Using Parallel Line Information on Building Ceilings for Indoor ALV Guidance and Automatic Environment Learning

Zhi-Fang Yang (楊致芳) and Wen-Hsiang Tsai (蔡文祥)
Institute of Computer and Information Science
Department of Computer and Information Science
National Chiao Tung University, Hsinchu, Taiwan 300, Republic of China

Abstract

An approach using parallel line information from building ceilings for indoor autonomous land vehicle (ALV) guidance and automatic environment learning is proposed. The objective is to steer an ALV in a straight corridor based on the information of the orientation and the horizontal relative position of the ALV. The approach has the advantage of utilizing ceiling line shapes which are usually not occluded. The approach is developed by treating a corridor as a cube. Two sets of parallel lines on a corridor ceiling is used to detect the vanishing line of the ceiling to derive the orientation of the ALV. A model of the parallel ceiling lines is built by automatic environment learning during a navigation stage. The horizontal relative position of the ALV is determined by model matching, in which the variation of parallel lines carries the information about the relative ALV position. Experiments have been conducted on a prototype ALV and acceptable results have been obtained to prove the feasibility of the proposed approach.

1. Introduction

Autonomous land vehicle (ALV) navigation by computer vision techniques is an intensive research field, which not only requires development of mature image processing methods, but also deep exploration of navigation environments. In addition, wide applications point out its importance.

Among a variety of visual information for use in ALV guidance in indoor environments, the information located on the ceiling, wall, and floor is quite stable and useful. The ceiling undoubtedly contains the most dependable data since it possesses the properties of being less occluded, providing the light source, and having more uniform styles. Building ceilings are thus utilized as the visual information source in this study for ALV guidance.

Vanishing points were derived from building ceilings to steer robots in Schuster, et al. [1]. However, not only vanishing points but also vanishing lines can be derived from ceilings. Hence the vanishing lines of ceilings are used in this study for ALV guidance. Furthermore, only the orientation of the robot is detected in [1] because of the intrinsic limitation of vanishing points. To improve this, a process of automatic environment learning and model matching are proposed to determine the relative position of the ALV.

On the other hand, the ceiling, walls, and floor of a building corridor form the shape of a cube. Viewing a problem world as a cube is also seen in Wang and Tsai [6], in which the cube is used as a calibration target. While the calibration is done outside a cube in [6], we attempt to study navigation inside a cube and this facilities coordinate system setup for ALV location.

Some related works are as follows. Line features in real indoor images are utilized, and elegant low-level image processing algorithms are developed for robot navigation in Lebegue and Aggarwal [2]. In Yagi, et al. [3], a conic projection image sensor (COPIS) acquires an omnidirectional view around the robot, from which the robot extracts path information to avoid collision and determines its locations and velocities. And in Yagi, et al. [4], the azimuth of each object in the omnidirectional image is detected to determine the robot location. In Atiya and Hager [5], an algorithm, which determines the correspondence between a stored map and detected landmarks, is given, and the location of the robot is computed based on the correspondence.

The proposed approach is now described briefly as follows. First, the coordinate systems are constructed from the viewpoint of treating the corridor as a cube. A model is built next automatically as an auxiliary for navigation during a learning stage. Then, we use the property of the vanishing line of the corridor ceiling to find the orientation of the ALV, and compute the horizontal relative position by analyzing the variation of the lines in the current image from those in the model. Finally, the turn angle of the ALV wheel is determined.

In the remainder of this paper, the descriptions of the coordinate systems and the transformations between them are described in Section 2. Section 3 includes the discussion on the use of vanishing points and vanishing lines for calculating the orientation of the ALV. Section 4 includes the description of a learning method, the process of detecting the horizontal relative position of the ALV with respect to the corridor wall, and the process for determination of the turn angle of the ALV wheel. Experimental results can be found in Section 5. And some conclusions are given in Section 6.

2. Viewing Corridor as a Cube and Corresponding Coordinate Systems

Viewing the navigation environment, a corridor, as a cube leads to a corresponding coordinate system representation. The global coordinate system for a given corridor set up for use in this study is shown in Fig. 1. The three main axes are parallel to the three main edges of the corridor, respectively. The camera coordinate system is built accordingly as shown in Fig. 2. The camera lens center points to the origin of the camera coordinate system.

