Robotica (2010) volume 28, pp. 97–106. © Cambridge University Press 2009
doi:10.1017/S026357470900561X

Feature map management for mobile robots in dynamic environments

Se-Jin Lee†, Byung-Jae Park†, Jong-Hwan Lim‡* and Dong-Woo Cho†,§*

†*Department of Mechanical Engineering, Pohang University of Science and Technology, San 31 Hyoja-dong, Pohang 790-784, Korea*
‡*Department of Mechatronics, Cheju National University, #1 Ara-dong, Jeju 690-756, Korea*
§*Department of Integrative Bioscience and Bioengineering, Pohang University of Science and Technology, San 31 Hyoja-dong, Pohang 790-784, Korea*

(Received in Final Form: March 27, 2009. First published online: April 28, 2009)

SUMMARY

This paper presents a new approach to the management of the environmental map for mobile robots in dynamic environments. The environmental map is built of primitive features, such as lines, points, and even circles, extracted from ambiguous data captured by the robot's sonar sensor ring. The feature map must be managed because the indoor surroundings where mobile robots operate are continuously changing due to nonstationary objects, such as wastebaskets, tables, and people. The features are processed by trimming, division, or removal, depending on the dynamic circumstances. All processing refers to the occupancy probabilities of grid squares generated for the map features. The occupancy probabilities of the squares are updated using the Bayesian updating model with the sonar sensor data. Experimental results demonstrate the validity of the proposed method.

KEYWORDS: Map management; Dynamic environments; Feature map; Sonar sensor ring; Mobile robots.

1. Introduction

Autonomous navigation of a mobile robot requires several vital functions, such as map building, localization, path planning, and obstacle avoidance. In most cases, accurate environmental map building is essential because it has a great effect on the performance of the other functions.[1–3] However, regardless of the quality of the initial map building, using only the static initial map is ineffective in dynamic environments. Therefore, a method for managing the environmental map is essential for integrating the map constructed earlier into the subsequently changing environment.

Sensors, such as laser scanners, vision sensors, infrared scanners, and sonar sensors, have been used to build environmental maps. The sonar sensor is most commonly used because it is cheap and gives direct depth information about the location of an object. However, the specular reflection of a sonar beam gives rise to the multipath phenomenon, which, together with the wide directivity, makes it very difficult to determine the exact locations of objects.[4–7] Nevertheless, building the environmental map using cheap sonar sensors does merit some attention.[1,6–8] In particular, a fixed sonar sensor ring is preferred because the measurement speed of a single rotating sonar sensor is too slow to be practical.

Feature-based map building using a set of geometric primitives, such as lines and points, is the general method used with sonar sensors. It has the advantage of efficiently representing the surroundings. A circle feature is a relatively new requirement for map building in residential surroundings because such areas often contain round objects or rounded walls. However, circle feature extraction from ambiguous sonar sensor ring data has received little attention. We extract and maintain regular circle features as well as lines and points in our work.

While the mapping of static environments has received considerable recent attention, in most cases, these algorithms cannot be directly applied to dynamic environments. Crowley[9] developed one of the earliest feature-based approaches by introducing the concept of the composite local model. Leonard and Durrant-Whyte[10,11] showed two implementations of extended Kalman filter (EKF) localization with a feature model that used a simple vehicle with a single rotating sonar sensor and a mobile robot with six static sonar transducers. Meng *et al.*[12] used the adaptive EKF (AEKF) to localize robot poses using eight Polaroid sonar sensors with four kinds of landmarks: planes, concave corners, convex edges, and cylinders. Wijk and Christensen[8] presented a sensor fusion scheme for sonar data called triangulation-based fusion (TBF). The TBF algorithm, which delivers stable point landmarks, was implemented as a voting scheme that groups sonar measurements that are likely to have resulted from common objects in the environment. Using 24 Polaroid sonar sensors, Tardos *et al.*[13] found groups of sonar arcs all tangent to the same line, and grouped sonar arcs intersecting the same point using the Hough transform, which is a voting scheme where each

* Corresponding authors. jhlim@cheju.ac.kr and dwcho@postech.ac.kr

https://doi.org/10.1017/S026357470900561X Published online by Cambridge University Press

piece of sensor information accumulates evidence about the presence of certain features compatible with the actual measurement in a discretized parametric space. Heale and Kleeman[14] detected line and point features by analyzing and estimating sonar echoes with a real-time digital signal processing sonar echo processor they developed. Bank and Kampke[15] used arrays of sensors for taking high-resolution ultrasonic environment images and determining the boundary of objects with 24 wide-angled Robert Bosch sensors to solve the ambiguity problem. Fazli and Kleeman[16] introduced a delayed classification algorithm to categorize line and point features using a heavy sonar sensor ring equipped with 48 Polaroid sensors during a simultaneous localization and mapping (SLAM) process.

Laser and vision sensors have also been used to build maps and localize robot poses. Arras *et al.*[17] introduced the concept of relative features and implemented SLAM with a laser scanner. Wolf and Sukhatme[18] were successful in differentiating static and dynamic parts of the environment using a laser scanner. Davison[19] accomplished bearing-only SLAM with a single camera. Lee *et al.*[20] developed a position prediction simulator using a ceiling-mounted CCD camera.

The presence of moving objects usually causes errors and compromises the overall quality of the map. This is a considerable problem since many realistic robot applications are in nonstatic environments. This paper suggests a new approach to managing the environmental map. The environmental map is built of primitive features, such as lines, points, and even circles, extracted from ambiguous sonar sensor ring data. The features are processed by trimming, division, or removal, depending on the dynamic circumstances. All operations refer to the occupancy probabilities of grids generated on the map features. The occupancy probabilities of grids are updated using the Bayesian updating model with the sonar sensor data. We have demonstrated the validity of the proposed method using real experiments.

2. Feature Map Building

2.1. Data association filter

To scan the environment densely, in general, a rotating sonar sensor has been used. To get densely scanned data, however, a robot has to stop at a stationary position. From a practical application point of view, it is very inefficient and unrealistic because the measurement speed of a single rotating sonar sensor is too slow to be used. Therefore we used a fixed sonar sensor ring equipped with eight pairs of the piezoelectric ceramic low-performance sensors. Since the sonar sensor ring produces data that are less dense than would be obtained from a single rotating sonar sensor or a laser scanner, it is impossible to use powerful data filters like regions of constant depth (RCDs) based on a scan data.[10,11] This configuration therefore requires a type of filter that can reject false indications in sparsely sampled data. The footprint association (FPA) filter classifies sonar data into groups so that features can then be extracted from the groups.[21] The FPA filter basically determines whether two sets of sonar data from different sensor locations relate to the same object.

