Automatic 3D House-layout Construction by a Vision-based Autonomous Vehicle Using a Two-camera Omni-directional Imaging System

Bo-Jhih You (尤柏智)
Institute of Multimedia Engineering,
National Chiao Tung University,
Hsinchu, Taiwan 30010
E-mail: yokihiro.cs97g@nctu.edu.tw

Wen-Hsiang Tsai (蔡文祥)
Dept. of Computer Science,
National Chiao Tung University,
Hsinchu, Taiwan 30010
E-mail: whtsai@cis.nctu.edu.tw

***Abstract*—An automatic house-layout construction system via vision-based autonomous vehicle navigation without human involvement is proposed. The system can be used to acquire the floor layout and flat objects on walls like doors and windows of an empty room. First, a new type of omni-directional camera system is designed, which consists of two omni-cameras aligned coaxially and back to back. The proposed house layout construction process consists of three stages, vehicle navigation, floor layout construction, and 3-D graphic house layout display. The vehicle navigation stage is conducted to follow the wall mopboards. Then, a global optimization method is proposed to construct a floor layout from all the mopboard edges in the second stage. In the last stage, doors and windows are detected from the omni-images taken in the vehicle navigation stage. With the above-mentioned data, a graphic form of the house configuration for 3-D display from any viewpoint can be obtained. Finally, experimental results are shown to to demonstrate the feasibility of the proposed approach.**

Keywords-autonomous vehicle navigation, omni-directional camera, automatic house layout construction.

I. INTRODUCTION

In recent years, related applications of the autonomous vehicle have been developed intensively by many researchers to help human beings in various areas of automation. There are many empty pre-owned houses with uncertain interior space configurations, and a house agent may want to obtain the layout of each empty room in a house in advance before getting the house ready for sale. It is inconvenient and sometimes dangerous for the house agent to measure the house layout line by line and room by room *by hand*. Besides, this task usually includes measurement of the positions of doors and windows in rooms, which might be so numerous that the task becomes too time-consuming to be endurable. A possible way out is to conduct the task *automatically* without human involvement!

To achieve this goal, it is desired to design a vision-based autonomous vehicle to possess an ability to navigate by *mopboard following* in an unknown empty room space and acquire automatically during navigation the information of the room layout and the structures (positions and heights) of the doors and windows in the room. This is feasible because most houses have mopboards at the roots of the walls. Use of the mopboard for vehicle guidance also simplifies the design of the vehicle navigation work for solving the above-mentioned *automatic house-layout construction* problem. Therefore, object localization for estimating mopboard positions is required.

Various visual sensing equipments such as cameras and ultrasonic sensors were used to acquire the 3-D information in vehicle surroundings in the past studies [1-2]. For depth information estimation, the most common stereo matching method may be used, which is based on the triangulation principle to obtain the relation between a 3-D point and multiple cameras, but matching of corresponding points between images is often a difficult problem in this approach. Use of the laser range finder together with conventional imaging sensors were also used frequently [3-4]. However, the laser range finder has a drawback, that is, it can only scan a 2D plane at a specific height. Jeng and Tsai [5] proposed a space-mapping method for the omni-camera image, which can be used to estimate the location of an object. In this study, we propose a method which is based on two concepts. The first is the use of the triangulation principle to measure depths of objects (i.e, their *ranges*). The other is the use of the space-mapping technique proposed by Jeng and Tsai [5] to get the location for a concerned object. In this way, we can estimate the positions of objects on the floor and walls in a room.

Autonomous vehicles or mobile robots usually suffer from accumulations of mechanical errors during navigation, which cause inaccurate measures of the moving distances and orientations yielded by the odometer in the vehicle. Chen and Tsai [6] proposed a mechanical error correction method by curve-fitting the erroneous deviations from correct paths. In this study, we utilize the curve-fitting method proposed by Chen and Tsai [6] to correct the positions of the vehicle and use the edge information estimated from images to correct the direction of the vehicle. Analyzing the omni-images taken by different types of camera is an important topic. Kim and Oh [4] proposed a method for extracting vertical lines from an omni-image with the help of the horizontal lines generated by a laser range finder.

The goal of this study is to design a system for automatic house-layout construction in an empty indoor room space using a vision-based autonomous vehicle. It is assumed that the adjacency walls in the room are perpendicular to each other. In order to achieve this goal, the system needs not only a capability of automatic navigation but also one to acquire environment information automatically. For a vehicle to have such capabilities, an imaging device with two vertically-aligned omni-cameras connected in a bottom-to-bottom fashion is designed in this study. With the environment information collected from the taken images, a method for constructing a 3-D house layout in graphic form by estimating the locations of mopboard edges, doors, and windows is also proposed. The system consists of three major phases: setup of the imaging system, vehicle navigation by mopboard following, and 3-D construction of room space. The details will be introduced in the following sections.

II. SYSTEM CONFIGURATION

In the proposed system, we make use of a Pioneer 3-DX vehicle made by MobileRobots Inc. as a test bed. The vehicle is equipped with an imaging system composed of two catadioptric omni-directional cameras which are connected and vertically-aligned in a bottom-to-bottom fashion. The imaging system is not only part of the vehicle system but also plays an important role of gathering environment information and locating the vehicle. A diagram illustrating the configuration of this system is shown in Fig. 1(a), and Figs. 1(b) and 1(c) show an example of the omni-images captured by the imaging system.

