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 comes mainly from the use of a new camera model which can be calibrated using only one space line without knowing its position and direction, as well as from the allowance of convenient changes of the detected parking space boundaries. After camera calibration based on the new camera model is completed, parking-space boundary lines are extracted automatically from input wide-angle images by a modified Hough transform with a new cell accumulation scheme, which can generates more accurate equal-width curves using the geometric relations of line positions and directions. Also, 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 by 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—**wide-angle cameras, space line analysis, Hough Transform, parking lot analysis, parking lot system.**

I. INTRODUCTION

Nowadays, more and more cameras are equipped in our environment for various applications like video surveillance, event recording, evidence collection, etc. In this study, we investigate the use of cameras for parking lot management. Most cameras used in parking lot systems are perspective ones; wide-angle cameras, such as fisheye-lens or catadioptric ones, are not commonly adopted yet. No matter what types of camera systems are 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 technical knowledge. In addition, finding vacant parking

spaces is often a trouble for people, especially in large indoor parking lots — one has to spend much time with concentration to drive through the entire parking area to search visually 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 two 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.

About the existing vision-based parking lot systems which 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 3D 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* in order to construct an easy-to-set-up system.

As to camera calibration which also influences the convenience of the system usage, the goal of this study is to provide a smart calibration process for a wide-angle camera, which can be performed easily by a normal 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 to calibrate a central (single-viewpoint) catadioptric camera by the use of a 1D 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.* [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 uses 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.
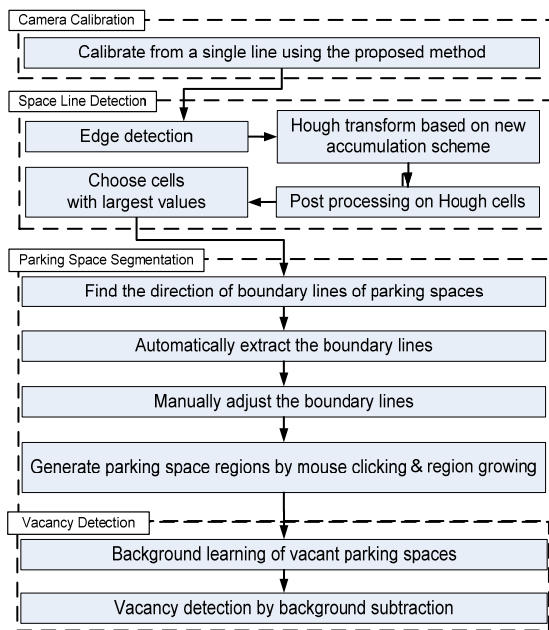

Fig. 1. The proposed parking lot system.

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

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

II. PROPOSED CAMERA MODEL AND CALIBRATION METHOD

To design an easy-to-setup vision-based parking lot system, the camera calibration process must be easy to carry out by normal users with no technical knowledge. In this study, 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 Sec. II.A. Then, the proposed camera model is described in Sec. II.B, and the proposed calibration method is presented in Sec. 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 illustrated in Fig. 2. Specifically, point P is firstly projected onto a point P_s on the unit sphere like being viewed from the *effective viewpoint O* through the sphere. Then, P_s is projected perspectively into the pixel p on the image plane Π like being viewed from the "pinhole" point O_c (usually the lens center). The two involved parameters l and f_e as illustrated in Fig. 2 are defined in the same way as suggested by Ying and Hu [9].
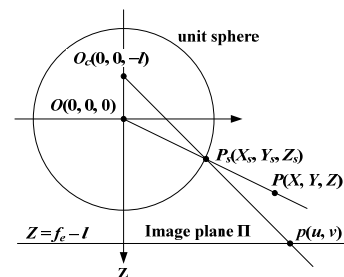

Fig. 2. An illustration of a two-step spherical mapping.

The unifying model described above has been proved 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 study, we simplify it to model fisheye-lens and hyperbolic catadioptric cameras, as described next.

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 the 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 calibrated easily by the use of a single space line as described in Sec. 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 Eq. (2) derived later in this section) with different values of the parameter l while the two vanishing points are fixed (marked in yellow in the figure). This phenomenon leads to two conclusions: 1) the parameter l *cannot* be well calibrated from line images; and 2) reversely, the value of the parameter l *did not* affect the space line detection process. The first conclusion is consistent with some previous studies [9][11]. Specifically, the parameter l was fixed in the simulation experiments described in Geyer and Daniilidis [9], so the parameter l was not derived in the calibration process; and as seen in Deng et al. [11], the parameter l is assumed to be known before the calibration. The second conclusion makes it possible to find an optimal value of the parameter l to approximate that of any kind of wide-angle camera, without affecting the space line detection process.
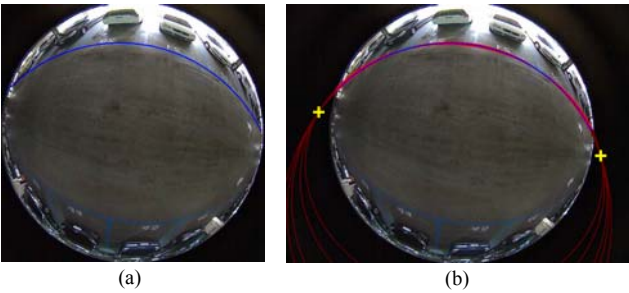

Fig. 3. Fitting a space line using different values of l. (a) Space line to be fit (marked as blue). (b) Fitting results using l = 2.0, 1.5, 1.0, 0.8, and 0.5 with larger ellipses corresponding to smaller values of l, and the two yellow crosses indicate the fixed vanishing points.

