Robotica (2007) volume 25, pp. 283–296. © 2006 Cambridge University Press
doi:10.1017/S0263574706003079 Printed in the United Kingdom

Simultaneous landmark classification, localization and map building for an advanced sonar ring

Saeid Fazli* and Lindsay Kleeman

ARC Centre for Perceptive and Intelligent Machines in Complex Environments (PIMCE), Intelligent Robotics Research Centre (IRRC), Department of Electrical and Computer Systems Engineering, Monash University, Australia. E-mail: lindsay.kleeman@eng.monash.edu.au

(Received in Final Form: July 28, 2006. First published online: September 4, 2006)

SUMMARY

An autonomous mobile robot operating in an unknown indoor environment often needs to map the environment while localizing within the map. Feature-based world models including line and point features are widely used by researchers. This paper presents a novel delayed-classification algorithm to categorize these features using a recently developed high-performance sonar ring within a simultaneous localization and map-building (SLAM) process. The sonar ring sensor accurately measures range and bearing to multiple targets at near real-time repetition rates of 11.5 Hz to 6 m range, and uses 24 simultaneously fired transmitters, 48 receivers and multiple echoes per receiver. The proposed algorithm is based on hypothesis generation and verification using the advanced sonar ring data and an extended Kalman filter (EKF) approach. It is capable of initiating new geometric features and classifying them within a short distance of travel of about 10 cm. For each new sonar reading not matching an existing feature, we initiate a pair of probational line and point features resulting from accurate range and bearing measurements. Later measurements are used to confirm or remove the probational features using EKF validation gates. The odometry error model of the filter allows for variations in effective wheel separation required by pneumatic robot tyres. The implementation of the novel classification and SLAM algorithm is discussed in this paper and experimental results using real sonar data are presented.

KEY WORDS: Sonar; Map building; Sonar ring; Classification; SLAM; Odometry

1. Introduction

Mapping is a fundamental requirement for a mobile robot to autonomously navigate in an *a priori* unknown indoor environment. The error growth of odometry dead-reckoning resulting from wheel distortion and slippage is usually unacceptable and environmental sensing is therefore needed. Building a map requires the interpretation of sensor information to estimate the locations of geometric features (landmarks) in a global reference map. This map is also used to determine the robot position in the environment. A large variety of sensor systems, such as sonar, radar, laser, infrared and vision are commonly used to make mobile robots capable of sensing their environment.[1–7] Also different world models such as feature-based and grid-based ones are widely used by researchers to map the environment.[8–13] To build a feature-based map, one has to decide what features to use to describe the robot's environment. Some researchers have used only line segments[14] and others have built maps containing lines and points.[2, 15, 16] Cylinders and arcs are rarely included in the feature set.[17] All measurements of geometric features are contaminated with background noise, false targets and phantom targets. The crucial problem in using sensory information to generate a map is the uncertainty in the origin of measurements and that in the robot position. Stochastic mapping using an extended Kalman filter (EKF) is widely used by researchers to solve this problem.[7, 11, 15, 16, 18–20]

Other crucial problem for sonar sensors is to classify the sensor information to discriminate planes, corners and edges. This is important in terms of being able to predict the measurements from a new position and hence apply an update procedure such as the Kalman filter. Some researchers have developed sonar sensors that allow target classification at one position.[16, 21–23] However, typical sonar sensors are commonly unable to classify the targets from one position because sonar observations are discrete points. Therefore, while navigating along a wall, the robot sees the wall not as a line but as a set of points. It is only through accumulation of observations over time that a correct feature classification can be made and a map can be constructed. Delayed decision-making techniques using Hough transforms are widely used by researchers to classify the landmarks.[8, 9, 24–28]

This paper presents a novel classification method within the simultaneous localization and map-building (SLAM) process. The proposed algorithm is based on delayed decision making using hypothesis generation and verification and accurate sonar data. The map considered in this paper is feature-based and consists of natural landmarks that occur in indoor environments; planes, corners and edges modeled into lines and points. The work presented here is the first reported mapping result from a recently developed fast and accurate sonar ring[29] mounted on the robot *Sombrero* shown in Fig. 1 in real indoor environments. Confirmation, deletion and initiation of new geometric features and SLAM process using EKF are included in the algorithm. If a new sonar reading is not associated with current features, a pair of probational features (a line and a point) will be initiated. The

*Corresponding author. E-mail: saeid.fazli@eng.monash.edu.au

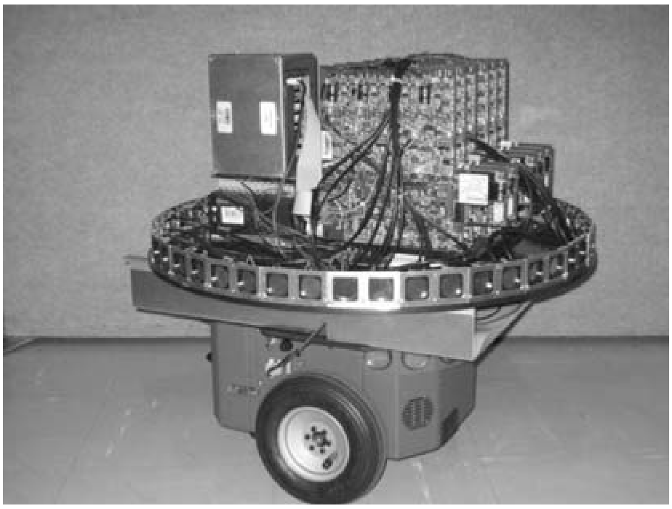

Fig. 1 The robot *Sombrero*.

point feature is initiated in the location of the observed target and the line feature, perpendicular to the line connecting the transmitter to the target. The validation gate condition of the Kalman filter is used to confirm or remove the probational features within the next 10 robot positions. Due to the high repetition rate of the advanced sonar ring of about 11.5 Hz, the confirmation and deletion process can be performed by the robot within 10 cm from where the feature is first sensed (or about 0.8 s at robot speeds of 10 cm/s). The high-speed classification helps to minimize the number of probational features and to speed up the mapping process. The odometry error model similar to the one in ref. 16 is used in this work, which incorporates variations in effective wheel separation and angle measurements. This model is suited to pneumatic tyre odometry errors where the wheel separation has been found to vary unpredictably with floor surface and path curvature. This paper also describes a novel sonar data-filtering technique to eliminate many phantom targets due to multipath echoes.

The paper is organized as follows: Section 2 contains an overview of the robot *Sombrero* and gives an overview of the software and hardware structure of the advanced sonar ring. Section 3 explains the details of the proposed landmark classification and the SLAM approach. Section 4 presents implementation details of the algorithm. Section 5 presents the experimental results to show the effectiveness of the proposed algorithm. Section 6 presents conclusions and a discussion of future work.

2. Overview of the Mobile Robot *Sombrero*

The robot *Sombrero* consists of an advanced sonar ring sensor mounted on an ActivMedia Pioneer 3 DX mobile robot. The locomotion method of the Sombrero is the differential drive method and the powered wheel steering (PWS) mechanism is used to control the robot motion. The advanced sonar ring works by the simultaneous firing of all transmitters emitting a burst of ultrasound in all directions, and then waiting for the echoes reflected from any objects within the sound beam. Then, the potential echo sample intervals are extracted from all 48 receivers using a thresholding method. These echo sample intervals are later processed on digital signal processors (DSPs) to obtain echo accurate arrival times.

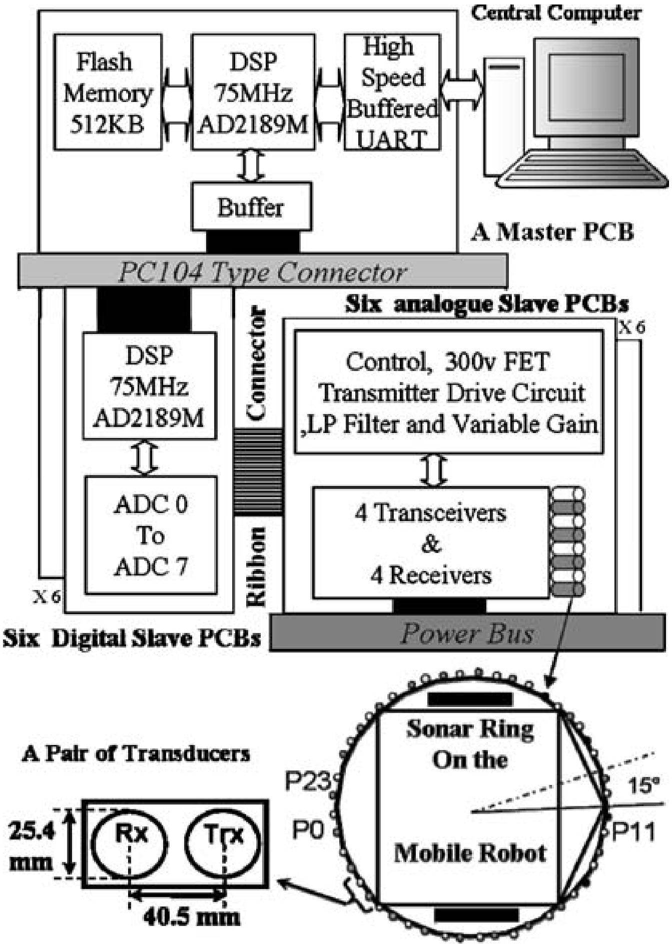

Fig. 2 The block diagram of the advanced sonar ring hardware.

To maximize the speed of the sensor and to be able to perform the task in the limited memory of the DSP, an interrupt service routine performs thresholding while the receivers are listening to echoes. The delays, known as the time-of-flight (TOF), are estimated for all the echoes reflected from different objects to every receiver in each firing. Then, the calculated distance-of-flight (DOF) of the returned echoes is twice the distance to the object. The bearing angle is determined by combining multiple measurements on different receivers. The basic idea is to calculate the TOF for each receiver by means of matched filtering (also called template matching), which is the minimum variance arrival time estimator in the presence of additive white Gaussian noise on the echo. A matched filter is based on finding the peak of the cross correlation of the echo with an *a priori* calculated template. This technique has been extensively used in refs. 15, 21, 22, and 30.

