Map-Based Navigation for a Mobile Robot with Omnidirectional Image Sensor COPIS

Yasushi Yagi, *Member, IEEE*, Yoshimitsu Nishizawa, and Masahiko Yachida

Abstract— **We designed a new omnidirectional image sensor COPIS (COnic Projection Image Sensor) to guide the navigation of a mobile robot. The feature of COPIS is passive sensing of the omnidirectional image of the environment, in real-time (at the frame rate of a TV camera), using a conic mirror. COPIS is a suitable sensor for visual navigation in a real world environment.**

We report here a method for navigating a robot by detecting the azimuth of each object in the omnidirectional image. The azimuth is matched with the given environmental map. The robot can precisely estimate its own location and motion (the velocity of the robot) because COPIS observes a 360° view around the robot, even when all edges are not extracted correctly from the omnidirectional image. The robot can avoid colliding against unknown obstacles and estimate locations by detecting azimuth changes, while moving about in the environment. Under the assumption of the known motion of the robot, an environmental map of an indoor scene is generated by monitoring azimuth change in the image.

I. Introduction

WHILE there has been much work on mobile robots with vision systems which navigate in both unknown and known environments [1]–[4], most robots view things only in front of them and avoid static obstacles and as a result, they may collide against objects moving from the side or from behind. To overcome these drawbacks, a sensor is needed to view the environment around the robot so that it may navigate safely.

Attempts have been made to acquire omnidirectional information on the environment, including acoustic sensing, passive and active vision.

An acoustic sensor can easily acquire a depth map of the environment around the robot. Moravec [5] developed the mobile robot with a number of Polaroid sonar ranging devices around its body. Many researchers have also proposed methods for analyzing range data, including hierarchical representation of a 2-D map for integrating sonar data from several viewpoints [6]–[8]. However, as ranging resolution of acoustic sensor is poor, it is difficult to measure locations of the object precisely. Thus, the Kalman filter is often applied for modeling the uncertainty of position measurement by the acoustic sensor [6]–[9]. It can be difficult to measure precise locations of the object outside of a certain range of the relative direction of object's surface because the intensity of the reflected rays depends on the surface direction relative to the sonar receiver.

Use of an active sensor provides a precision which is higher than that of the acoustic method. Gabriel *et al.* [10] investigated omnidirectional range imaging by rotating a time-of-flight range finder. Application of this to real-time problems is not easy because of the long (about 1 s) imaging time. Jarvis and Byrne [11] proposed another real-time omnidirectional active sensing method using a conic mirror and sheets of light, however, applications have not been reported because the image is not of a sufficient quality to be useful.

Passive imaging methods using a rotating camera [12], [13], a fish-eye lens [14], [15], a conic mirror [16] or a spherical mirror [17] have been used to obtain omnidirectional views of the environment. Although precise azimuth information is available in the omnidirectional view obtained using a rotating camera, the time-consuming imaging prevents application to real-time problems. A fish-eye lens provides for a wide view of a semi-sphere around the camera. However, the ground (floor) and objects on it appear along the boundary of the circular image. Because of the poor image resolution, the imaging is inadequate for monitoring objects around the robot. A conic mirror provides an image of the environment around it and a 360° view is attainable. Imagery taken by a spherical mirror provides a similar omnidirectional view of the environment and is used to locate the robot on route [17]. A disadvantage of the spherical mirror is that the resolution along a radial direction of spherical mirror is poor because the view angle along vertical direction is too wide; structures in an environment with walls and doors in a room appear along the circular boundary of the image, therefore, it is difficult to extract edge segments from the image as lengths of edge segments are short.

We have proposed a new omnidirectional image sensor COPIS (COnic Projection Image Sensor) for guiding navigation of a mobile robot [16], [19]. The key feature of COPIS, developed independently of the work of Jarvis and Byrne [11], is passive sensing of the omnidirectional environment in real-time (at the frame rate of a TV camera), using a conic mirror. Mapping of the scene onto the image by COPIS involves a conic projection and vertical edges in the environment appear as lines radiating from the image center. The field viewed by COPIS is limited by the vertex angle of conic mirror and the visual angle of the camera lens. Therefore, the main field

Manuscript received September 13, 1993; revised April 1, 1994.

Y. Yagi and M. Yachida are with the Department of Systems Engineering, Faculty of Engineering Science, Osaka University, Toyonaka, Osaka 560 Japan.

Y. Nishizawa is with the Department of Information and Computer Science, Faculty of Engineering Science, Osaka University, Toyonaka, Osaka 560 Japan.

IEEE Log Number 9411916.

viewed by COPIS is a side view and the resolution along a radial direction of conic mirror is sufficient to extract vertical edges.

We reported a method of collision avoidance, which is based on azimuth information in the image sequence [18], [19]. Since both robot and objects move in the environment, estimation of their locations and motions using a single camera is difficult. We assume the constant linear motions of objects and the robot, and estimate directions of the object velocities relative to the robot from the loci of object points in the image. Potentially dangerous objects are detected by observing their azimuth changes. The robot decelerates so as to avoid collision and it can also estimate location and motion of the object by the relative velocity change. Thus, COPIS is a suitable sensor for visual navigation in a real world environment.

For safe navigation, the robot must be able to avoid collision against objects and be able to estimate its own location in an environment.

There are two types of navigation; the robot moves in a known or an unknown environment. In a known environment, the robot can be navigated speedily and effectively using not only visual information from the input image but also by making use of the given map of a known environment. Theoretical and practical maps based on methods of navigation have been proposed [4], [20], [21]. Yachida *et al.* used prior knowledge on the environment for both navigation and monitoring [4]. While the robot is moving around, it stops every few meters, takes images and analyzes the images to find obstacles and estimate its own location. With their method, the vertical edges extracted in the image are matched with ones on the image model generated from the 3-D model. However, only the front region of the robot is observed and it may not avoid danger from the side or behind. In 1988, Sugihara reported a theoretical analysis concerning the possibility of finding the robot's own location [20]. He assumed that all vertical edges are identical and that the problem was to search for the point where the image is taken by establishing correspondence between the vertical edges in the image and those in the map. This is the general case for estimating the robot's location, but, a long computation time is needed.

We describe here a method for navigating the robot with a given environmental map; generation of an environmental map and estimation of location of the robot by matching observed vertical edges with a given map.

Under the assumption of known motion of the robot, locations of objects around the robot can be estimated by detecting their azimuth changes in the omnidirectional image [22]. One can recognize the surface of objects and estimate the free space for a mobile robot from the geometrical relation between object points in the image [23]. Using these methods, the robot generates an environmental map of an indoor scene while it is moving in the environment.

Since the environmental map is given, the location and the motion of the robot can be estimated by detecting the azimuth of each object, in the omnidirectional image [24]. We assume that the robot is initially parked at a standard position, driven around a room and a corridor of the building via a given route and rough movement of the robot can be measured using an internal sensor. There are, however, measurement errors caused by swaying motion of the robot. The sway motion we refer to in this paper means unobservable errors in orientation when tires slip on slightly rough ground. By matching azimuth information from both the input image and the environmental map in a predicted search region, the robot can always estimate its own location and move precisely because COPIS observes a 360° view around the robot, even if all edges are not extracted correctly from the omnidirectional image. We also present a method to estimate locations of unknown obstacles by detecting their azimuth changes while the robot is moving in the environment and can avoid collisions.

Experiments done in an actual environment support the validity of our method.

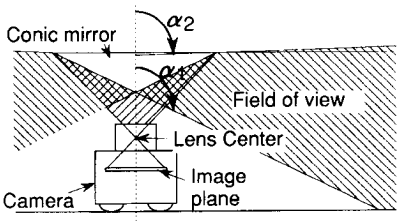


Fig. 1. Visual field of COPIS.

II. Conic Projection Image Sensor (COPIS)

A. Sensor Configuration

COPIS mounted on a robot consists of a conic mirror and a TV camera, with its optical axis aligned with the cone's axis, in a cylindrical glass tube, as shown in Fig. 1. By setting the axis of COPIS vertical, a 2π view around the robot is attained. The visual field covers between two view angles α_1 and α_2, which are determined from the visual angle γ of the camera and the vertex angle δ of the conic mirror as

$$\alpha_1 = \delta,$$
$$\alpha_2 = \delta - \tfrac{1}{2}\gamma. \tag{1}$$

COPIS provides an omnidirectional view of the floor and objects above it.

B. Conic Projection

COPIS maps the scene onto the image plane through a conic mirror and a lens. We call this mapping "conic projection." We describe here how features of the scene appear in the image.

Let us use the three-dimensional coordinate system O-XYZ, aligned with the image coordinate system o-xy and the Z-axis pointed toward the cone's vertex (see Fig. 2). We fix origin O at the camera center, thus the image plane is on a level of f, where f is the focal length of the camera.

A conic mirror yields the image of a point in space on a vertical plane through the point and its axis. Thus, the point P at (X, Y, Z) is projected onto the image point p at (x, y) such that

$$\tan\theta = \frac{Y}{X} = \frac{y}{x}. \tag{2}$$

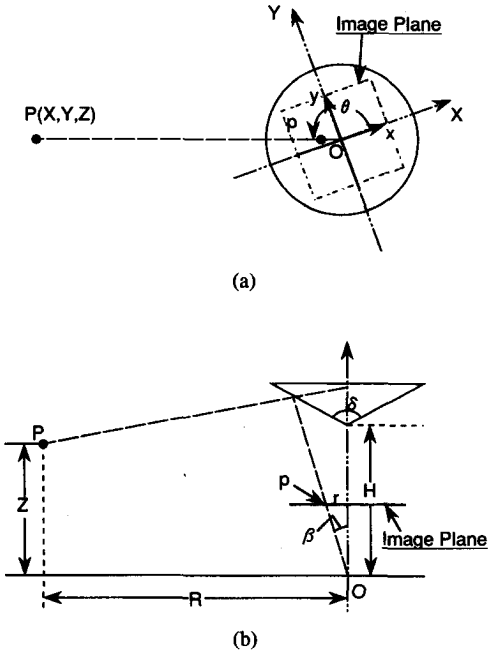


Fig. 2. Invariant relation of azimuth angle. (a) Invariability of azimuth angle in COPIS. (b) Linear relation of tilt angle in COPIS.