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 study, we define the

interesting range of the 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][11] are 0.8, 0.966 and 1.0, that derived in [12] is 0.9663, and that derived in [7] is 1.07.

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 1000×1000 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 the following steps 2.1) through 2.3):
   2.1) find the best-fitting curve I_{ij}^* to each line image I_i, where I_{ij}^* is computed by Eq. (2) with $l = l_j^*$ described in Sec. II.C and a Levenberg–Marquardt process;
   2.2) compute the distance $dist_{ij}$ between I_i and I_{ij}^* as

$$dist_{ij} = \frac{1}{n}\sum_{k=1}^{n}\left\| p_k - p_k^* \right\| \qquad (1)$$

   where n denotes the number of pixels on the line image I_i, p_k a pixel on I_i, p_k^* the nearest pixel on I_{ij}^* to p_k, and $\| p_k - p_k^* \|$ the distance between p_k and p_k^*;
   2.3) 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 $dist_j$ meaning 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 also the *average fitting error* and denoted as $dist$ subsequently.

An experimental result of the above process is shown in Fig. 4, where Fig. 4(a) shows a line image I_i marked in blue and a best-fitting curve I_{ij}^* marked in red; 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 Eq. (2) with $l = l^*$ resulting in an average fitting error $dist \approx 1.1941$ pixels.
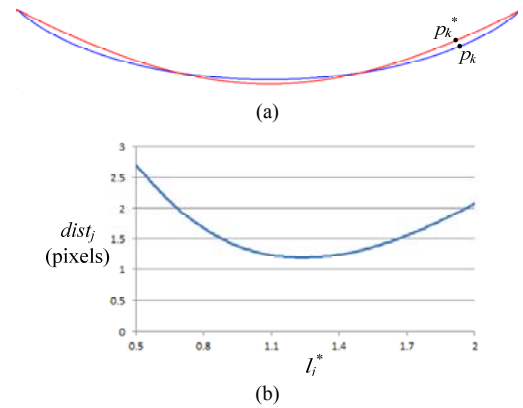

(a)

(b)

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.

As a result of utilizing the aforementioned second conclusion, we propose a camera model which is identical to the unifying one proposed in [8] but with its parameter l fixed

to be the optimal value 1.24 derived previously. 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 the conventional models [8][13], by which the camera cannot be calibrated reliably from line images as proved previously in Sec. II.B. It also facilitates a non-technical user to conduct the calibration process without difficulty as mentioned previously.

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) marked manually. Then, the best-fitting ellipse to this line image is computed, from which the unknown camera parameters and the space line are estimated roughly. 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 its line image, and E_L the best-fitting ellipse to I_L. As derived in [9][13], I_L can be expressed as

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

where (u, v) are coordinates of pixels on I_L, (n_x, n_y, n_z) is the normal vector of the plane Π 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 Sec. II.A. Also, let the ellipse E_L be described by

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

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

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

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

Now, the problem is to estimate *roughly* the values of the parameters (l, f_e, G, H) such that Eqs. (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 the parameters (f_e, G, H) which satisfy

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

where "~" means "equals up to a scale." Let λ 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 above equation for use as rough estimates of the parameters is:

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

A Levenberg–Marquardt process is conducted finally 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

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

with respect to all the pixels (u, v) on the line image I_L. After this optimization process is done, the 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 for Figs. 5(a) and 5(b) are 319.90 and 319.57, respectively, and those for Figs. 5(c) and 5(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.
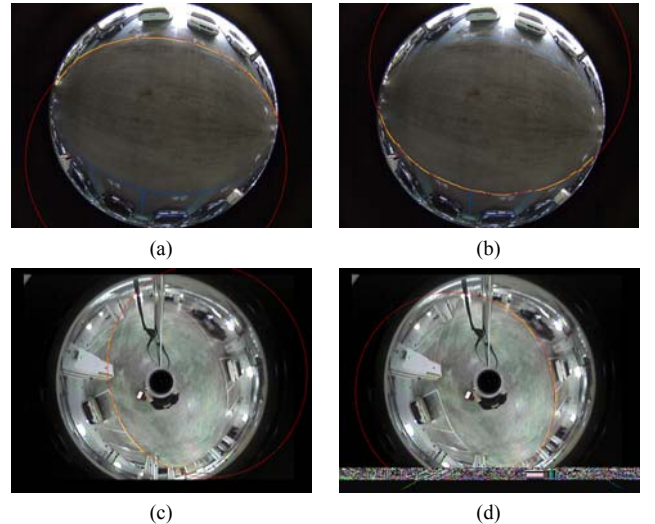

(a)　　　　　　　　　　(b)

(c)　　　　　　　　　　(d)

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

III. PROPOSED SPACE LINE DETECTION METHOD

One of the important features in man-made environments is 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 images* so far. In this section, the proposed method for detecting the conic-section curve is described, which, differently 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 Sec. II.B) for more accurate detection of parking-space boundary lines.
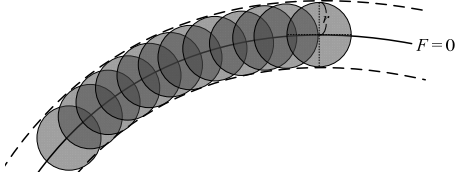

Fig. 6. An illustration of an equal-width curve along a curve $F = 0$ with width r, defined to be the regions of all the gray circles (not all drawn), or equivalently, the thick area bounded by the two dashed curves.

