Robust Sonar Feature Detection for the SLAM of Mobile Robot *

Jinwoo Choi†, Sunghwan Ahn†, and Wan Kyun Chung†

† *Robotics & Bio-Mechatronics Lab., Pohang University of Science & Technology (POSTECH), Pohang, Korea*

E-mail: {shalomi, ash, wkchung}@postech.ac.kr

***Abstract*— Sonar sensor is an attractive tool for the SLAM of mobile robot because of their economic aspects. This cheap sensor gives relatively accurate range readings if disregarding angular uncertainty and specular reflections. However, these defects make feature detection difficult for the most part of the SLAM. This paper proposes a robust sonar feature detection algorithm. This algorithm gives feature detection methods for both point features and line features. The point feature detection method is based on the TBF [1] scheme. Moreover, three additional processes improve the performance of feature detection as follows; 1) stable intersections, 2) efficient sliding window update and 3) removal of the false point features on the wall. The line feature detection method is based on the basic property of adjacent sonar sensors. Along the line feature, three adjacent sonar sensors give similar range readings. Using this sensor property, we propose a novel algorithm for line feature detection, which is simple and the feature can be obtained by using only current sensor data. The proposed feature detection algorithm gives a good solution for the SLAM of mobile robots because it gives an accurate feature information for both the point and line features even with sensor errors. Furthermore, a sufficient number of features are available to correct mobile robot pose. Experimental results of the EKF-based SLAM demonstrate the performance of the proposed feature detection algorithm in a home-like environment.**

***Index Terms*— Sonar sensors, Point feature, Line feature, Feature detection, SLAM, Mobile robot.**

I. INTRODUCTION

Robust feature detection is one of the crucial problems of simultaneous localization and map building (SLAM) of mobile robot. Because the detected feature is used as a component to construct a feature-based map and also as a landmark to localize a mobile robot. If failing to detect features or obtaining inaccurate information of feature position, the robot should excessively depend on the odometry data to estimate its pose. However, unboundness property of the odometry data makes the estimation divergent. Therefore, to perform the SLAM successfully, feature detection should give accurate feature information as much as possible.

*This research was supported in part by a grant (02-PJ3-PG6-EV04-0003) of Ministry of Health and Welfare, by the International Cooperation Research Program (M6-0302-00-0009-03-A01-00-004-00) of the Ministry of Science and Technology, by the National Research Laboratory (NRL) Program (M1-0302-00-0040-03-J00-00-024-00) of the Ministry of Science and Technology, and by a grant(M1-0214-00-0116) of the Ministry of Science and Technology Republic of Korea.

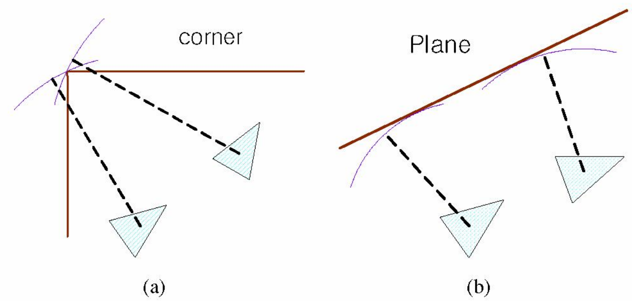


Fig. 1. Basic principles of sonar feature detection: (a) Point feature (b) Line feature.

Feature detection using accurate sensors such as laser range finders is comparatively easier because the obtained sensory information is quite dense and accurate. However, use of these sensors is restricted by their expensive cost. On the other hand, sonar sensors which are cheap and give relatively accurate range readings can be an alternative. When using sonar sensors for feature detection, two types of features are shown as follows; point features (fig. 1(a)) and line features (fig. 1(b)). However, in that case, this cheap sensor suffers from significant angular uncertainty (about $22.5°$) because of its large beam width. Moreover, the sonar beam has mirror-like reflecting property which makes the obtained range reading uncertain. These defects of sonar sensors make the feature detection very difficult. To overcome these problems, many researchers have studied the sonar feature detection.