Two types of homogeneous coordinates with an arbitrary constant k are defined as follows. First, let $\mathbf{P}_g = [kx_g, \ ky_g, \ kz_g, \ k]$ be the coordinates representing a point P_g in the global coordinate system, and $\mathbf{P}_c = [kx_c, \ ky_c, \ kz_c, \ k]$ the coordinates representing a point P_c in the camera coordinate system. The transformation between P_g and P_c is

$$\mathbf{P}_c = \mathbf{P}_g \mathbf{T} \mathbf{T}_\phi \mathbf{T}_\theta \mathbf{T}_\psi \tag{1}$$

where

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -l & -m & -n & 1 \end{bmatrix} \tag{2}$$

$$\mathbf{T}_\phi = \begin{bmatrix} \cos\phi & 0 & -\sin\phi & 0 \\ 0 & 1 & 0 & 0 \\ \sin\phi & 0 & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$\mathbf{T}_\theta = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$\mathbf{T}_\psi = \begin{bmatrix} \cos\psi & \sin\psi & 0 & 0 \\ -\sin\psi & \cos\psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

and ϕ, θ, and ψ are the pan, tilt, and swing angles of the camera coordinate system with respect to the global coordinate system. All illustration of the meanings of l, m, n, and the angles is shown in Fig. 2.

The transformation of the direction of a line from the global coordinate system to the camera coordinate system is also needed. Let points $\mathbf{p} = [p_x, \ p_y, \ p_z]$ and $\mathbf{q} = [q_x, \ q_y, \ q_z]$ lie on a line L, then we define the direction of line L as the vector $\mathbf{d}_{\mathbf{pq}} = [q_x - p_x, \ q_y - p_y, \ q_z - p_z]$. Let $\mathbf{d}_g = [ha_g, \ hb_g, \ hc_g, \ h]$ be the direction of a line in the global coordinate system. The transformation from $\mathbf{d}_g$ into $\mathbf{d}_c = [ha_c, \ hb_c, \ hc_c, \ h]$, which is the corresponding direction of the line in the camera coordinate system, is

$$\mathbf{d}_c = \mathbf{d}_g \mathbf{T}_\phi \mathbf{T}_\theta \mathbf{T}_\psi . \tag{6}$$

3. Detection of ALV Orientation by Vanishing Line Information from Building Ceiling

The proposed method begins with finding the formula describing the direction of the vanishing line of a corridor ceiling on the image plane of a certain camera coordinate system. With some reasonable constraints, it can be reduced to include only one unknown variable, the orientation of the ALV. In the mean time, the vanishing line of the corridor ceiling on an image is detected by image processing techniques. The image processing results and the formula constitute an equation. Solving the equation, we can get the orientation of the ALV.

3.1 Concept of Vanishing Line

How to infer 3-D information from 2-D images by computer vision techniques is necessary for ALV navigation. If we can get the relation between these two kinds of information, certain 3-D information can be recovered from 2-D images. In this study, vanishing lines on 2-D images are used to find 3-D information.

A vanishing line can be expected if we imagine a plane stretching infinitely. See Fig. 3 for the relation between vanishing points and vanishing lines. More details about vanishing points can be found in Schuster, et al. [1]. Having two vanishing points $\mathbf{p}_1$ and $\mathbf{p}_2$ on an image plane for two different sets of coplanar parallel lines, one can compute the direction of the vanishing line of the plane consisting of the two sets of coplanar parallel lines to be

$$\mathbf{l} = \mathbf{p}_1 - \mathbf{p}_2 . \tag{7}$$

3.2 The Vanishing Line of a Corridor Ceiling

We now determine the direction of the vanishing line of a corridor ceiling. The lines on the ceiling can be grouped into two sets of parallel lines. One has the direction $\mathbf{d}_x = (1, \ 0, \ 0)'$, and the other has the direction $\mathbf{d}_z = (0, \ 0, \ 1)'$. Both directions are defined in the global coordinate system. By (6), the corresponding directions $\mathbf{d}_{cx}$ and $\mathbf{d}_{cz}$ in the camera system can be found to be

$$\mathbf{d}_{cxh} = \mathbf{d}_{gxh}\mathbf{T}_{\phi}\mathbf{T}_{\theta}\mathbf{T}_{\psi} \tag{8}$$

and

$$\mathbf{d}_{czh} = \mathbf{d}_{gzh}\mathbf{T}_{\phi}\mathbf{T}_{\theta}\mathbf{T}_{\psi} \tag{9}$$

where $\mathbf{d}_{gxh} = [1,\ 0,\ 0,\ 1]'$ and $\mathbf{d}_{gzh} = [0,\ 0,\ 1,\ 1]'$, and the subscript h denotes homogeneous coordinates. And the corresponding vanishing points vp_x and vp_z on the image plane are

$$vp_x = \begin{bmatrix} f\dfrac{\cos\psi\cos\phi - \sin\theta\sin\psi\sin\phi}{-\cos\theta\sin\phi} \\ f\dfrac{\sin\psi\cos\phi + \sin\theta\cos\psi\sin\phi}{-\cos\theta\sin\phi} \end{bmatrix} \tag{10}$$

and

$$vp_z = \begin{bmatrix} f\dfrac{\cos\psi\sin\phi + \sin\theta\sin\psi\cos\phi}{\cos\theta\cos\phi} \\ f\dfrac{\sin\psi\sin\phi - \sin\theta\cos\psi\cos\phi}{\cos\theta\cos\phi} \end{bmatrix}. \tag{11}$$