The overall framework of the proposed system is illustrated in Fig. 2. The major stages in proposed 3D house-layout construction by vehicle navigation include: (1) vehicle navigation by mopboard following; (2) floor-layout construction; and (3) 3-D house-layout construction.

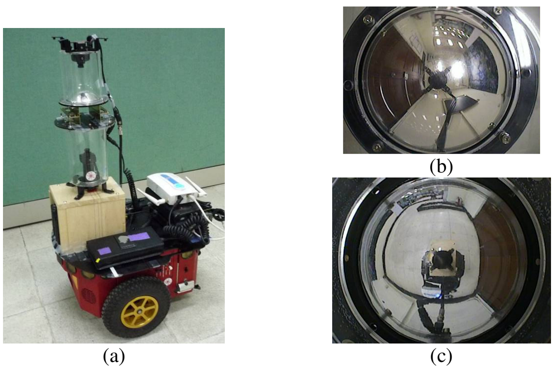


Figure 1. Illustration of system configuration. (a) The two-camera imaging system is equipped on the vehicle. (b) An acquired image of the ceiling using the upper camera. (c) An acquired image of the floor using the lower camera.

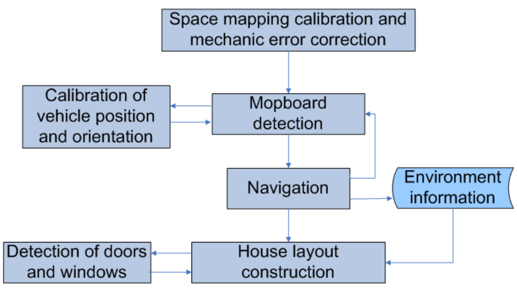


Figure 2. Flowchart of proposed system.

Before conducting vehicle navigation, we must perform the setup of the system which includes space-mapping calibration and mechanic error correction. For camera calibration, we establish a space-mapping table which is proposed by Jeng and Tsai [5] for each omni-camera by finding the relations between specific points (on a specially-designed calibration object) in 2-D omni-images and the corresponding points in 3-D space. In other words, the space-mapping table specifies a relation among the corresponding space-image coordinates pairs (P_i, p_i). The relation between the omni-camera and image coordinate systems is shown in Fig. 3, and the example of mapping table is shown in Table 1. As a result, the vehicle which carries the imaging system has the abilities to estimate the distances between the vehicle and an object (the mopboard here) utilizing the omni-images.

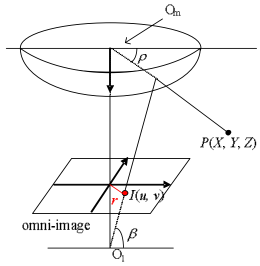


Figure 3. Relation between omni-camera and image coordinate systems.

Table 1. Example of pano-mapping table of size $M\times N$ [5].

	θ_1	θ_2	θ_3	θ_4	…	θ_M
ρ_1	(u_{11}, v_{11})	(u_{21}, v_{21})	(u_{31}, v_{31})	(u_{41}, v_{41})	…	(u_{M1}, v_{M1})
ρ_2	(u_{12}, v_{12})	(u_{22}, v_{22})	(u_{32}, v_{32})	(u_{42}, v_{42})	…	(u_{M2}, v_{M2})
ρ_3	(u_{13}, v_{13})	(u_{23}, v_{23})	(u_{33}, v_{33})	(u_{43}, v_{43})	…	(u_{M1}, v_{M3})
ρ_4	(u_{14}, v_{14})	(u_{24}, v_{24})	(u_{34}, v_{34})	(u_{44}, v_{44})	…	(u_{M1}, v_{M4})
…	…	…	…	…	…	…
ρ_N	(u_{1N}, v_{1N})	(u_{2N}, v_{2N})	(u_{3N}, v_{3N})	(u_{4N}, v_{4N})	…	(u_{MN}, v_{MN})

When creating the space mapping table, the focal point of the hyperboloidal mirror is viewed as the origin in the CCS. However, the focal point may not be located on the bottom of the omni-camera. We propose in this study a method to find the focal point of the hyperboloidal mirror. As shown in Fig. 4, we use two different landmarks L_1 and L_2 which have the same corresponding image point p with known heights and horizontal distances from the transverse axis of the hyperboloidal mirror. We assume that O_w is at (0, 0, 0). Then, as shown in Fig. 4, the position of the focal point can be computed by the following equation:

$$\tan\theta = \frac{H_1 - \overline{O_m O_w}}{D_1} = \frac{H_2 - H_1}{D_2 - D_1}; \quad (1)$$

$$\overline{O_m O_w} = H_1 - D_1 \times \tan\theta = H_1 - D_1 \times \frac{H_2 - H_1}{D_2 - D_1}. \quad (2)$$

Accumulations of mechanical errors during navigation usually will lead to wrong control instructions and inaccurate vehicle localization information, and it is desired to conduct a calibration task to eliminate such errors. Chen and Tsai [6] proposed a mechanical error correction method by curve-fitting the erroneous deviations from correct paths. In this study, we use this method for correcting the vehicle odometer readings.

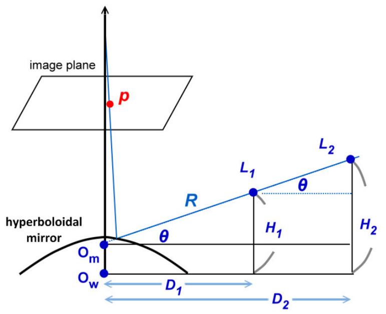


Figure 4. Finding out the focal point O_m.