Leonard and Durrant-Whyte [2] obtained region of constant depth (RCD) using a rotating sonar sensor. They extracted the RCDs corresponding to the planes, corners, edges and cylinders from a single dense scan. Wijk and Christensen [1] developed a point feature detection method, Triangulation-Based Fusion (TBF) algorithm. In the TBF algorithm, each sonar information was represented as an arc and the point feature is obtained from the average point of arc intersections between current and previous sonar data. Tardós et al. [3] used the Hough Transform for the point and line feature detection and the detected features are used for the SLAM of mobile robot in indoor environment. In different way from above methods, some researchers used signal processing with various types of sensor arrays for the sonar feature detection. These approaches were mainly performed by Kuc and Kleeman [4], [5]. However, these aforementioned approaches have limitation to be used for the SLAM. The obtained feature information is too sensitive to the sonar sensor error or threshold values. Besides, the number of obtained feature is not enough to implement the SLAM of mobile robot.

In this paper, we propose a robust feature detection scheme for both point features and line features by using sonar sensors. First, point feature detection is based on the TBF algorithm [1] which gives a good framework for the point feature detection. On the basic TBF scheme, we added three additional processes to improve the performance of point feature detection; 1) stable intersections, 2) efficient sliding window update and 3) removal of false point features on the wall. They can guarantee the robustness of point feature detection even with sonar sensor error and increase the number of detected features comparing with the original TBF scheme. Additionally, line feature detection is based on the basic property of adjacent sonar sensors of a sonar ring: Three adjacent sonar sensors give similar range readings on same line feature. Accordingly, line features can be easily obtained by checking this physical property. The proposed line feature detection method can increase the number of robust features to estimate robot pose.

This paper is organized as follows. Point feature detection method is given in Section II. In Section III, line feature detection is described. To show the performance of the proposed feature detection methods, Section IV shows experimental results of the EKF-based SLAM in a home-like environment. Finally, conclusion follows in Section V.

II. Point Feature Detection

Point feature detection method is based on the TBF algorithm [1] which gives feature information for corners, edges and pole-type features. In this section, basic principle of the TBF algorithm is described and three additional processes for the improvement of point feature detection are proposed.

A. TBF algorithm

The TBF algorithm is a point feature detection method. Main processes of the TBF algorithm are as follows;

1) Each sonar information is represented as an arc whose central angle is 22.5°.
2) Arc intersections between specific number of previous sonar data and the current sonar data are calculated.
3) If the number of obtained intersections is more than a threshold value, average point of intersections is used as point feature.

TBF algorithm is implemented with a sliding window which contains sonar data acquired for a certain number of steps. When new sensor data are obtained, sliding window is updated according to the sensor position.

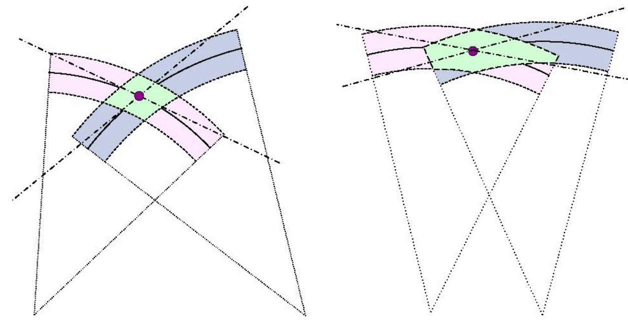


Fig. 2. Stability of intersections: (a) Stable Intersection (b) Unstable Intersection.

This algorithm gives a good framework for point feature detection. However, there are several problems to be improved for implementation of the SLAM. These problems are as follows;

1) The location of obtained features should be more accurate and robust for the range error of sonar sensors.
2) It is necessary to remove some false point features which are not extracted from a specific point.
3) To localize the mobile robot successfully, the feature detection should give good features as many steps as possible.

The accuracy of detected features can be improved by applying more strict condition. However, it decreases the number of successful features because there's a trade-off between the applied condition and the number of obtained features. To improve above problems of the original TBF, additional processes for point feature detection are suggested in the following subsections. We describe the first and the second processes in II-B and the last one in II-C.

B. Improvement I

1) Stable Intersections: Stable intersection is an intersection whose angle between two sonar arcs is larger than a given threshold [6].

$$\theta_{int} > \theta_t \tag{1}$$

Using the stable intersections gives two advantages in point feature detection. The one is robustness for the sensor error. While a stable intersection point is not significantly disturbed by errors of sonar sensor (fig. 2(a)), an unstable intersection point can be significantly changed by sensor errors (fig. 2(b)). Therefore, the obtained features using stable intersections can be robust for the sensor errors. The other is removal of false point features on the wall. Intersections obtained for the wall might be unstable intersections not stable ones. Consequently, the false point features on the wall can be removed by using the stable intersections.