By (7), the direction $\mathbf{d}_{xz}$ of the vanishing line composed of vp_x and vp_z is

$$\mathbf{d}_{xz} = \begin{bmatrix} f(\tan\phi + \cot\phi)\dfrac{\cos\psi}{\cos\theta} \\ f(\tan\phi + \cot\phi)\dfrac{\sin\psi}{\cos\theta} \end{bmatrix}. \tag{12}$$

3.3 Determination of ALV Orientation

The orientation of the ALV is determined by the pan, tilt, and swing angles of the camera which is fixed on the ALV. The tilt angle θ and the swing angle ψ of the ALV are constants during a navigation process. With these constraints, (12) can be written as

$$\mathbf{d}_{xz} = \begin{bmatrix} c_1(\tan\phi + \cot\phi) \\ c_2(\tan\phi + \cot\phi) \end{bmatrix}, \tag{13}$$

where $c_1 = f\dfrac{\cos\psi}{\cos\theta}$ and $c_2 = f\dfrac{\sin\psi}{\cos\theta}$ with three constants f, θ, and ψ.

We know that a set of parallel lines in the 3-D space must have an identical vanishing point on a 2-D image plane. By using this property, the candidates of a vanishing point can be chosen from the 2-D image. For this, we compute all intersections of the detected lines and collect the intersections falling in a tolerable range of a predicted vanishing point location as the candidates of that vanishing point. The range is set as a window centered at the predicted vanishing point position in the 2-D image. And by (10), the upper left point $\mathbf{W}_{ul-vpx}$ and the lower right point $\mathbf{W}_{lr-vpx}$ of the window for selecting the candidates of the vanishing point vp_x are computed with $\phi = \phi_0 - \phi_r$ and $\phi = \phi_0 + \phi_r$ respectively. And by (11), the upper left point $\mathbf{W}_{ul-vpz}$ and the lower right point $\mathbf{W}_{lr-vpz}$ of the window for selecting the candidates of the vanishing point vp_z are computed with $\phi = \phi_0 - \phi_r$ and $\phi = \phi_0 + \phi_r$ respectively, where ϕ_0 is the corresponding pan angle of a predicted vanishing point, and ϕ_r indicates the tolerable error range of the pan angle. One example is shown in Fig. 4. After all intersections of the lines located in the window are collected as the candidates of the vanishing point, their coordinates are averaged finally to get the desired vanishing point. And the direction of the vanishing line determined by vp_x and vp_z is computed directly by (7). Suppose that the direction of the vanishing line is found to be $(m,\ n)$ by the above procedure. With (13), we can compute the pan angle, yielding

$$\phi = \arctan\left(\frac{1}{2}\left(c_1^{-1}m \pm \left(c_1^{-2}m^2 - 4\right)^{\frac{1}{2}}\right)\right) \tag{14}$$

or

$$\phi = \arctan\left(\frac{1}{2}\left(c_2^{-1}n \pm \left(c_2^{-2}n^2 - 4\right)^{\frac{1}{2}}\right)\right). \tag{15}$$

The differences between the pan angle of the predicted vanishing point and the pan angles found by (14) and (15) are calculated. The calculated pan angle with the smallest difference is chosen as the desired pan angle. The idea behind such a selection is that the pan angle is not changed dramatically during the navigation process. Then, the orientation of the ALV, denoted as $\Delta\phi$, is:

$$\Delta\phi = \phi - \phi_{CA} \tag{16}$$

where ϕ_{CA} is the pan angle of the camera with respect to the ALV, which is obtained by calibration.

Sometimes it may fail to detect the existence of two vanishing points, i.e., only one vanishing point is found. In this situation, the single vanishing point is used to determine the orientation of the ALV, which can be directly derived from (10) and (16) or from (11) and (16).

4. Detection of Horizontal Relative Position of ALV

In order to keep the ALV in a straight path in the corridor, the horizontal relative position of the ALV with respect to the corridor wall must be known. While the vanishing line only contains information about the orientation of the ALV, the information of the variation in slope and position of a set of parallel lines on the ceiling provides certain hint of the relative position of the ALV. For example, when we move left or right horizontally and look at the set of parallel lines pointing forward on the ceiling, we can see those lines varying in their positions and slopes (see Fig. 5). Based on this phenomenon, a model of a set of parallel lines on the ceiling is established in advance and the difference between the parallel lines on an image and those in the model is analyzed to get the horizontal relative position of the ALV with respect to the corridor wall in this study. The method consists of automatic environment learning and model matching and all the needed information is retrieved from the parallel lines on building ceilings.