III. 3-D DATA MEASUREMENT MEHTOD

In this study, we design a new type of omni-directional camera to achieve acquisition of environment images. Note that the two omni-cameras are connected and vertically-aligned in a bottom-to-bottom fashion, which means that none of the space points can be projected on both image planes at the same time. As a result, the most common stereo matching method that is based on the triangulation principle to obtain the relation between a 3-D point and multiple cameras does not work here at all. We need another theory to deal with the stereo matching problem here. One way proposed in this study is to estimate the horizontal distance to the desired object point, and subsequently to utilize both the distance and the elevation angle of the space point with respect to the local camera coordinate system CCS_{local} to get relevant 3-D data. As shown in Fig. 5, assume that both space points P_1 and P_2 lie on the vertical line L which is perpendicular to the x-y plane in the global coordinate system GCS and that P_1 is the intersection point of L and the x-y plane in the GCS. Assume also that P_1 and P_2 lie on the light rays with the azimuth-elevation angel pairs of (θ_1, ρ_1) and (θ_2, ρ_2) with respect to the upper camera coordinate system CCS_1 and the lower camera coordinate system CCS_2, respectively, where O_{m1} is the focal point of the lower mirror, and O_{m2} is the focal point of the upper mirror. Assume finally that the point O_{local} with coordinates (0, 0, 0) is the origin of the CCS_{local}. Referring to Fig. 5 and by the triangulation principle, we have

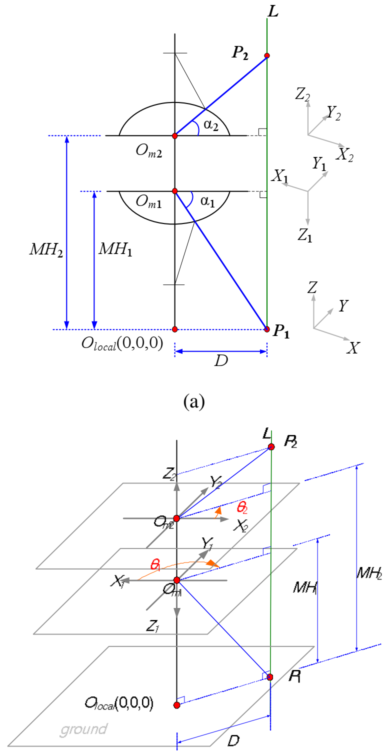
$$D = \frac{MH_1}{\tan \alpha_1}, \tag{3}$$

where D is the horizontal distance between O_{local} and P_1, and MH_1 is the height of the focal point O_{m1}. As a result, the height H_2 of the space point P_2 can be computed by

$$H_2 = MH_2 + D \times \tan \alpha_2 \tag{4}$$

where MH_2 is the height of the focal point O_{m2}.

According to the rotational invariance property of the omni-camera, we can derive the following equations:



(a)

(b)

Figure 5. (a) Computation od depth using the two-camera omni-directional imaging system. (b) Details of (a).

$$\cos \theta_1 = \frac{X_1}{\sqrt{X_1^2 + Y_1^2}} = \frac{u_1}{\sqrt{u_1^2 + v_1^2}}; \tag{5}$$

$$\sin \theta_1 = \frac{Y_1}{\sqrt{X_1^2 + Y_1^2}} = \frac{v_1}{\sqrt{u_1^2 + v_1^2}}; \tag{6}$$

$$\cos \theta_2 = \frac{X_2}{\sqrt{X_2^2 + Y_2^2}} = \frac{u_2}{\sqrt{u_2^2 + v_2^2}}; \tag{7}$$

$$\sin \theta_2 = \frac{Y_2}{\sqrt{X_2^2 + Y_2^2}} = \frac{v_2}{\sqrt{u_2^2 + v_2^2}}. \tag{8}$$

Because the vertical line L is perpendicular to the X_1-Y_1 and X_2-Y_2 planes and the positive direction of the X_1-axis in the CCS_1 is opposite to the positive direction of the X_2-axis in the CCS_2, it can be found that $X_1 = -X_2$ and

$$\cos \theta_1 = -\cos \theta_2; \ \sin \theta_1 = \sin \theta_2. \tag{9}$$

Now, we can utilize (3), (4), and (9) to calculate the coordinates $(X_{p_1}, Y_{p_1}, Z_{p_1})$ of P_1 with respect to the CCS_{local} as follows:

$$\begin{aligned} X_{p_1} &= -D \times \cos \theta_1 = -(MH_1 / \tan \alpha_1) \times \cos \theta_1; \\ Y_{p_1} &= \ \ D \times \sin \theta_1 = -(MH_1 / \tan \alpha_1) \times \sin \theta_1; \\ Z_{p_1} &= \ \ 0, \end{aligned} \tag{10}$$

and the coordinates $(X_{p_2}, Y_{p_2}, Z_{p_2})$ of P_2 with respect to the CCS_{local} as follows:

$$\begin{aligned} X_{p_2} &= D \times \cos\theta_2 = (MH_1 / \tan\alpha_1) \times \cos\theta_2 = -X_{p_1}; \\ Y_{p_2} &= D \times \sin\theta_2 = (MH_1 / \tan\alpha_1) \times \sin\theta_2 = Y_{p_1}; \\ Z_{p_2} &= H_2 = MH_2 + D \times \tan\alpha_2 \\ &= MH_2 + (MH_1 / \tan\alpha_1) \times \tan\alpha_2. \end{aligned} \tag{11}$$

IV. House Layout Construction

As mentioned previously, it is desired to conduct the house-layout construction task by driving the vehicle around the room and collect environment data. More specifically, we discuss the major stages of the house-layout construction task in the following.