(a) (b)

Fig. 3. Experimental Environment: (a) Environment with pole-type point features (b) Home-like Environment.

2) Efficient Sliding window update: The second additional process for the point feature detection is an efficient sliding window update method. In the original TBF algorithm, the sliding window is updated when the sensor positions are changed more than a certain minimum distance with respect to the sensor position of previous sensor data.

$$\sqrt{(x_{s_{new}} - x_{s_{pre}})^2 + (y_{s_{new}} - y_{s_{pre}})} > d_t \qquad (2)$$

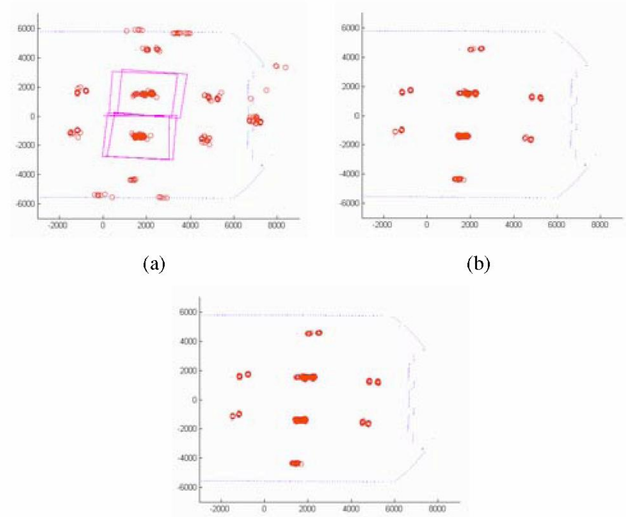
However, this method decreases the number of detected features. Especially for the rotational motion of mobile robot, only few point features can be detected. When the robot rotates at a fixed point, the sliding window is updated continuously because the sensor position is changed. Then, all the sensor data in the sliding window is acquired from a fixed robot position and there might be no intersection between sonar sensors in the sliding window. Therefore, point features can not be detected because there's no arc intersection. If the robot fails to detect features when rotating at a fixed position, heading angle error can not be corrected adequately.

To solve this problem, we suggest a new sliding window update that is more efficient than the previous one. The proposed method updates the sliding window according to the robot position where the sensor data are acquired. In other words, the sliding window is updated when the distance between robot positions where the previous sensor data and new sensor data are acquired is more than a threshold distance.

$$\sqrt{(x_{R_{new}} - x_{R_{pre}})^2 + (y_{R_{new}} - y_{R_{pre}})} > d'_t \qquad (3)$$

If the distance is smaller than the given threshold, only the most recent data in the sliding window are changed by new ones. As a result, the updated sliding window keeps that the distance between robot positions where the sensor data is acquired is larger than the given threshold. And it can increase the number of successful features, especially for the rotational motion of mobile robot.

3) Experimental Verification: To evaluate the effectiveness of stable intersections and efficient sliding window update, experiments are performed with pole-type point features



(a) (b) (c)

Fig. 4. Point feature detection : (a) Original TBF, feature detected at (208/565 steps) (b) Stable intersections (214/565 steps) (c) Stable intersections with proposed window update (418/565 steps).

(fig. 3(a)). The results of point feature detection are shown in fig. 4. As shown in fig. 4(a), the original TBF gives some false features on the wall and outliers. Moreover, locations of the detected features are widespread. Comparatively, fig. 4(b) shows a result using stable intersections. This result verifies that stable intersections can give accurate features and remove false features well. As shown in fig. 4(b), the locations of features are concentrated on the real position closely and all the false features and outliers are rejected. Furthermore, the number of steps for successful feature detection is increased by almost twice via efficient sliding window update (fig. 4(c)). Because the increased number of detected feature might be induced from the rotational motion of mobile robot, the increased features are very useful to correct the heading angle error of mobile robot.

C. Improvement II

As mentioned above, point features should be detected from specific points such as corners, edges and pole-type features. However, some point features are obtained along the wall especially for following two cases; 1) when the mobile robot navigates close to a wall 2) the environment is complex such as a home-like environment. For those cases, the false features can not be removed by using only stable intersections. Furthermore, these false point features on the wall make the SLAM divergent because features are obtained not at a specific point but along the wall with wide range. In this subsection, the last additional process, removal of such false point features is suggested.