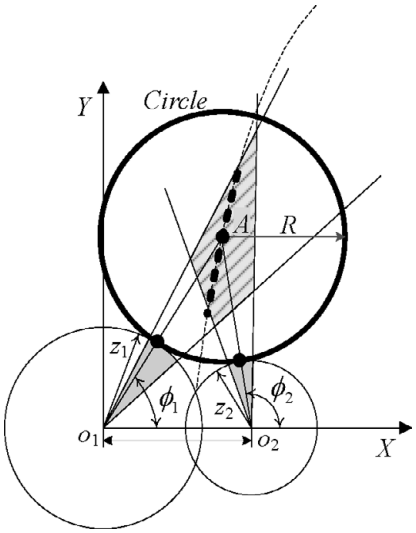


Fig. 1. Geometric relationships between two individual sonar measurements. Two sonar ranges are assumed to be returned from the identical circular feature. The bold dashed line segment on the dashed line is defined as the trace of centers that represents possible locations of the centers of the hypothesized circle feature.

In Fig. 1, two circles centered at sensor locations o_1 and o_2 are defined with radii equal to the range values z_1 and z_2, respectively. The effective sonar beam aperture is defined as the constraint angle of each footprint, which is represented by the shaded fan shapes. Angles ϕ_1 and ϕ_2 are the bearings from the positive x-axis of the local coordinate system to the center of a hypothesized circle, A. The FPA filter assumes that if any two sets of range data, z_1 and z_2, relate to the same hypothesized feature, that feature will define a third circle that is tangential to the two circles defined by the sonar footprints. The general problem, therefore, is to find radius R of the third circle that defines the hypothesized feature.

Since both ϕ_1 and R are unknowns, we cannot determine uniquely the values of ϕ_1 and R. For this reason, Leonard tried to extract features by taking the limit of the equation as $R \to \infty$ and $R = 0$.[10,11] This is a very special case of the third circle, i.e., $R \to \infty$ is a line feature and $R = 0$ is a point feature. Using this method, we cannot extract circle features that have finite R.

In order to determine R, we used the angle constraints (angular range) of each footprint. As shown in Fig. 1, each range value (z_1 or z_2) has angular uncertainty which is identical to the effective beam aperture of the sonar sensor. This angular range is represented as gray area in the figure. The location of actual object can be any point within the arc defined by the angular range of each footprint. Therefore, there can be infinite numbers of third circles with unknown radii that are tangent to the two circles the footprints define. However, we can find the limit of the radii of the third circles, which can be calculated by using the lower angle and upper angle of the angular range of each footprint. As a result, we can find the trace of centers that represent possible locations of centers of the circumcircles (the third circles) within the effective sonar beam aperture of each footprint. The trace is drawn as a bold dot line in Fig. 1.

https://doi.org/10.1017/S026357470900561X Published online by Cambridge University Press

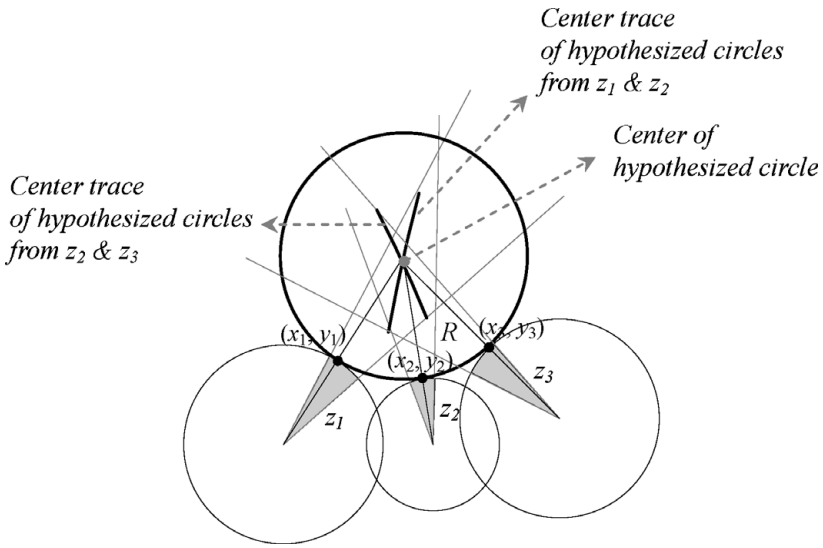


Fig. 2. The geometrical relationship between the hypothesized circle and three circles the footprints define for determining the hypothesized circle feature.

Using the second law of cosines, we can get the following relations between R and the angles (ϕ_1, ϕ_2):

$$\phi_1 = \cos^{-1}\left(\frac{(z_1+R)^2-(z_2+R)^2+d^2}{2d(z_1+R)}\right)$$
$$\phi_2 = \pi - \cos^{-1}\left(\frac{(z_2+R)^2-(z_1+R)^2+d^2}{2d(z_2+R)}\right) \quad (1)$$
$$\text{if } \theta_l^1 \le \phi_1 \le \theta_u^1 \text{ and } \theta_l^2 \le \phi_2 \le \theta_u^2, \text{ then they are associated with a circle.}$$

In Eq. (1), θ_l^i and θ_u^i are lower and upper limit angles of the effective beam width respectively. If we let θ_m^i be the sensor bearing and ω be the effective width of the sonar beam, then θ_l^i and θ_u^i are:

$$\theta_l^i = \theta_m^i - \frac{\omega}{2},\ \theta_u^i = \theta_m^i + \frac{\omega}{2}. \quad (2)$$

Substituting θ_l^i and θ_u^i for ϕ_1 and ϕ_2 in Eq. (1), the limit and the trace of the values of R can be found.

It is possible to determine ϕ_1 and ϕ_2 uniquely for some special cases. If R is very large or infinity, the hypothesized feature should be a line. In contrast, if R is very small or zero, the hypothesized feature should be a point. Except these two special cases, however, it is impossible to determine uniquely the radius of the hypothesized circle with only two range footprints. Therefore, we consider more than two pairs of footprints as shown in Fig. 2 to fine the radius of the hypothesized circle. The intersection point of more than two center traces was used for determining the center of the hypothesized circle.