A. Vehicle navigation by mopboard following

Most houses have mopboards at the roots of their walls. The mopboard feature is used for vehicle navigation in this paper. In this section, we describe the method we use to detect mopboard edges in omni-images and the entire navigation strategy.

According to the rotational invariance property of the omni-image, all points lying on each straight line L, which is parallel to the Z_1-axis in the CCS_1, are projected onto the image plane of the upper image coordinate system ICS_1 with the azimuth θ_1, forming a straight line in the ICS_1 and passing the origin I_1 of ICS_1, as shown in Fig. 6. And because mopboards occupy the bottom parts of walls, it results in a fact that the projected pixels of the mopboards in the omni-image also have an obvious band. We utilize these two properties to find out the mopboard edge points which are located on the edge between the mopboard and the floor.

Assuming that the house is empty, we can transform an acquired grayscale image into a binary one by a pre-defined threshold and perform erosion and dilation operations on it. In this way, the mopboard will remain in the image and the mopboard edge points can be detected by using a scanning line starting from the image center. An algorithm for the detection of the mopboard edges is described as follows.

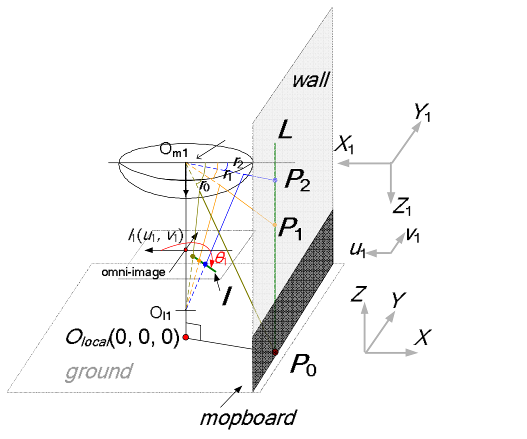


Figure 6. Vertiine L in the OCS_{local} and corresponding line l in the ICS_1

Algorithm 1: Mopboard detection

***Input*:** An omni-image I_k taken from the lower omni-camera; a desired scanning range; and a vehicle *mask*.
***Output*:** A set M_k of image coordinates of mopboard edges.
***Steps*:**

1) Transform I_k into a grayscale version I_G.
2) Reset the gray values g_{pi} of each pixel p_i in the gray image I_G by comparing g_{pi} with a pre-defined threshold T as follows:
   if $g_{pi} \geq T$, set g_{pi} as a white pixel;
   else, set g_{pi} as a black pixel.
3) Perform erosion on I_G to obtain a new image I_G'.
4) Perform dilation on I_G' to obtain a second new image I_G''.
5) For each scan line L_θ in a pre-selected angle scanning range with certain angular scanning steps, perform the following steps.
   a) Start from the image center of I_G'', traverse along the line L_θ, and find the first intersection black pixel p of I_G'' with L_θ.
   b) Check if p is followed by 10 or more consecutive black pixels along L_θ: if so, then add the coordinates of p into M_k (which is set empty initially).
6) Repeat Step 5 until the scan range is exhausted.

After detecting the mopboard edge points, we have a set M_k which includes the image coordinates of the detected mopboard edge points. Denote the set M_k in more detail by $M_k = \{(u_t, v_t)\}$, where (u_t, v_t) are the detected mopboard edge coordinates in the ICS_1. In order to estimate the corresponding locations of these detected edge points, we first choose an element (u_t, v_t) from M_k and calculate the corresponding radius r_t and azimuth θ_t by transforming (u_t, v_t) into the corresponding polar coordinates (r_t, θ_t). Then, we determine a column index of the pano-mapping table by finding out which *azimuth interval* the θ_t lies on. For example, if there exists i satisfying the following equation:

$$\theta_i = i \times (2\pi / M) \leq \theta_t < (i+1) \times (2\pi / M) = \theta_{i+1}, \tag{12}$$

then the column index is taken to be i. Also, we compare r_t with r_s in the entry E_{is} of the pano-mapping table, and keep the nearest radius r_j. As a result, the corresponding ρ_j can be found out. Once the ρ_j is known, the 3-D position estimation task can be carried out.

For vehicle navigation, the distance to the wall is important navigation information. The main idea is to navigate and keep the navigation path parallel to the wall. In this way, we choose three regions R_L, R_R, and R_F on the CCS_{local}, as shown in Fig. 7, so that the walls which are close to the vehicle can be detected. And we define the three sets S_L, S_R, and S_F by

$$\begin{aligned} S_L &= \{P_L \mid P_L \in M_k \text{ and } P_L \in R_L\}; \\ S_R &= \{P_R \mid P_R \in M_k \text{ and } P_R \in R_R\}; \\ S_F &= \{P_F \mid P_F \in M_k \text{ and } P_F \in R_F\} \end{aligned} \tag{13}$$

where the set M_k includes the detected mopboard edge points with coordinates in the CCS_{local}.