1) Removal of false point features on the wall: Removal of false point features on the wall can be performed by

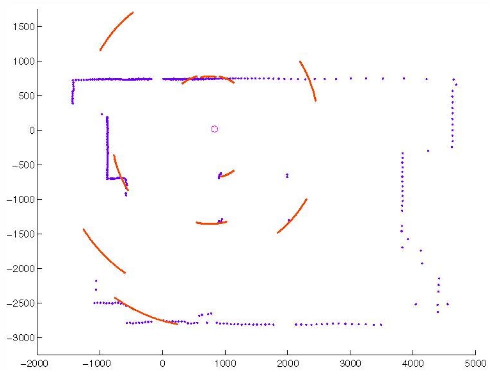
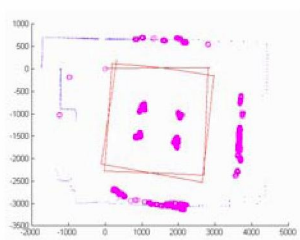
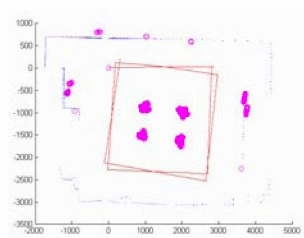


Fig. 5. Sonar sensor data represented as uncertainty arc with laser map data



(a)

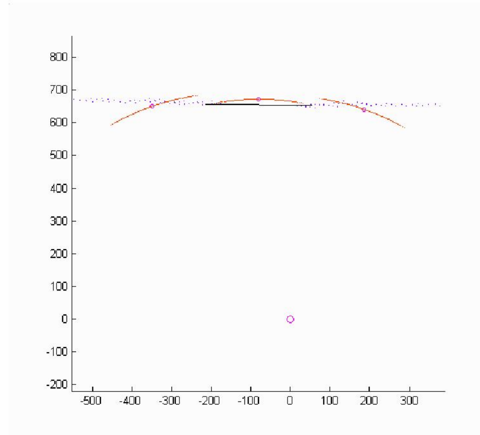


(b)

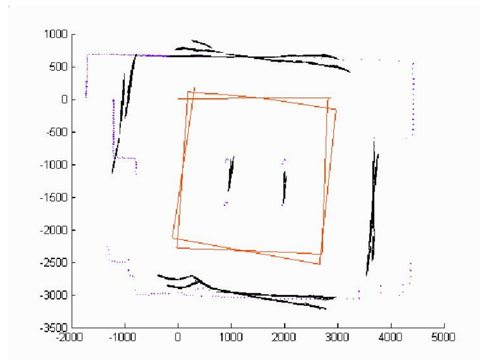
Fig. 6. Effect of removing false features on wall : (a) Before removing, feature detected at (648 / 2323 steps) (b) After removing false features (880 / 2323 steps).

understanding the sensor property of sonar returns from the wall. Fig. 5 shows sonar data which are represented as uncertainty arcs with the laser map data which show real environment of mobile robot. Three adjacent sonar sensors give similar ranges on the wall because sonar sensor has wide range of aperture angle (Top in fig. 5). From point features, however, every single feature is detected by only one sensor. Consequently, if three sonar sensors which are adjacently located show similar ranges, we can be sure that the sonar returns are not obtained from the point feature but from the line feature such as walls. Using this sensor property, false features on the wall can be removed by rejecting these sensor data in calculating arc intersections in advance.

2) Experimental Verification: To verify the performance of removing false point features on the wall, experiments are executed in a home-like environment (fig. 3(b)). The results of point feature detection are shown in fig. 6. Before applying the proposed method, obtained features contain many false point features on the wall and these features can not be used as point features because they are placed along the surfaces (fig. 6(a)). On the other hand, fig. 6(b) shows the effectiveness of the proposed method. As a result of the proposed method, false point features are removed almost clearly. The remaining features are real point features extracted from legs of table and corners of furniture.



(a)



(b)

Fig. 7. Line feature detection : (a) Line fitting using three sonar data (b) Result of line feature detection.

III. Line Feature Detection

Line features describe walls and planes of furniture. Using line feature gives some advantages for the SLAM. First of all, mobile robot can perform successful self-localization using line features even though the environment has insufficient number of point features. Moreover, the obtained map from the result of the SLAM using line features can represent a real environment better than the case of using only point features.