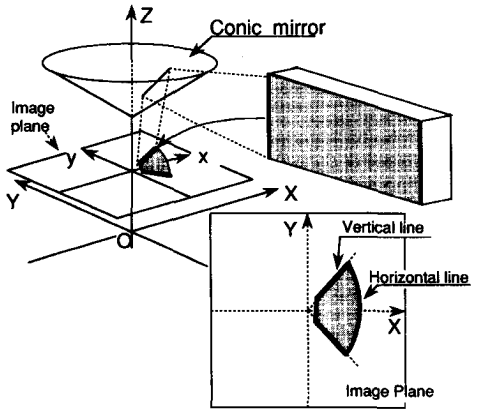


Fig. 3. Conic projection of 3-D lines.

In other words, all points with a same azimuth in space appear on a radial line through the image center and its direction θ indicates the azimuth.

The vertical lines provide a useful cue in a man-made environment such as a room, containing many objects with vertical edges; for example, doors, desks and shelves. If a line is vertical, it appears radially in the image plane, as shown in Fig. 3. Thus, COPIS can easily find and track the vertical edges by searching consecutive images for radial edges. The locations of the robot and objects are estimated by monitoring only loci of azimuths of the 3-D vertical lines in the image. For further details of COPIS optics, we refer to reader to our preceding report [19].

III. Assumptions

The following properties of the environment and the mobile robot are assumed for image analysis.

The floor is almost flat and horizontal while walls and static objects such as desks or shelves have vertical planes. The robot moves in a man-made environment such as a room or a corridor. Its motion parameters are two translational components (U, V) and one rotational component φ (pan angle).

IV. Generation of an Environmental Map

A. Principle of Location Measurement

Let us denote the robot motion by $[u(t), v(t)]$ and $\varphi(t)$. $\varphi(t)$ is the pan angle at time t. Defining the position of P at time $t = 0$ by $P_1[X(0), Y(0), Z(0)]$, the relative velocity of the point P in the environment at time t is represented by $[-u(t), -v(t), 0]$. We get the location of point P at time t as

$$\begin{aligned} X(t) &= \int_{\tau=0}^{t} -u(\tau)\,d\tau + X(0) \\ Y(t) &= \int_{\tau=0}^{t} -v(\tau)\,d\tau + Y(0) \\ Z(t) &= Z(0). \end{aligned} \tag{3}$$

Thus, from (2) and (3), the relation between an azimuth angle of an object and time t is obtained as follows,

$$\tan[\theta(t) - \varphi(t)] = \frac{\displaystyle\int_{\tau=0}^{t} -v(\tau)\,d\tau + Y(0)}{\displaystyle\int_{\tau=0}^{t} -u(\tau)\,d\tau + X(0)}. \tag{4}$$

The locus of the azimuth angle in consecutive image is represented by (4). Thus, if the azimuth angle θ is observed at two points, the relative location between the robot and the object point is calculated by triangulation as follows,

$$\begin{bmatrix} X(0) \\ Y(0) \end{bmatrix} = \frac{1}{\tan\theta'(2) - \tan\theta'(1)} \begin{bmatrix} -1 & 1 \\ -\tan\theta'(2) & \tan\theta'(1) \end{bmatrix} \cdot \begin{bmatrix} V(1) - U(1)\tan\theta'(1) \\ V(2) - U(2)\tan\theta'(2) \end{bmatrix} [\tan\theta'(1) \neq \tan\theta'(2)]$$

$$\theta'(t) = \theta(t) - \varphi(t)\,(t = 1, 2)$$

$$U(t) = \int_{\tau=0}^{t} -u(\tau)\,dt,$$

$$V(t) = \int_{\tau=0}^{t} -v(\tau)\,d\tau \quad (t = 1, 2) \tag{5}$$

where $\theta(1)$ and $\theta(2)$ are the azimuth angle θ after time $t = 1$ and 2, respectively. In case of $\theta(1) = \theta$ (2), the matrix of (5) is singular. Now, when the object lies on the axis of the robot's motion, the conditions of $\tan\theta'(1) \neq \tan\theta'(2)$ are not satisfied. In this case, it is impossible to calculate the location. However, as the object has a certain size, it is unlikely that all points on the object move on the same axis as the robot movement. Therefore, at least, locations of a few points on the object can be calculated. The matching method in consecutive images is described in Section VI-B.

The azimuth angle has an observational error due to the swaying motion of the robot. This swaying motion may change the pan, tilt and roll angles of the camera. While the tilt and

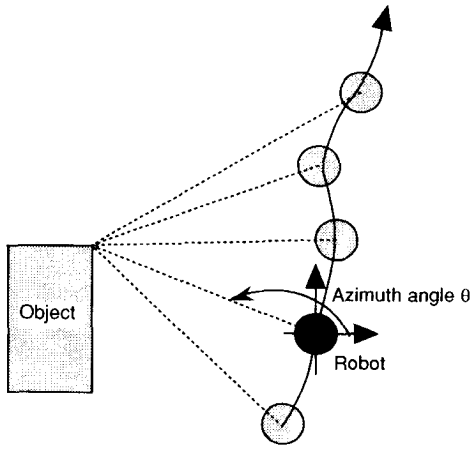


Fig. 4. Estimation of location of an unknown object.

roll angles have little influence on the observed azimuth angle, the pan angle has a strong effect on the angle. Therefore, as shown in Fig. 4, we estimate the more precise location using consecutive measurements, by the least squares method. Mathematically, if we define the squared error $d(t)$ of the least squares method using (4) and (5), values of location $[X(0), Y(0)]$ can be found by solving the following partial differential equations.

$$\begin{aligned} d(t) &= \{[U(t) - X(0)] \sin\theta'(t) \\ &\quad - [V(t) - Y(0)] \cos\theta'(t)\}^2\theta'(t) \\ &= \theta(t) - \varphi(t) \end{aligned}$$

$$\frac{\partial}{\partial X(0)} \sum_{\tau=0}^{n} d(t) = 0, \quad \frac{\partial}{\partial Y(0)} \sum_{\tau=0}^{n} d(t) = 0. \tag{6}$$

In a real environment, there are objects with vertical edges and also curved objects. As the method proposed in this paper is based on vertical edges that often appear indoors, it cannot estimate locations of curved objects with no vertical edges such as cubes. However, such objects rarely appear indoors. Many curved objects, which often appear in a man-made environment, are cylindrical ones such as pipes and extinguishers. Apparent vertical edges appear on both sides of the cylindrical object. We call the apparent edge the extremal boundary. When the cylindrical object is viewed from two different camera positions, the location of its extremal boundaries can be estimated by triangulation. The estimated location has measurement error because the intersecting point of two lines of sight is not actually on the extremal boundary, as shown in Fig. 5. However, this gap between the intersecting point and the boundary is small and the obtained precision of location of cylindrical objects can be considered to be sufficient for robot navigation.

B. Estimation of Free Space

The location data derived from an edge-based algorithm yields a wire frame description of the scene, i.e., a set of vertical line segments in three-dimensional space. However, the navigation needs information to determine the free space and its boundary such as surfaces and walls. Faugeras *et al.* proposed the representation of free space from three-dimensional data obtained by stereo vision [25]; the data are obtained from several positions of view. Their method is based on the use of constrained Delaunay triangulation, and the data are represented by polyhedral surfaces. After moving the camera's position, new data are merged with previous data and the representation is updated. As this method takes time, it is not applicable to real time navigation.

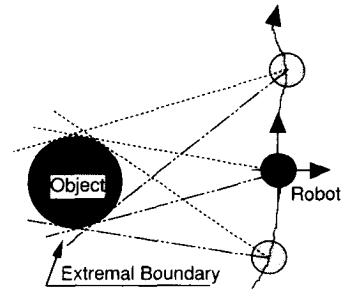


Fig. 5. Extreme boundary of cylindrical object

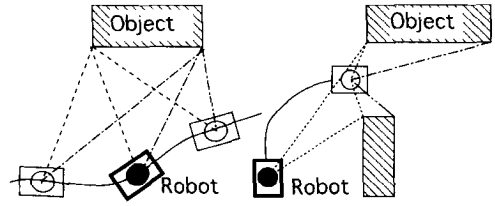


Fig. 6. Azimuth angle relation. (a) Case of object surface. (b) Case of passage.

One can estimate the minimum set of free space for a mobile robot from the geometrical relation between object points in the image. Both sides of edges of the object surface are contiguous in the image and have an invariant relation with the robot motion, as shown in Fig. 6(a). On the other hand, in case of passage, other edges appear between contiguous ones with movement of the robot, as shown in Fig. 6(b). Thus, by using these relations, one can predict the object surfaces and estimate the free spaces. The basic algorithm is made according to the following arrangements:

STEP 0: The obtained azimuth angle data at the first frame are put in order of the angular size and stored in a data file. The data file is a two-dimensional array as shown in Fig. 7, where the row means each edge and the obtained location data (x, y) are stored in the first two columns. The azimuth angles in each frame are stored after the third column. For instance, the first frame data (10, 30, 50, 80, $\cdots$) are stored at the third column in the data file.