Most of the researchers have used the Centerline Model in which the location of an object is assumed to be on the sensor measurement axis even if the angular resolution of a sonar beam is very low. However, the FPA filter can decide the possible location of an object for each sonar measurement. Figure 2 shows the contact points (x_i, y_i) with the hypothesized feature and the corresponding sonar footprints, and it will be used as parameters for least square optimization in feature extraction procedure.

2.2. *Feature extraction*

Whenever a new measurement comes in, FPA is executed for all data groups which are the groups of measurements that support the same hypothesized features accumulated in the data storage. If the correspondence is satisfied to one of the data group, it is added to that group. If not, it becomes a new data group.

As stated above, 3 or more sets of data are necessary in order to determine the parameters of the hypothesized feature except the special case of a line or point one. In the real application, however, even the special case also needs 3 or more sets of sonar data to get the more precise parameters of the hypothesized feature because of measurement noise and robot's position uncertainty. The sets of data that are associated with the same hypothesized feature are clustered into a group using FPA filter.[22] Figure 3 shows some experimental results of groups that were clustered from sets of data measured by using the fixed-type sensor ring equipped with eight Murata sonar sensors whose directivity is about 45°. The true object was a wall, a slim bar, and a basket for Fig. 3(a), (b), and (c), respectively. The traveled distances were about 0.4 m, 1m, and 1.5 m for the line, point, and circle feature extractions.

Sonar footprints reflected from the wall or the basket are all tangent to the wall or basket, while those reflected from the bar intersect in a point where the bar stands. However, because of the measurement noise and robot's position uncertainty, all sets of sonar data do not always intersect in an exact one point as shown in Fig. 3(b). This is also true for the cases of lines and circles. Therefore, least square optimization scheme was used to estimate optimal parameters of the feature extracted from the group of sonar footprints where measurement noise and robot's position uncertainty are involved.

https://doi.org/10.1017/S026357470900561X Published online by Cambridge University Press

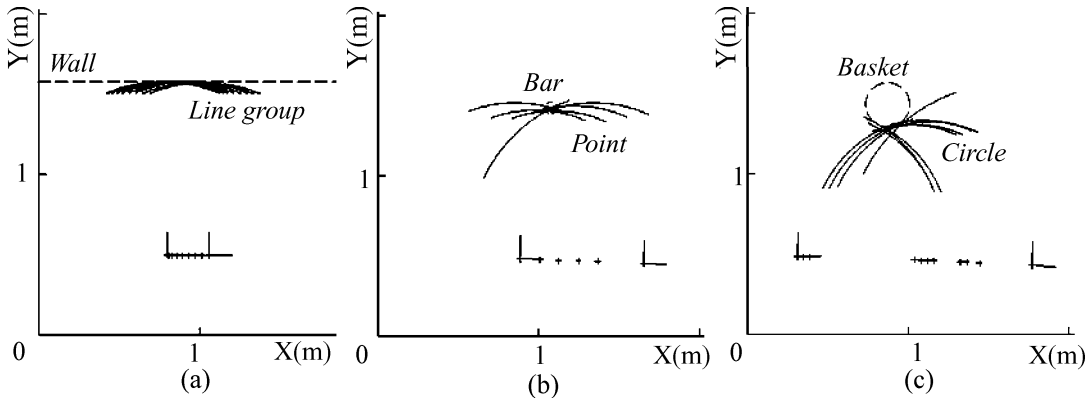


Fig. 3. Groups of sonar measurements classified by using FPA filter for (a) a hypothesized line feature, (b) point and (c) circle. Arcs represent the front edges of sonar footprints corresponding to the effective beam width of the sonar beam, dotted features represent the true shapes of objects, and cross marks represent robot positions where the corresponding sets of data were acquired.

The linear function of feature model for least square optimization is

$$A\hat{C} \approx \bar{b}, \tag{3}$$

where $\hat{C}$ is the coefficient vector of the feature model. If the number of observation, sets of sonar data, is m and the number of coefficient for the feature model is n, A is defined as a known $m \times n$ dimensional matrix, $\hat{C}$ is defined as an unknown n dimensional vector. Here, $\bar{b}$ is defined as a known m-dimensional vector. The feature model equations, A and $\bar{b}$ used in this paper are listed in Table I.

The solution, $\hat{C}$, for least squares of Eq. (3) through the singular value decomposition is given by,

$$A^T A\hat{C} = A^T\bar{b}, \tag{4}$$

$$\hat{C} = V\Sigma^{+}U^T\bar{b}. \tag{5}$$

In Eq. (5), V is $n \times n$ orthogonal matrix whose columns are eigenvectors of A^TA, U is $m \times m$ orthogonal matrix whose columns are eigenvectors of AA^T. Here, $\sum^{+}$ is defined by transposing $\sum$ and inverting the nonzero diagonal entries.

Figure 4 shows the features extracted through least square optimization from the groups shown in Fig. 3. In this figure, r and θ are defined as the distance and bearing with respect to the origin of the global reference frame, respectively. This result shows that the performance of a feature extraction is good enough to use practical applications even if sparse sonar data were used. Furthermore, it shows the possibility of extracting a circle feature as well as a line or a point one by using the proposed FPA filter. These initialized segments are merged into previously registered features for building a global map.

3. Map Management

The constructed map has to be managed well because it has a great effect on the performance of the other functions. We expect better performance in localization, path planning, map junction, and SLAM during autonomous navigation of a mobile robot when the environmental map is well constructed and maintained according to the specific circumstances at hand.

Localization: The EKF is commonly used to localize robot poses with landmarks of environmental features. If the constructed feature map does not reflect the true shape of the environment very well, the correspondence between feature

Table I. Feature model equations and the coefficient matrices. (x_i, y_i) is the contact point through FPA filter as shown in Fig. 2.