A. Line Feature Detection

Line feature detection using sonar sensors is usually performed by using accumulated sensor data obtained from a specific number of steps because sonar data are very sparse. However, this causes delayed registration to the state matrix of the EKF-based SLAM and the odometry error along the path is included in the feature information. In this subsection, line feature detection method using only current sensor data is proposed. The proposed method is very simple and reflects characteristics of sonar sensor well.

As shown in fig. 5, three adjacent sensors detect the same wall. To detect line features, this property of sonar sensors is used. However, this condition is not enough for line feature detection, i.e., not all of three sensor data which are adjacently located and give similar range readings are obtained from line feature. The sonar data can be classified to line feature using the following conditions;

1) Three adjacent sonar sensors have similar range readings.
2) The middle one has minimum range among them.

The second condition is used because the direction of middle sensor is closer than those of other two sensors. From the sensor data classified by above conditions, the line features can be extracted by the following procedures;

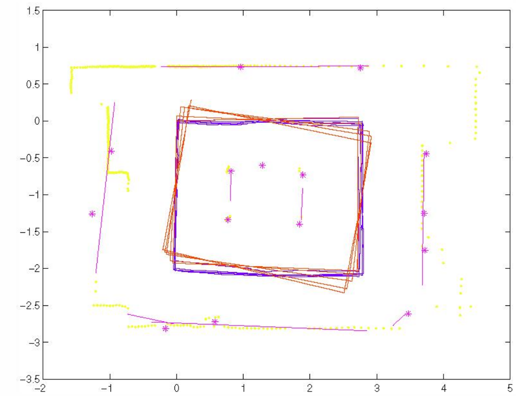
1) Obtaining three points using centerline sensor model [6],
2) Least square line fitting with obtained points,
3) Determining boundaries for line feature.

The last step determines the end points of detected line feature. In the proposed line feature detection method, end points of each line feature can be determined by sonar beam width. Because three sonar sensors detect the same line feature, we can conclude that the entire angular range ($\pm 11.25°$) of middle sensor is within the detected line feature. Thus, we can determine end points of the detected line feature from the direction of middle sensor and beam aperture angle (fig. 7(a)). This information for line features can be used usefully for line matching or complete map generation.

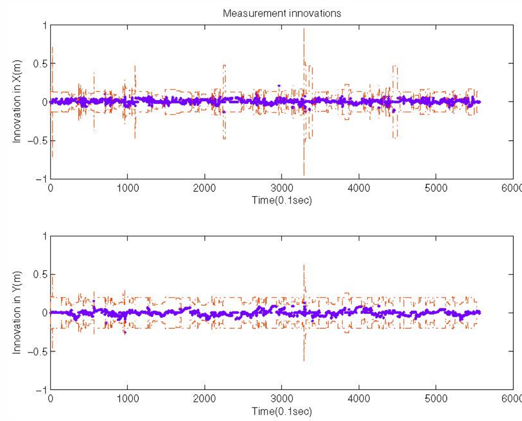
The proposed line feature detection has some advantages with respect to other line feature detection methods. First of all, the proposed algorithm is very simple and its computational time is very fast because it is not a voting scheme such as the Hough Transform. The second advantage is that the line feature detection can be implemented by using only current sensor data. This is a significant advantage when the detected feature is used for the realtime EKF-based SLAM as mentioned above. The last advantage is the determination of boundaries for detected line feature using the physical property of sonar sensor. Other line feature detection including the Hough Transform can not provide this information.

B. Experimental Verification

To verify the proposed line feature detection, an experiment is performed in a home-like environment (fig. 3(b)). Result of the line feature detection is shown in fig. 7(b). Even though detected lines are mis-aligned to the real line features because of accumulated odometry error, the result shows that the feature detection works well. As the robot rotates the same trajectory twice, line features are extracted successfully and the end points are also well-determined.



(a)



(b)

Fig. 8. EKF-based SLAM : (a) EKF-based SLAM result ; Red lines for Odometry robot pose, blue lines for estimated robot pose and magenta for estimated map data (b) Innovation for x and y with their 2σ uncertainty bounds.

IV. Experimental Results

To verify the proposed feature detection method, we performed the standard EKF-based SLAM using the detected features [7]. Experiments were carried out using our BSR-I differential-drive robot equipped with 2 SICK laser scanners (not used in the estimations) and a sonar ring of 16 Polaroid sensors. The robot navigated a rectangular path in a home-like environment where a table, a sofa and other furniture are set as shown in fig. 3(b). The robot navigated the path five times with 0.15 m/s and returned to the initial pose and the total navigated length was about 50 m. The robot ran the given path during about 10 minutes and sonar data acquisition and the EKF update were performed by 10 Hz. The data associations were verified by testing the joint compatibility between the obtained feature data and the corresponding features in the map data [8]. For each step, at most one point feature and one line feature were used for updating the robot pose and map data in the EKF algorithm.