STEP 1: After the second frame, the obtained azimuth angle data are also put in order of the angular size, then the edge data matched with the previous frame are stored at each row in the data file. When the unmatched edge datum appears, the new row is sorted between two rows, including both sides of the adjacent edges to the unmatched edge in the

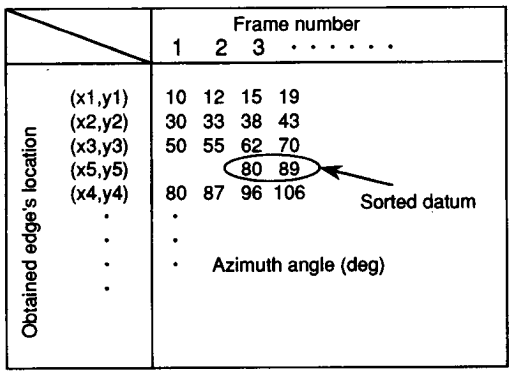


Fig. 7. Data structure of azimuth angle in consecutive images.

current frame. For instance, angle data {15, 38, 62, and 96} at the 3rd frame are matched with the angle data {12, 33, 55, and 87} at the 2nd frame. However, since the angle datum {80} at the 3rd frame is not matched, this datum is sorted between {62} and {96}.

STEP 3: The free space is estimated by connecting the vertical edges according to the adjacent relation, as shown in Fig. 8(a). Thus, data on location loaded from the first two columns are only connected according to the row number. The final datum on location is connected to the first datum.

As shown in Fig. 8(b), the new candidates of surfaces are merged with the previous ones and the free space is expanded while the robot moves. However, it cannot guarantee whether the candidate surface is a true one as the free space is estimated by only connecting the vertical edges according to the adjacent relation. This free space means space in which the robot can move freely. Therefore, there is no problem when the robot moves inside this minimum free space. However, as shown in Fig. 8(c), if the robot moves toward its goal position that is the outside of this minimum free space, the robot cannot go through the passage because the passage (the inside of region between two objects) is estimated as the surface (wall). This problem is a common one related to the edge-based algorithm that yields a wire frame description of the scene. Generally, when the robot moves parallel and closer between two objects in the environment in which the background is constant, the inside of the region between two objects is estimated as the wall. Therefore, one needs to use not only the edge-based algorithm but also a surface-based one with range data from an acoustic sensor or an active vision which can measure location of the inside of the surface.

An acoustic sensor can readily acquire a depth map of the environment around the robot. As mentioned in introduction, since the ranging resolution of the acoustic sensor is poor and the intensity of the reflected rays depends on the surface direction relative to the sonar receiver, it is difficult to measure precise locations of the object.

We integrate the acoustic sensor with COPIS; we estimate precise locations of vertical edges and a minimum set of free space, using COPIS. When the robot finds the candidate surface along this path, candidates of surfaces are verified

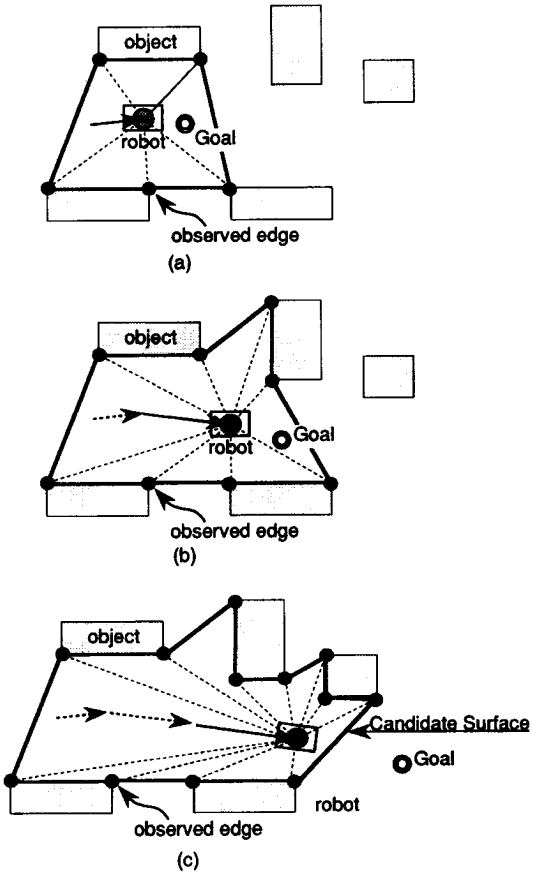


Fig. 8. Estimation of free space.

using the acoustic sensor. The acoustic sensor need not scan the whole region of the candidate surface because the surface region is estimated by COPIS and the purpose of the acoustic sensor is only to verify whether or not the candidate surface is a true one. Therefore, the acoustic sensor measures distance at only one point on the candidate. Furthermore, the intensity of the reflected rays depends on the surface direction relative to the sonar receiver. To solve this problem, the robot changes its direction perpendicular to the surface direction when it moves closer to the candidate surface. Let us denote the distance from the robot to the candidate surface by L. This distance L can be estimated by COPIS. If the candidate surface is a false one (empty), the range data from ultrasonic sensor is continuously higher than the distance L when the robot moves toward the candidate, thus, the robot can go through there. When the candidate is the true one, the measured range is similar to the distance L; then the robot changes its path and avoids collision against the true surface. Therefore, one can verify whether the candidate surface is true or false using ultrasonic range data. For further details, see our preceding report [26].

V. Navigation Algorithm with a Given Map

Fig. 9 shows an outline of the navigation algorithm. The robot is initially parked at a standard position and is driven around a room and a corridor of the building via a given route. The robot is equipped to know the standard position (starting position) and its own movement; however, there

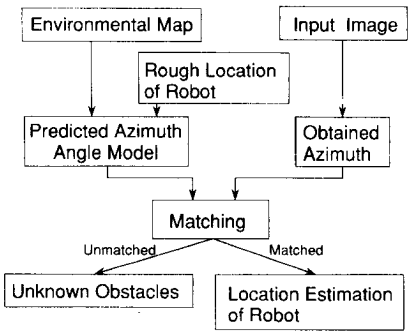


Fig. 9. Outline of navigation algorithm.

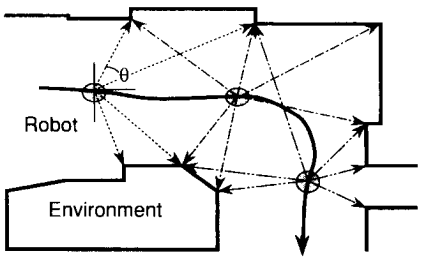


Fig. 10. Estimation of location of robot.

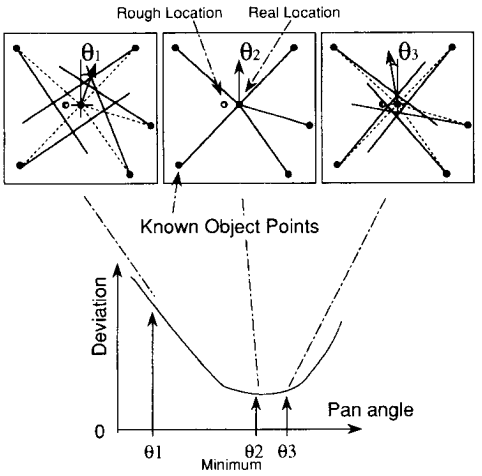


Fig. 11. Process of estimating pan angle of robot.

are measurement errors caused by swaying motion. A rough location of the robot can be calculated from the starting position and from movement. Thus, the azimuth angles of vertical edges are estimated using the given map and rough location. Using azimuth information from both the input image and the environmental map, one can estimate location and motion of the robot. After matching the observed edges with the environmental map, the undetectable vertical edges are recognized as unknown obstacles and their locations are estimated.

When there is an obstacle along a path toward a given goal position, the robot has to change its path. By evaluating the distance between the robot and the goal position, the robot plans a new minimum length path without colliding against the obstacle and moves toward the goal position.

A. Location and Motion Estimation of the Robot

Essentially, the location of a mobile robot can be defined by the azimuth angle θ, as shown in Fig. 2. Thus, as illustrated in Fig. 10, if three and more azimuth angles of the object are observed in the given environmental map where the robot moves, the location of the robot is calculated by matching the obtained azimuth angles with the environmental map. Actually, one can estimate a more precise location using the least squares method.

When (X_i, Y_i) are locations of objects in the given map at time t, the azimuth angle of the vertical edge is represented by

$$\tan\left[\theta(i) - \varphi(t)\right] = \frac{Y_i + Y(t)}{X_i + X(t)}. \tag{7}$$

Thus, the locations $[X(t), Y(t)]$ of the robot are estimated by solving the following partial differential equations. $\theta(i)$ is the observed azimuth angle of the vertical edge.

$$e(i) = \{[X_i + X(t)] \sin\theta(i) - [Y_i + Y(t)] \cos\theta(i)\}^2$$

$$\frac{\partial}{\partial X(t)_{i=0}} \sum_{i=0}^{m} e(i) = 0, \quad \frac{\partial}{\partial Y(t)_{u=0}} \sum_{i=0}^{m} e(i) = 0. \tag{8}$$

$e(i)$ is the squared error of the least squares method, and m is the number of vertical edges which can match the given map. However, the pan angle $\varphi(t)$ of the robot is an unknown parameter. Thus, we have to know the precise pan angle to estimate a more precise location of the robot. By changing the pan angle of the robot in a certain margin of error with the swaying motion of the robot, one can find the precise pan angle when the deviation of the least square method becomes minimum. The location of the robot can then be detected. Fig. 11 shows the process of finding the minimum deviation by changing the pan angle. In this work, we changed the pan angle every 0.5°, in the range of 10°.