Type of feature	Feature model	A	$\bar{b}$
Line	$\alpha x + \beta = y$	$\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_m & 1 \end{bmatrix}$	$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$
Point	$\alpha = x, \beta = y$	$\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_{m\times 1}$	$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}, \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$
Circle	$\alpha(x^2 + y^2) + \beta x + \gamma y = 1$	$\begin{bmatrix} x_1^2 + y_1^2 & x_1 & y_1 \\ x_2^2 + y_2^2 & x_2 & y_2 \\ \vdots & \vdots & \vdots \\ x_m^2 + y_m^2 & x_m & y_m \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_{m\times 1}$

https://doi.org/10.1017/S026357470900561X Published online by Cambridge University Press

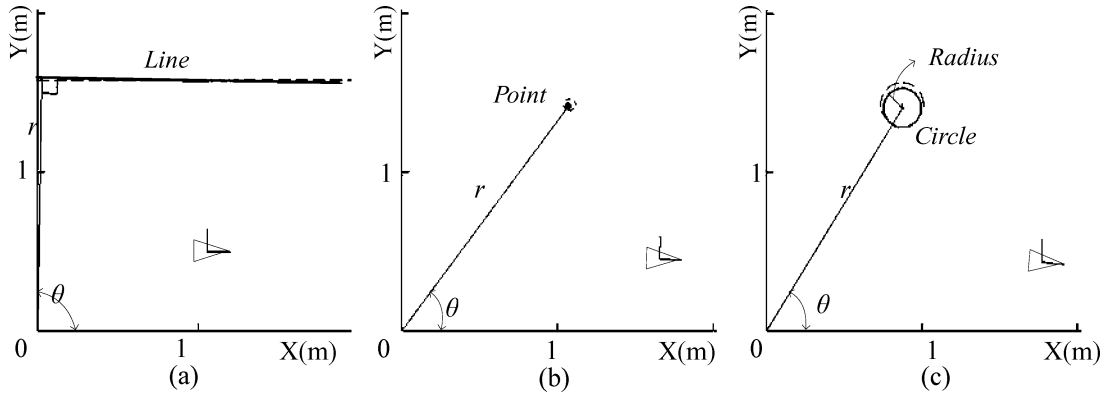


Fig. 4. Feature segments extracted through least square optimization.

landmarks and observations will be misleading, and this can eventually lead to the failure of localization. Therefore, successful performance of a correspondence test can be guaranteed by good map management.

Path Planning: The best path to a destination is based on map information alone. Therefore, if the map is not well managed, it is unreasonable to expect a successfully optimized path to the destination.

Map Junction: We can use extracted features to join various segments to construct the complete map. If features are extracted using uncertain sensor data, it is necessary to cope with the effect of the wide sonar beam that causes features to be overextracted around the edges of objects. Therefore, managing and controlling inaccurate features is helpful for joining feature segments as well as executing the other functions.

SLAM: Anything that can be done to increase the computational efficiency of the SLAM algorithms is important because SLAM consumes the greatest computational load.[23] A practical implementation will be able to remove needless landmarks and reduce the computational requirements.

3.1. Feature processing

We used three strategies for feature processing to make the feature map more realistic, even when using sonar sensors in dynamic environments. These strategies are trimming, division, and removal.

Trimming: Because of the wide sonar beam, lines or circle features tend to be overextracted around the ends of objects. Trimming the uncertain parts of features so they resemble the true object shapes is necessary to overcome the drawbacks of the sonar sensors.

Division: If a door that was initially closed during map building times is subsequently opened, the single line feature extracted for the wall, including the door, must be divided into two parts.

Removal: All features extracted for movable objects are subject to removal. Therefore, we need to determine the existence of these features.

Figure 5 shows each type of line feature processing. In Fig. 5(a), the line feature was overextracted in comparison with the true length of the wall because of the width of the sonar beam. The invisible part of a circle feature needs to be trimmed as well in Fig. 6. Also, when a door that was closed during the initial map building is subsequently opened, the single line feature extracted for the wall, including the door, must be divided into two parts, as shown in Fig. 5(b). If the line feature is not divided into two parts, optimized path planning is disrupted. Finally, features extracted for movable objects, such as wastebaskets, tables, and people, must handled with care. If an object disappears, all features corresponding to that object must also be removed, as shown in Fig. 5(c).

3.2. Occupancy probability updating of grids

3.2.1. Data reliability test. Trimming, division, and removal processing are based on the occupancy probabilities of grid squares generated on the extracted features. The occupancy probabilities of the squares are updated using the Bayesian updating model with the sonar data. Before applying the sonar data to update the occupancy probability of the grid squares, we must evaluate the data reliability to distinguish reliable measurements from spurious multipath data caused by the specular reflection of the sonar beam. The data reliability is also tested using the FPA filter to determine how geometrically acceptable the sonar data is at an object boundary. As shown in Fig. 7, the sonar data measured at the current step t can be matched with the data in the buffer from other sonar scans taken at different robot positions. If the associations of the other data with the current sonar data are successful, it qualifies for participation in the occupancy probability for updating the grid squares of the map features. The test is successful if the percentage of data association, λ_g, is greater than data reliability factor that will be given in Section 3.4.

3.2.2. Updating model. The optimum occupancy grid-updating algorithm calculates the posterior over prior states given the data. We can derive the log odds form of an updating model according to the Bayes filter algorithm, as given by Thrun *et al.*[24]:

$$l_t(m_i) = \log \frac{p\,(m_i\,|z_t\,)}{1 - p\,(\bar{m}_i\,|z_t\,)} - l_o + l_{t-1}(m_i). \qquad (6)$$

https://doi.org/10.1017/S026357470900561X Published online by Cambridge University Press

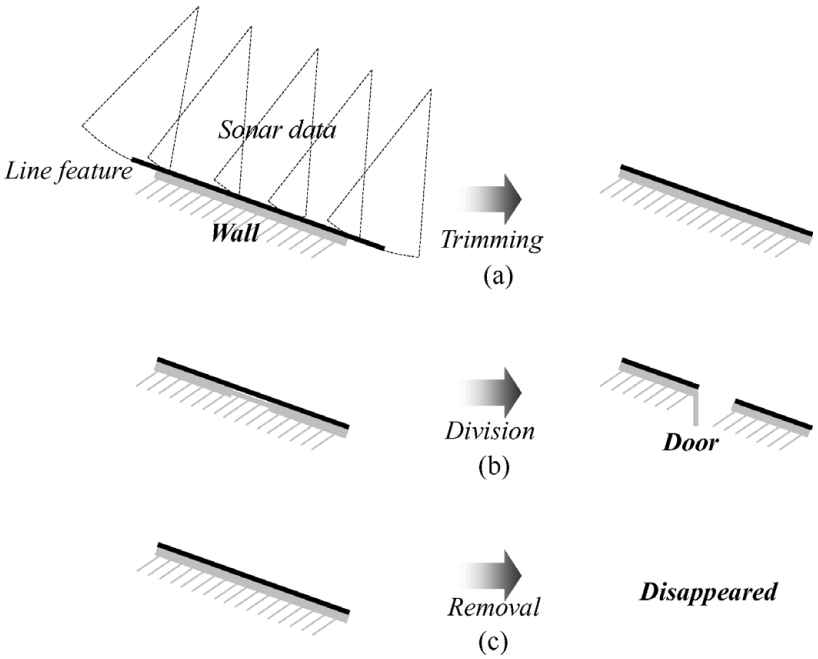


Fig. 5. Feature processing of a line feature showing (a) trimming, (b) division, and (c) removal.