Fig. 8 shows results of the EKF-based SLAM after nav-

	x_{err} (mm)	y_{err} (mm)	θ_{err} $(^\circ)$
Odometry	218.7	283.4	9.6
Estimation	29.6	21.5	0.5

TABLE I
ROBOT POSE ERROR

igating 5 cycles. In fig. 8(a), the odometry path and the estimated path from the EKF-based SLAM are represented as red and blue lines respectively. The real map data obtained from two SICK laser scanners are shown as yellow points to verify the estimated map data. The magenta asterisks and the same color of lines represent estimated point features and line features, respectively. As shown in fig. 8(a), robot path using odometry data is inaccurate because of the unbounded pose error of odometry data. The error for heading angle is more serious than position error. In contrast, the EKF-based SLAM estimates the rectangular robot path accurately and the obtained features are matched with real environment well. Four legs of table, corners of furniture and some specific points such as knobs are detected as point features. The line feature is obtained from the wall and surfaces of table, sofa and other furniture.

Fig. 8(b) shows innovations for x and y with their 2σ uncertainty bounds. The innovations are clearly remain inside their 2σ uncertainty bounds. This facts guarantee the consistency of the EKF-based SLAM.

The final robot pose errors with respect to the initial pose are shown in table I. The estimated robot pose is better than the odometry data with about 10 times for position and almost 20 times for heading angle, which is natural as a result of the EKF-based SLAM with robust feature detection.

V. CONCLUSION

This paper addressed a robust sonar feature detection for the SLAM of mobile robot. The point feature detection and line feature detection methods were suggested. For point feature detection, we used the TBF scheme and three additional processes which are 1) stable intersections, 2) efficient sliding window update and 3) removal of false point features on the wall. The proposed point feature detection is robust for the range error of sonar data. Moreover, successful features can be obtained for more steps than the original TBF algorithm, thus, the obtained point feature can be used for the EKF-based SLAM well. Line feature detection was simply induced from the characteristics of adjacent sonar sensors. The proposed line feature detection method was performed by using only current sensor data. Thus, the detected feature can be registered to the state of the EKF-based SLAM without delay. Moreover, end points for each detected line feature can be determined by the aperture angle of sonar beam.

Experiments were conducted by the realtime EKF-based SLAM in a home-like environment. The robot path was estimated accurately and the obtained map represented real home-like environment well with satisfactory of innovation propagation.

REFERENCES

[1] O. Wijk and H. I. Christensen, "Triangulation-Based Fusion of Sonar Data with Application in Robot Tracking," *IEEE Trans. on Robotics and Automation*, vol. 16, no. 6, pp. 740–752, 2000.

[2] J. J. Leonard and H. F. Durrant-Whyte, "Mobile Robot Localization by Tracking Geometric Beacons," *IEEE Trans. on Robotics and Automation*, vol. 7, no. 3, pp. 513–523, 1991.

[3] J. D. Tardós, J. Neira, P. M. Newman, and J. J. Leonard, "Robust mapping and localization in indoor environments using sonar data," *International Journal of Robotics Research*, vol. 21, no. 4, pp. 311–330, 2002.

[4] B. Barshan and R. Kuc, "Differentiating sonar reflections from corners and planes by employing an intelligent sensors," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, no. 6, pp. 560–569, 1990.

[5] L. Kleeman and R. Kuc, "An optimal sonar array for target localization and classification," in *International Conference on Robotics and Automation*, pp. 3130–3135, 1994.

[6] H. Choset, K. Nagatani, and N. A. Lazar, "The arc-transversal median algorithm: A geometric approach to increasing ultrasonic sensor azimuth accuracy," *IEEE Trans. on Robotics and Automation*, vol. 19, no. 3, pp. 513–523, 2003.

[7] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. Boston, MA: Academic Press, 1988.

[8] J. Neira and J. D. Tardós, "Data association in stochastic mapping using the joint compatibility test," *IEEE Trans. on Robotics and Automation*, vol. 17, no. 6, pp. 890–897, 2001.