The motion of the robot can be estimated by measuring its location in consecutive images. As COPIS can take an omnidirectional image, this system can estimate its location and motion even when it is turning.

B. Matching Radial Lines to Predicted Azimuth Angle Model

In this section, we describe the matching method between azimuth angles from the input image and the environmental map. Fig. 12(a) shows an example of the environmental map of a two-dimensional model viewed from the vertical direction. If the rough location of the robot is known, one can generate a predicted azimuth angle model from the environmental map, as shown in Fig. 12(b).

Next, we describe how rough location of the robot can be obtained. The rough location is given at the starting position. When the robot moves, the rough location can be calculated by adding the robot movement from the encoder of the robot and the estimated location at the prior frame. As the locus of the azimuth angle in consecutive images is represented by (4),

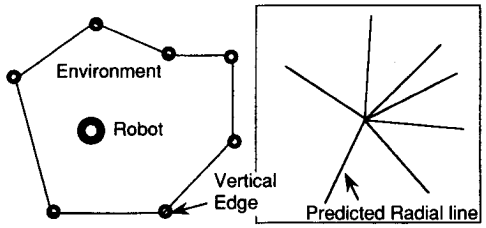


Fig. 12. Environmental map and prediction of the azimuth angle.

the azimuth angle at the current frame can be predicted using the rough location of the robot. This predicted azimuth angle of each edge is then compared with the azimuth angle of the radial line obtained from the input image.

As the azimuth angle at the current frame can be predicted, one can set a narrow search region around this predicted azimuth angle of the vertical edge, which is an estimated position from the environmental map. As a radial line is sought in a narrow region, reliable correspondence can be established.

C. Estimation of Unknown Obstacles

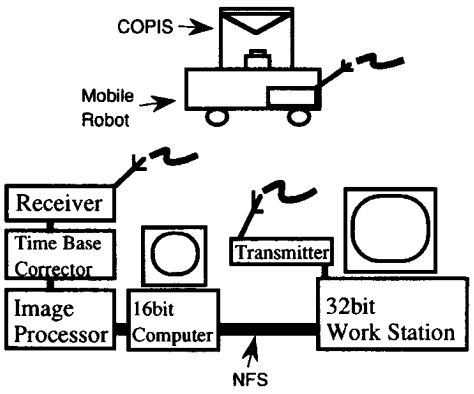
After matching the observed objects with the environmental map, the objects which are undetectable in the environmental map are recognized as unknown ones. If there are edges which have not been matched to the environmental map, the robot considers that they are caused by unknown obstacles, and it estimates their locations.

The relative location between the robot and the obstacle is calculated by triangulation if the azimuth angle is detected at two positions while the robot is moving. The principle for estimating location of obstacles is the same as for generating an environmental map. In this case, the movement of the robot can be calculated from estimated locations of the robot, in consecutive frames. The method for estimating the object region is the same as for estimating free space. If two obstacles are present in the environment, other edges appear between contiguous ones. Candidates of vertical edges can be classified to each obstacle.

VI. The COPIS System

A. System Components

A prototype of the COPIS system shown in Fig. 13(a) and (b) has three essential components; an imaging subsystem COPIS, an image processing subsystem and a mobile robot. Wireless modems are used for serial communication between the robot and the image processing subsystem. COPIS mounted on the robot consists of a conic mirror with the parameters given in Table I, and a TV camera in a cylindrical glass tube with a diameter of 200 mm and a height of 200 mm. The image is transmitted by UHF video transmitter and receiver. The speed of the robot is about 5 cm/s and size is approximately 25 cm (Width) × 45 cm (Length) × 42 cm (Height). The image processing sub-system consists of a monitor, an image processor which converts each omnidirectional image into a 512 × 480 8 bit digital image and a 32 bit workstation. The image analysis is implemented on an image processor and the processing time is about 0.7 s/frame. The total processing cycle, including communication is about 1 s/frame. Fig. 14(a) is an example of an environment and Fig. 14(b) is an example of input images.



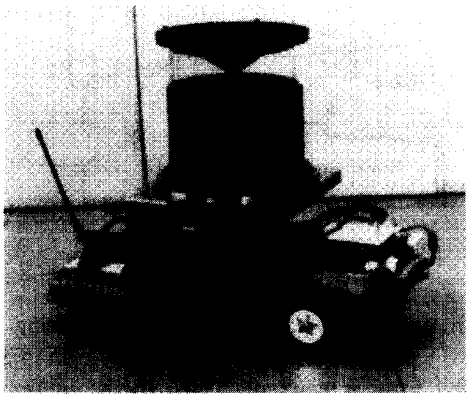


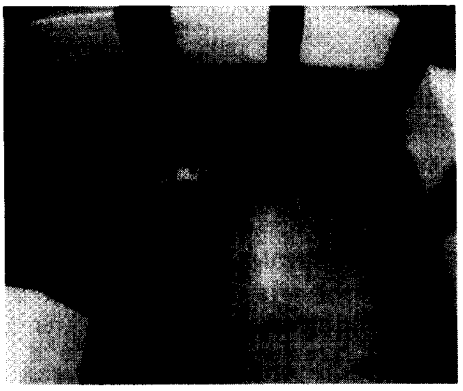
Fig. 13. (a) COPIS system configuration. (b) Prototype of COPIS.

TABLE I
Configuration and Parameters of COPIS

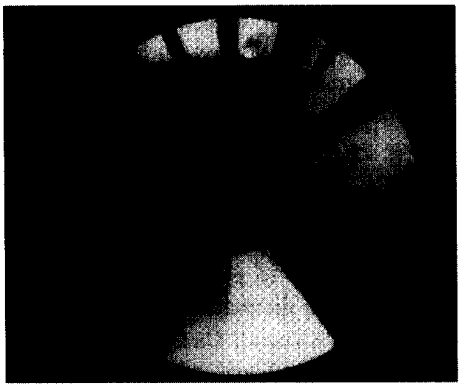
Vertical angle of conic mirror θ	112.6(deg)
Focal length f	8.5(mm)
F number of camera lens	16
Distance between lens center and vertex of conic mirror l	approx. 110(mm)
Lower view angle $\alpha 1$	112(deg)
Upper view angle $\alpha 2$	90(deg)

B. Image Processing

Fig. 15 shows the procedure used to estimate loci of vertical edges. The robot begins to move and takes an image sequence. As shown in Fig. 16, we apply the 3 × 3 Sobel operator to each input image [Fig. 14(b)] to find edge points and their directions. Next we apply thresholding to each edge picture. To find the vertical edges, radial edges in the image are



(a)



(b)

Fig. 14. (a) Experimental environment. (b) An example of input image.

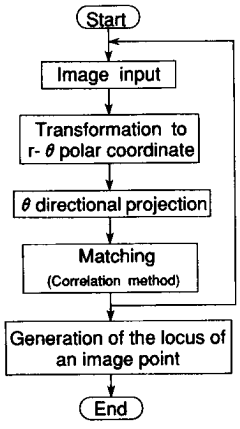


Fig. 15. Process sequence of locus estimation.

transformed into 2-D polar coordinates (r, θ) $(r = \sqrt{x^2 + y^2})$ and projected onto the θ-axis to get the 1-D projection. In this projection, the data are segmented into regions of each edge and the azimuth of the edges is estimated by calculating the average angle of each region. Fig. 17 shows the 1-D projection of edges.

To estimate the loci of azimuths of vertical edges, the correspondence of edge length between edges in the 1-D projection of consecutive images is established by using a correlation method in the restricted search field. As consecutive images are sampled densely, one can consider that the azimuth angle

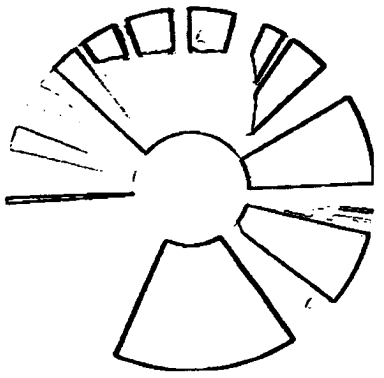


Fig. 16. Edge picture of Fig. 14(b).

Fig. 17. Projection of edge points onto theθ axis.

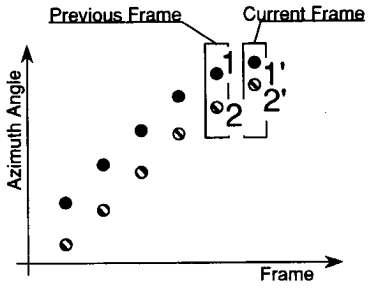


Fig. 18. Matching by evaluating the conformation of neighboring relation.

of the vertical edge in next frame is in the neighborhood of the azimuth angle in the current frame. Therefore, a certain margin of search field in the next frame is set around the current azimuth angle of the obtained vertical edge. After matching a few frames, the search region can be limited to a narrowed one by calculating the locus of each edge from (4). We then evaluate the conformation of neighboring relations. For instance, if there is some difference in matching score by correlation, the order of azimuths of edges becomes a prior consideration. As shown in Fig. 18, the edge 1(2) matches with edge 1′(2′). Fig. 19 shows results of loci of vertical edges.