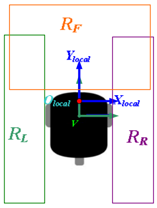


Figure 7. An illustration of the pre-defined detect regions.

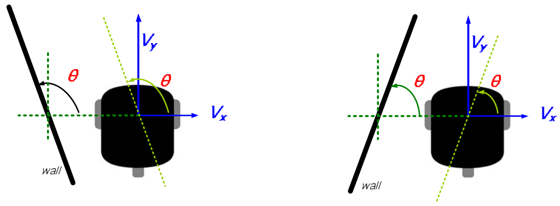
However, for example, the edge points which are on the left wall with respect to the vehicle may appear both in R_L and in R_F at the same time, resulting in the wrong distance estimation. In order to transform the detected mopboard edge point locations into useful navigation information and environment data, a *pattern classification* technique is proposed. Once we can assign each of the edge points to its corresponding wall by this classification technique, we can adjust the direction of the vehicle to keep the vehicle's navigation path parallel to each wall and estimate the distances to the nearby walls. Let the new sets derived from (13) after the pattern classification process be described as follows:

$$
\begin{aligned}
S_L' &= \left\{ P_L' \mid P_L' \in S_L \text{ and } P_L' \in W_L \right\};\\
S_R' &= \left\{ P_R' \mid P_R' \in S_R \text{ and } P_R' \in W_R \right\}; \qquad (14)\\
S_F' &= \left\{ P_F' \mid P_F' \in S_F \text{ and } P_F' \in W_F \right\}.
\end{aligned}
$$

As a result, the distances to each wall can be estimated more accurately, and are used as navigation information to guide the vehicle.

As mentioned above, the estimated positions of the detected mopboard edge points are derived in a raster-scanning order in the omni-image. Based on this order, the classification process would be simplified because any of these estimated points can be easily divided into two parts. The purpose of using a pattern classification technique in this study is to find an edge point locating at the corner of two adjacent walls that can divide the edge points into two parts. With a property that the adjacent walls are perpendicular to each other, we can conduct the pattern classification process by the LSE criterion to find such a corner point, if it exists. Let such a corner point, denoted as S_k, divides the set of edge points in different walls into two sets, denote as S_{k1} and S_{k2}. A reasonable assumption is made here, that is, the two lines which are used to fit the points in S_{k1} and S_{k2} by the LSE criterion, respectively, are perpendicular to each other. In this way, such a corner point will yield a minimum error.

Before driving the vehicle to go forward or to turn around, it is desired that the vehicle can keep its path to be parallel with the walls. Otherwise, the vehicle may hit the wall. For this purpose, the detected mopboard edges can be used, by a line fitting process, to generate a straight line which describes the direction of these detected edges. With the direction of the line, the vehicle can adjust its pose such that it can keep an appropriate distance with respect to the wall. The entire process is illustrated in Fig. 8.



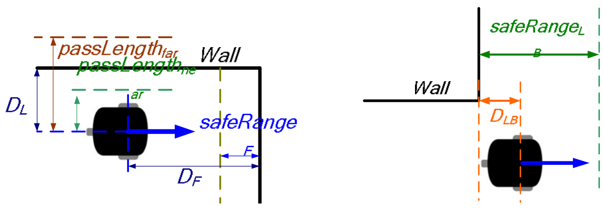
(a) (b)

Figure 8. Vehicle direction adjustment. (a) $\theta > 90^o$. (b) $\theta < 90^o$.

Before the vehicle starts a navigation session, it will estimate the distances to the nearby walls which are within the relevant angular scanning ranges. Based on the distances, we can know if it can turn to the left, turn to the right, or go forward, and then issue an appropriate control instruction to the vehicle to drive it to navigate safely. The major rules for controlling the vehicle are described as follows:

- For the work of the imaging system, keeping an appropriate distance with the wall is needed.
- The vehicle can go forward until the left wall is not detected further or until it is blocked by the frontal wall.

Both situations are illustrated in Fig. 9, and the algorithm of issuing a control instruction to the vehicle for each navigation cycle is described as follows. Note that the distance D_{LB} is assigned a value only under the condition that the vehicle exceeds the corner point, as shown in Fig. 9(b). For gathering enough information, it is desired that the vehicle goes forward a fixed distance $moveDist_{fix}$ if it is moveable, and then collect information. Refer to Fig. 9 for notations used in the algorithm. D_L, D_F, and D_R are the distances between the vehicle and the left wall, the frontal wall, and the right wall, respectively, if the corresponding walls are detected. The threshold values $passLength_{near}$ and $passLength_{far}$ are used to decide whether the vehicle can move forward or not.



(a) (b)

Figure 9. Illustration of the safe ranges. (a) Approaching to the frontal wall. (b) Exceeding the corner.

B. *Floor-layout construction*

As mentioned, it is reasonable to use two mutually perpendicular lines to fit the edge points which belong to adjacent mutually perpendicular walls. It means that if we choose one fitting line and adjust the direction of it, the directions of other fitting lines will change, too. As a result, using the fitting lines to fit the edge points will incur fitting errors. Based on this idea, we propose a global LSE optimization method to minimize the fitting error and the entire process is described in the following algorithm.

Algorithm 2: Floor-layout construction

Input: n sets of detected mopboard edge points, $S_1, S_2, \ldots, S_n$ of n walls, $W_1, W_2, \ldots, W_n$.

Output: A floor layout.