The field of view of the Polaroid 7000 transducers used in the advanced sonar ring is about 15°; therefore, 24 pairs of transducers are considered to cover the robot enabling it to estimate the bearing using one transceiver and one receiver in every pair. Figure 2 shows an overview of the hardware structure of the advanced sonar ring.

Six DSP boards called slave PCBs are designed to manage all 48 transducers. Each DSP processes the echoes returned

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

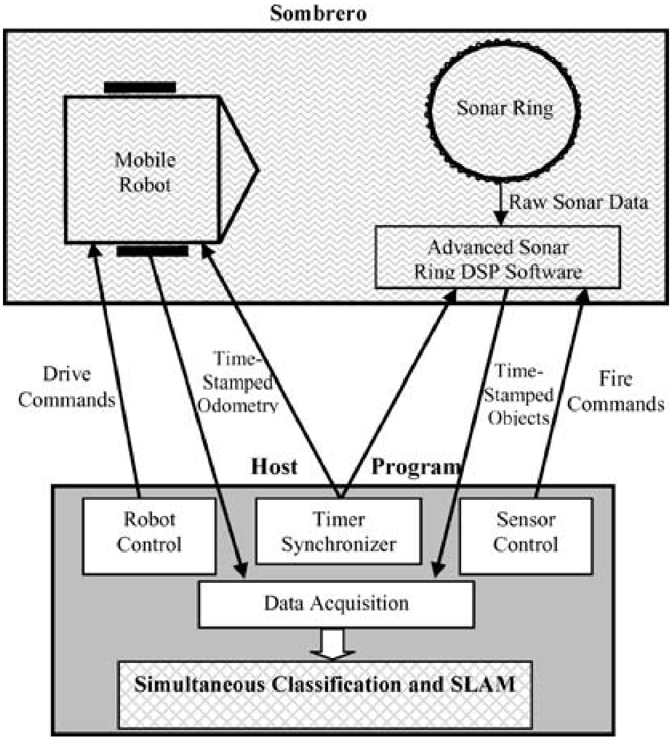

Fig. 3 Overview of the host software of *Sombrero*.

to the eight receiver channels. Each slave board is responsible for controlling the transmission and data acquisition process for four pairs of transceiver and receivers. Also, the board contains a high-voltage dc–dc converter to produce a 300-V bias on the eight transducers.

Finally, a master DSP is designed to communicate with and manage all the slave PCBs. A master DSP also in turn relays results of all slaves to a host computer over a serial line. One of the advantages of this configuration is that it relieves the computational burden of the host computer allowing computationally intensive applications to take place on a moving platform. Figure 3 shows an overview of the software structure of the host computer that controls the robot *Sombrero* and communicates with the sonar ring. The sonar ring can sense smooth targets, such as planes and right-angled corners, with an accuracy of approximately 0.6 mm and 0.2° for ranges from 0.2 to 4 m.[29]

3. Landmark Classification and SLAM for Autonomous Navigation

We assume that the actual three-dimensional (3-D) world model is orthogonal to the horizontal plane of the sonar ring, therefore, the environment can be adequately represented by a two-dimensional (2-D) map. A stochastic feature-based mapping method based on EKF is used for simultaneous localization of *Sombrero* and mapping of the environment. The method is based on the description by Davison,[6] where all map features are updated on each measurement result. In each robot pose, only one set of sensor results are applied to the SLAM in order to speed up the process. This means that if multiple firings occur in the same robot pose, the results of the first firing are applied to the algorithm and the rest are ignored. In the following Sections 3.1 and 3.2, a vehicle model, and its associated errors, is developed that allows odometry inputs to predict the motion of the robot. The Kalman filter facilitates the fusion of these odometry inputs with sonar measurements (discussed in Section 3.3) by predicting environmental features (discussed in Section 3.4), and correcting them using an error model of the sonar measurements as described in Section 3.5. Section 3.6 describes how the sonar features are classified and integrated into the map-building process.

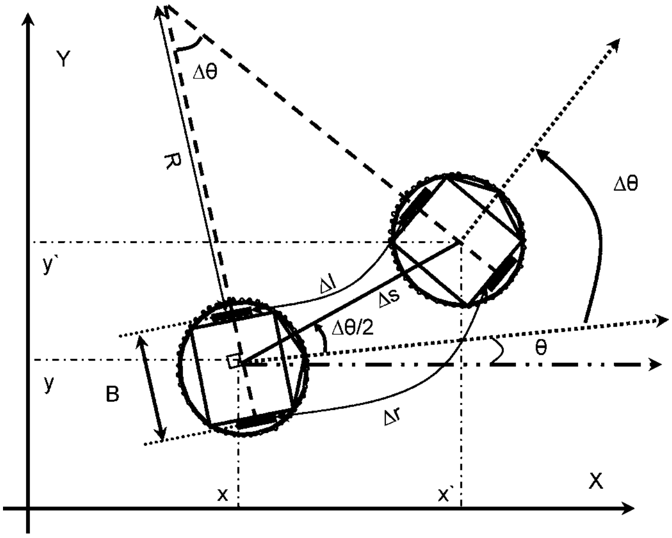

Fig. 4 Geometry of the robot movement.

3.1. *Mobile vehicle model*

The robot state vector is denoted as

$$\mathbf{x}_R = [\theta \quad x \quad y]^T \tag{1}$$

comprising a heading and the Cartesian location of the centre of the drive wheel axis defined with respect to the global coordinate frame shown in Fig. 4. The state vector of the robot is initialized to $[0 \quad 0 \quad 0]^T$.

Internal sensors (encoders) give information about the speed and the distance the robot wheels move. This information must be integrated to give the robot's global position and direction. On *Sombrero*, the encoder counts are accumulated for 20 ms, and then used to update the position of the robot. Based on the encoder counts of the wheel, the distance moved by each wheel is estimated. Therefore, at the end of 20 ms, the right wheel has traveled a distance Δr and the left wheel has traveled a distance Δl.

Figure 4 shows the path of the robot when it moves to a new position. The combined rotation and translation motion is shown in the figure. We assume that the robot travels in an arc while it translates a distance Δs and turns an angle $\Delta\theta$. By using the notation of Fig. 4, the distances traveled can be expressed as

$$\begin{cases} \Delta l = \Delta\theta R \\ \Delta r = \Delta\theta(R + B) \end{cases} \tag{2}$$

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

and therefore

$$\begin{cases}\Delta\theta = (\Delta r - \Delta l)/B \\ R = (\Delta l B)/(\Delta r - \Delta l).\end{cases} \tag{3}$$

From the law of cosines

$$\Delta s^2 = 2(R + B/2)^2 - 2(R + B/2)^2 \cos(\Delta\theta). \tag{4}$$

Using Eq. (3), Δs can be rewritten as

$$\Delta s = \frac{B(\Delta r + \Delta l)}{(\Delta r - \Delta l)} \sin(\Delta\theta/2) \overset{\Delta\theta\to 0}{\approx} (\Delta r + \Delta l)/2. \tag{5}$$

It can be proved that the angle between the translation vector (Δs) and the heading of the robot is $\Delta\theta/2$ as shown in Fig. 4. Therefore, if in the next time increment the robot moves a distance Δs and turns $\Delta\theta$, then the new position of the robot $[\theta' \quad x' \quad y']^T$ can be calculated as

$$\begin{cases}\theta' = \theta + \Delta\theta \\ x' = x + \Delta s \cos(\theta + \Delta\theta/2) \\ y' = y + \Delta s \sin(\theta + \Delta\theta/2).\end{cases} \tag{6}$$

3.2. The odometry error propagation under robot motion

We denote the robot state vector at time step k by $\mathbf{x}_{\mathrm{R}}(k) = [\theta(k) \quad x(k) \quad y(k)]^T$. Using Eqs. (3)–(6), the robot's motion through the environment is described as

$$\begin{aligned}\mathbf{x}_{\mathrm{R}}(k+1) &= \mathbf{x}_{\mathrm{R}}(k) + \mathbf{f}(\mathbf{x}_{\mathrm{R}}(k), \mathbf{y}(k)) \\ \mathbf{y}(k) &= [\Delta r \quad \Delta l \quad B]^T \\ \mathbf{f}(\mathbf{x}_{\mathrm{R}}(k), \mathbf{y}(k)) &= \begin{bmatrix} \dfrac{\Delta r - \Delta l}{B} \\ \dfrac{\Delta r + \Delta l}{2} \cos\left(\theta(k) + \dfrac{\Delta r - \Delta l}{2B}\right) \\ \dfrac{\Delta r + \Delta l}{2} \sin\left(\theta(k) + \dfrac{\Delta r - \Delta l}{2B}\right) \end{bmatrix}.\end{aligned} \tag{7}$$

As the robot moves, the odometry errors accumulate to make the robot's overall position very uncertain with respect to the global frame. This uncertainty is produced by two mechanisms: the addition of new position integration errors and the development of previous errors under motion. The error model explained in this section, which is similar to the one presented in ref. 16, considers noise on Δr, Δl and B. *Sombrero* has inflatable tyres that provide good traction at the expense of wheel separation uncertainty. Using a first-order Taylor expansion of equations given in Eq. (7), the iterative equation can be derived in terms of the Jacobians $\partial\mathbf{f}/\partial\mathbf{x}$ and $\partial\mathbf{f}/\partial\mathbf{y}$ (see ref. 16 for more details)

$$\Delta\mathbf{x}_{\mathrm{R}}(k+1) \cong \left(\mathbf{I} + \frac{\partial\mathbf{f}}{\partial\mathbf{x}}\right) \Delta\mathbf{x}_{\mathrm{R}}(k) + \frac{\partial\mathbf{f}}{\partial\mathbf{y}} \Delta\mathbf{y}(k). \tag{8}$$