4.1 Model Learning

Automatic environment learning (or model learning) in this study means to collect environment information automatically by an ALV for use in later navigation. In the learning stage, an operator drives the ALV through a desired path and takes images to collect path information at first. The ALV system then processes the images and builds a model of the path environment automatically for use in the navigation stage.

The learning stage is as follows.

Step 1: *Acquiring images:* Drive the ALV manually in the corridor and acquire images continuously.

Step 2: *Detecting edge points:* Detect edge points in each acquired image by scanning the image line by line horizontally. A pixel p_i is selected as an edge point if it satisfies the following relation:

$$|p_i - p_{i-1}| > t, \tag{17}$$

where p_{i-1} is the left neighbor of p_i, and t is a threshold constant.

Step 3: *Detecting line candidates:* Apply the Hough transform [8] on the edge points to detect lines in each acquired image. The normal representation of a line, $x\cos\theta + y\sin\theta = \rho$, is used, and a $\theta-\rho$ parameter space is set up. The contribution to each corresponding accumulator cell of an edge point p_r at the rth row of the image (the r value is numbered from bottom to top) is a weight w depending on the r value:

$$w = (row - r)/s, \tag{18}$$

where row is the number of rows of the image, and s is a scaling constant.

Step 4: *Selecting a candidate model line:* Choose the line $x\cos\theta_{\max} + y\sin\theta_{\max} = \rho_{\max}$ as a candidate model line if the corresponding accumulator cell $(\theta_{\max}, \rho_{\max})$ has the maximum value (denoted as max) in the entire $\theta-\rho$ parameter space.

(The following three steps aim to remove the "plateau" associated with the "peak" of the candidate model line found in Step 4.)

Step 5: *Detecting "plateaus" in the parameter space:* Scan the entire parameter space in the order of increasing θ_i values, and mark a θ_i value with bit "1" ("1" means a high spot in the "plateaus") if there exits an accumulator cell (θ_i, ρ_j) with cell value k satisfying the following constraint:

$$k > max \times c, \tag{19}$$

where c is a constant; otherwise mark θ_i with bit "0". The c value used in this study is 0.8.

Step 6: *Filling holes in detected "plateaus":* List the marks of all θ_i values derived from Step 5 as a string. Scan the string and change the "101", "1001", and "10001" patterns in the string into "111", "1111", and "11111", respectively.

Step 7: *Removing the plateau containing θ_{max}:* Search the string obtained from Step 6 to detect the range $(\theta_{begin}, \theta_{end})$ of the 1's run containing θ_{max}. Reset all the accumulator cells (θ_i, ρ_j) to 0 if θ_i satisfies the following constraint:

$$\begin{cases} \theta_{begin} \le \theta_i \le \theta_{end} & \text{when } \theta_{end} - \theta_{begin} > \theta_{range}; \\ \theta_{max} - \frac{1}{2}\theta_{range} \le \theta_i \le \theta_{max} + \frac{1}{2}\theta_{range} & \text{otherwise,} \end{cases} \tag{20}$$

where θ_{range} is a constant.

Step 8: *Repeating the candidate model line selection steps:* Repeat Steps 4 through 7 until i candidate model lines are selected in an acquired image. In this study, i is chosen as 2.

Step 9: *Ordering the candidate model lines:* Sort the i candidate model lines according to their θ values in the increasing order, based on the assumption that the θ values of the ceiling lines increases from left to right on the acquired images.

Step 10: *Grouping all the candidate model lines:* Collect the kth lines in all acquired images as the kth group.

Step 11: *Choosing the desired model lines:* Sort the lines in each group according to their Hough parameter θ, and average all the ρ values with the same θ value. Collect all the lines whose parameters θ are the medians in their respective groups as the desired model.

4.2 Determination of Horizontal Relative Position of ALV by Model Matching

In a navigation stage, the acquired images are matched with the learned model. The difference between the current image and the model contains information about the horizontal relative position of the ALV with respect to the corridor wall. In fact, two factors influence the difference: the relative angles and the relative position of the ALV with respect to the global coordinate system. Because the camera is fixed on the ALV which navigates on a flat floor, the tilt angle and the swing angle of the camera with respect to the global coordinate system are constants. Furthermore, the only translation considered is the x axis direction of the global coordinate system. Thus, the two factors are actually the relative pan angle and the horizontal relative position of the ALV with respect to the global coordinate system. Since the relative pan angle of the ALV can be detected by vanishing line information, the horizontal relative position of the ALV can thus be derived as follows.