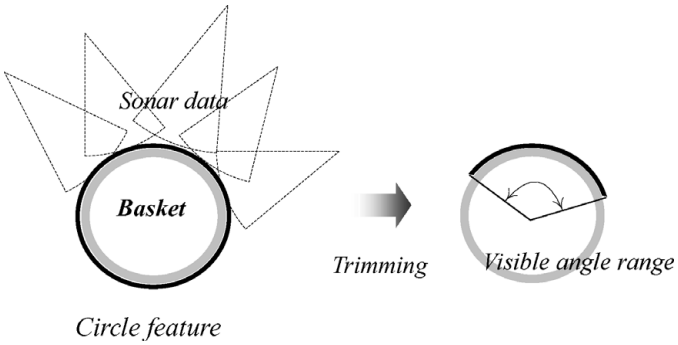


Fig. 6. Trimming the circle feature.

As usual, m_i is the ith grid on the map features and z_t is the measurement at time t. The constant l_o and the prior $l_{t-1}(m_i)$ are represented as a log odds ratio:

$$l_o = \log \frac{p(m_i)}{1 - p(m_i)}, \tag{7}$$

$$l_{t-1}(m_i) = \log \frac{p\,(m_i\,|z_{t-1}\,)}{1 - p\,(m_i\,|z_{t-1}\,)}. \tag{8}$$

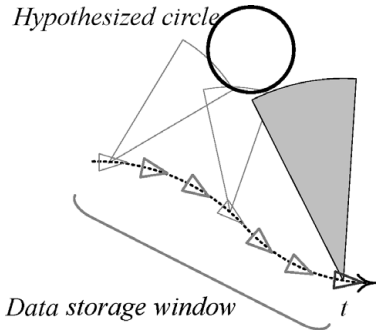


Fig. 7. Sets of sonar data associated with the hypothesized circle by the FPA model in the sonar returns taken from different robot positions.

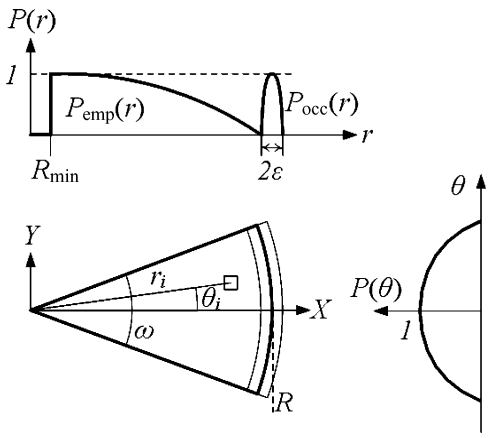


Fig. 8. Probability profiles of a sonar beam for P_{emp} and P_{occ}.

The left term can be a sensor model according to m_i and z_t in the log odds form

$$\begin{aligned}
&\text{sensor model}(m_i, z_t) = -\log \frac{P_{\text{emp},i}}{1 - P_{\text{emp},i}}, \ \text{cell } i \in \text{empty region} \\
&\text{sensor model}(m_i, z_t) = \log \frac{P_{\text{occ},i}}{1 - P_{\text{occ},i}}, \ \textit{cell } i \in \text{occupied region}
\end{aligned} \tag{9}$$

where $P_{\text{emp},i}$ and $P_{\text{occ},i}$ are characteristics of the sensor, i.e., $P_{\text{emp},i}$ and $P_{\text{occ},i}$ are functions of both the distance from the transducer and the angle from the transducer axis. The advantage of the log odds over the probability representation is that it avoids numerical instabilities for probabilities near zero or one.

Figure 8 shows the probability profiles of a sonar beam. The parameter ε is a gap for the occupancy region.

3.3. Grid association

The occupancy probability of grids squares estimated around the extracted features can be a good reference for determining

https://doi.org/10.1017/S026357470900561X Published online by Cambridge University Press

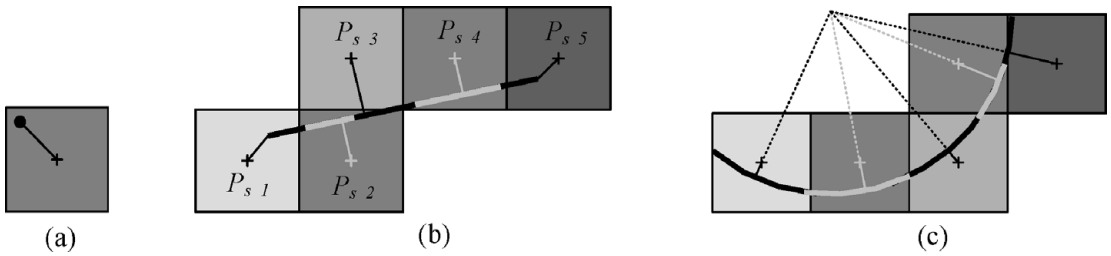


Fig. 9. Grid association for feature processing in the case of a (a) point, (b) line, and (c) circle.

whether the feature confidence is good or not. We divided the features into several segments determined by the grid array to evaluate the reliability of each segment, as shown in Fig. 9. The reliability of each segment is the same as the occupancy probability of the grid square for that segment. As shown in Fig. 9 (a), the reliability of the point feature can be related to the occupancy probability of the square that is the closest to the point location assuming that the center of the square is the location of that square. If the probability of the square under the point is greater than 0.7, the point is assumed to exist there. On the other hand, if the reliability of the point is less than 0.7, the feature is removed from the map.