The covariance of the error in the robot pose at step $k+1$ is now derived in terms of step k

$$\begin{aligned}\mathbf{P}_{\mathrm{RR}}(k+1) &\equiv E\langle \Delta\mathbf{x}_{\mathrm{R}}(k+1)\Delta\mathbf{x}_{\mathrm{R}}(k+1)^T\rangle \\ &= \left(\mathbf{I} + \frac{\partial\mathbf{f}}{\partial\mathbf{x}_{\mathrm{R}}}\right) E\langle \Delta\mathbf{x}_{\mathrm{R}}(k)\Delta\mathbf{x}_{\mathrm{R}}(k)^T\rangle \left(\mathbf{I} + \frac{\partial\mathbf{f}}{\partial\mathbf{x}_{\mathrm{R}}}\right)^T \\ &\quad + \frac{\partial\mathbf{f}}{\partial\mathbf{y}} E\langle \Delta\mathbf{y}(k)\Delta\mathbf{y}(k)^T\rangle \frac{\partial\mathbf{f}}{\partial\mathbf{y}}^T \\ &= \underbrace{\mathbf{P}_{\mathrm{RR}}(k) + \mathbf{P}_{\mathrm{RR}}(k)\frac{\partial\mathbf{f}^T}{\partial\mathbf{x}_{\mathrm{R}}} + \frac{\partial\mathbf{f}}{\partial\mathbf{x}_{\mathrm{R}}}\mathbf{P}_{\mathrm{RR}}(k) + \frac{\partial\mathbf{f}}{\partial\mathbf{x}_{\mathrm{R}}}\mathbf{P}_{\mathrm{RR}}(k)\frac{\partial\mathbf{f}^T}{\partial\mathbf{x}_{\mathrm{R}}}}_{\text{covariance propagation}} \\ &\quad + \underbrace{\frac{\partial\mathbf{f}}{\partial\mathbf{y}}\mathbf{Q}(k)\frac{\partial\mathbf{f}}{\partial\mathbf{y}}^T}_{\text{new error}}\end{aligned} \tag{9}$$

where E is the expectation and $\mathbf{Q}(k)$ is the error covariance of $\mathbf{y}(k)$, the wheel measurements and separation. We assume that in Eq. (9), the errors in robot pose $\mathbf{x}$ at time k and the new errors in odometry parameters $\mathbf{y}$ at time k are independent. The errors in Δr, Δl and B are assumed to be independent

$$\mathbf{Q} = \begin{bmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_l^2 & 0 \\ 0 & 0 & \sigma_B^2 \end{bmatrix} \tag{10}$$

The covariance update Eq. (9) can be calculated in a computationally effective manner as explained in ref. 16. The values for the variances σ_r^2 and σ_l^2 must be proportional to Δr and Δl at each step to make the system consistent, which means that for a given range or angle change, the final covariance should be independent of the number of steps to traverse a path.[16] Since the covariances add when independent noise segments are concatenated, these covariances must be proportional to the distance traveled as

$$\begin{cases}\sigma_r^2 = E^2|\Delta r| \\ \sigma_l^2 = E^2|\Delta l|\end{cases} \tag{11}$$

where E is the error standard deviation for a 1-m travel. Also from ref. 16 σ_B^2 can be written as

$$\sigma_B^2 = \frac{A^2B^2}{2\pi|\Delta\theta|} \tag{12}$$

where A is the angle error standard deviation for a full 2π revolution of the robot in one step that is attributed to variations in the wheel separation B.

3.3. Geometric feature models

The environmental landmarks are modeled as lines and points. The lines represent plane reflectors while the point features are used for corners and edges. The point feature is characterized by its 2-D Cartesian coordinates $[x_p \quad y_p]^T$. The plane feature is represented by a line with an angle φ and the minimum distance d to the global coordinate system origin as shown in Fig. 5. The line $[\varphi \quad d]^T$ can be represented

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

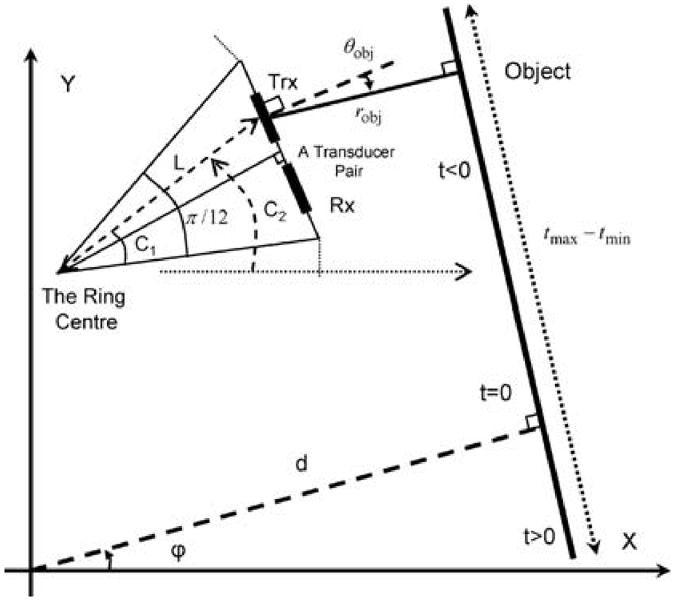

Fig. 5 Line representation.

by a parametric equation

$$\begin{cases} x = d\cos\varphi + t\sin\varphi \\ y = d\sin\varphi - t\cos\varphi \end{cases} \tag{13}$$

where t is the signed distance from the nearest point on the line to the origin. In order to model a partial plane, parameters $t_{\min}$ and $t_{\max}$ are saved to determine line endpoints. The length of the partial line is also represented by $t_{\max} - t_{\min}$. The parametric Eq. (13) can also be written as

$$g(x, y) = y\sin\varphi + x\cos\varphi - d = 0. \tag{14}$$

The representation $[\varphi \quad d]^T$ has the alternative of $[\varphi + \pi - d]^T$ that results in the same parametric equations except that t reverses sign. Here, the two lines are considered different since they correspond to sensing from different sides of the line. If the line were sensed from the other side of the coordinate origin, a negative distance parameter would be recorded. This representation differentiates between two sides of a partial plane. That means

$$\begin{cases} \text{if } (\operatorname{sign}(g(0,0)) == \operatorname{sign}(g(\mathrm{trx}_x, \mathrm{trx}_y))) \\ \qquad \Rightarrow \text{change } d \text{ to positive} \\ \text{else} \quad \Rightarrow \text{change } d \text{ to negative} \end{cases} \tag{15}$$

where $(\mathrm{trx}_x, \mathrm{trx}_y)$ are the coordinates of the transceiver that observes the partial plane (Fig. 5). New sonar measurements are only fused with the line if their t parameter lies in the line or in the extension of the line by less than 0.2 m. When an extension of the line is accepted by the EKF as described by the validation gate of Eq. (41) given later the values of $t_{\min}$ and $t_{\max}$ are updated.

3.4. Prediction of geometric feature positions

The advanced sonar ring consists of 24 transducer pairs numbered from 0 to 23 (see Fig. 2). Each pair has a transceiver and a receiver. The sensor fires all transceivers simultaneously. Then, the results of echo signal processing including the pair number that senses the reflector and the DOFs of the transducers are sent to the host program. This information is used to estimate the reflector state parameters as follows.

The geometry of sensing a plane, a corner and an edge are shown in Fig. 6. The label Trx is used for a transceiver and Rx for a receiver. The DOFs are labeled $\mathrm{dof}_{\mathrm{trx}}$ and $\mathrm{dof}_{\mathrm{rx}}$. Based on the geometry shown in Fig. 6, the incidence angle θ_{obj} can be calculated by applying the law of cosines

$$\theta_{\mathrm{obj}} = \begin{cases} \dfrac{\pi}{2} - \cos^{-1}\left(\dfrac{D^2 + \mathrm{dof_{trx}}^2 - \mathrm{dof_{rx}}^2}{2D\mathrm{dof_{trx}}}\right) \\ \text{for a corner or a plane} \\ \dfrac{\pi}{2} - \cos^{-1}\left(\dfrac{D^2 + (0.5\mathrm{dof_{trx}})^2 - (\mathrm{dof_{rx}} - 0.5\mathrm{dof_{trx}})^2}{2D(0.5\mathrm{dof_{trx}})}\right) \\ \text{for an edge} \end{cases} \tag{16}$$

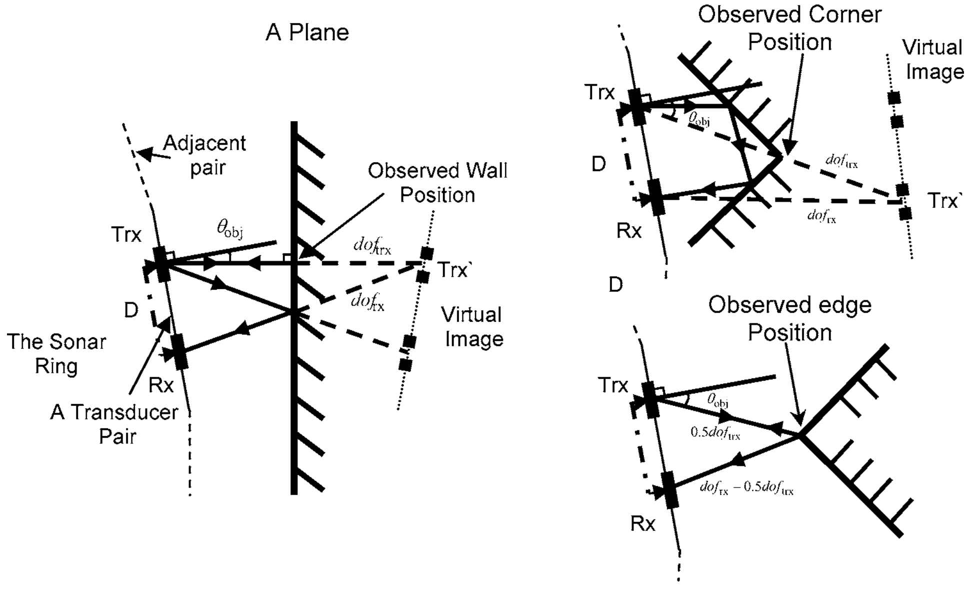

Fig. 6 The geometry of sensing a plane, a corner and an edge.