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 the plane Π formed by the corresponding space line and the origin of the camera. The normal vector can be expressed as

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

where

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

and $0 \leq A, B \leq 1$. The 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 the parameter (A, B), then the cell value at (A, B) is incremented by one.

In more detail, by referring to Fig. 7, it can be figured out that a pixel (u, v) is in the equal-width region with width r if and only if the curve $F = 0$ intersects the circle C centered at (u, v) with radius r. Furthermore, according to the intermediate value theorem [16], the curve $F = 0$ intersects the 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 *positive*, we try to find the 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 the 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 the gradient ∇F specifies the direction of *increasing* the function value $F(u, v)$, the vector $(\Delta u, \Delta v)$ should be in the *negated* direction of the gradient ∇F. So, under the constraint that the length of $(\Delta u, \Delta v)$ is the radius r, the 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 a similar reasoning that the vector $(\Delta u, \Delta v)$ should be with the *same* direction of the gradient ∇F so that it can be expressed as $[\nabla F(u, v)/\|\nabla F(u, v)\|] \times r$. As a summary, we have

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

Accordingly, the function value $F(u', v')$ can now be derived by the 2nd-order Taylor expansion to be:

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

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

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

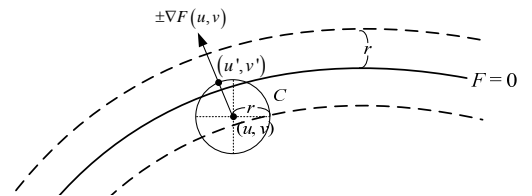whose details may be derived from (2) and are omitted here.

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

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 the parameters (A, B), and increment the value of the Hough cell with (A, B) by one.
3) Find out the parameters (A, B) of the desired line images (conic-section curves) from the resulting Hough space by the following steps.

3.1) Enhance the local maximums of the Hough space by applying the following sharpening filter:

$$\frac{1}{25}\begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 24 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \end{bmatrix}. \tag{14}$$

3.2) Pick out all the Hough cells with values larger than a pre-selected threshold δ 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.

By the way, 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 the parameter l, as mentioned in Sec. II.B. The former is approximated to be 1 pixel, and the latter 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 above 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 the filter (14) 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 Figs. 14(e) and (f), where the numbers of drawn lines are 30 and 50, respectively.
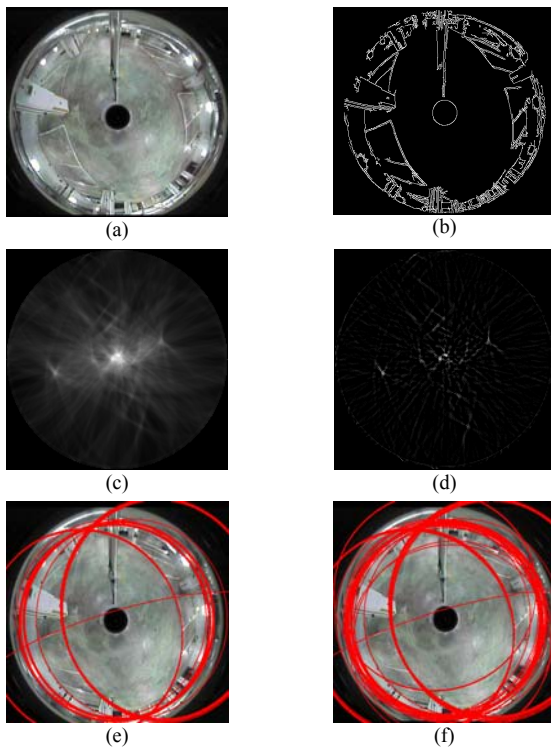

Fig. 8. Results of the proposed space line detection method. (a) The input image. (b) Canny edge result. (c) The Hough space. (d) The Hough space after applied the filter (14). (e)(f) Results of 30 and 50 lines, resepctively.

B. Effectiveness of Proposed Cell Accumulation Scheme

As stated previously, 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 the parameters A and B and the 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 a threshold T and consider pixels with coordinates (u, v) satisfying the following inequality as being on the conic section described by $F(u, v) = 0$:

$$\left| F(u,v) \right| \le T . \tag{15}$$

Two results yielded by this method are shown in Figs. 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 which 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 Sec. III.A yields equal-width curves as shown in Figs. 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.
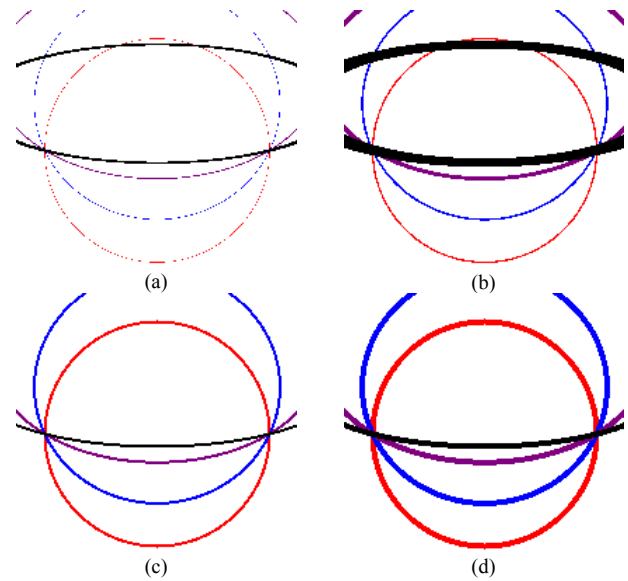

Fig. 9. Comparison of conventional and proposed accumulation method. (a)(b) Non-equal-width results yielded by conventional method with thresholds $T = 40$ and 150, respectively. (c)(d) Equal-width results yielded by proposed method with width $r = 1$ and 2, respectively.