Vertical edges sometimes disappear and reappear due to occlusion by other obstacles. In case of the short interval of occlusion, it can continue to match the vertical edges in consecutive images because the azimuth angle at the next frame is predicted from the locus of azimuth angle of the vertical edge in consecutive images. However, when the occlusion becomes longer, the error between the predicted and the real observed azimuth angle will increase, therefore, it is difficult to find the corresponding one. Usually, both estimated locations, before and after occlusion, are close. Therefore, if

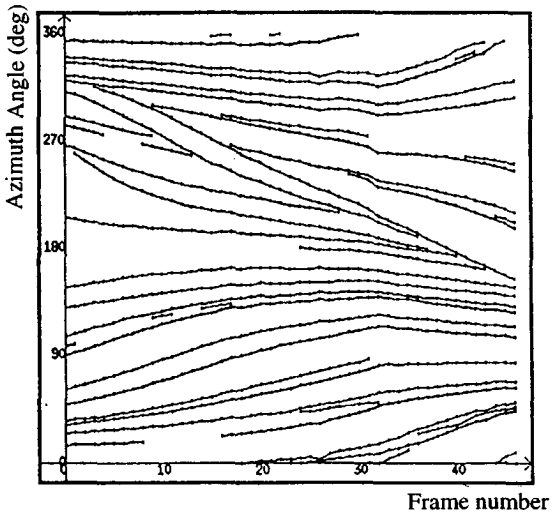


Fig. 19. Locus map of vertical edges.

the distance between two estimated locations is shorter than a certain length, we connect them and reestimate the location. If the distance is longer than a certain length, we consider that each locus of azimuth angles occurred from different vertical edges.

C. Experimental Results

Using the COPIS system, we did several experiments. In Section VI-C-1 and VI-C-2, we show experimental results for generating maps and navigating with a given map, by a real-time control. Results of surface verification using range data from an ultrasonic sensor is shown in Section VI-C-3. In Section VI-C-4, we show experimental results for map generation in a room and a corridor.

1) Results of Map Generation: As shown in Fig. 14(a), the first one was carried out in a simple room 2.5 m by 2.5 m, to evaluate the fundamental performance of the system. Observing the loci of azimuth angles of vertical edges, one can estimate locations of these edges and object planes, thus, the environmental map can be built. First, the robot moves straight from the 1st to the 16th frame, and during the next 16 frames, the robot changes its direction and moves round the arc toward right side, and finally moves straight again. Fig. 20(a)–(h) shows results at the 5th, 10th, 15th, 30th, 35th, 40th, 45th, and the final frame. The vertical edges matched in more than 4 frames were plotted as black points and the estimated surfaces were drawn by a straight line between the vertical edges. In this experiment, as shown in Fig. 20(a), a large error occurred in front of the robot. Generally, the error of the measurement by triangulation is in inverse proportion to the trigonometric parallax and the distance between view positions. The large error in front of the robot occurred because azimuth angles of edges in front of the robot did not change significantly and the movement of the robot was short. When the robot moved continuously straight, the error decreased in front of the robot because of the increase of distance between view positions with the movement of the robot and because of effect of the least square method as the number of measurements increased, as shown in Fig. 20(b) and (c). Then, when the robot moved further and turned round toward the right side, as shown in Fig. 20(d)–(h), the locations of these edges are estimated precisely because the trigonometric parallax increased. The average error of the edge location which matched more than 15 frames was approximately 3 (cm) and the average error of location measurements which matched in more than 4 frames was approximately 6 (cm), as shown in Fig. 20(h). The obtained precision is considered to be sufficient for robot navigation in a man-made environment such as a room, although the edge positions were calculated from projection data for every 1°.

Other errors relate to the precision of the obtained azimuth angles. Because edge positions were calculated from projection data of every 1°, it is difficult to distinguish the projection data of all edges in the area where there are dense vertical edges. In this system, the resolution of azimuth angle is approximately 0.25° in the peripheral area of the image and approximately 1° at the circumference of a 60 pixels radius from the image center. A higher precision in location measurement may be obtained by improving resolution of the azimuth angle. To precisely define the vertical edges, radial edges in the image are transformed into the 2-D polar coordinates (r, θ) and projected onto the θ-axis to get the 1-D projection. In this projection, we did not distinguish edge sign (edges from black to white and vice versa). Using edge signs, edges from black to white (plus edge) and edges from white to black (minus edge) are projected separately. Therefore, edges can be discriminated. However, in this experiment, to examine the potential of COPIS against effects of observational errors in real-time navigation, we simplified the projection method, without edge sign. The projection method with edge sign will be shown in Section VI-C-4.

2) Results of Visual Navigation with Given Map: An experiment was done in a room 4 m by 4 m. Each image was taken every 6 cm of the robot movement. Observing the locus of the azimuth angle of vertical edges, COPIS can estimate its own location and motion. Fig. 21 shows plots of the azimuth angle of vertical edges in the environment, and the result is shown in Fig. 22(a)–(e). First, the robot moves straight from the 1st to the 20th frame [Fig. 22(a) and (b)] and at this 20th frame, the robot finds the unknown object, front side, changes direction and moves round the arc toward the left side, as shown in Fig. 22(c). Next the robot changes direction and moves round the arc toward the right side [Fig. 22(d)], and finally moves toward the goal position [Fig. 22(e)]. The given vertical edges in the environmental map were plotted as small black thin rings, and given edges used for estimating robot's locations were plotted as black thick squares. The obtained vertical edges on the unknown obstacles, matched in more than 15 frames, were plotted as black points. The estimated locus of the robot was drawn by a black thick line.

An average error of the location measurement of the robot was approximately 3 cm, the maximum error was approximately 7 cm and the final error at the goal position approximately 6 cm. In this experiment, after the robot moves round the arc toward the right side, a large error occurred. During the first 60 frames, there are many vertical edges matched

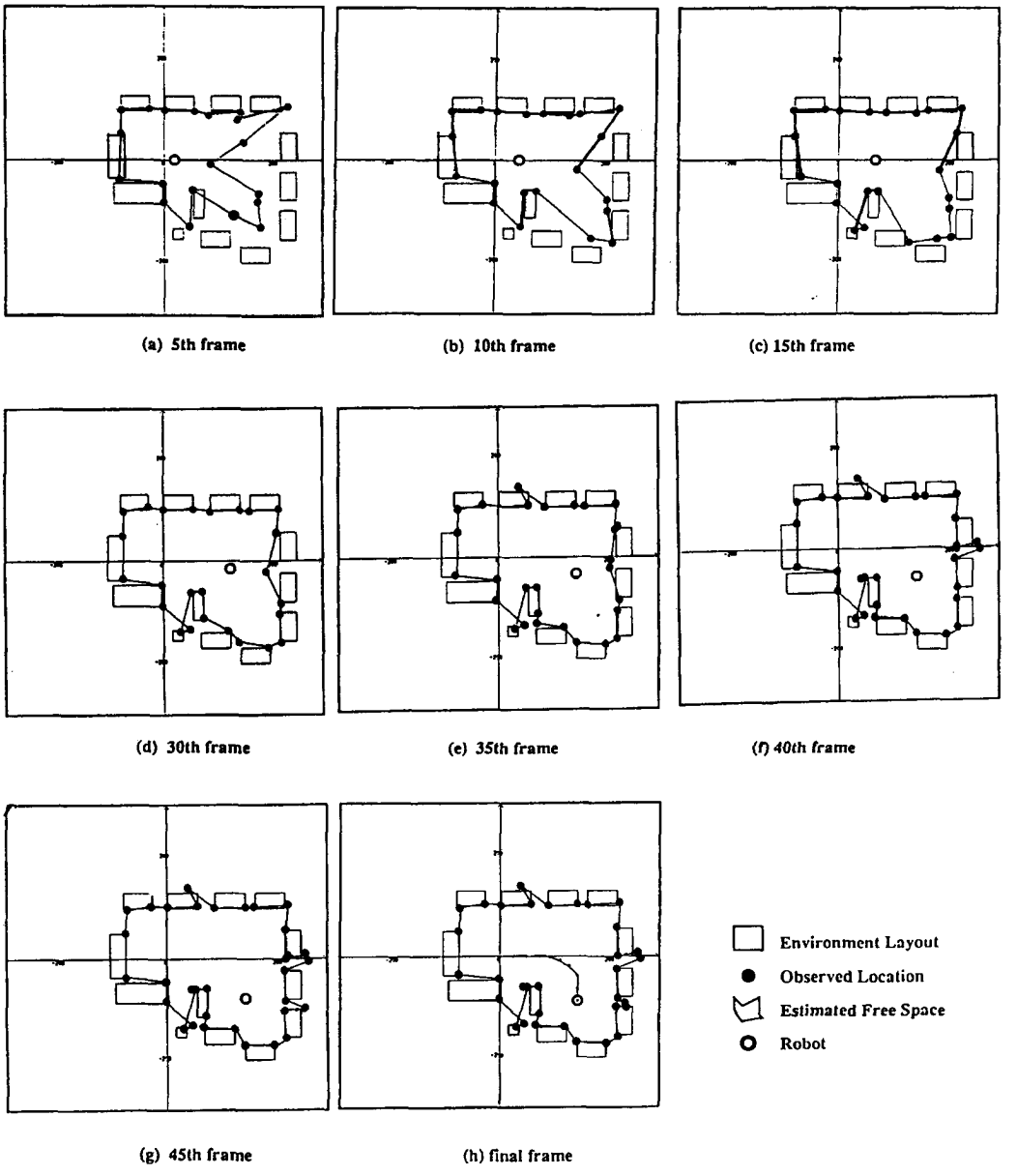


Fig. 20. Results of location estimation of object and free space for generating of environmental map.

with the environmental map; the location of the robot can be calculated precisely from these vertical edges. However, when the robot moves along a final course, these vertical edges move toward the posterior region of the robot, and some vertical edges are occluded by an unknown obstacle. In such cases, the number of vertical edges matched with the environmental map will decrease. Therefore, the location of the robot cannot be calculated so precisely.