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

where D is the separation between the transceivers of each pair. The distance of an object to a transceiver, r_{obj} is determined by

$$r_{\text{obj}} = 0.5\,\text{dof}_{\text{trx}}. \tag{17}$$

As the map primitives are lines and points and there is no differentiation between corners and edges, the first equation in Eq. (16) is used to calculate the bearing angle of the reflector. However, the sensor does not classify the objects at the first step; therefore, the application assumes that the reflector is a corner or a plane feature and uses the same equation for all sonar measurements. This assumption is acceptable because the difference in results of both equations in Eq. (16) is usually small. For example, the difference is less than 0.4° for ranges greater than 1.5 m. When accuracy is important, a target classified as an edge can be adjusted in angle accordingly.

The coordinates of the transceiver that senses the object with respect to the global system is determined using the robot state $[\theta' \quad x' \quad y']^T$ and the definitions shown in Fig. 5

$$\begin{cases} \text{Trx}_x = x + L\cos(\theta + C_2) \\ \text{Trx}_y = y + L\sin(\theta + C_2) \\ C_2 \equiv \dfrac{\pi}{12}\,\text{pair}_{\text{no}} + \pi + C_1 \end{cases} \tag{18}$$

where pair_{no} is the sequence number of the transducer pair (P0 to P23 in Fig. 2). The parameters L, C_1 and C_2 are shown in Fig. 5 and are determined by the geometric design and construction of the sonar ring. The measured values of L and C_1, used in this work are 0.3083 m and 0.1966 rad, respectively.

Therefore, the position of the object can be written as

$$\begin{cases} \text{object}_x = \text{Trx}_x + r_{\text{obj}}\cos(\theta - \theta_{\text{obj}} + C_3) \\ \text{object}_y = \text{Trx}_y + r_{\text{obj}}\sin(\theta - \theta_{\text{obj}} + C_3) \\ C_3 \equiv \dfrac{\pi}{12}\,\text{pair}_{\text{no}} + \pi + \dfrac{\pi}{24}. \end{cases} \tag{19}$$

If we assume that the measured reflector is a partial plane, then using Eqs. (14), (18) and (19), we can calculate the parameters φ and d of the line feature that is perpendicular to the line between Trx and the object for the object position

$$\begin{cases} \varphi = \tan^{-1}\left(\dfrac{\text{object}_y - \text{trx}_y}{\text{object}_x - \text{trx}_x}\right) \\ d = \text{object}_y \sin\varphi + \text{object}_x \cos\varphi \end{cases} \tag{20}$$

Equation (15) is then applied to achieve a consistent representation.

If we assume that the target is a point feature then the coordinates of the target can be used as the feature parameters, and from Eqs. (18) and (19)

$$\begin{cases} x_p = x + L\cos(\theta + C2) + r_{\text{obj}}\cos(\theta - \theta_{\text{obj}} + C_3) \\ y_p = y + L\sin(\theta + C2) + r_{\text{obj}}\sin(\theta - \theta_{\text{obj}} + C_3). \end{cases} \tag{21}$$

3.5. Measurement model

At time step k, the robot obtains sonar measurements dof_{trx} and dof_{rx} from a pair of transducers. Then, using Eqs. (16) and (17), the measurement vector is formed as

$$\mathbf{z}(k) = [\theta_{\text{obj}}(\text{radian})\ r_{\text{obj}}(\text{metre})]^T. \tag{22}$$

However, the value of measurement $\mathbf{z}(k)$ is a function of the robot state at time k and the location of the feature from which it originated, subject to a noise disturbance, as given by the measurement model

$$\begin{cases} \mathbf{z}(k) = \mathbf{h}(k, \mathbf{x}(k)) + \mathbf{w}(k) \\ \mathbf{x}(k) = [\mathbf{x}_{\text{R}}(k)\,\mathbf{x}_1(k)\ldots\mathbf{x}_n(k)]^T \end{cases} \tag{23}$$

where $\mathbf{x}(k)$ is the system state vector including the robot state vector and all n geometric features. The measurement model $\mathbf{h}$, takes different forms depending on the type of feature, and $\mathbf{w}$ is the measurement noise with error covariance which is defined as

$$\mathbf{R}(k) = E\left\langle \mathbf{w}(k)\mathbf{w}(k)^T \right\rangle = \begin{bmatrix} \left(2^\circ \times \dfrac{\pi}{180}\right)^2 & 0 \\ 0 & (0.004\,\text{m})^2 \end{bmatrix}. \tag{24}$$

To define the measurement model $\mathbf{h}$, of a plane reflector, we assume that the line state $[\varphi \quad d]$ is known. Then, by using the robot state and pair_{no} at time step k, the measurement elements θ_{obj} and r_{obj} are calculated as

$$\begin{cases} \theta_{\text{obj}} = \theta - \varphi + C_3 \\ r_{\text{obj}} = |(x + L\cos(\theta + C_2))\cos\varphi \\ \qquad\quad + (y + L\sin(\theta + C_2))\sin\varphi - d|. \end{cases} \tag{25}$$

To define measurement model $\mathbf{h}$ for a corner or an edge reflector, we assume that the point state $[x_{\text{p}} \quad y_{\text{p}}]$ is known. Then by using the robot state and pair_{no} at time step k, the measurement elements θ_{obj} and r_{obj} are derived as

$$\begin{cases} \theta_{\text{obj}} = \theta + C_3 - \tan^{-1}\left(\dfrac{y_{\text{p}} - y - L\sin(\theta + C_2)}{x_{\text{p}} - x - L\cos(\theta + C_2)}\right) \\ r_{\text{obj}} = \sqrt{(x_{\text{p}} - x - L\cos(\theta + C_2))^2 + (y_{\text{p}} - y - L\sin(\theta + C_2))^2}. \end{cases} \tag{26}$$

3.6. Proposed classification and SLAM algorithm

Accurate measurements of the range and bearing angle are performed within the multi-DSP architecture of the advanced sonar ring as explained in Section 2. Each measurement introduces a dot in the world frame that can be part of a partial plane or the position of a corner (or an edge). The advantage of accurate bearing angles provided by the advanced sonar ring[29, 31, 32] is that it allows us to precisely calculate the state parameters of the predicted partial plane from each measurement result. The predicted plane feature is a line perpendicular to the line between the transmitter

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

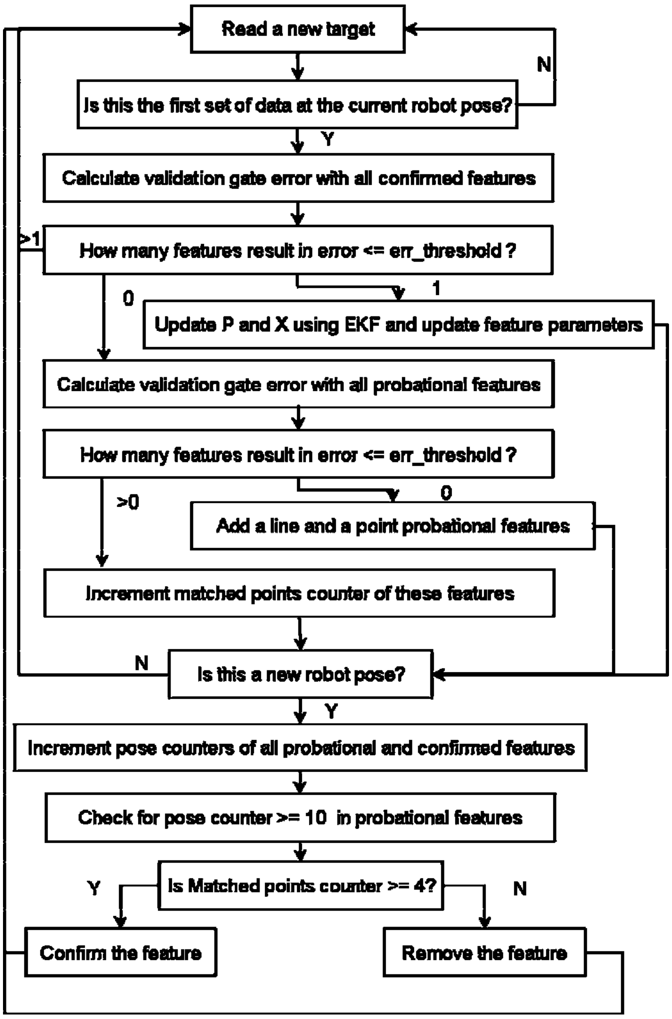
Read a new target
Is this the first set of data at the current robot pose?
N
Y
Calculate validation gate error with all confirmed features
>1
How many features result in error <= err_threshold ?
0
1
Update P and X using EKF and update feature parameters
Calculate validation gate error with all probational features
How many features result in error <= err_threshold ?
>0
0
Add a line and a point probational features
Increment matched points counter of these features
N
Is this a new robot pose?
Y
Increment pose counters of all probational and confirmed features
Check for pose counter >= 10 in probational features
Y
Is Matched points counter >= 4?
N
Confirm the feature
Remove the feature

Fig. 7 The block diagram of the proposed classification and SLAM algorithm. The *error* and *err_threshold* are explained after Eq. (41) and relate to the validation gate of the Kalman filter. The 'matched points counter' is incremented when a measurement falls in the validation gate.

and the object in the object position. A predicted corner or edge feature is a point in the measured reflector position as explained in Section 3.4. This prediction of the object position from one sonar result is not possible for conventional sonar sensors due to a large uncertainty in the bearing angle. The proposed classification and SLAM algorithm uses an EKF to update the robot position and the features' positions and also to classify the targets. It uses the method introduced by Davison,[6] where all map features are updated on each measurement result. The method here differs from that of Davison[6] in the sensor employed (i.e. sonar ring) and the method for classifying the sonar targets as the robot moves, as described later. Figure 7 shows a flow chart of the proposed classification and SLAM method that is detailed later.