In the case of a line feature, we can draw perpendicular lines from the centers of the squares around the feature, as shown in Fig. 9(b). The squares under each end of the line feature cover each end segment. The other segments are covered by each square by drawing a perpendicular line to the segment to evaluate the reliability. The line feature can be also treated using the same approach as in the point feature case. An overextracted line feature can therefore be made similar to the real object by removing the ends of the feature. One line feature can be divided into two lines by removing the middle portion of the feature.

In the case of a circle feature, we can draw the lines from centers of the grid squares around the feature toward the center of the circle feature, as shown in Fig. 9(c). The segments are covered by each grid square by drawing the line to the segment to evaluate the reliability, as in the case of a line feature. Therefore, the invisible part of the circle feature can be removed.

3.4. Mapping and map management

The methods in this paper are composed of three main procedures: feature mapping, occupancy grid updating and feature management. The procedure of feature mapping has some threshold parameters. The first one is the percentage of data association among data storages for each feature, $\lambda_{l,p,c}$, which represents the level of feature. It can be tuned according to the performance of a sensor and does not have to be adapted to new environments. The second one is a threshold for merging each feature segment. It is the geometric parameter for a simple feature association. It can be found according to the feature accuracy based on the performance of a sensor and also it does not have to be changed according to the environments. The feature extraction is executed at every 2.5 s.

The procedure of the occupancy grid update also has some design parameters. The first one is the data reliability, λ_g, which is defined as the number of data associated with the same hypothesized feature over the number of robot step within the data storage window. It also depends on not the environment but the performance of the sensor. The second parameter is a grid square size. It must be chosen by considering the environmental size and the computational load. The updating frequency for grids is about 5 Hz.

The threshold parameter for the procedure of the feature management is the threshold probability as mentioned in Section 3.3. That is, if the probability of the grid under the feature segment, p_{s_n}, is less than 0.7, the feature segment is removed. This parameter also does not depend on the environment. The process of trimming and division is executed automatically as the segments with probability less than 0.7 are removed. If the remaining segments of the feature are less than 30% by comparing to that of the original feature, then the feature is removed from the map. The feature management is executed at every 5 s.

4. Experimental Results

Experiments were carried out using the Pioneer-3DX differential drive robot equipped with a sonar ring composed of eight Murata sensors, each with a beam directivity of 45°, as shown in Fig. 11(a). The radius of the sonar sensor ring was 16 cm and the height above the ground was 37 cm. The Polaroid sonar sensors mounted on the robot by the manufacturer were not used. The robot explored two loops around the experimental environment, controlled manually at an average speed of 0.2 m/s. During this exploration phase, the sonar ring acquired range data at frequency of 5 Hz as shown in Fig. 10. Figure 11(b) shows the experimental environment of the corridor in the engineering building at Cheju National University. The gray circular and rectangular objects in Fig. 11(b) were removed after the first exploration of the robot to observe the performance of feature management in a dynamic environment. In Fig. 11(c), the small points represent all the raw sonar returns acquired by the sonar sensor ring. As expected, the shape of the data cloud was quite different from the true corridor shape because of the wide directivity and the specular reflection of the sonar beams. Table II shows the parameters used in the experiments.

Figure 12 shows the results of the feature map building without any map management after the first exploration loop and after both loops. All features were extracted at acceptable positions for the objects, as shown in Fig. 12(a). Most

https://doi.org/10.1017/S026357470900561X Published online by Cambridge University Press

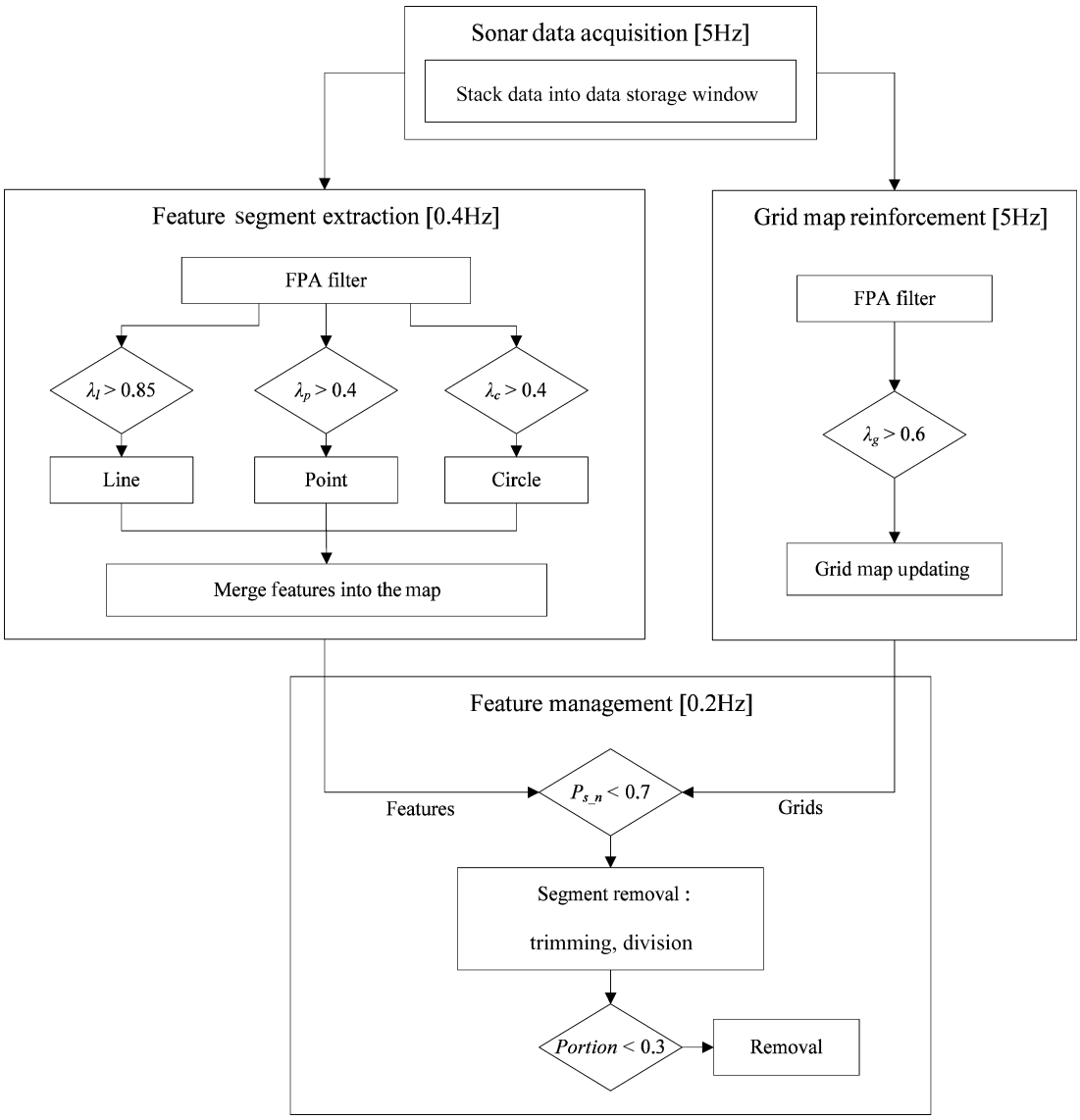


Fig. 10. Flowchart of feature extraction and management.