Steps:

1) (*Line fitting for each wall*) Fit the points in S_k of W_k with a line L_k by the LSE curve fitting scheme and compute its mean point M_k where $1 \le k \le n$.
2) (*Optimal fitting with respect to a chosen line*) Perform the following steps to obtain *globally* optimal fitting with respect to a selected fitting line of a certain wall.
   a) Choose a fitting line L_k, starting from $k = 1$ until $k = n$, and compute its direction angle θ_k.
   b) *Adjust* θ_k by adding a small angle ε (initially adding -10^{o} and then adding 0.1^{o} each time later until adding up to $+10^{o}$), resulting in θ_k', to generate a new line L_k' such that L_k' passes the point M_k with direction angle θ_k'.
   c) (*Generation of other fitting lines*) Generate a sequence of lines $L_{k+1}', L_{k+2}', \ldots, L_{(k+n-2) \bmod n}'$ with each L_{k+i}' perpendicular to its former line $L_{k+(i-1)}'$ and passing its original mean point M_{k+i} (i.e., every two neighboring lines are mutually perpendicular).
   d) (*Computing the sum of fitting errors of all lines*) Compute the error e_i of fitting all the points in S_i of W_i to line L_i' obtained in the last two steps (Steps b) and c) above), and sum the errors up to get a *total error* e_k' for L_k'.
   e) Repeat b) through d) until the range of angular adjustment, $(-10^{o}, +10^{o})$, is exhausted.
   f) Find the minimum of all the total errors e_k' and denote it as $e_{\min, k}$.
   g) Repeat Steps a) through f) to compute the $e_{\min, k}$ for all $k = 1, 2, \ldots, n$.
   h) Find the global minimum error $e_{\min, k_o}$ as the one which is the minimum of all the $e_{\min, k}$.
3) Take all the lines with adjusted angles corresponding to $e_{\min, k_o}$ as the desired floor layout.

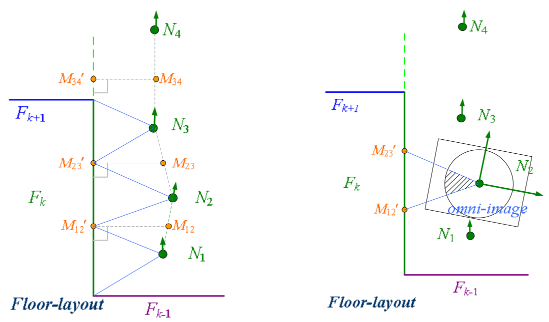
C. 3-D house-layout construction

Only creating a floor layout is insufficient for use as a 3-D model of the indoor room space. The objects on walls such as doors and windows must also be detected and be drawn to appear in the desired 3-D room model. Therefore, it is indispensible to analyze the omni-images taken by the imaging system to extract such objects. However, an omni-image covers a large range, but the further the distance is, the larger the estimation error is. How to retrieve the desired information is an important task. For this, we propose a method to determine a scanning range with two direction angles for each pair of omni-images based on the floor layout edge equation. With the scanning region for each omni-image, we can retrieve appropriate 3-D information from different omni-images. Note that each object which is detected by the scanning region of each omni-image is regarded as an individual one. Therefore, adjacent objects which appear in the consecutive omni-images taken from the same omni-camera (the upper or the lower one) must be combined into one based on, e.g., the information of their positions. Also, due to the configuration of the imaging system, some objects on the wall, such as windows and doors, may appear in the pair of omni-images (the upper and the lower ones). To solve it, we propose a method to recognize doors and windows from the combined objects.

As shown in Fig. 10, for each wall W_k, there are multiple navigation imaging spots $N_1, N_2, \ldots$, and N_m for taking omni-images as the environment information. In order to determine a scanning region for each omni-image taken at N_1 through N_m, we calculate the midpoint M_{ij} of N_i and N_j for all $i = 1, 2, \ldots, n - 1$ and $j = i + 1$. Then, we project each midpoint M_{ij} onto the line L_k which is the result of the floor-layout construction process mentioned previously, resulting in a projection point M_{ij}'. And based on the positions of the projection points and the navigation imaging spots, we can determine the cover ranges. With the cover range and the direction of the vehicle at the navigation spot, we can determine the scanning range with two direction angles for each pair of omni-images.

However, if we conduct the detection of objects within each scanning range, the mopboard will be detected as an object on the wall. Because the objects to be detected are doors and windows, and the mopboard is regarded as a feature of a wall, the mopboard should not appear in the detected objects. For this reason, we define a scanning region excluding the mopboard in the scanning range. In order to exclude the mopboard part in the scanning range, we use the space mapping table in a reverse way to estimate the image coordinates of a concerned space point at a known position.

More specifically, we use the distance to a detected edge point, and a pre-defined height of the mopboard to calculate the elevation of a point which is on the top edge of the mopboard with respect to the CCS_1. Then, we can look up the space mapping table to find its radius in the omni-image.



(a) (b)

Figure 10. Illustration of determining scanning range. (a) Decide the cover range. (b) Relevant scanning range on omni-image.

As a result, the mopboard part can be excluded in the omni-image, forming a desired scanning region. In this way, the object detection process can be carried out within this region.