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 Figs. 10(a) and (b), respectively. The 10 and 30 detected space line images yielded by the conventional scheme are drawn in Figs. 10(c) and (d), respectively, and those yielded by the proposed method are shown in Figs. 10(e) and (f),

respectively. As can be found from the figures, 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 can be seen from Fig. 9(b), the conventional scheme generates a thicker curve when the real curve appears nearer to the image center, and vice versa. This means that each Hough cell corresponding to a curve nearer to 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 the image center are *more* likely to be chosen first, yielding the results as shown in Figs. 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 as shown in Figs. 10(e) and (f).
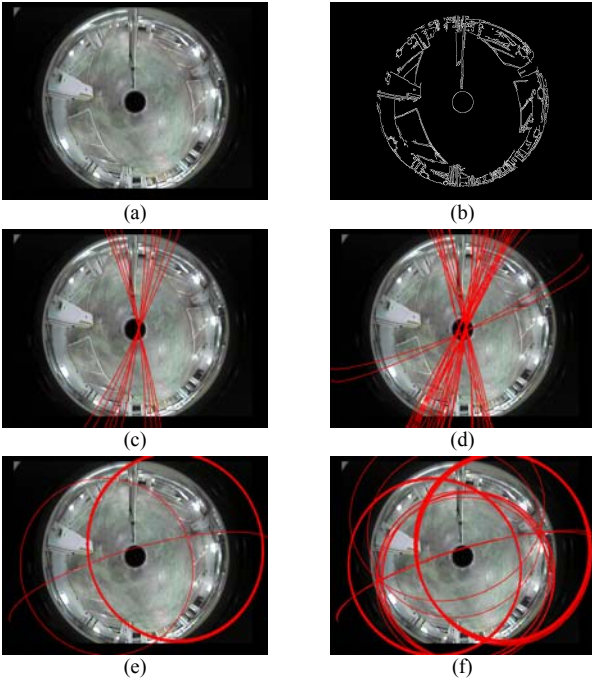

Fig. 10. Comparisons of space line detection results yielded by conventional and proposed method. (a) Input image. (b) Edge detection result. (c)(d) 10 and 30 detected curves using conventional method. (e)(f) 10 and 30 detected curves using proposed method.

IV. DETECTION OF PARKING SPACE VACANCY

Each parking space is usually marked by some colored boundary lines. In this section, some properties of such boundary lines are first derived in Sec. IV.A. Then, a method to detect such boundary lines is proposed in Sec. IV.B. Finally, in Sec. 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 Sec. I, the wide-angle cameras used in the proposed parking lot system are affixed on the ceiling to look right downward; so, 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, so we have

$$d_{1z} = 0. \tag{16}$$

Let (A_1, B_1) be the Hough parameters of line L_1 found by the space line detection method proposed in Sec. 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

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

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

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

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

$$A_1 d_{1x} + B_1 d_{1y} = 0, \tag{19}$$

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

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

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 derived similarly to be

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

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

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

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

$$\left(A_1, B_1\right) = \lambda \left(A_2, B_2\right), \tag{23}$$

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

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

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

$$\left(A_1, B_1\right) \cdot \left(A_2, B_2\right) = 0. \tag{25}$$

Moreover, in order to ignore boundary lines which are too far away, we check the *elevation angle* ψ of each boundary line, which is defined in a way as illustrated in Fig. 11 so that if the larger the angle Ψ is, the farther the boundary line is from the camera. According to our experimental experience, if the

camera is mounted at 2.5 meters high, the farthest usable boundary line is at a distance about 10.0 meters from the camera. As a result, the maximum value of Ψ is $\tan^{-1}(10.0/2.5)$ = 75.96°, which is equivalent to $n_{1z} \leq \sin(\Psi) = 0.970$. Using this result with (17), we can derive another constraint on the parameters (A, B) as $n_{1z} = \sqrt{1 - A_1^2 - B_1^2} \leq 0.970$, or equivalently, as
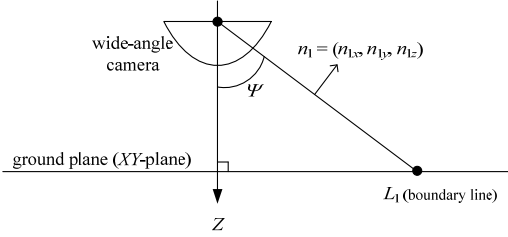
$$A_1^2 + B_1^2 \geq 0.059 . \tag{26}$$

Fig. 11. Definition of the elvation angle Ψ of a boundary line L_1.

B. Automatic Detection of Boundary Lines

As can be seen from Figs. 8(e) and (f), the results of the line image detection method proposed in Sec. III are too noisy to be used directly. In this section, a method is proposed to make use of the geometric properties mentioned previously in Sec. IV.A to generate a better segmentation result.