After the robot moves straight from the 1st to the 20th frame, a large error of the location of an unknown obstacle occurred in the front region of the robot, as shown in Fig. 22(b). The error diminishes while the robot moves, as shown in Fig. 23. Finally, the average error of the location of the unknown obstacle was approximately 3 cm.

3) Verification of the Candidate Surface using Range Data from an Ultrasonic Sensor: Using the COPIS system with a given map, we did experiments in a 4 m by 4 m room. The robot was going to move from the start position to the goal position. Given vertical edges matched with the environmental map exist around the robot at various places, as shown in Fig. 24(a), (b), thus the location of the robot could be calculated from these vertical edges, both were done in the same room with unknown obstacles. The first experiment did not make use of range data from an ultrasonic sensor and the second one uses both azimuth information in the image sequence and range data from an ultrasonic sensor. The result is shown in Fig. 24(a) and (b). The given vertical edges in the environmental map were plotted by white squares. Observing the locus of the azimuth angle of vertical edges, COPIS can estimate its own location and motion. The estimated locus of the robot was drawn by a black line. As shown in Fig. 24(a), the inside of the region (passage) between two objects is estimated as the wall, and the robot could not go through there. However, the range data is continuously higher than the

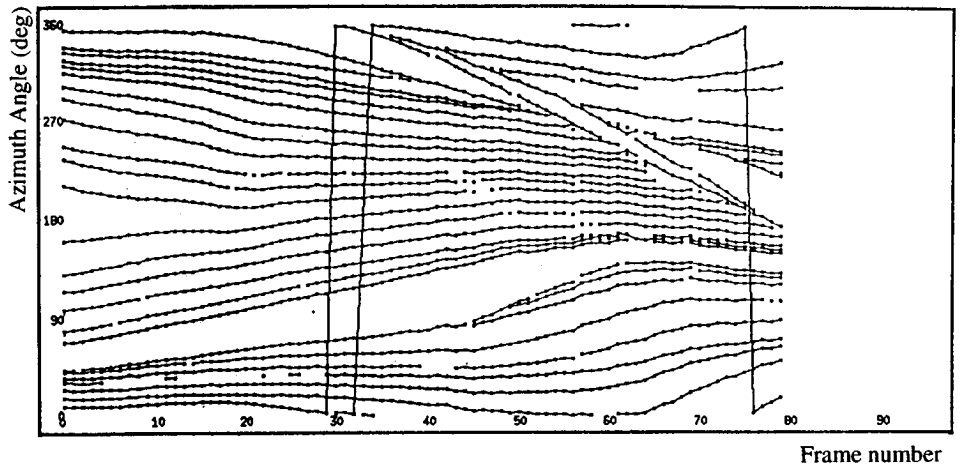


Fig. 21. Locus map of vertical edges.

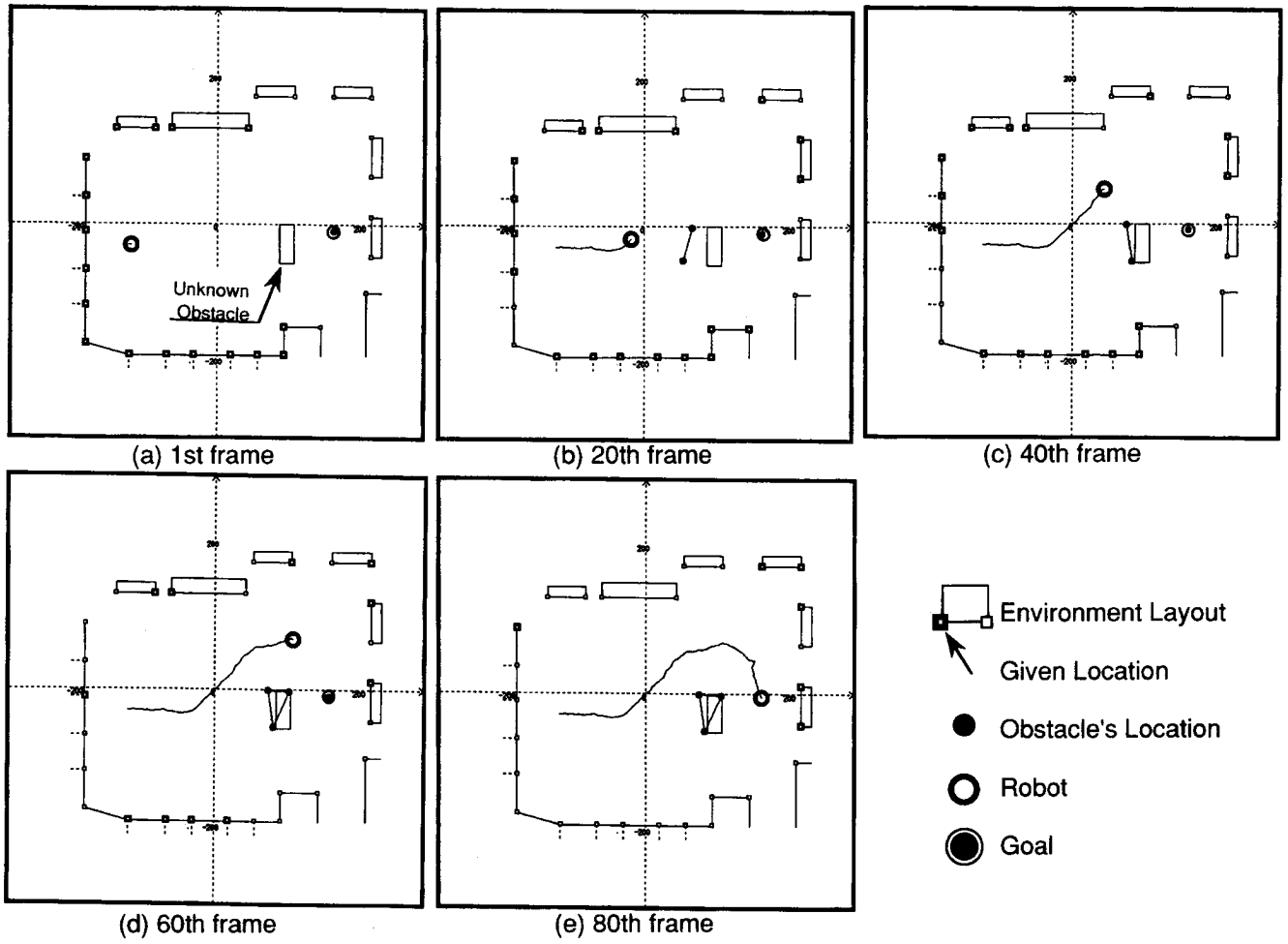


Fig. 22. Measurement of location of obstacle and robot's location and motion.

distance L estimated by COPIS, thus, the robot can go through the inside of the region (passage) between two objects, as shown in Fig. 24(b). These results show that sensor integration is useful for robot navigation.

4) Evaluation of Effectivity in a Real Environment: Results were obtained with the robot moving in a corridor and in a computer room. In the corridor, the robot moves straight during 40 frames. Fig. 25(a) and (b) are examples of input image and loci of azimuths of vertical edges, respectively. Fig. 25(c) is the result of estimated locations of vertical edges, at the final frame. The estimated locations of vertical edges were plotted by black squares and the estimated surfaces were drawn by a solid line between the estimated vertical edges. A large error of location occurred in the front region of the

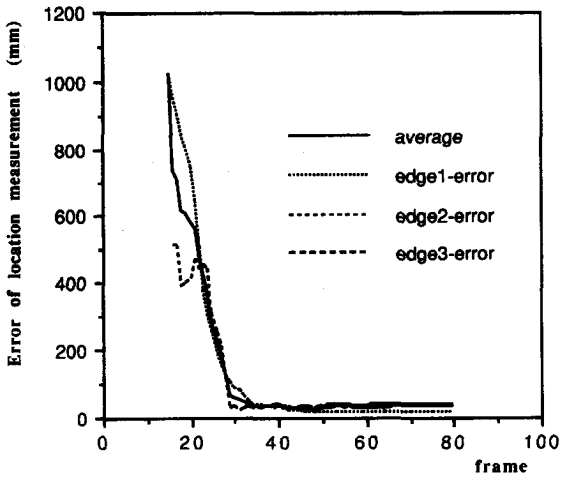
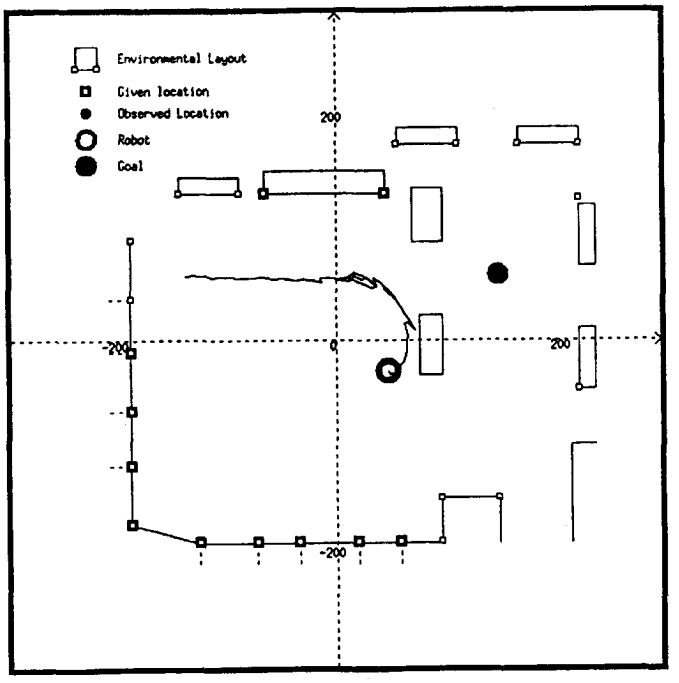


Fig. 23. Measurement error of location of obstacle.