Consider a point in the global coordinate system. It has different coordinates for two different camera coordinate systems. Given a point $\mathbf{p} = (x,\ y,\ z)$ which has coordinates $\mathbf{p}_0' = (x_0',\ y_0',\ z_0')$ for the first camera coordinate system $\mathbf{C}_0$, and coordinates $\mathbf{p}_1' = (x_1',\ y_1',\ z_1')$ for the second camera coordinate system $\mathbf{C}_1$ which rotates through a pan angle $\Delta\phi$ and shifts for a distance of Δx with respect to the $\mathbf{C}_1$ coordinate system, by (1) we have

$$\begin{aligned}&(x_0',\ y_0',\ z_0',\ 1)\\ &= (x_1',\ y_1',\ z_1',\ 1)\mathbf{T}_{-\psi}\mathbf{T}_{-\theta}\mathbf{T}_{-\Delta\phi}\mathbf{T}_{-\phi_0}\mathbf{T}_{-\Delta x}\mathbf{T}_{\phi_0}\mathbf{T}_{\theta}\mathbf{T}_{\psi},\end{aligned} \tag{21}$$

and the coordinates of the corresponding image point of $\mathbf{p}_0'$, denoted as $\left(f\frac{x_0'}{z_0'},\ f\frac{y_0'}{z_0'}\right)$, can be written in terms of those of the corresponding image point of $\mathbf{p}_1'$, denoted as $(x_{1i}',\ y_{1i}')$, as follows (the details omitted) :

$$\left(f\frac{x_0'}{z_0'},\ f\frac{y_0'}{z_0'}\right) = \left(f\frac{U'+R\Delta x}{W'+T\Delta x},\ f\frac{V'+S\Delta x}{W'+T\Delta x}\right) \tag{22}$$

where

$$U' = \frac{1}{f}[A\cdot x_{1i}' + B\cdot y_{1i}' + C\cdot f]$$

$$V' = \frac{1}{f}[D\cdot x_{1i}' + E\cdot y_{1i}' + F\cdot f]$$

$$W' = \frac{1}{f}[G\cdot x_{1i}' + H\cdot y_{1i}' + I\cdot f]$$

and A through I, and $R,S,$ and T are computed according to the following equality:

$$\mathbf{T}_{-\psi}\mathbf{T}_{-\theta}\mathbf{T}_{-\Delta\phi}\mathbf{T}_{-\phi_0}\mathbf{T}_{-\Delta x}\mathbf{T}_{\phi_0}\mathbf{T}_{\theta}\mathbf{T}_{\psi} = \begin{bmatrix} A & D & G & 0\\ B & E & H & 0\\ C & F & I & 0\\ \Delta x\cdot R & \Delta x\cdot S & \Delta x\cdot T & 1\end{bmatrix}$$

where all the $\mathbf{T}$ matrices on the left-hand side have been described previously.

Suppose $\mathbf{P}$ is a point on some line L parallel to the z axis of the global coordinate system. Let L_0' be the projected line on the image plane of the $\mathbf{C}_0$ camera coordinate system. The vanishing point of the z axis of the global coordinate system on the image plane of the $\mathbf{C}_0$ system, denoted as $\mathbf{vp} = (vp_x,\ vp_y)$, must lie on L_0'. Since the projected point of $\mathbf{P}$ is also on the line L_0', the direction of line L_0', denoted as $\mathbf{S}_0$, can be derived from these two points. That is,

$$\mathbf{S}_0 = \left(f\frac{x_0'}{z_0'},\ f\frac{y_0'}{z_0'}\right) - \mathbf{vp}. \tag{23}$$

By (22), the slope $\tan\theta_0$ for $\mathbf{S}_0$ can be shown in terms of the corresponding image point $\mathbf{p}_1'$ in the $\mathbf{C}_1$ camera system as follows :

$$\tan\theta_0 = \left(\frac{f\dfrac{V'+S\Delta x}{W'+T\Delta x} - vp_y}{f\dfrac{U'+R\Delta x}{W'+T\Delta x} - vp_x}\right). \tag{24}$$

Let $\tan\theta_0$ be the slope of the line in the model. Note that both $\tan\theta_0$ and $\mathbf{vp}$ can be obtained from the learned model. Then, Δx can be solved from (24). That is,

$$\Delta x = (-z_1')\frac{(fV' - W'vp_y) - \tan\theta_0(fU' - W'vp_x)}{(fS - Tvp_y) - \tan\theta_0(fR - Tvp_x)}. \tag{25}$$

Note that z_1' is treated as a scalar since we don't want to know it. Recall that if the ALV navigates in the middle path of the corridor, its x coordinate in the global coordinate system should be l. But now the x coordinate of the ALV in the global coordinate system is actually $l+\Delta x$, which means that the ALV has shifted for the distance of Δx in the direction of the x axis of the global coordinate system.