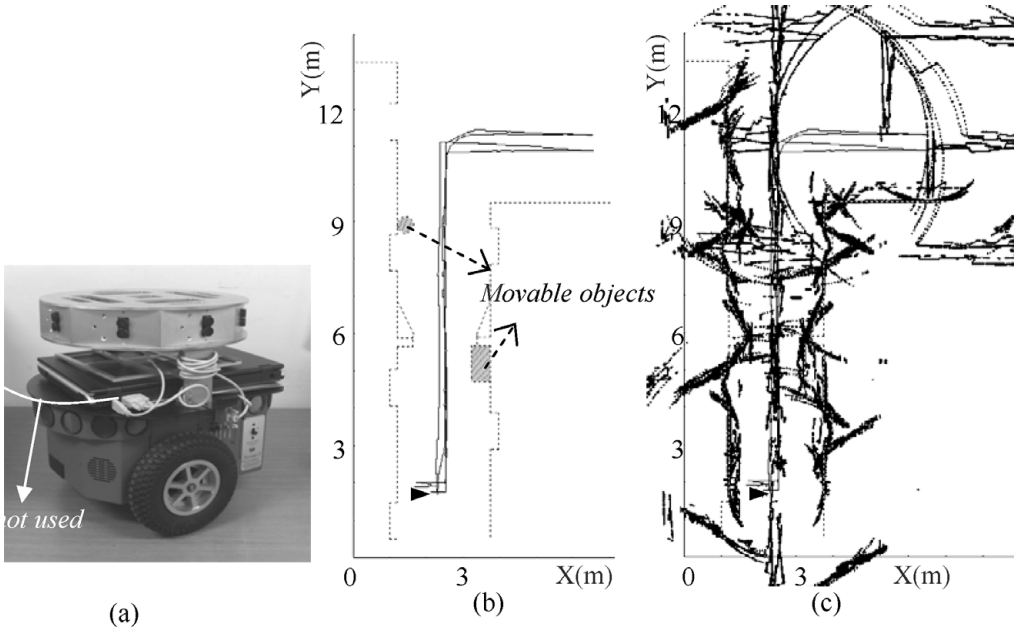


Fig. 11. (a) Pioneer-3DX mobile robot with a fixed sensor ring equipped with eight Murata sonar sensors, each with a beam directivity of roughly 45°. (b) True shape of the corridor and the robot path during the exploration phase. (c) Plot of all sonar returns based on the robot path.

https://doi.org/10.1017/S026357470900561X Published online by Cambridge University Press

Table II. Threshold parameters in all procedures.

Feature mapping	Percentage of data association for extracting a line	85%
	Percentage of data association for extracting point and circle	40%
	Line interval, angle difference of lines	0.2 m, 3°
	Point interval	0.3 m
	Circle's center interval	0.3 m
Occupancy grid updating	Percentage of data association for evaluating the data reliability	60%
	Grid square size	0.05 m
Feature managing	Grid probability threshold	0.7

features were also extracted and enhanced by observing the same surroundings more than once, as shown in Fig. 12(b). However, although one circular and one rectangular-shaped object were removed after the first exploration loop, the features extracted for these objects were not removed in Fig. 12(b). Furthermore, several line features were overextracted at the ends of the walls because of the characteristic wide beam width of the sonar beams.

Figure 13 compares the results of feature mapping after both exploration loops executed by the robot. As shown in Fig. 13, the features were processed by trimming, dividing, and removal. In particular, by removing the features for objects that had been removed, the overall map state was reasonably controlled in a dynamic environment that included movable objects. Moreover, the features overextracted by the wide sonar beam width were adjusted closely to the actual boundaries of the real objects. All processing referred to the occupancy probabilities of grids estimated for the features by the sonar sensor data. The gray regions in Fig. 13 show the grids near the features that had an occupancy probability of greater than 0.7. In this experiment, the robot trajectory was such that the position error cannot effect large on the performance of the feature extraction and

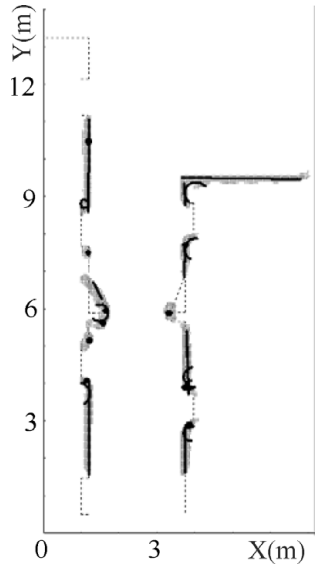


Fig. 13. Mapping results after finishing two exploration loops to check the feature evaluation performance.

grid updates. For very wide area where large distance drift can be a problem, division of some local area will be necessary to apply the method effectively.