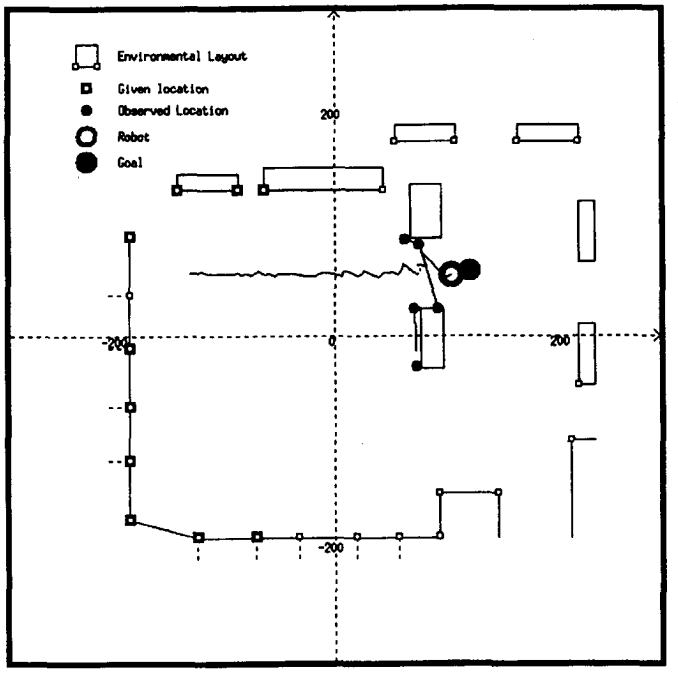
robot, because the azimuth angles of the edges in front of the robot did not change significantly. By changing the direction of movement, precision can be increased as in the same case described in Section VI-C-1. In case of cylindrical objects such as an extinguisher or a pipe, as shown in Fig. 25(d) and (e), estimated locations were adequate for navigation. In case of a four legged stool, as shown in Fig. 25(f), regions between legs are estimated as for a candidate surface because we assume the surface by using only a simple geometrical relation. However, the false surfaces could be removed by using an acoustic sensor as described in the Section IV-B.

Next, we show the result in the computer room with several objects. The top view of the environment is shown in Fig. 26. Close up pictures of portions labeled as (A), (B), (C), and (D) of Fig. 26 are shown in Fig. 27(a)–(d), respectively. Fig. 28(a) is one of the input images and Fig. 28(b) is the result of estimated locations of vertical edges at the final frame. The estimated locations of vertical edges were plotted by black points and the estimated surfaces were drawn by a dashed line between the estimated vertical edges. In this case, the height of the object (desk) is higher than that of COPIS. Therefore, COPIS observed the lower part the desk and estimated the desk configuration to be a concave object as label (B). Therefore, one can estimate both convex and concave ones.

The method proposed in this paper uses properties common to many indoor environments. The floor is almost flat and many visible vertical edges are present about the room. Therefore, the method can be used for mobile robots working in most buildings or plants. The obtained precision of environmental maps is sufficient for robot navigation in a man-made environment such as a room.



(a)



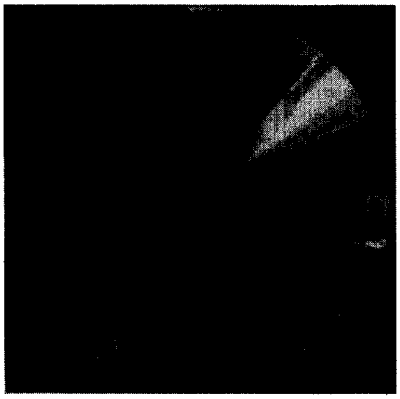
(b)

Fig. 24. (a) Results of map-based navigation without ultrasonic sensor. (b) Results of map-based navigation by integrating ultrasonic sensor with COPIS.

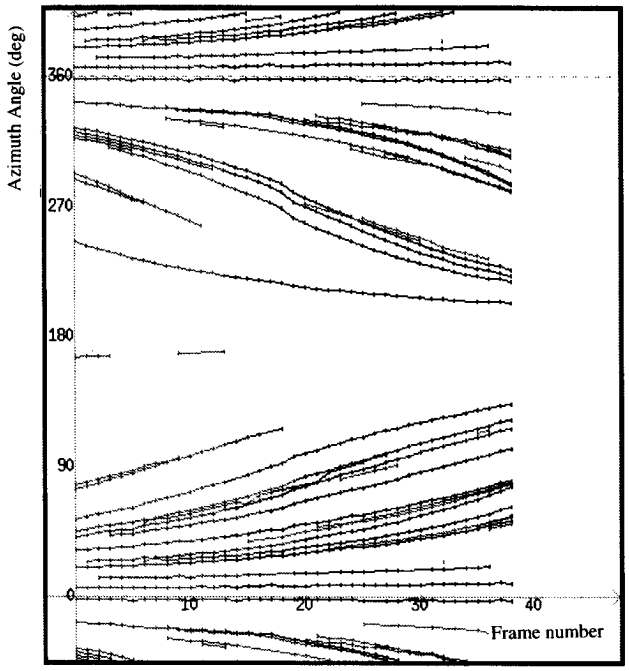
VII. Conclusion

We described the map generating and the map-based navigating algorithms, using the omnidirectional image sensor COPIS.

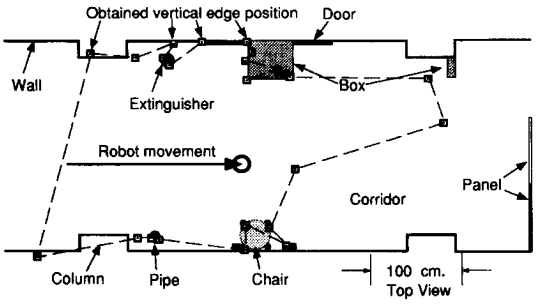
In the first half of this paper, we described a method for generating an environmental map. Assuming that the robot moves at a known velocity, COPIS could estimate locations of vertical edges and the free space for a mobile robot from the geometrical relation between vertical edges in the image. This free space means space where the robot can move freely. Therefore, there is no problem when the robot moves in this free space. When the robot moves toward its goal position that is out of this free space, it cannot go through the passage because it is estimated as the surface. This problem is a common one related to the edge-based algorithm that yields



(a)



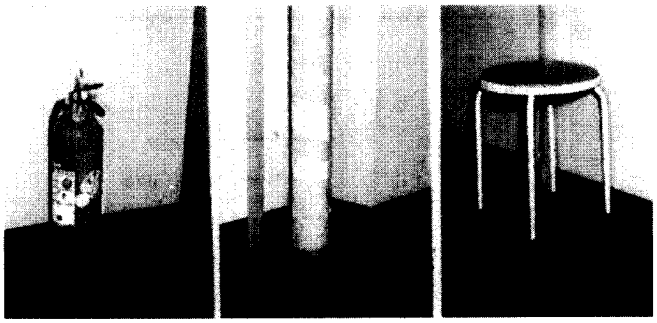
(b)



(c)

Fig. 25. (a) An example of input image. (b) Locus map of vertical edges. (c) Results of generation of an environmental map.

a wire frame description of the scene. To solve this problem, we integrated COPIS with the acoustic sensor. It was difficult to distinguish projection data of all edges in the area where there are dense vertical edges. This problem can be solved by improving resolution of the azimuth and using edge sign.



(d) (e) (f)

Fig. 25. *(Continued)* (d) Extinguisher. (e) Pipe. (f) Four legged stool.

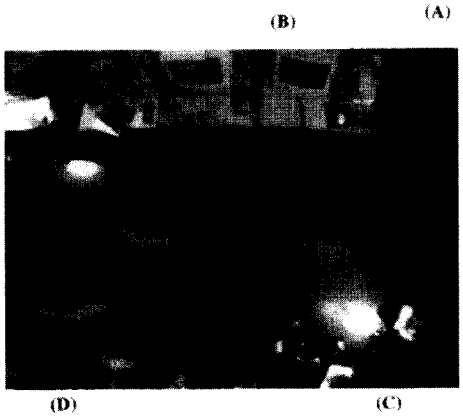
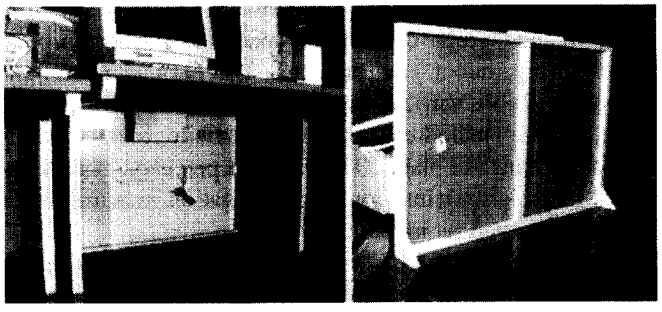


Fig. 26. Aerial view of the scene in the computer room.

(a) (c)



(b) (d)

Fig. 27. (a) Right side panel of desk and TV table (label A). (b) Desk (label B). (c) Box and paper bag (label C). (d) Wood panel (label D).

In the latter part, we described a method for estimating motion of the robot and the locations of unknown obstacles.