Measurement to feature association plays an important role for both the classification and the SLAM processes. In fact, data association (which feature a measurement is generated from), can significantly affect the map. The validation gate condition for a Kalman filter is used for data association. This is explained in detail in Section 4. The new measurements can either generate new map features or be fused to the existing ones. After each association, all error covariance and cross-covariance matrices are updated by the fusion of a new result.

The classification algorithm is based on multi-hypothesis generation and verification. If a new result is not associated with the existing features, two probational features as hypotheses, are generated, i.e. a line and a point. The state parameters of these features are calculated from single measurement result as explained in Section 3.4. The robot inserts them into its probational feature matrix. The features are partitioned into probational and confirmed features, with only the confirmed features being used for SLAM. The verification is performed based on the association results on sensor measurements within the robot's next 10 positions. The repetition rate of the sensor is 11.5 Hz.[29] Therefore, the verification process is performed within about 1 s when the robot is moving. The method is capable of initiation, confirmation and deletion of the geometric features. After confirmation, the feature is moved from probational to the confirmed partition and is used in the SLAM process.

4. Implementation Details

As explained earlier, the robot state vector at time step k, $\mathbf{x}_\mathrm{R}(k)$ and state vectors of the confirmed features $\mathbf{x}_i(k)$ comprise the system state vector $\mathbf{x}(k) = [\mathbf{x}_\mathrm{R}(k) \quad \mathbf{x}_1(k) \cdots \mathbf{x}_n(k)]^T$, where n is the number of the confirmed features and $\mathbf{x}_i(k)$ is $[\varphi \quad d]^T$ for line features and $[x_\mathrm{p} \quad y_\mathrm{p}]^T$ for point features. The measurement model of the system $\mathbf{z}(k)$ is defined in Eq. (23). The objective of SLAM is to use the EKF to recursively compute an estimate for $\mathbf{x}(k)$

$$\hat{\mathbf{x}}(k) = [\hat{\mathbf{x}}_\mathrm{R}(k) \quad \hat{\mathbf{x}}_1(k) \cdots \hat{\mathbf{x}}_n(k)]^T \tag{27}$$

which is called the system state estimate, and its covariance is called the system state prediction covariance matrix

$$\mathbf{P}(k \mid k) = \begin{bmatrix} \mathbf{p}_{\mathrm{RR}}(k \mid k) & \mathbf{p}_{\mathrm{R1}}(k \mid k) & \ldots & \mathbf{p}_{\mathrm{R}n}(k \mid k) \\ \mathbf{p}_{\mathrm{R1}}(k \mid k) & \mathbf{p}_{11}(k \mid k) & \ldots & \mathbf{p}_{1n}(k \mid k) \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ \mathbf{p}_{\mathrm{R}n}(k \mid k) & \mathbf{p}_{1n}(k \mid k) & \ldots & \mathbf{p}_{nn}(k \mid k) \end{bmatrix} \tag{28}$$

where $\mathbf{p}_{\mathrm{R}i}(k \mid k)$ is a robot-to-feature cross-covariance matrix at time step k given measurements $\mathbf{z}(k)$, $\mathbf{z}(k-1)$, ..., $\mathbf{p}_{ij}(k \mid k)$ is a feature-to-feature cross-covariance matrix and a covariance of error in the feature state is denoted as $\mathbf{p}_{ii}(k \mid k)$.

The EKF can be formulated into a two-stage process as

(1) System state prediction: $\hat{\mathbf{x}}(k+1 \mid k)$—estimate of $\mathbf{x}$ at step $k+1$ given measurements $\mathbf{z}(k)$, $\mathbf{z}(k-1), \ldots$;
(2) System state update: $\hat{\mathbf{x}}(k+1 \mid k+1)$ and $\mathbf{p}(k+1 \mid k+1)$.

In order to perform the first stage, such a system state is estimated that contains the robot position vector and all the features states. Robot state prediction $\hat{\mathbf{x}}_\mathrm{R}(k+1 \mid k)$ is calculated using Δr and Δl, given by the odometry system as explained in Section 3.2, Eq. (7). The prediction of features

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

states $\hat{\mathbf{x}}_i(k+1\,|\,k)$, due to the assumption of a stationary environment, is taken to be the same as $\hat{\mathbf{x}}_i(k\,|\,k)$.

Stage two of the EKF can be divided into state estimation and state covariance estimation.

State estimation consists of

(1) measurement prediction:

$$\hat{\mathbf{z}}(k+1\,|\,k) = \mathbf{h}(k+1,\ \hat{\mathbf{x}}(k+1\,|\,k)); \tag{29}$$

(2) measurement residual:

$$\mathbf{v}(k+1) = \mathbf{z}(k+1) - \hat{\mathbf{z}}(k+1\,|\,k); \tag{30}$$

(3) updated state estimate:

$$\hat{\mathbf{x}}(k+1\,|\,k+1) = \hat{\mathbf{x}}(k+1\,|\,k) + \mathbf{W}(k+1)\mathbf{v}(k+1) \tag{31}$$

where $\mathbf{W}(k+1)$ is called the Kalman gain defined later in the state covariance estimation.

State covariance estimation consists of

(1) state prediction covariance:

$$\mathbf{P}(k+1\,|\,k) = \begin{bmatrix} \mathbf{p}_{\mathrm{RR}}(k+1\,|\,k) & \mathbf{p}_{\mathrm{R}1}(k+1\,|\,k) & \ldots & \mathbf{p}_{\mathrm{R}n}(k+1\,|\,k) \\ \mathbf{p}_{\mathrm{R}1}(k+1\,|\,k) & \mathbf{p}_{11}(k+1\,|\,k) & \ldots & \mathbf{p}_{1n}(k+1\,|\,k) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{p}_{\mathrm{R}n}(k+1\,|\,k) & \mathbf{p}_{1n}(k+1\,|\,k) & \ldots & \mathbf{p}_{nn}(k+1\,|\,k) \end{bmatrix} \tag{32}$$

where $\mathbf{p}_{\mathrm{RR}}(k+1\,|\,k)$ is calculated based on Eq. (9), and other elements are taken from the previous time step;

(2) residual covariance or innovation covariance:[33]

$$\mathbf{S}(k+1) = \mathbf{R}(k+1) + \mathbf{H}(k+1)\mathbf{p}(k+1\,|\,k)\mathbf{H}(k+1)^T \tag{33}$$

where $\mathbf{R}(k+1)$ is defined in Eq. (24) and $\mathbf{H}(k+1)$ is a Jacobian evaluated as

$$\mathbf{H}(k+1) = \left.\frac{\partial \mathbf{h}(k+1)}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}(k+1|k)}; \tag{34}$$

(3) filter gain:

$$\mathbf{W}(k+1) = \mathbf{p}(k+1\,|\,k)\mathbf{H}(k+1)^T\mathbf{S}(k+1)^{-1}; \tag{35}$$

(4) updated state covariance based on Joseph's form:

$$\begin{aligned} \mathbf{p}(k+1\,|\,k+1) = {} & [\mathbf{I} - \mathbf{W}(k+1)\mathbf{H}(k+1)]\ \mathbf{p}(k+1\,|\,k) \\ & \times [\mathbf{I} - \mathbf{W}(k+1)\mathbf{H}(k+1)]^T \\ & + \mathbf{W}(k+)\mathbf{R}(k+1)\mathbf{W}(k+1)^T. \end{aligned} \tag{36}$$

Based on Eqs. (25) and (26), the Jacobians of $\mathbf{h}(k+1)$ have different forms for line and point features. For a line feature, $\mathbf{H}$ is calculated as

$$\begin{aligned} \mathbf{H} &= \begin{bmatrix} \mathbf{H}_{11} & \mathbf{H}_{12} & \mathbf{H}_{13} & \mathbf{H}_{14} & \mathbf{H}_{15} \\ \mathbf{H}_{21} & \mathbf{H}_{22} & \mathbf{H}_{23} & \mathbf{H}_{24} & \mathbf{H}_{25} \end{bmatrix} \\ \mathbf{H}_{11} &= \frac{\partial \theta_{\mathrm{obj}}}{\partial \theta} = 1 \\ \mathbf{H}_{12} &= \frac{\partial \theta_{\mathrm{obj}}}{\partial x} = 0 \\ \mathbf{H}_{13} &= \frac{\partial \theta_{\mathrm{obj}}}{\partial y} = 0 \\ \mathbf{H}_{14} &= \frac{\partial \theta_{\mathrm{obj}}}{\partial \varphi} = -1 \\ \mathbf{H}_{15} &= \frac{\partial \theta_{\mathrm{obj}}}{\partial d} = 0 \\ \mathbf{H}_{21} &= \frac{\partial r_{\mathrm{obj}}}{\partial \theta} = \operatorname{sign}(A)(-L\cos\varphi\sin(\theta+C_2) \\ &\quad + L\sin\varphi\cos(\theta+C_2)) \\ \mathbf{H}_{22} &= \frac{\partial r_{\mathrm{obj}}}{\partial x} = \operatorname{sign}(A)\cos\varphi \\ \mathbf{H}_{23} &= \frac{\partial r_{\mathrm{obj}}}{\partial y} = \operatorname{sign}(A)\sin\varphi \\ \mathbf{H}_{24} &= \frac{\partial r_{\mathrm{obj}}}{\partial \varphi} = \operatorname{sign}(A)(-(x+L\cos(\theta+C_2))\sin\varphi \\ &\quad + (y+L\sin(\theta+C_2))\cos\varphi) \\ \mathbf{H}_{25} &= \frac{\partial r_{\mathrm{obj}}}{\partial d} = -\operatorname{sign}(A) \end{aligned} \tag{37}$$

where A is defined as

$$\begin{aligned} A \equiv {} & (x + L\cos(\theta+C_2))\cos\varphi \\ & + (y + L\sin(\theta+C_2))\sin\varphi - d. \end{aligned} \tag{38}$$