As defined above, 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 θ 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 θ, we first generate the Hough space and apply the filter as stated in Sec. III. From this Hough space, about 0.3% of the Hough cells with the largest values and satisfying (26) are chosen and put into a set S. Also, we create 90 bins in the range 0°~90° for different values of θ. For each cell with parameters (A_i, B_i) in the set S, we find the *closest* bin b_i with value θ_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 θ.

An experimental result of the above 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 the line image detection method proposed in Sec. III, the 2D Hough space, after applying the filter, becomes as that shown in Fig. 8(d). The results of the accumulated bin values of θ are

shown in Fig. 12(a), where the largest bin is with value $\theta =$ 76.5°, which is taken finally as the direction of the boundary lines.

After the direction θ of the boundary lines is derived, the boundary lines which do not satisfy the direction θ are removed as follows. For each Hough cell in the set S with parameters (A_i, B_i), if there exists a real number θ_{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 θ and so kept in the set S; otherwise, it is removed. Afterwards, the connected components S_i in the 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

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

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 line of the parking spaces have been detected.
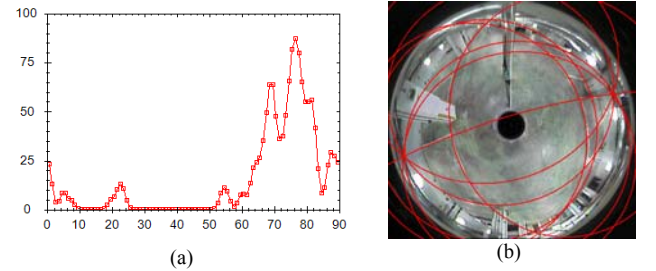

(a)   (b)

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

C. Parking Space Segmentation and Vacancy Detection

Two more steps are proposed in this section 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 performed easily 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 boundary line, one line 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 θ found previously, the boundary line can now be uniquely determined with only *one* point marked by the user; without θ, 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

added and one line 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 performed immediately to find the region of the parking space using the boundary lines. Two results are shown in Figs. 13(c) and (d).

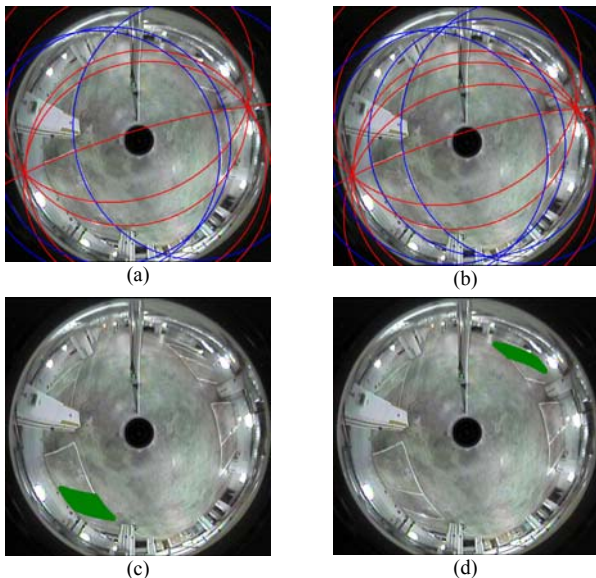(a)   (b)

(c)   (d)

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

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 as 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. And 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 to be *occupied*; otherwise, *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 Sec. V.D.
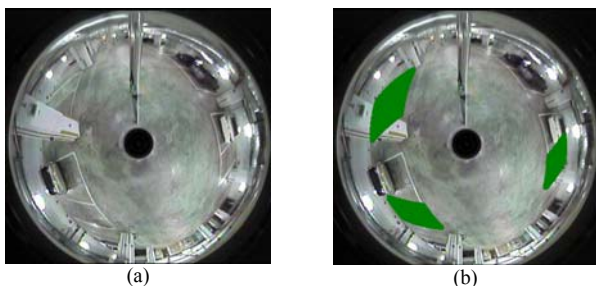

(a)   (b)

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

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 in this section.

A. Experimental Results of Proposed Calibration Method

The proposed camera calibration method described in Sec. 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 already been 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 one to be calibrated. For this, a series of experiments were conducted using space lines lying on the ground at different positions, with the camera affixed on the ceiling and looking downward. Specifically, as illustrated in Fig. 15, one of the parameters of a space line L is its elevation angle ψ, so several simulated space lines with different elevation angles ψ in the range of 0°~90° were generated for uses in the experiments. The parameters f_e of these simulated lines were all set to be $f_e^* = 500$, and their line images generated to be of the size 800×800 identically. Also, the coordinates of the pixels on each line image were disturbed by Gaussian noise with zero mean and varying standard deviations σ. Then, the proposed calibration method was applied to derive the 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 Figs. 16(a), (b), and (c) for different noise levels σ = 1.0, 3.0, and 5.0 pixels, respectively. Recalling that the actual value of the parameter f_e is $f_e^* = 500$, one can see from the figures that the parameter f_e can be robustly calibrated under a large noise level when the elevation angle $\Psi > 15°$. Note also that a space line with an elevation angle Ψ near 0° will appear to be a short and nearly *straight* line segment going through the image center, so that the parameter f_e cannot be calibrated reliably using it, as expected according to the theory of the proposed calibration method described in Sec. II.C.
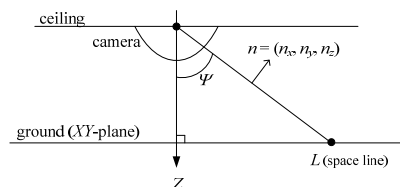

Fig. 15. Definition of the elevatation angle Ψ of a space line L on the ground.

