Algorithm Fusion for Feature Extraction and Map Construction From SONAR Data

Hesham Ismail and Balakumar Balachandran

***Abstract*— Feature extraction is an important aspect of simultaneous localization and mapping, a process, in which a mobile platform is used to create a map of an unknown environment, while simultaneously locating the platform's own position within the constructed map or environment. Geometric shapes or features, such as lines, circles, and interior and exterior corners, are determined as a part of this process, and these features may be used as landmarks. In this paper, an original feature extraction algorithm specific to distance measurements obtained through SONAR sensor data is presented. This algorithm has been put together by combining the SONAR salient feature extraction algorithm and the triangulation Hough-based fusion with the point-in-polygon detection. The reconstructed maps obtained through simulations and experimental data with the fusion algorithm are compared with the maps obtained with existing feature extraction algorithms. Based on the results, it is suggested that the proposed algorithm can be considered as an option for the data obtained from SONAR sensors in environments, where the other forms of sensing are not viable.**

***Index Terms*— Triangulation based fusion (TBF), Hough transform (HT), triangulation Hough based fusion (THF), SONAR salient feature extraction algorithm, point-in-polygon detection, fusion algorithm, feature extraction, sound navigation and ranging (SONAR), indoor environment, simultaneous localization and mapping (SLAM).**

NOMENCLATURE

n_t	Threshold counter value
x_s	Sensor x position
y_s	Sensor y position
x_T	Landmark x position
y_T	Landmark y position
x_{est}	Moving average of landmark's x position
y_{est}	Moving average of landmark's y position
γ	Orientation of sensor reading
r	Range from sensor reading
ρ	Length of vector that is normal to line and passes through origin
θ	Orientation of ρ with respect to x axis
A	Matrix used to store the vote counts (ρ, θ)
o_1	SONAR location 1
z_1	SONAR range reading for location 1
A	Center of accepted hypothetical circle
ϕ_1	Bearing from positive x axis to center A at location 1
θ_{1min}	Minimum orientation of the SONAR at location 1
d	Distance between SONAR location 1 and SONAR location 2
β	Angular uncertainty of SONAR sensor
q_r	Radius of hypothetical circle
q_o	Center of hypothetical circle
m	Total number of SONAR readings from one location
n	Total number of locations which mobile vehicle visited or took scans from
δ	Degree value used to determine number of arc segments
D	Distance from SONAR to center of hypothetical circle
σ	Threshold value
R	Radius of accepted hypothetical circle

Manuscript received February 3, 2015; revised June 16, 2015 and July 5, 2015; accepted July 9, 2015. Date of publication July 15, 2015; date of current version September 4, 2015. The associate editor coordinating the review of this paper and approving it for publication was Prof. Janice Limson.

The authors are with the Department of Mechanical Engineering, University of Maryland, College Park, MD 20742 USA (e-mail: hismail@umd.edu; balab@umd.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSEN.2015.2456900

I. INTRODUCTION

SIMULTANEOUS localization and mapping (SLAM) is a process, which consists of a number of components including motion sensing, environment sensing, vehicle pose estimation, feature extraction, data association, and filtering. Here, the authors have focused on environment sensing as well as feature extraction. Different types of features include artificial features and geometric features. Artificial features can be well-placed beacons (landmarks) in an environment, which are used to help the mobile vehicle localize itself with respect to them. On the other hand, geometric features are features that naturally exist in the environment, and these features can be described in terms of geometric parameterization. There are several types of sensors that can be used to obtain distance measurements, such as laser sensors, digital cameras, infrared sensors, and SOund Navigation And Ranging (SONAR) sensors. Here, the focus is on determining geometric features with SONAR sensing.

SONAR sensors have been used in autonomous mobile vehicles, since the sensors are inexpensive when compared to laser sensors and digital cameras. In the present work, another reason for considering SONAR sensing is the goal to build a mobile vehicle for oil storage tank inspection. An environment such as the one in an oil tank limits the types of sensors that one can use. Laser sensors are avoided for safety reason, and camera sensors are undesirable since oil is opaque. SONAR sensors are safe to use and functional in this harsh environment. However, a key limitation of a SONAR sensor is the high angular uncertainty $22.5\% - 30\%$ [1]. To overcome the angular limitation of SONAR sensors, different algorithms have been developed.