The model matching algorithm is summarized as follows:

Step 1: *Detecting edge points:* A pixel p_i in each acquired image is selected as an edge point if it satisfies (17).

Step 2: *Detecting line candidates:* Apply the Hough transform [8] on the edge points to detect lines in each acquired image. The normal representation of a line, $x\cos\theta + y\sin\theta = \rho$, is used, and a $\theta-\rho$ parameter space is set up. The contribution to each corresponding accumulator cell of an edge point p_r at the rth row of the image (the r value is numbered from bottom to top) is a weight w computed by (18).

Step 3: *Selecting a candidate model line:* Choose the line $x\cos\theta_{\max} + y\sin\theta_{\max} = \rho_{\max}$ as a candidate model line if the corresponding accumulator cell $(\theta_{\max}, \rho_{\max})$ has the maximum value (denoted as max) in the entire $\theta-\rho$ parameter space.

(The following three steps aim to remove the "plateau" associated with the "peak" of the candidate model line found in Step 3.)

Step 4: *Detecting "plateaus" in the parameter space:* Scan the entire parameter space in the order of increasing θ_i values, and mark a θ_i value with bit "1" ("1" means a high spot in the "plateaus") if there exits an accumulator cell (θ_i, ρ_j) with cell value k satisfying (19).

Step 5: *Filling holes in detected "plateaus":* List the marks of all θ_i values derived from Step 5 as a string. Scan the string and change the "101", "1001", and "10001" patterns in the string into "111", "1111", and "11111", respectively.

Step 6: *Removing the plateau containing θ_{max}:* Search the string obtained from Step 5 to detect the range $(\theta_{begin}, \theta_{end})$ of the 1's run containing θ_{max}. Reset all the accumulator cells (θ_i, ρ_j) to 0 if θ_i satisfies (20).

Step 7: *Repeating the candidate model line selection steps:* Repeat Steps 3 through 6 until i candidate model lines are selected in the acquired image. In this study, i is chosen as 2.

Step 8: *Ordering the candidate model lines:* Sort the i candidate model lines according to their θ values in the increasing order.

Step 9: *Computing the Δx by the matched model line:* The selected line , the kth line, with the maximum value among all the Hough accumulator cells is chosen as the matched line. Select an arbitrary point on it and substitute its coordinates on the image plane into (25) with the parameters of the corresponding kth model line to obtain the value of Δx for the desired shift x value of the ALV in the x direction.

4.3 Determination of ALV Wheel Turn Angle

The pan angle of the ALV and the horizontal relative position of the ALV with respect to the corridor wall now can be used to determine the turn angle of the ALV wheel. To this aim, we propose to use the following formula :

$$\Delta\phi_{turn} = f(\Delta\phi) + g(\Delta x), \tag{26}$$

where $\Delta\phi$ derived from (16) is the pan angle of the ALV, Δx derived from (26) is the horizontal relative translation of the ALV with respect to the corridor wall, and f and g are two functions to be described next.

Since vanishing point and vanishing lines are sensitive to noise, it is not proper to use $\Delta\phi$ directly. The value $\Delta\phi$ should be adjusted in order to provide reliable information. Therefore, the function f is taken to be as follows in this study :

$$f(\Delta\phi) = \begin{cases} -c_1\Delta\phi & if \quad c_1\Delta\phi \le \Delta\phi_{\max}; \\ -\dfrac{\Delta\phi_{\max}\Delta\phi}{|\Delta\phi|} & if \quad c_1\Delta\phi > \Delta\phi_{\max}. \end{cases} \tag{27}$$

where c_1 and $\Delta\phi_{\max}$ are two constants. That is, the system scales $\Delta\phi$ first and then limit the influence of $\Delta\phi$ in a predefined range $\Delta\phi_{\max}$. The function g takes the horizontal relative translation of the ALV into consideration. And it is chosen in this study as

$$g(\Delta x) = -c_{2i}(\Delta x + s_i), \tag{28}$$

where c_{2i} >0 is the scaling constant for the ith model line and s_i is the constant for the ith model line.

5. Experimental Results

Experiments were conducted in a building corridor and acceptable results have been obtained to prove the feasibility of the proposed scheme. The images were taken by a CCD camera mounted on the ALV. Image processing was done with a PC486 on the ALV. The maximum speed of the ALV is about 25 cm/sec.