For a point feature, the Jacobian matrix $\mathbf{H}$ is calculated as

$$\begin{aligned} \mathbf{H}_{11} &= \frac{\partial \theta_{\mathrm{obj}}}{\partial \theta} \\ &= 1 - \left(\frac{1}{1+A_4^2}\,\frac{-L\cos(\theta+C_2)A_3 - L\sin(\theta+C_2)A_2}{A_3^2}\right) \\ \mathbf{H}_{12} &= \frac{\partial \theta_{\mathrm{obj}}}{\partial x} = \frac{-1}{1+A_4^2}\,\frac{A_2}{A_3^2} \\ \mathbf{H}_{13} &= \frac{\partial \theta_{\mathrm{obj}}}{\partial y} = \frac{-1}{1+A_4^2}\,\frac{-1}{A_3} \\ \mathbf{H}_{14} &= \frac{\partial \theta_{\mathrm{obj}}}{\partial x_{\mathrm{p}}} = \frac{-1}{1+A_4^2}\,\frac{-A_2}{A_3^2} \\ \mathbf{H}_{15} &= \frac{\partial \theta_{\mathrm{obj}}}{\partial y_{\mathrm{p}}} = \frac{-1}{1+A_4^2}\,\frac{1}{A_3} \\ \mathbf{H}_{21} &= \frac{\partial r_{\mathrm{obj}}}{\partial \theta} = \frac{1}{2\sqrt{A_1}}(2L\sin(\theta+C_2)A_3 \\ &\quad -2L\cos(\theta+C_2)A_2) \\ \mathbf{H}_{22} &= \frac{\partial r_{\mathrm{obj}}}{\partial x} = \frac{1}{2\sqrt{A_1}}(-2A_3) \\ \mathbf{H}_{23} &= \frac{\partial r_{\mathrm{obj}}}{\partial y} = \frac{1}{2\sqrt{A_1}}(-2A_2) \\ \mathbf{H}_{24} &= \frac{\partial r_{\mathrm{obj}}}{\partial x_{\mathrm{p}}} = \frac{1}{2\sqrt{A_1}}(2A_3) \\ \mathbf{H}_{25} &= \frac{\partial r_{\mathrm{obj}}}{\partial y_{\mathrm{p}}} = \frac{1}{2\sqrt{A_1}}(2A_2) \end{aligned} \tag{39}$$

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

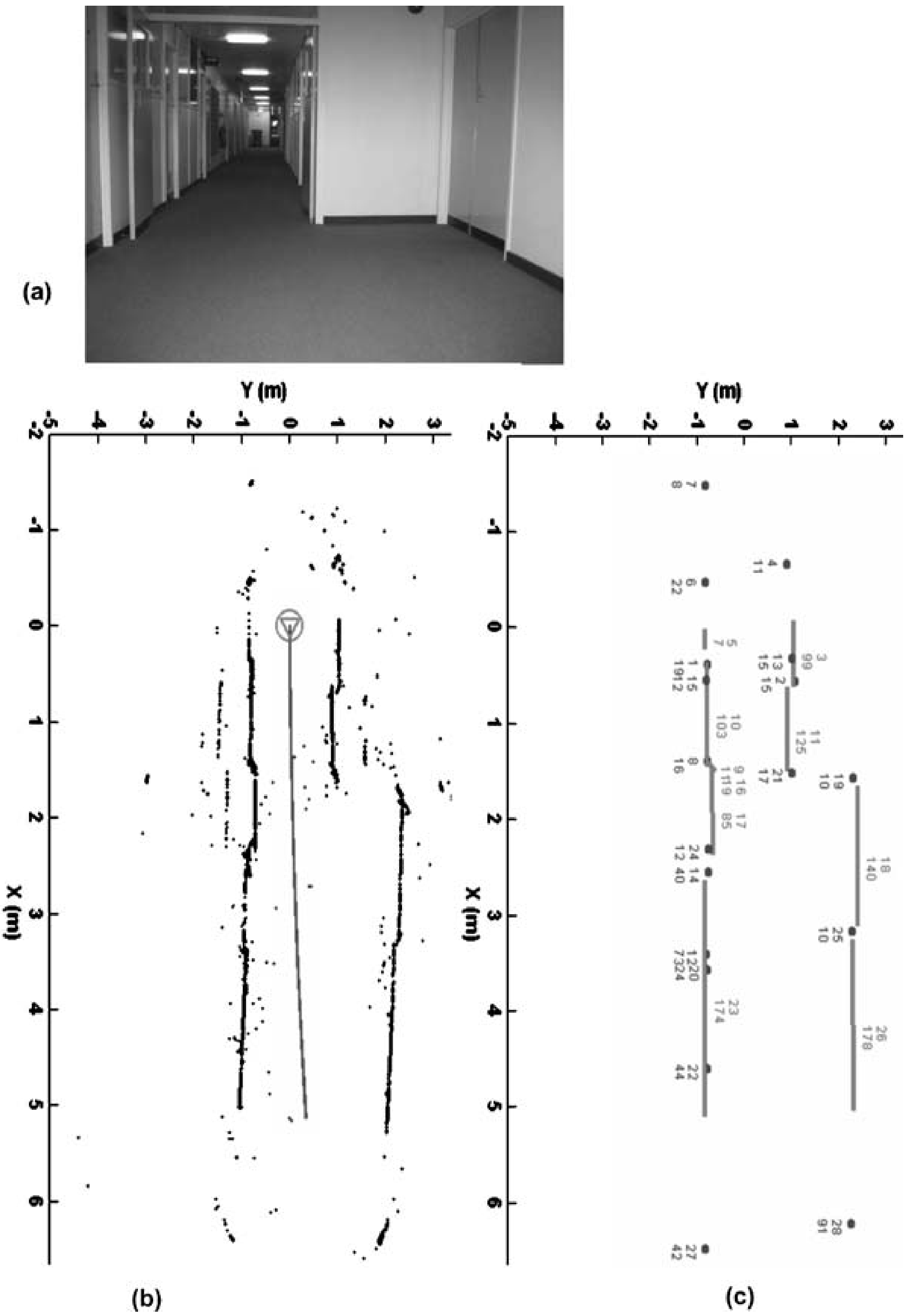

Fig. 8 (a) A corridor where the robot travels about 5 m, classifies line and point features and generates a map. The map features are labeled with a sequence number and the number of associations. (b) The robot trajectory based on odometry and the raw sonar data. (c) The map after application of SLAM EKF filtering and classification.

where A_1 to A_4 are defined as

$$\begin{aligned} A_2 &\equiv y_{\mathrm{p}} - y - L\sin(\theta + C_2) \\ A_3 &\equiv x_{\mathrm{p}} - x - L\cos(\theta + C_2) \\ A_1 &\equiv A_2^2 + A_3^2 \\ A_4 &\equiv A_2/A_3. \end{aligned} \tag{40}$$

A validation gate is used for association as follows

$$e^2 = \mathbf{v}(k+1)^T \mathbf{S}(k+1)^{-1} \mathbf{v}(k+1) \leq 9. \tag{41}$$

The validation gate threshold, referred to as *err_threshold* in Fig. 7, is 9 in Eq. (41) and the *error* that is compared to this threshold in Fig. 7 refers to e in Eq. (41).

5. Experimental Results

Experiments have been carried out in different real indoor environments. The robot *Sombrero* traveled at different speeds and collected on-the-fly range and bearing measurements using the advanced sonar ring. The sonar data were used in the offline classification and the SLAM algorithm was implemented with Matlab.

5.1. A corridor

Figure 8 shows a corridor where the robot travels about 5 m, classifies the line and point features and generates a map. Raw sonar data and the results of the proposed algorithm are shown in the figure. The map features are labeled with