5. Conclusions

This paper describes a new approach to managing the environmental map. The environmental map is built of primitive features, such as lines, points and even circles, extracted from ambiguous data from the sonar sensor ring. The features extracted in the dynamic environment that included nonstationary objects were processed for map management by trimming, division, or removal, depending on the circumstances. All processing referred to the occupancy probabilities of the grid squares generated for the features. The occupancy probability of the grid squares

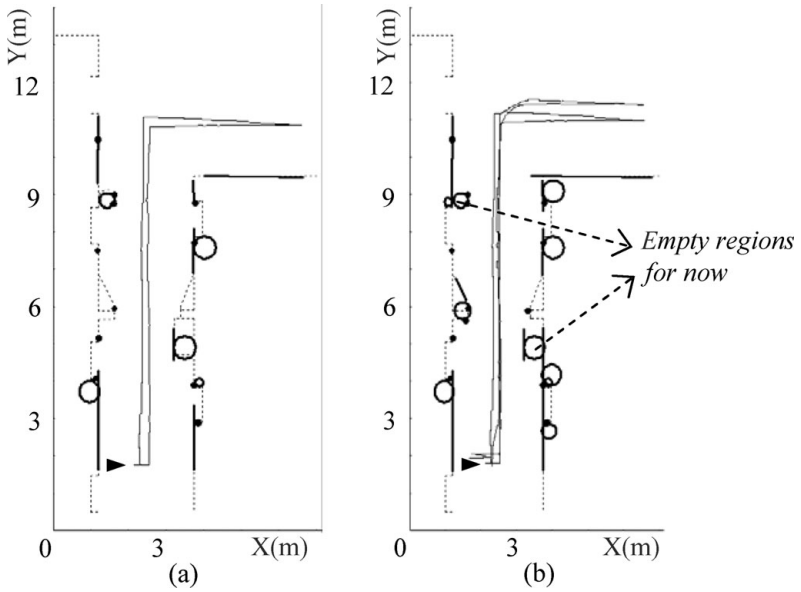


Fig. 12. Mapping results after finishing (a) one and (b) two exploration loops to check the feature evaluation performance.

https://doi.org/10.1017/S026357470900561X Published online by Cambridge University Press

was updated using the Bayesian updating model with the sonar sensor data. The experimental results demonstrated the validity of the proposed method.

Acknowledgments

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD, Basic Research Promotion Fund) (KRF-2008-521-D00032).

References

1. J. H. Lim, Map Construction, Exploration, and Position Estimation for an Autonomous Mobile Robot Using Sonar Sensors *Ph.D. Dissertation* (Korea: Pohang Institute of Science and Technology, 1994).
2. A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE J. Robot. Autom.* **RA-3**(3), 249–265 (1987).
3. S. Thrun, C. Martin, Y. Liu, D. Hahnel, R. Montemerlo, D. Chakrabarti and W. Burgard, "A real-time expectation-maximization algorithm for acquiring multiplanar maps of indoor environments with mobile robots," *IEEE Trans. Robot. Autom.* **20**, 433–442 (2004).
4. J.-H. Lim and J. J. Leonard, "Mobile robot relocation from echolocation constraints," *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(9), 1035–1041 (2000).
5. J.-H. Lim and C.-U. Kang, "Grid-based localization of a mobile robot using sonar sensors," *J. Mech. Sci. Technol.* **16**(3), 302–309 (2002).
6. S.-J. Lee, J.-H. Lim, C.-U. Kang, W.-K Chung and D.-W. Cho, "Feature Based Map Building Using Sparse Sonar Data," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Alberta, Canada (August, 2005) pp. 492–496.
7. S.-J. Lee, Y. Lee, W.-S. Yun, C.-U. Kang, J.-H. Lim, W.-K. Chung and D.-W. Cho, "Evaluation of Features Through Grid Association for Building a Sonar Map," *2006 International Conference on Robots and Automation*, Orlando, FL (May, 2006) pp. 2615–2620.
8. O. Wijk and H. I. Christensen, "Triangulation-based fusion of sonar data with application in robot pose tracking," *IEEE Trans. Robot. Autom.* **16**(6), 740–752 (2000).
9. J. L. Crowley, "Navigation for an intelligent mobile robot," *IEEE J. Robot. Autom.* **RA−1**(1), 31–41 (1985).
10. J. J. Leonard, *Direct Sonar Sensing for Mobile Robot Navigation* (Kluwer Academic Publishers, Dordrecht, The Netherlands, 1992).
11. J. J. Leonard and H. F. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," *IEEE Trans. Robot. Autom.* **7**(3), 376–382 (1991).
12. Q. Meng, Y. Sun and Z. Cao, "Adaptive extended Kalman filter (AEKF)-based mobile robot localization using sonar," *Robotica* **18**, 459–473 (2000).
13. J. D. Tardos, J. Neira, P. M. Newman and J. J. Leonard, "Robust mapping and localization in indoor environments using sonar data," *Int. J. Robot. Res.* **21**(4), 311–330 (2002).
14. A. Heale and L. Kleeman, "A Real Time DSP Sonar Echo Processor," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Takamatsu, Japan (2000) pp. 1261–1266.
15. D. Bank and T. Kampke, "High-resolution ultrasonic environment image," *IEEE Trans. Robot.* **23**(2), 370–381 (2007).
16. S. Fazli and L. Kleeman, "Simultaneous landmark classification, localization and map building for an advanced sonar ring," *Robotica* **25**, 283–296 (2007).
17. K. O. Arras, J. A. Castellanos, M. Schilt and R. Siegwart, "Feature-based multi-hypothesis localization and tracking using geometric constraints," *Robot. Auton. Syst.* **1056**, 1–13 (2003).
18. F. D. Wolf and G. S. Sukhatme, "Mobile robot simultaneous localization and mapping in dynamic environments," *Auton. Robots* **19**, 53–65 (2005).
19. A. J. Davison, "Real-time simultaneous localization and mapping with a single camera," *IEEE Int. Conf. Comput. Vis.* **2**, 1403–1410 (2003).
20. K. H. Lee, S. H. Kim and Y. K. Kwak, "Mobility improvement of an Internet-based robot system using the position prediction simulator," *Int. J. Precis. Eng. Manuf.* **6**(3), 29–36 (2005).
21. S.-J. Lee, J.-H. Lim and D.-W. Cho, "Feature-based map building using sparse sonar data in a home-like environment," *J. Mech. Sci. Technol.* **21**(1), 74–82 (2007).
22. S.-J. Lee, J.-H. Lim and D.-W. Cho, "Development of robust feature detector using sonar data," *J. Korean Soc. Precis. Eng.* **25**(2), 35–42 (2008).
23. G. Dissanayake, S. B. Williams, H. Durrant-Whyte and T. Bailey, "Map management for efficient simultaneous localization and mapping (SLAM)", *Auton. Robots* **12**, 267–286 (2002).
24. S. Thrun, W. Burgard and F. Fox, *Probabilistic Robotics* (MIT Press. Cambridge, MA, 2005).

https://doi.org/10.1017/S026357470900561X Published online by Cambridge University Press