With the above-mentioned scanning region, we can detect the objects in each omni-image by the proposed two-way angular scanning scheme which extends the method in Algorithm 1. The two-way scheme is used instead of the original one described in Algorithm 1 because the scheme can achieve a better estimation. We first perform Steps 1 through 4 of Algorithm 1 for the pair of images, then traverse along the line L_θ from the outer boundary to the inner one and from the inner to the outer boundary within the scanning regions, and find the first intersection black pixel which is followed by 10 or more consecutive black pixels, respectively. The first found black pixel in each direction will be regarded as an element of the boundary of the detected object. According to the two first scanned pixels which are detected by traversing along the scanning line with opposite directions, we utilize the information within the scanning line bounded by the two scanned pixels to determine whether the two pixels are boundary points of the object or not. Besides, objects may occupy the omni-image and are detected for some certain continuous angular interval. We utilize the property to combine the detected objects which are detected from certain continuous scanning lines into an individual one. In this way, there may be some individual objects in the scanning region. As a result, according to these pixels which are boundary points of the object, and by utilizing the 3-D position estimation method by looking up the space mapping table, the average heights at the bottom and the top of the object can be estimated.

There are some major rules which may be adopted for combing the detected objects.

- First, we traverse along the scanning line L_θ within the scanning region R' from opposite directions and find the pixels p_i and p_o of the object boundaries, respectively.
- By counting the numbers n_b and n_{sum} of black pixels and all pixels between the two detected pixels along L_θ, if found, and by a threshold and the ratio n_b / n_{sum}, we can determine whether the pixels p_i and p_o are on the boundary of the object or not.
- In R', if all the pixels p_i and p_o for each scanning line L_θ in an angular interval are all on the boundary of an object, then such a largest interval can used to describe the same object. Besides, in R', those combined objects may be combined again according to their positions.

With the above rules, all individual objects can be determined in the same scanning region of an omni-image.

However, an object on a wall may appear to cross two or more scanning regions of corresponding source omni-images taken by the same upper or lower omni-camera. Besides, an object also has a possibility to appear in the pairs of omni-images simultaneously. For the above reason, we have to combine those detected individual objects which belong to the same one. Here, we denote O_1 and O_2 as the sets of the individual objects detected from the omni-images taken by the lower and upper omni-cameras, respectively. As shown in Fig. 11, at first, we conduct the combination task for all the objects in O_1 and O_2, according to their positions to form two new sets O_1' and O_2', respectively. Then, we conduct the reorganization task. The process is described in the following algorithm.

Algorithm 3: Objects reorganization

Input: O_1' and O_2' including the combined objects on walls.
Output: Window objects set O_W, and door objects set O_D.
Steps:

1) (*Objects recognition for each wall*) For each floor-layout edge F_k, perform the following steps to recognize the objects on wall W_k.
   a) Choose an object $o_{2,i}$ from O_2', to find an object in O_1' at a similar location.
      i. If such object $o_{1,j}$ is found, then check if it is connected to the mopboard by its location:
         1. if yes, then recognize $o_{1,j}$ together with $o_{2,i}$ as a door, and add it to O_D;
         2. otherwise, recognize $o_{1,j}$ together with $o_{2,i}$ as a window, and add it to O_W.
      ii. If such an object is not found, recognize $o_{2,i}$ as a window, and add it to O_W.
   b) Repeat Step (a) until the objects of O_2' are exhausted.
   c) Recognize the remaining objects in O_1' as widows, and add it to O_W.
2) Repeat Step (1) until all the floor-layout edges are exhausted.

V. EXPERIMENTAL RESULTS

We show some experimental results of the proposed automatic house-layout construction system by autonomous vehicle navigation based on mopboard following in indoor environments. The experimental environment was an empty indoor room with mopboards at the bottom of the walls. Fig. 12 shows an example of the resulting images of mopboard detection in which the red dots show the detected mopboard edge points and the result of the pattern classification procedure. Fig. 13 shows the estimated mopboard edge points of all walls and the floor layout which is constructed by the globally optimal fitting method. In Table 2, we show the errors in percentage between the actual widths of the walls and the estimated data of 9 times of navigations using the proposed system. The average error of the wall widths is 2.75% and the average error in percentage of the estimated total perimeter of the floor layout with respect to the real data is 0.21%. We simulated a window in our experimental environment by creating a black frame and attaching it on a wall. The window is not so low that only the upper omni-camera can "see" it. An example of the images of the door and window and the detection results are shown in Fig. 14. An example of house-layout construction in graphic form is shown in Fig. 15.

VI. CONCLUSSIONS

A system for automatic house-layout construction by vision-based autonomous vehicle navigation in an empty room space has been proposed. To achieve acquisition of images, a new type of omni-directional camera has been designed for this study, which consists of two omni-cameras aligned coaxially and back to back. The proposed automatic house layout construction process consists of three major stages: (1) vehicle navigation by mopboard following; (2) floor layout construction; and (3) 3-D house layout construction. he entire house layout construction process is fully automatic, requiring no human involvement, and so is very convenient for real applications. The experimental results show the feasibility of the proposed method.

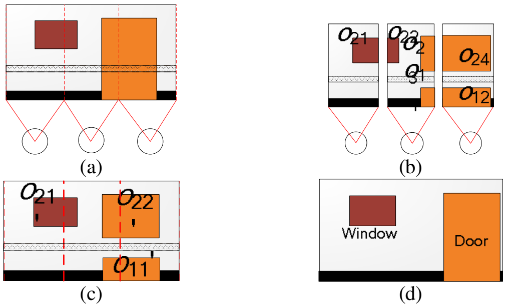


Figure 11. Illustration of object combinations. (a) Scanning regions. (b) Individual objects. (c) Combined objects for each omni-camera. (d) Reorganization of objects.