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

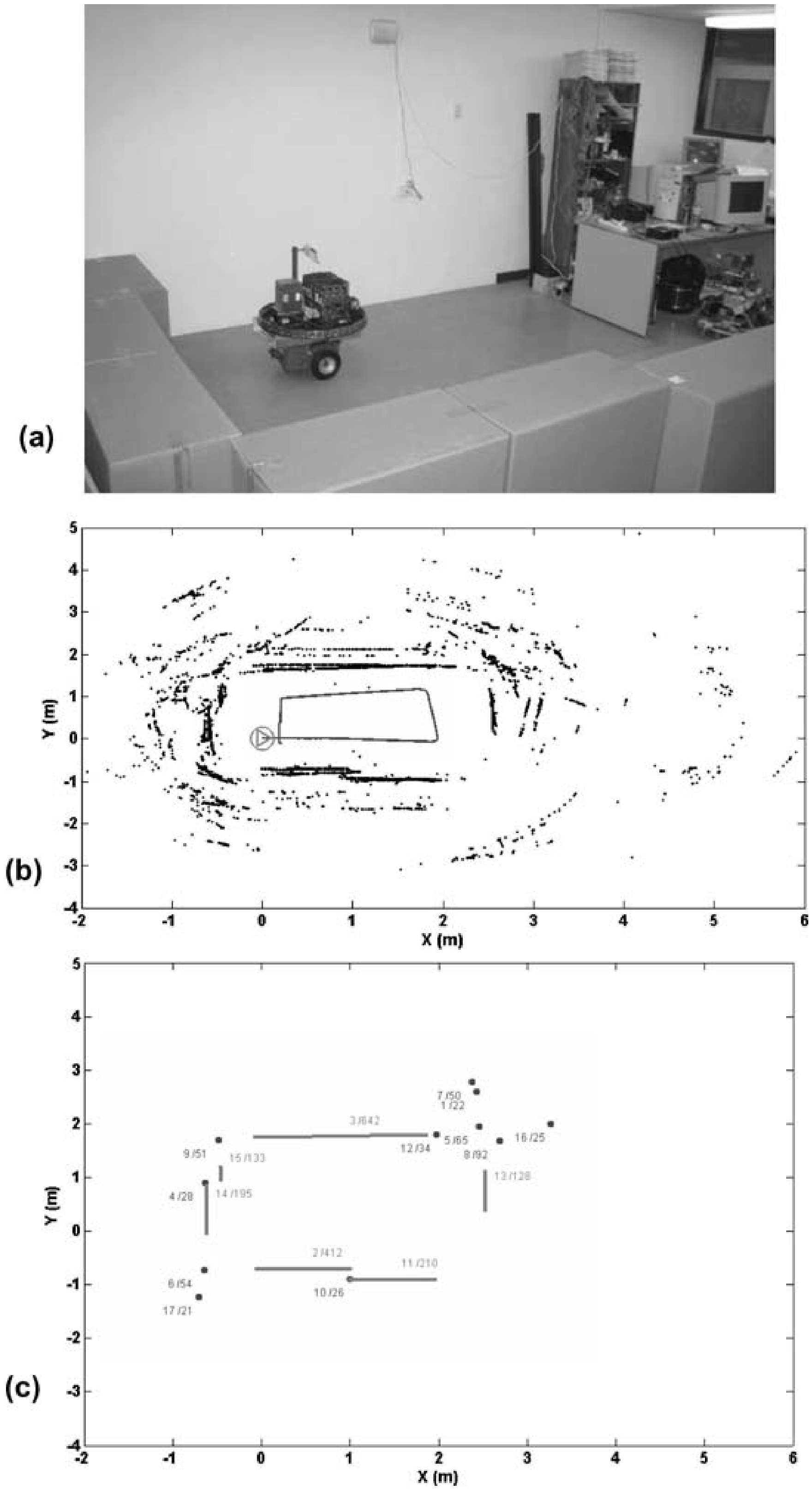

Fig. 9 (a) The robot travels in a rectangular path in the small lab environment consisting of a wall, cardboard boxes and lab equipment. (b) The robot trajectory based on odometry only and the raw sonar data. (c) The SLAM map.

a sequence number and the numbers of associations. The numbers of the associations for the line features are quite large depending on the length of the feature. This is due to the low travel speed of 10 cm/s and high sensing rate of 11.5 Hz. However, for the point features, the numbers of associations are not as large because of the limited viewing angle of the point features and weak reflections from the edges that make them invisible in some robot positions even inside their viewing angle.

5.2. *Small indoor environment*

Figure 9 shows the outlines of the testing environment. Raw sonar data and the generated 2-D map are also shown in the figure. The robot travels in a rectangular path in the

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

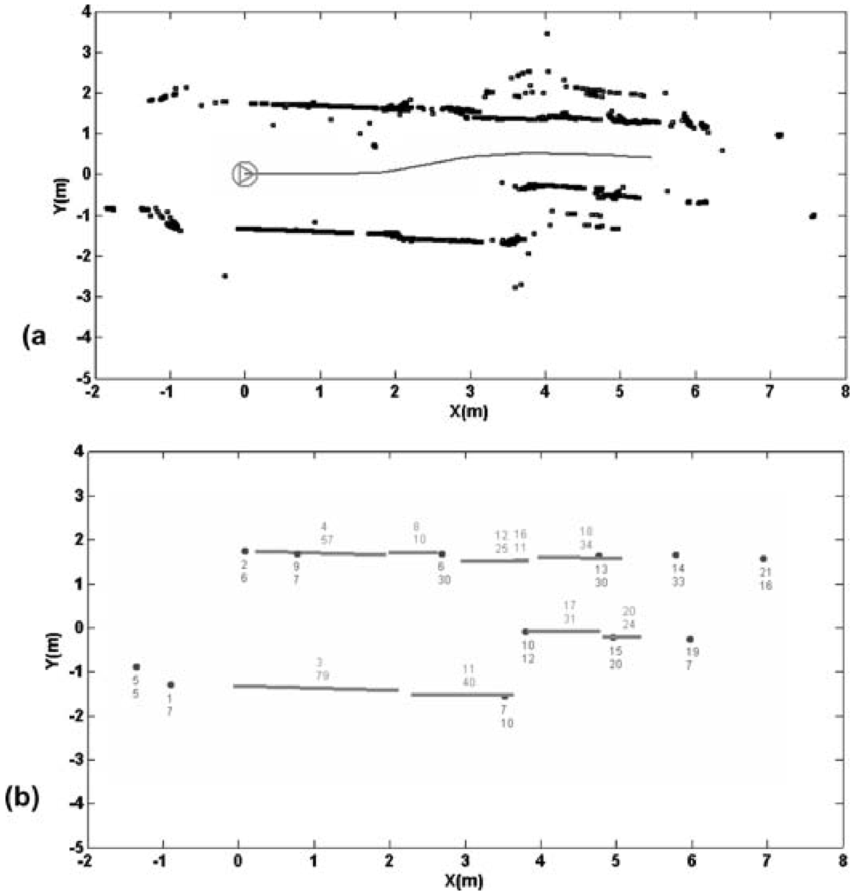

Fig. 10 A high-speed classification and SLAM of the environment shown in Fig. 8. (a) The robot trajectory and raw sonar data when the robot travels at 30 cm/s. (b) The map.

small environment consisting of cardboard boxes, a wall and some lab equipment. The raw measurements contain many phantom objects that result from the large number of multi-path echoes due to the small size and rectangular form of the environment. Phantom objects are eliminated by ignoring echoes that have a distance of flight that is approximately an integer multiple of a nearer echo from a confirmed partial plane in the path of the phantom target. Dislocations of the walls are also due to the accumulating odometry errors of the platform. However, these errors are eliminated in the generated map by applying the proposed method.

5.3. *High-speed mapping*

The high repetition rate of the advanced sonar ring enables the robot to travel faster while sensing the environment. The speed of the robot was set to 30 cm/s and the experiment performed in the same corridor as shown in Fig. 8. The raw sonar data and the map are shown in Fig. 10. The numbers of the associations are smaller than those in Fig. 8. Some of the point features, such as number 19 and 8 in Fig. 8, have disappeared in Fig. 10. A higher speed has less effect on the line features due to the very large association numbers that are still big enough even at the high speed of 30 cm/s. Due to the limited viewing angle of the point features, some of them are eliminated in high speed due to the small sensing positions and therefore small numbers of associations. However, the map generated at the speed of 30 cm/s contains enough landmarks for navigation tasks. At higher speeds, more point features and some short line features will disappear. The execution time under Matlab for the 1511 sonar readings and 27 landmarks generated during a 17-s robot travel time was 5 s on a Pentium IV 2.4-GHz laptop computer. That is, the processing time would allow for a real-time implementation for this number of landmarks and this is the case in all the experiments above; however, the processing time per measurement increases with the square of the number of landmarks. Reducing the SLAM processing time in large-scale maps is dealt with in many other research works, for example in ref. 34.

5.4. *Map quality*

In order to show the quality of the maps, hand tape measurements of the point features in an environment are compared to the ones from a robot-generated map shown in Fig. 11. The point features are vertical bars of diameter 2.5 cm. The sonar ring measures the position of the bars from angles varying approximately within −45 to +45° as it moves past and the map process integrates these into a single point. The measured lengths are labeled L1 to L14 as shown in Fig. 11. Table I presents the error in the measured lengths. The results show a small bias in that the distances are overestimated in the mapping process. This effect can be attributed to an error in the speed of sound.

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

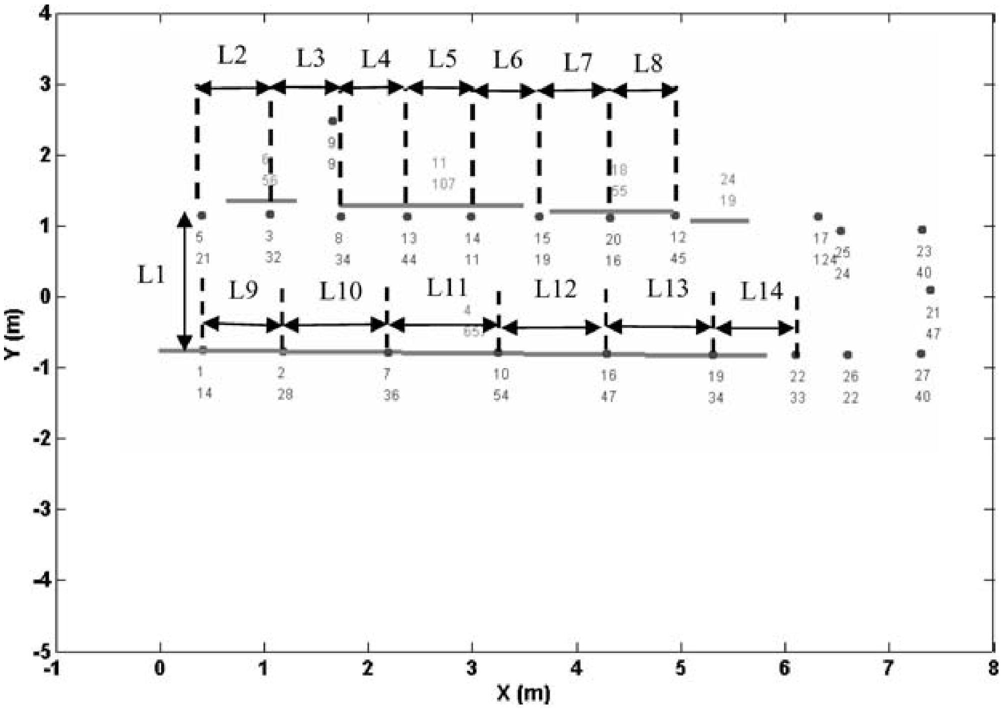

Fig. 11 Generated map of the corridor as in Section 5.3.

6. Conclusions and Future Work

This paper has presented a novel approach to feature classification within the SLAM process using a recently developed advanced sonar ring. The advanced sonar ring produces accurate measurements and allows reliable measurement association with map features. The error in range measurement has been measured to be less than 0.6 mm and for bearing angle it is less than 0.17°.[29] The proposed algorithm produces high-quality sonar maps with a Kalman filter SLAM approach. This paper highlights the benefit of accurate sonar measurements in robust feature classification using a fast hypothesis-generation and verification technique. The performance of the algorithm has been illustrated by experimental results.