B. Calibration of Camera Parameters

The main objective of the proposed methods is the ease for a non-technical user to set up the system. In this section, achievement of this goal is demonstrated — the user only has to select a single space line with its position and direction unknown. Images like 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 Figs. 17(a)-(d). In each experiment, the user only has to choose arbitrarily an obvious space line in the captured image, and marked *roughly* the pixels on the line as yellow points by mouse dragging. Then, the proposed calibration method was 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 Figs. 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. Also 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 experiment 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.
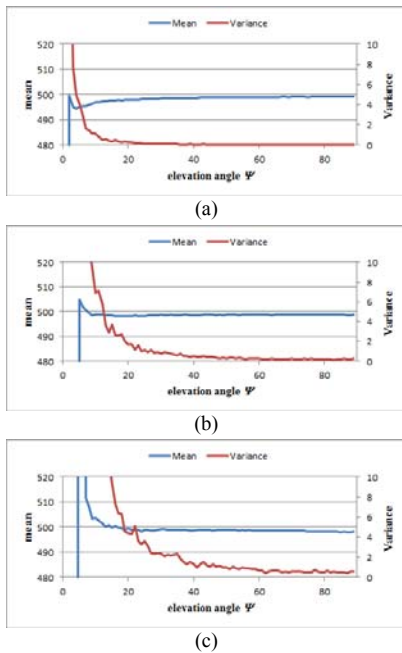

(a)

(b)

(c)

Fig. 16. Computed means and variances of calibrated values f_e for simulated space lines with different elevation angles Ψ where standard deviaion of Gaussian noise is 1.0 pixel for (a), 3.0 pixels for (b), and 5.0 pixels for (c).

C. Experimental Results of Boundary Line Detection

In the boundary line detection method proposed in Sec. IV.B, the direction of the boundary lines, denoted as θ, is computed first and the boundary lines are derived accordingly. Some experiments were conducted to test the capability of this scheme of finding θ 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 the angles 30°, 60°, and 90° as shown in Figs. 19(b), (c) and (d), respectively. The boundary line directions θ in these

four images were firstly derived. Then, the boundary lines were detected automatically and superimposed on the original images as shown in Figs. 19(a)-(d), respectively. From the figures, one can see that the boundary lines fit well to the real ones appearing in the captured images, meaning that the directions of the boundary lines were found correctly by the proposed method.
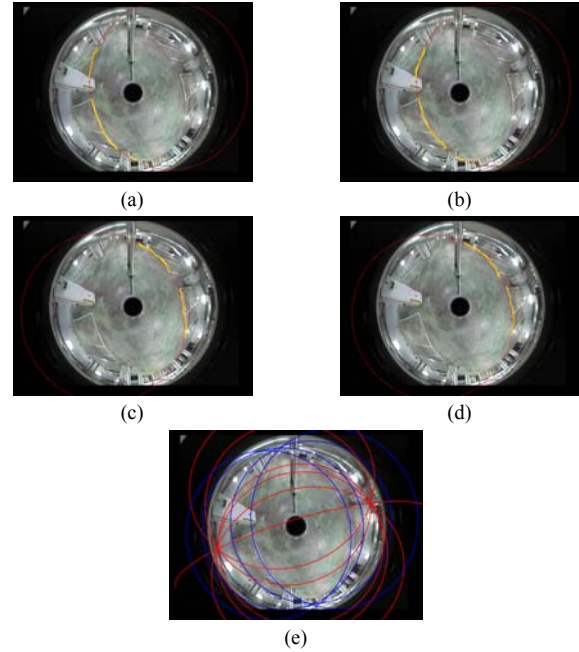

(a)

(b)

(c)

(d)

(e)

Fig. 17. Applying proposed calibrate 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.
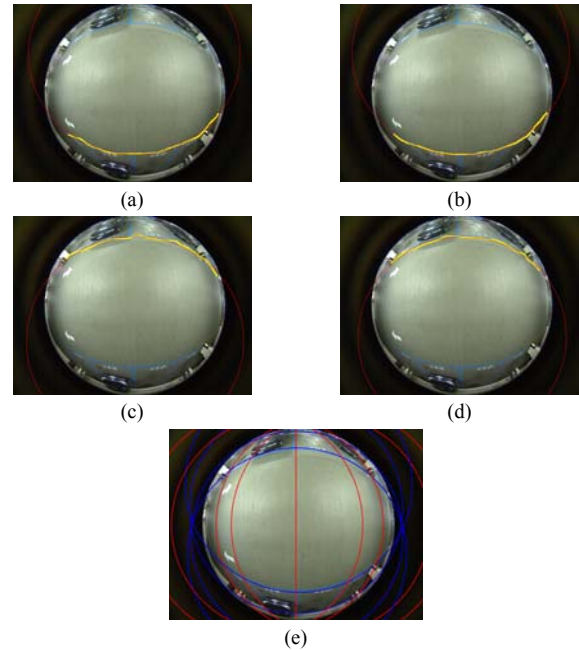

(a)

(b)

(c)

(d)

(e)

Fig. 18. Use of 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) The boundary lines found by proposed detection method with f_e = 331.12.