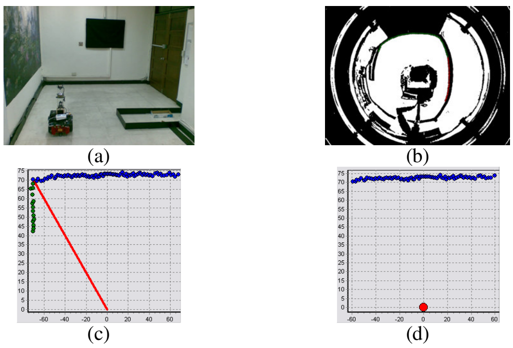


Figure 12. Environment data collection. (a) Experimental environment. (b) Detected mopboard edge points. (c) The detected mopboard points. (d) Result of the classification (the points belonging to the upper wall).

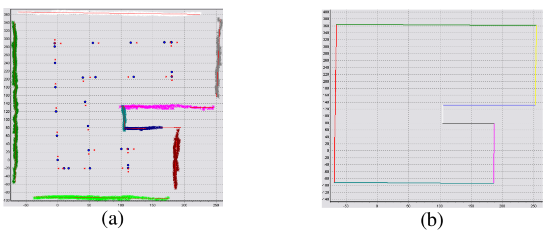


Figure 13. Illustration of global optimization. (a) Etimated mopboard edge points of all walls. (b) A floor layout fitting the points in (a).

Table 2. Precision of estimated wall widths and their error percentages.

Exp. No.	Average estimated wall width error %	Estimated perimeter error %
1	2.345	0.36
2	2.586	0.03
3	2.948	0.02
4	3.129	0.35
5	2.090	0.37
6	3.137	0.28
7	2.387	0.09
8	2.749	0.23
9	3.335	0.13
average	2.745	0.21

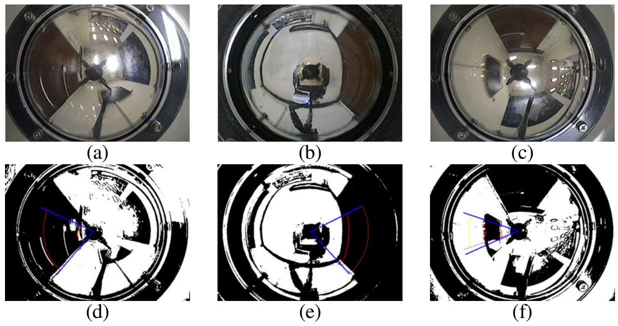


Figure 14. Images and door detection result. (a) Image of the door taken by the upper omni-camera. (b) Image of the door taken by the lower omni-camera. (c) Image of the window taken by the upper omni-camera. (d) Door detection result of (a). (e) Door detection result of (b). (f) Window detection result of (c).

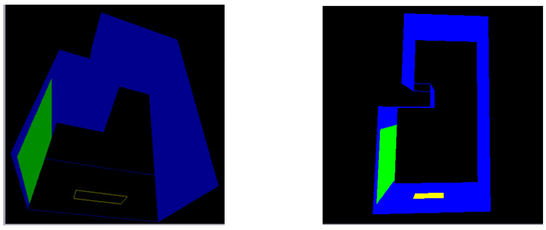


Figure 15. Graphic display of constructed house layout. (a) Viewing from the top (green rectangle is a door and yellow one is a window). (b) Viewing from the back of the window

REFERENCES

[1] A. Ohya, A. Kosaka, and A. Kak, “Vision-based Navigation by a Mobile Robot with Obstacle Avoidance Using Single-Camera Vision and Ultrasonic Sensing,” *IEEE Transactions on Robotics and Automation*, Vol. 14, No. 6, pp. 969-978, 1998.

[2] J. Gluckman, S. K. Nayar, and K. J. Thoresz, “Real-Time Omnidirectional and Panoramic Stereo,” *Proceedings of DARPA98*, pp.299–303, 1998.

[3] B. S. Kim, Y. M. Park and K. W. Lee, “A experiment of 3D reconstruction using laser range finder and CCD camera,” *Proceedings of IEEE 2005 International Geoscience and Remote Sensing Symposium* , Seoul, Korea, pp.1442-1445 , July 25-29, 2005.

[4] S. Kim and S. Y. Oh, “SLAM in Indoor Environments using Omni-directional Vertical and Horizontal Line Features,” *Journal of Intelligent and Robotic Systems*, v.51 n.1, p.31-43, January 2008.

[5] S. W. Jeng and W. H. Tsai, “Using Pano-mapping Tables for Unwrapping of Omni-images into Panoramic and Perspective-view Images,” *Journal of IET Image Processing*, Vol. 1, No. 2, pp. 149-155, June 2007.

[6] M. C. Chen and W. H. Tsai, "Vision-based Security Patrolling in Indoor Environments Using Autonomous Vehicles, "*Proceedings of 2005 Conference on Computer Vision, Graphics and Image Processing,* pp. 811-818, Taipei, Taiwan, Republic of China, August 2005

[7] C. J. Wu and W. H. Tsai, “Location Estimation for Indoor Autonomous Vehicle Navigation by Omni-Directional Vision Using Circular Landmarks on Ceilings,” *Robotics and Autonomous Systems*, Vol. 57, No. 5, pp. 546-555, May 2009.