The system described in this paper allows reliable measurement association with map features. However, failure can rarely occur due to erroneous associations when odometry errors exceed expectations, and errors occur in the generation of probational features.

The advanced sonar ring senses the environment at the high repetition rate of 11.5 Hz without mechanical problems and delays associated with scanning sonar sensors as in refs. 7 and 16. As a result, this system has the advantage of providing high-density maps that enable the robot to travel faster (30 cm/s) compared to the previous works.[7, 16, 19, 21] However, in the previous works, object classification was performed in one sensing cycle, which is not the case for the advanced sonar ring. This is due to the sensor design that employs one transmitter per pair. To enable the sensor to classify objects in each measurement cycle, each direction must contain at least two transmitters and two receivers.[30, 35] To reduce the cost and complexity, the sensor design was planned based on in-movement-classification. This paper has presented a novel delayed classification algorithm that categorizes features within the SLAM process. The method is capable of classifying the objects within a short distance of travel of about 10 cm. The main advantage of the proposed system is that it enables the robot to travel faster while providing dense maps of the indoor environment.

Future work will concentrate on the real-time implementation of the algorithm on *Sombrero*, improving association and recovering from failure of SLAM. The related problems of kidnapping, loop closure, large-scale SLAM and a map matching strategy to re-establish the robot's position when its uncertainty is too large, will also be investigated.

Table I Errors in map generation.

	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10	L11	L12	L13	L14
Map value (cm)	190.3	65.4	65.9	63	61.6	65.4	62.9	64	76.6	101.4	102.6	102.1	101.7	77.2
Real value (cm)	189.5	64.5	64	62.6	62.1	64.5	62.6	65.1	76.4	101.2	101.2	101.2	101.2	76.7
Error (cm)	0.8	0.9	1.9	0.4	−0.5	0.9	0.3	−1.1	0.2	0.2	1.4	0.9	0.5	0.5

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

Acknowledgements

We gratefully acknowledge Mr. Steven Armstrong for his assistance in the design and construction of the hardware and basic communication infrastructure of the sonar ring, funding from the ARC Centre for Perceptive and Intelligent Machines in Complex Environments and funding from the Monash University Postgraduate Publication Award.

References

1. L. Kleeman, "Fast and accurate sonar trackers using double pulse coding," *Proceedings of the 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'99): Human and Environment Friendly Robots with High Intelligence and Emotional Quotients*, Kyongju, South Korea (Oct. 17–21, 1999) (IEEE, Piscataway, NJ) pp. 1185–1190.
2. R. Kuc and V. B. Viard, "Physically based navigation strategy for sonar-guided vehicles," *Int. J. Robot. Res.* **10**, 75–87 (1991).
3. D. Murray and C. Jennings, "Stereo vision based mapping and navigation for mobile robots," *Proceedings of the 1997 IEEE International Conference on Robotics and Automation (ICRA)*, Albuquerque, NM (Apr. 20–25, 1997) (IEEE, Piscataway, NJ) pp. 1694–1699.
4. C. Schroeter, H.-J. Boehme and H.-M. Gross, "Robust map building for an autonomous robot using low-cost sensors," *Proceedings of the 2004 IEEE International Conference on Systems, Man and Cybernetics (SMC 2004)*, The Hague, The Netherlands (Oct. 10–13, 2004) (IEEE, New York) pp. 5398–5403.
5. T. Yata, L. Kleeman and S. I. Yuta, "Fast-bearing measurement with a single ultrasonic transducer," *Int. J. Robot. Res.* **17**, 1202–1213 (1998).
6. A. Davison, Mobile Robot Navigation Using Active Vision *D. Phil. Thesis* (Oxford, UK: University of Oxford, (1998).
7. A. Diosi and L. Kleeman, "Advanced sonar and laser range finder fusion for simultaneous localization and mapping," *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japan (Sep. 28–Oct. 2, 2004) (IEEE, New York) pp. 1854–1859.
8. M.-H. Li, B.-R. Hong and R.-H. Luo,"Simultaneous localization and map building for mobile robot," *Harbin Gongye Daxue Xuebao/J. Harbin Inst. Technol.* **36**, 874–876 (2004).
9. A. Grossmann and R. Poli, "Robust mobile robot localization from sparse and noisy proximity readings using Hough transform and probability grids," *Robot. Auton. Syst.* **37**, 1–18 (2001).
10. R. Gartshore, A. Aguado and C. Galambos, "Incremental map building using an occupancy grid for an autonomous monocular robot," *Proceedings of the 7th International Conference on Control, Automation, Robotics and Vision (ICARC)*, Singapore (Dec. 2–5, 2002) (Nanyang Technological University, Singapore) pp. 613–618.
11. J. J. Leonard, H. F. Durrant-Whyte and I. J. Cox, "Dynamic map building for an autonomous mobile robot," *Int. J. Robot. Res.* **11**, 286–298 (1992).
12. D. Maksarov and H. F. Durrant-Whyte, "Mobile vehicle navigation in unknown environments: A multiple hypothesis approach," *IEE Proc. Control Theory Appl.* **142**, 385–400 (1995).
13. T. Tsubouchi, "Nowadays trends in map generation for mobile robots," *Proceedings of the 1996 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Osaka, Japan (Nov. 4–8, 1996) (IEEE, Piscataway, NJ) pp. 828–833.
14. K.-T. Song and C. C. Chang, "Ultrasonic sensor data fusion for environment recognition," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Yokohama, Japan (Jul. 26–30, 1993) (IEEE, Piscataway, NJ) pp. 384–390.
15. K. S. Chong and L. Kleeman, "Feature-based mapping in real, large scale environments using an ultrasonic array," *Int. J. Robot. Res.* **18**, 3–19 (1999).
16. L. Kleeman, "Advanced sonar and odometry error modeling for simultaneous localization and map building," *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Las Vegas, NV (Oct. 27–31, 2003) (IEEE, Piscataway, NJ) pp. 699–704.
17. J. J. Leonard and H. F. Durrant-Whyte, "Simultaneous map building and localization for an autonomous mobile robot," *Proceedings of the IEEE International Workshop on Intelligent Robots and Systems*, Osaka, Japan (Nov. 1991) pp. 1442–1447.
18. O. Wijk and H. I. Christensen, "Triangulation-based fusion of sonar data with application in robot pose tracking," *IEEE Trans. Robot. Autom.* **16**, 740–752 (2000).
19. W. D. Rencken, "Autonomous sonar navigation in indoor, unknown and unstructured environments," *Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, Munich, Germany (Sep. 12–16, 1994) (IEEE, Piscataway, NJ) pp. 431–438.
20. G. Zunino and H. I. Christensen, "Simultaneous localization and mapping in domestic environments," *Proceedings of the International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Baden-Baden, Germany (Aug. 20–22, 2001) (IEEE, Piscataway, NJ) pp. 67–72.
21. K. S. Chong and L. Kleeman, "Mobile-robot map building from an advanced sonar array and accurate odometry," *Int. J. Robot. Res.* **18**, 20–36 (1999).
22. A. Heale and L. Kleeman, "A real time DSP sonar echo processor," *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Takamatsu, Japan (Oct. 31–Nov. 5, 2000) (IEEE, Piscataway, NJ) pp. 1261–1266.
23. H. J. Jeon and B. K. Kim, "Study on world map building for mobile robots with tri-aural ultrasonic sensor system," *Proceedings of the 1995 IEEE International Conference on Robotics and Automation*, Nagoya, Japan (May 21–27, 1995) (IEEE, Piscataway, NJ) pp. 2907–2912.
24. J. Howell and B. R. Donald, "Practical mobile robot self-localization," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, CA (Apr. 24–28, 2000) (IEEE, Piscataway, NJ) pp. 3485–3492.
25. J. M. P. Lorenzo, R. Vazquez-Martin, P. Nunez, E. J. Perez and F. Sandoval, "A Hough-based method for concurrent mapping and localization in indoor environments," *Proceedings of the 2004 IEEE Conference on Robotics, Automation and Mechatronics*, Singapore (Dec. 1–3, 2004) (IEEE, New York) pp. 840–845.
26. R.-H. Luo and B.-R. Hong, "Simultaneous localization and mapping based on multisensor fusion," *Harbin Gongye Daxue Xuebao/J. Harbin Inst. Technol.* **36**, 566–569 (2004).
27. W. Niblack and D. Petkovic, "On improving the accuracy of the Hough transform," *Mach. Vis. Appl.* **3**, 87–106 (1990).
28. X. Yun, K. Latt and J. S. Glennon, "Mobile robot localization using the Hough transform and neural networks," *Proceedings of the 1998 IEEE International Symposium on Intelligent Control, ISIC*, Gaithersburg, MD (Sep. 14–17, 1998) (IEEE, Piscataway, NJ) pp. 393–400.
29. S. Fazli and L. Kleeman, "Sensor design and signal processing for an advanced sonar ring," *Robotica*, **24**, 433–446 (2006).
30. L. Kleeman and R. Kuc, "Mobile robot sonar for target localization and classification," *Int. J. Robot. Res.* **14**, 295–318 (1995).
31. S. Fazli and L. Kleeman, "A low sample rate real time advanced sonar ring," *Proceedings of the 2004 Australasian Conference on Robotics and Automation*, Canberra, Australia (Dec. 6–8 2004).

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press

32. S. Fazli and L. Kleeman, "A real time advanced sonar ring with simultaneous firing," *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan (Sep. 28–Oct. 2, 2004) pp. 1872–1877.
33. Y. Bar-Shalom, X. Rong Li and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, [electronic resource] 1st ed. (Wiley-Interscience, New York, (2001).
34. M. Bosse, P. Newman, J. Leonard and S. Teller, "Simultaneous localization and map building in large-scale cyclic environments using the atlas framework," *Int. J. Robot. Res.* **23**, 1113–1139 (2004).
35. L. Kleeman and R. Kuc, "Optimal sonar array for target localization and classification," *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, San Diego, CA (May 8–13, 1994) (IEEE, Piscataway, NJ) pp. 3130–3135.

https://doi.org/10.1017/S0263574706003079 Published online by Cambridge University Press