To demonstrate that the direction θ of the boundary lines can be robustly found even under bad conditions, two more experiments were conducted with input images shown in Figs. 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 Figs. 20(c), (e), and (g), and those of the second are shown in Figs. 20(d), (f), and (h). Specifically, the edge detection results are shown in Figs. 20(c) and (d), respectively, in which one can see that the boundary lines were poorly detected; also, some of the boundary lines are missing, as shown in Figs. 20(e) and (f). However, the directions of the detected and drawn boundary lines are correct, meaning that the direction θ can be derived precisely even under bad conditions. With this correct value of θ, the boundary lines can be adjusted easily by the proposed method as demonstrated by the results shown in Figs. 20(g) and (h).
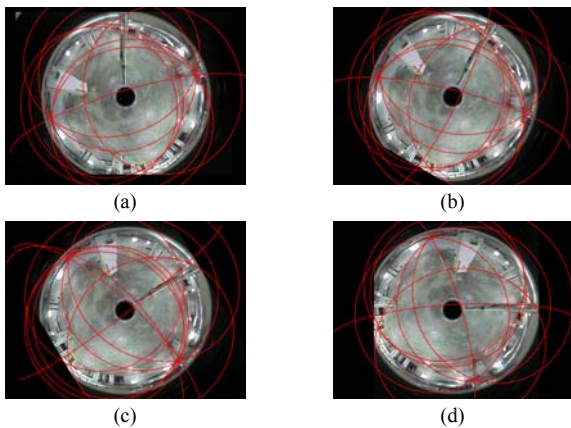

Fig. 19. Automatic detection results of boundary lines. Rotation angles of these images are (a) 0°, (b) 30°, (c) 60°, and (d) 90°, respectively. The detected boundary lines are drawn and superimposed on the images.

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 in this section. A fisheye-lens camera manufactured by Hunt Electronic with model No. HLC-1NAD was affixed on the ceiling of the parking lot. An image as shown in Fig. 21(a) was acquired at midnight firstly and used both for calibration and for detecting the boundary lines of the parking spaces. Specifically, a space line in the image was chosen arbitrarily 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 the parameter f_e was calibrated to be 331.1.

Afterwards, the direction θ of the boundary lines was derived, and the boundary lines detected automatically, with the results drawn in Fig. 21(c). Subsequently, boundary lines were added or removed by *single* clicks on the image as described previously. A result of such adjustments is shown in Fig. 21(d). At last, 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 Figs. 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.
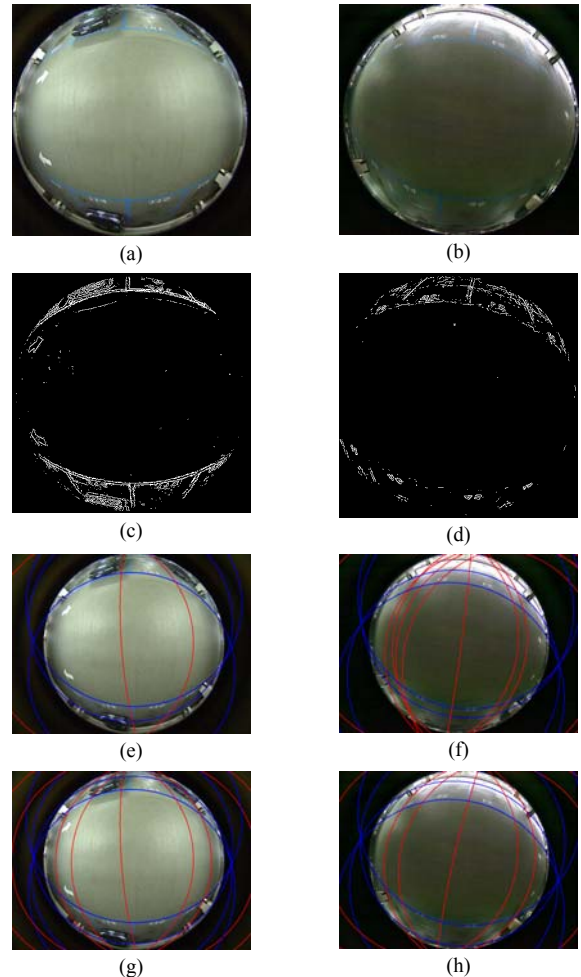

Fig. 20. Results of two experiments under bad conditions with those of the first shown in (a)(c)(e)(g), and those of the second shown in (b)(d)(f)(h). (a)(b) Input images. (c)(d) Edge detection results. (e)(f) Automatic detection results of boundary lines. (g)(h) Adjusted boundary lines.

VI. CONCLUSIONS 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, like fisheye-lens or catadioptric ones 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 also been proposed to calibrate the camera model using only one

space line without knowing its location and direction, so that the calibration process can be done easily 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 more robust and accurate detection result of parking-space boundary lines. A convenient adjustment method has also been 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 in order to conduct parking-space vacancy detection based on background subtraction. More intelligent methods may be developed to remove this weakness in the future study.
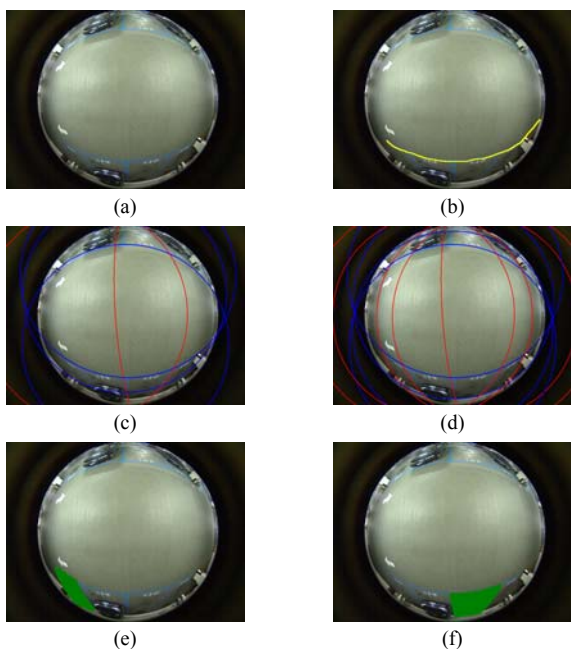

