feb. 2010.
- [7] S. Gasparini, P. Sturm, and J. P. Barreto, "Plane-based calibration of central catadioptric cameras," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, 2009, pp. 1195–1202.
- [8] C. Geyer and K. Daniilidis, "A unifying theory for central panoramic systems and practical implications," in *Proc. 6th Eur. Conf. Comput. Vis.*, 2000, pp. 445–462.
- [9] X. Ying and Z. Hu, "Catadioptric camera calibration using geometric invariants," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 10, pp. 1260–1271, Oct. 2004.
- [10] X. Ying and Z. Hu, "Can we consider central catadioptric cameras and Fisheye cameras within a unified imaging model?" in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 442–455.
- [11] X. M. Deng, F. C. Wu, and Y. H. Wu, "An easy calibration method for central catadioptric cameras," *Acta Autom. Sinica*, vol. 33, no. 8, pp. 801–808, Aug. 2007.
- [12] Y. Bastanlar, L. Puig, P. Sturm, J. J. Guerrero, and J. Barreto, "DLT-like calibration of central catadioptric cameras," in *Proc. 8th Workshop Omnidirectional Vis., Camera Netw. Non-Classical Cameras*, Oct. 2008.
- [13] C. Geyer and K. Daniilidis, "Catadioptric projective geometry," *Int. J. Comput. Vis.*, vol. 45, no. 3, pp. 223–243, Dec. 2001.
- [14] J. P. Barreto and H. Araujo, "Geometric properties of central catadioptric line images," in *Proc. 7th Eur. Conf. Comput. Vis.*, 2002, pp. 237–251.
- [15] J. Fabrizio, J. P. Tarel, and R. Benosman, "Calibration of panoramic catadioptric sensors made easier," in *Proc. 3rd IEEE Workshop Omnidirectional Vis.*, 2002, pp. 45–52.
- [16] T. Apostol, *Calculus*, 2nd ed. Hoboken, NJ, USA: Wiley, Jun. 1967.



**Shen-En Shih** received the B.S. degree in computer science from National Chiao Tung University, Hsinchu, Taiwan, in 2009, where he is currently working toward the Ph.D. degree with the College of Computer Science.

Since August 2009, he has been a Research Assistant with the Computer Vision Laboratory, Department of Computer Science, National Chiao Tung University. His current research interests include computer vision, image processing, human-machine interfacing, and stereo vision.



**Wen-Hsiang Tsai** (S'78–M'79–SM'91) received the B.S. degree from National Taiwan University, Taipei, Taiwan, in 1973; the M.S. degree from Brown University, Providence, RI, USA, in 1977; and the Ph.D. degree from Purdue University, West Lafayette, IN, USA, in 1979, all in electrical engineering.

Since 1979, he has been with National Chiao Tung University (NCTU), Hsinchu, Taiwan, where he is currently a Chair Professor of computer science. At NCTU, he has served as the Head of the Department of Computer Science, the Dean of General Affairs, the Dean of Academic Affairs, and a Vice President. From 2004 to 2007, he was the President of Asia University, Taichung, Taiwan. He has published 157 journal papers and 245 conference papers. His current research interests include computer vision, information security, video surveillance, and autonomous vehicle applications.

Dr. Tsai has served as the Editor or Editor-in-Chief of several international journals, including *Pattern Recognition*, the *International Journal of Pattern Recognition and Artificial Intelligence*, and the *Journal of Information Science and Engineering*. From 1999 to 2000, he was the Chair of the Chinese Image Processing and Pattern Recognition Society of Taiwan, and from 2004 to 2008, he was the Chair of the Computer Society of the IEEE Taipei Section in Taiwan. He has received many awards, including the Annual Paper Award from the Pattern Recognition Society of the United States; the Academic Award of the Ministry of Education, Taiwan; the Outstanding Research Award of the National Science Council, Taiwan; the ISI Citation Classic Award from Thomson Scientific, and more than 40 other academic paper awards from various academic societies. He is a Life Member of the Chinese Pattern Recognition and Image Processing Society, Taiwan.