The model was built during a learning stage in which 17 images were taken. Two lines passing through the left and right edges of the ceiling lamps respectively are selected in the model. The values $\tan\theta_0$ and vp used in (25) were found to be 31.821 for the left model line, -4.165 for the right model line, and $(-230.523, -9.27)$ respectively. Several representative images taken during the navigation stage together with the extracted lines are shown in Fig. 6. Fig. 6(a) shows an image taken at 60 cm left from the middle path of the corridor. The computation results include

matched model line → the right one;

$\Delta x = -3120$ with $z_1' = -10000$

(The ALV is at the left side.);

$\Delta\phi = -0.0524$ (The ALV navigates towards right;

it should turn to left by 3.002°.);

$\Delta\phi_{turn} = 0.0349$ (The ALV turns left less than 3.002°).

Fig. 6(b) shows an image taken at 10 cm left from the middle path of the corridor. The computation results include

matched model line → the right one;

$\Delta x = -2677$ with $z_1' = -10000$

(The ALV is almost in the middle.);

$\Delta\phi = 0.0253$ (The ALV navigates towards left,

and it should turn to left by 1.45°.);

$\Delta\phi_{turn} = -0.0253$

(The ALV turns left without considering Δx).

And Fig. 6(c) shows an image taken at 60 cm right from the middle path of the corridor. The computation results include

matched model line → the right one;

$\Delta x = -1870$ with $z_1 = -10000$

(The ALV is at the right side.);

$\Delta\phi = 0.0524$ (The ALV navigates towards left;

it should turn to right by 3.002°.);

$\Delta\phi_{turn} = -0.0349$

(The ALV turns right less than 3.002°.).

From these data, we can find the Δx increases from the left side to the right side. In all of these cases, the ALV tunes to the right side successfully.

6. Conclusions

An approach has been proposed for indoor ALV guidance and automatic environment learning by computer vision techniques. All necessary information comes from the parallel lines on building ceilings. Interesting results or observations have been found in this research. First, building ceilings are indeed a good kind of navigation information source. Ceilings are quite uniform and less occluded. Especially, the parallel lines on ceilings contain various valuable information: the vanishing line of the corridor ceiling determines the orientation of the ALV, and the variation of a set of parallel lines on the ceiling determines the horizontal relative translation of the ALV. Second, viewing a corridor as a cube helps us construct necessary coordinate systems. And it is an interesting viewpoint to treat the ALV walking in a big cube. Third, a method of automatic environment learning has been developed to extract information automatically. A model is built as an auxiliary data in the navigation stage. Finally, there is always a tradeoff between time-consuming low-level image processing and image processing errors. Here we tolerate the image processing errors to meet the real time constraint. Although our method is quite satisfactory for indoor ALV navigation, future works may be devoted to more complicated works such as cross turning in a path, navigation under other ceiling shapes, etc.

References

[1] R.Schuster, N.Ansari, and A.Bani-Hashemi, "Steering a robot with vanishing points," *IEEE Trans. on Robotics and Automation*, Vol.9, No.4, pp. 491-498, Aug. 1993.

[2] X. Lebegue and J.K. Aggarwal, "Significant line segments for an indoor mobile robot," *IEEE Trans. on Robotics and Automation*, Vol. 9, No. 6, pp. 801-815, Dec. 1993.

[3] Y. Yagi, S. Kawato, and S. Tsuji, "Real-time omnidirectional image sensor (COPIS) for vision-guided navigation," *IEEE Trans. on Robotics and Automation*, Vol.10, No.1, pp. 11-22, Feb. 1994.

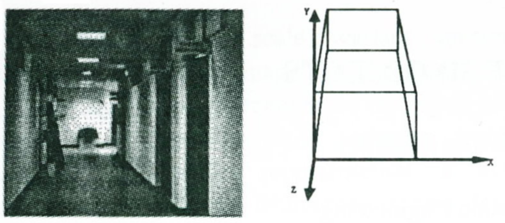
[4] Y.Yagi, Y. Nishizawa, and M. Yachida, "Map-based navigation for a mobile robot with omnidirectional image sensor COPIS," *IEEE Trans. on Robotics and Automation*, Vol.11, No.5, pp.634-648, Oct. 1995.

[5] S. Atiya and G.D. Hager, "Real-time vision-based robot localization," *IEEE Trans. on Robotics and Automation*, Vol. 9, No. 6, pp.785-800, Dec. 1993.

[6] L.L.Wang and W.H.Tsai, "Camera calibration by vanishing lines for 3D computer vision," *IEEE Trans. PAMI*, Vol.13, No3, pp.370-376, 1991.

[7] F.M.Pan and W.H.Tsai, "Automatic environment learning and path generation for indoor autonomous land vehicle guidance using computer vision techniques," *Proceedings of 1993 National Computer Symposium*," Chiai, Taiwan, Republic of China, Dec. 1993, pp.311-321.

[8] R.C. Gonzalez, and R.E. Woods, [1992]. *Digital Image Processing*, Addison-Wesley, Reading, Mass.



(a) A corridor image.