Fig. 21. Steps of setting up proposed system. (a) An image captured at midnight. (b) Manually-marked pixels (in yellow) of a space line for calibration task. (c) Results of automatic detection of boundary lines. (d) Result of boundary line adjustment. (e) and (f) Regions of two found parking spaces (marked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

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," *Proc. Neural Information Processing*, pp. 506-515, 2012.

[2] C. C. Huang and S. J. Wang, "A Hierahical Bayesian Generation Framework for Vacant Parking Space Detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 12, pp. 1770-1785, 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," *Proc. 2010 IEEE International Conference on Fuzzy System*, pp. 1950-1956, 2010.

[4] W. Lixia and J. Dalin, "A Method of Parking Space Detection based on Image Segmentation and LBP," *Proc. Fourth IEEE International Coneference on Multimedia Information Networking*, pp. 229-232, 2012.

[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," *Proc. 18th IEEE International Coneference on Image Processing*, pp. 637-640, 2011

[6] S. Ramalingam, P. Sturm, and S. K. Lodha, "Generic Self-Calibration of Central Cameras," *Computer Vision and Image Understanding*, vol. 114, no. 2, pp. 210-219, 2010.

[7] S. Gasparini, P. Sturm, and J. P. Barreto, "Plane-Based Calibration of Central Catadioptric Cameras," *Proc. IEEE 12th International Conference on Computer Vision*, pp. 1195-1202, 2009.

[8] C. Geyer and K. Daniilidis, "A Unifying Theory for Central Panoramic Systems and Practical Implications," *Proc. Sixth European Conf. Computer Vision*, pp. 445-462, 2000.

[9] X. Ying and Z. Hu, "Catadioptric Camera Calibration Using Geometric Invariants," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 10, pp. 1260-1271, 2004.

[10] X. Ying and Z. Hu, "Can We Consider Central Catadioptric Cameras and Fisheye Cameras within a Unified Imaging Model?" *Proc. European Conference on Computer Vision*, pp. 442-455, 2004.

[11] X. M. Deng, F. C. Wu, and Y. H. Wu, "An Easy Calibration Method for Central Catadioptric Cameras," *Acta Automatica Sinica*, vol. 33, no. 8, pp. 801-808, 2007.

[12] Y. Bastanlar, L. Puig, P. Sturm, J. J. Guerrero, and J. Barreto, "DLT-Like Calibration of Central Catadioptric Cameras," *Proc. 8th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras*, Oct. 2008.

[13] C. Geyer and K. Daniilidis, "Catadioptric Projective Geometry," *Int'l Journal of Computer Vision*, vol. 45, no. 3, pp. 223-243, 2001.

[14] J. P. Barreto and H. Araujo, "Geometric Properties of Central Catadioptric Line Images," *Proc. Seventh European Conference on Computer Vision*, pp. 237-251, 2002.

[15] J. Fabrizio, J. P. Tarel, and R. Benosman, "Calibration of Panoramic Catadioptric Sensors Made Easier," *Proc. IEEE Third Workshop on Omnidirectional Vision*, pp. 45-52, 2002.

[16] T. Apostol, *Calculus, Vol 1: One-Variable Calculus with an Introduction to Linear Algebra*, Wiley, 2nd edition, June, 1967.

Shen-En Shih received the B.S. degree in computer science from National Chiao Tung University, Taiwan, in 2009, and is currently pursuing the Ph.D. degree at the College of Computer Science, National Chiao Tung University. He has been a research assistant at the Computer Vision Laboratory in the Department of Computer Science at National Chiao Tung University from August 2009. His current research interests include computer vision, image processing, human-machine interfacing, and stereo vision.
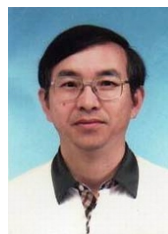
Wen-Hsiang Tsai received the B.S. degree in EE from National Taiwan University, Taiwan, in 1973, the M.S. degree in EE from Brown University, USA in 1977, and the Ph.D. degree in EE from Purdue University, USA in 1979. Since 1979, he has been with National Chiao Tung University (NCTU), Taiwan, where he is now a Chair Professor of Computer Science. At NCTU, he has served as the Head of the Dept. of Computer Science, the Dean of General Affairs, the Dean of Academic Affairs, and a Vice President. From 1999 to 2000, he was the Chair of the Chinese Image Processing and Pattern Recognition Society of Taiwan, and from 2004 to 2008, the Chair of the Computer Society of the IEEE Taipei Section in Taiwan. From 2004 to 2007, he was the President of Asia University, Taiwan.

Dr. Tsai has been an Editor or the 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*. He has published 157 journal papers and 245 conference papers and received many awards, including the Annual Paper Award from the Pattern Recognition Society of the USA; 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. His current research interests include computer vision, information security, video surveillance, and autonomous vehicle applications. He is a Life Member of the Chinese Pattern Recognition and Image Processing Society, Taiwan and a Senior Member of the IEEE.