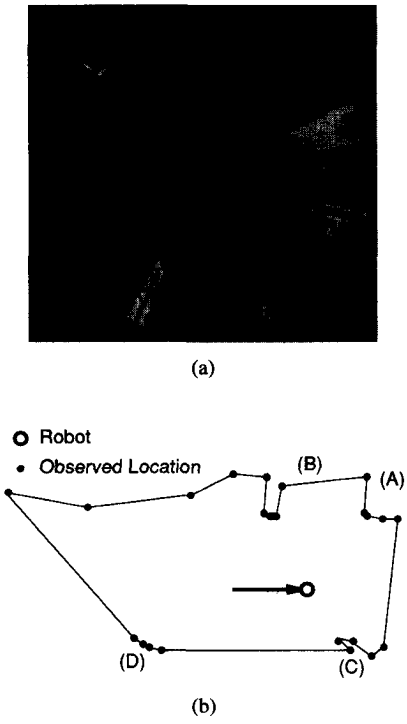


Fig. 28. (a) An example of input image. (b) Results of generation of an environmental map.

The robot's motion was estimated by matching azimuth information of vertical edges from both the input image and the given environmental map. The robot motion could successfully be estimated even when only a part of the given environmental map could be detected from the omnidirectional image. The locations of unknown obstacles were estimated by monitoring loci of azimuth angles of the vertical edges. Since we used triangulation for location measurement of edges, a large error occurred in the front region of the robot when the robot moved straight forward, but the error diminishes when the robot changes direction.

Since COPIS can observe a 360° view around it in real-time, and the precision of the obtained location of the robot and unknown obstacles is sufficient, we believe that COPIS is a suitable sensor for navigation. However, a resolution of the image taken by COPIS is not sufficient to understand details of shapes of objects because a 360° view is projected on one image. The binocular vision using the usual lens with a limited view is useful for understanding details. Therefore, the sensor system integrated with COPIS and the binocular vision will be efficient for navigation and manipulation. For example, a global view taken by COPIS is used for navigation and for finding candidates of interesting objects. Local views taken by binocular vision with the usual lens are used to analyze detailed 3-D structures of only interesting objects. We are now considering integrating COPIS with binocular vision.

ACKNOWLEDGMENT

The authors would like to thank M. Ohara for helpful advice on presentation of this material and R. Hiura for doing the experiments in a real environment.

REFERENCES

[1] A. M. Waxman, J. J. LeMoigne, and B. Scinvasan, "A visual navigation system for autonomous land vehicles," *IEEE J. Robotics Automat.*, vol. RA-3, no. 2, pp. 124–141, 1987.

[2] M. Turk, K. D. Morgenthaler, K. D. Gremban, and M. Marra, "VITS-A vision system for autonomous land vehicle navigation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-10, no. 3, pp. 342–360, 1988.

[3] C. Thrope, M. H. Hebert, T. Kanade, and S. A. Shafer, "Vision and navigation for the Carnegie-Mellon Navilab," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-10, no. 3, pp. 362–373, 1988.

[4] M. Yachida, T. Ichinose, and S. Tsuji, "Model-guided monitoring of a building environment by a mobile robot," in *Proc. 8th IJCAI*, Aug. 1983, pp. 1125–1127.

[5] H. P. Moravec, "The Stanford Cart and the CMU Rover," *Proc. IEEE*, vol. 71, no. 7, pp. 872–884, 1983.

[6] A. Elfes, "Sonar-based real world mapping and navigation," *IEEE J. Robotics Automat.*, vol. RA-3, pp. 249–265, 1987.

[7] J. L. Crowley, "World modeling and position estimation for a mobile robot using ultrasonic ranging," in *Proc. Int. Conf. Robotics Automat.*, 1989, pp. 674–680.

[8] J. F. G. Lamadrid and M. L. Gini, "Path tracking through uncharted moving obstacles," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, no. 6, pp. 1408–1422, 1990.

[9] J. J. Leonard, H. F. Durrant-Whyte, and I. J. Cox, "Dynamic map building for an autonomous mobile robot," *Int. J. Robotics Res.*, vol. 11, no. 4, pp. 286–298, 1992.

[10] G. L. Miller and E. R. Wagner, "An optical rangefinder for autonomous robot cart navigation," in *Proc. SPIE Mobile Robots II*, 1987, vol. 852, pp. 132–144.

[11] R. A. Jarvis and J. C. Byrne, "An automated guided vehicle with map building and path finding capabilities," in *Proc. 4th ISSR*, 1988, pp. 497–504.

[12] H. Ishiguro, M. Yamamoto, and S. Tsuji, "Omni-directional stereo for making global map," in *Proc. 3rd ICCV*, 1990.

[13] K. B. Saraclik, "Characterizing an indoor environment with a mobile robot and uncalibrated stereo," in *Proc. IEEE Int. Conf. Robotics Automat.*, 1989, pp. 984–989.

[14] Z. L. Cao, S. J. Oh, and E. L. Hall, "Dynamic omnidirectional vision for mobile robots," *J. Robotic Syst.*, vol. 3, no. 1, pp. 5–17, 1986.

[15] S. J. Oh and E. L. Hall, "Guidance of a mobile robot using an omnidirectional vision navigation system," in *Proc. SPIE 852 Mobile Robots II*, 1987, pp. 288–300.

[16] Y. Yagi and S. Kawato, "Panorama scene analysis with conic projection," in *Proc. IEEE Int. Workshop Intelligent Robots & Syst.*, 1990, pp. 181–187.

[17] J. Hong, X. Tan, B. Pinette, R. Weiss, and E. M. Riseman, "Image-based homing," in *Proc. Int. Conf. Robotics Automat.*, Apr. 1991, pp. 620–625.

[18] Y. Yagi, S. Kawato, and S. Tsuji, "Collision avoidance using omni-directional image sensor (COPIS)," in *Proc. IEEE Int. Conf. Robotics Automat.*, Apr. 1991, pp. 910–915.

[19] ——, "Real-time omnidirectional image sensor (COPIS) for vision-guided navigation," *IEEE Trans. Robotics Automat.*, vol. 10, no. 1, pp. 1–12, 1994.

[20] K. Sugihara, "Some location problems for robot navigation using a single camera," *Computer Vision, Graphics and Image Processing 42*, pp. 112–129, 1988.

[21] S. Atiya and G. Hager, "Real-time vision-based robot localization," in *Proc. Int. Conf. Robotics Automat.*, Apr. 1991, pp. 639–644.

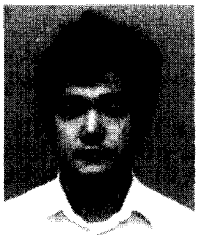
[22] Y. Yagi and M. Yachida, "Real-time generation of environmental map and obstacle avoidance using omni-directional image sensor with conic mirror," in *Proc. IEEE Conf., Computer Vision and Pattern Recog.*, June 1991.

[23] Y. Yagi, Y. Nishizawa, and M. Yachida, "Estimation of free space for mobile robot using omnidirectional image sensor COPIS," in *Proc. IEEE/IES Int. Conf. Industrial Electronics, Control and Instrumentation (IECON'91)*, Kobe, Oct. 1991, vol. 2, pp. 1329–1334.

[24] ——, "Map based navigation of the mobile robot using omnidirectional image sensor COPIS," in *Proc. IEEE Int. Conf. Robotics Automat.*, Nice, May 1992, vol. 1, pp. 47–52.

[25] O. D. Faugeras, E. L. Bras-Mehlman, and J. D. Boissonnat, "Representing stereo data with the delaunay triangulation," *Int. J. Artificial Intell.*, vol. 44, pp. 41–87, 1990.

[26] Y. Yagi, Y. Nishizawa, and M. Yachida, "Obstacle avoidance for mobile robot integrating omnidirectional image sensor COPIS and ultrasonic sensor," in *Proc. 2nd Int. Conf. Automation Robotics and Computer Vision*, 1992, vol. 3, pp. RO-11.7.1–RO-11.7.5.



Yasushi Yagi (M'91) received the B.E. and M.E. degrees in control engineering, in 1983 and 1985, respectively, and the Ph.D. degree in 1991, from Osaka University.

In 1985, he joined the Product Development Laboratory, the Mitsubishi Electric Corporation, where he was working on robotics and inspections. In 1990, he was a Research Associate of Information and Computer Science, Faculty of Engineering Science, Osaka University. Since 1993, he has been a Lecturer of Systems Engineering, Faculty of Engineering Science, Osaka University. Computer vision and robotics are his research subjects.

Yoshimitsu Nishizawa received the B.E. and M.E degrees in information and computer science from Osaka University in 1991 and 1993, respectively.

He is presently working at Fujitsu Limited. His research interests are in the fields of computer vision and robotics.

Masahiko Yachida received the B.E., M.Sc. degrees in electrical engineering and the Ph.D. degree in control engineering, all from Osaka University, Osaka, Japan, in 1969, 1971, and 1976, respectively.

He joined the Department of Control Engineering, Faculty of Engineering Science, Osaka University in 1971 as an Assistant Professor, became an Associate Professor at the same department and moved to the Department of Information and Computer Science of the same university as a Professor in 1990. Since 1993, he has been a Professor at the Department of Systems Engineering of the same university. He was a Research Student of the research laboratory Riso, Danish Atomic Energy Commission, in 1967, joined the Intelligent Industrial Robot Project at the Electrotechnical Laboratory of Japanese Government from 1969 to 1970, was a Research Associate at the Coordinated Science Laboratory, University of Illinois, Urbana, from 1973 to 1974, a Research Fellow at the Fachbereich Informatik, Hamburg University from 1981 to 1982, and a CDC Professor at the Department of Computer Science, University of Minnesota, in 1983. He is an author of *Robot Vision* (Shoukoudou) and an Editor of *Computer Vision* (Maruzen). His interests are in the fields of computer vision, image processing, mobile robot and artificial intelligence.