(b) The global coordinate system.

Fig. 1. Corridor representation.

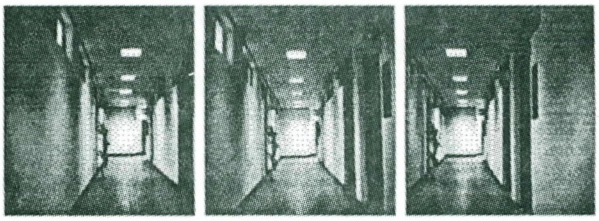
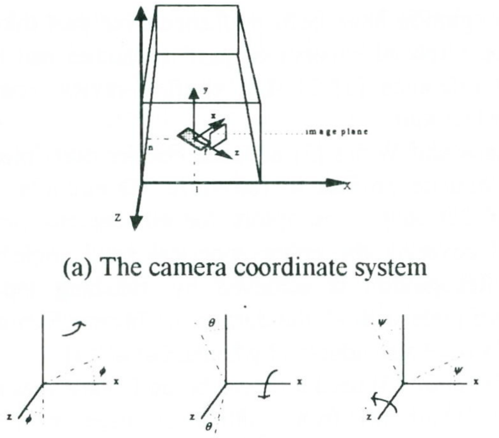


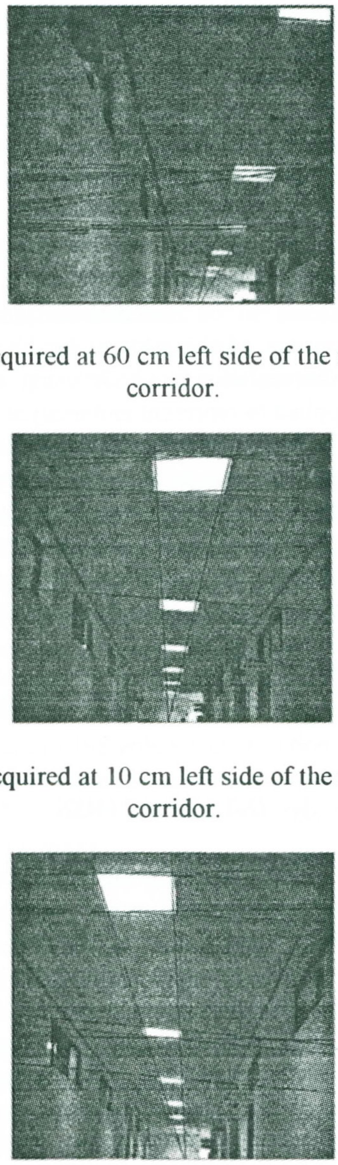
Fig. 5. Images acquired at different horizontal relative positions. Note the lines composed of the edges of the ceiling lamps have different slopes and positions.



(a) The camera coordinate system

(b) Definition of the pan, tilt, and swing angles.

Fig. 2. Coordinate systems and the definitions of rotation and parameters.



(a) Image acquired at 60 cm left side of the middle of the corridor.

(b) Image acquired at 10 cm left side of the middle of the corridor.

(c) Image acquired at 60 cm right side of the middle of the corridor.

Fig. 6. Processed images acquired at different relative positions with respect to the middle of the corridor.

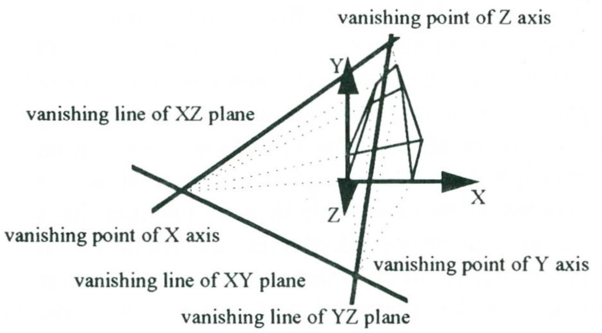

vanishing point of Z axis
vanishing line of XZ plane
Y
X
Z
vanishing point of X axis
vanishing point of Y axis
vanishing line of XY plane
vanishing line of YZ plane

Fig. 3. The vanishing lines and vanishing points of a cube in the global coordinate system.

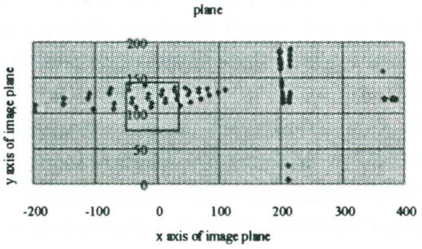


Fig. 4. The intersections falling in the window (the rectangle) are chosen as vanishing point candidates and their coordinates are averaged finally as the vanishing point. The dots represent the intersections of the detected lines on the image plane.