In many research efforts, the extraction of meaningful features from raw SONAR data has been examined. These features are planes (lines), corners, and edges (points). Whyte and Leonard [2] used Region of Constant Depth (RCD) to detect planes, corners, and cylindrical features in an environment. They found that geometric feature such as planes (lines), corners (points) and cylindrical features are unique features that can be extracted from most indoor environments. Wijk *et al.* [1] developed a point-feature extraction algorithm called Triangulation-Based Fusion (TBF). Continuing the work of Wijk *et al.* [1], Choi *et al.* improved the TBF algorithm by adding the following: i) stable intersections, ii) efficient sliding window updates, and iii) removal of false features on the boundary [3]. In addition, Choi *et al.* [3] added a novel line feature detection scheme. This line feature detection works as follows. First, readings from three adjacent SONAR sensors are gathered. If the range from these three readings is low, for the readings in close agreement, a line feature is registered. Weiqin [4] also extended the work of Wijk *et al* by focusing specifically on line feature detection. In the corresponding algorithm, each SONAR reading is represented by an arc, and there is a tangent line for each arc. If the tangent line passes through two tangent points, then a line feature is registered. Tardós *et al.* [5] used the Hough Transform (HT) to determine features such as points and lines. They used a winner-takes-all strategy to distinguish between lines and points, if the number of votes from the Hough Transform are the same. Yap and Shelton [6] used the Randomized Hough Transform (RHT) to detect line features. The RHT algorithm is based on the assumption that the detected lines are orthogonal to each other. If a detected line is not perpendicular to another, then, it is removed. Yap and Shelton tested the RHT algorithm in a large environment setting. Baolong *et al.* [7] used the processed data from TBF as input to the Hough Transform to obtain line features in the considered environment. Lee and Song [8] and Lee *et al.* [9] developed a new feature extraction for SONAR data, known as SONAR salient feature. This algorithm is effective in determining corners and edges. If an environment has a low number of natural features, artificial features are added to help the use of the SONAR salient feature algorithm.

In previous research efforts, fixed SONAR and rotary SONAR sensors have been commonly used. In the case of rotary ones, the sensor base is rotated by using a servomotor to get a full scan of the environment at the considered location. Kleeman and Kuc [10] used a different SONAR arrangement wherein two transmitters and two receivers were used rather than fixed or servo-mounted SONAR configurations. Through their work, it was shown that a low number of SONAR sensors are needed to distinguish between different features in an environment such corners, edges, and planes. They used amplitude and range values from a SONAR sensor, whereas previous researchers had only used range values from a SONAR sensor. In a relatively recent effort of Fazli and Kleeman [11], 24 simultaneously-fired transmitters and 48 receivers were used. This SONAR configuration was tested on a vehicle in motion. It was found that for speeds above 30 cm/s, most of the point features and some of the line features were not detected; yet, there were still enough features detected for navigation purposes. In another different use of SONAR sensors, Steckel and Peremans [12] used biomimetic SONAR sensors to perform SLAM. However, in this effort, features in the environment were not extracted, since the focus was on the localization of a mobile vehicle navigating through a complex environment. The algorithm was tested in a corridor environment. As expected, this environment was hard to navigate because of the lack of features.

An original fusion algorithm for feature extraction is proposed in this work. The authors use a combination of available algorithms and exploit their strengths for feature extraction. This fused algorithm can be used to capture points, lines, and cylindrical features, whereas the TBF and SONAR salient algorithms are limited to point features, and the Hough Transform is used best for capturing line features. In this new combined algorithm, the different features in the environment are determined and the redundant or repeated, features are removed. This algorithm is different from the one presented in the work of Yap and Shelton [6], since the authors do not assume orthogonal intersections of lines, and they can determine lines intersecting at arbitrary angles. In addition, the present algorithm can be used to detect corners, cylindrical aspects, and lines. The TBF algorithm is used to determine point features from the environment, and the Hough transform algorithm is used to determine line features. Furthermore, if the Hough Transform is applied to TBF data, better line detection can be achieved; this is known as Triangulation Hough based fusion (THF). Also, for the new line detected, TBF data are used to draw line segments. Here, in the combined algorithm, TBF results are used again to improve the overall performance and accuracy of the feature extraction and remove redundant features. Noting that the SONAR salient algorithm is better for detecting edges, points, and cylindrical shaped objects, here, it has been used to determine point and cylindrical features. It is mentioned that the inside of oil storage tanks consists of cylindrical features, which can be detected with the SONAR salient algorithm. Through the use of point-in-polygon (PIP), the proposed algorithm can separate the data into interior and exterior (boundary) groups. For example, the Hough transform and TBF algorithm can be used to reconstruct the environment's boundaries, while the SONAR salient algorithm is used to reconstruct the interior features. The separation of groups ensures that each part of the algorithm works on "prefered" features. The new fusion algorithm has been studied through simulations and experiments, and the results are reported here.

This manuscript follows previous studies reported by the authors [13], [14]. The authors have organized the material into five sections. First, the TBF, Hough Transform, SONAR salient and fusion algorithms are briefly introduced and explained. Next, simulation results obtained with the use of TBF, Hough Transform, SONAR salient and proposed fusion algorithms are presented and compared. Following

Algorithm 1 TBF Algorithm Developed by Wijk *et al.* [1]

1: **for** $i = 1 \rightarrow m$ **do**
2: $n_t = 0$
3: **for** $j = (n-1) \rightarrow 1$ **do**
4: **if** (x_T, y_T) =find intersection $\left(x_{si}, y_{si}, r_i, \gamma_i, x_{sj}, y_{sj}, r_j, \gamma_j\right)$ **then**
5: $x_{est} = \dfrac{n_t x_{est} + x_t}{n_t + 1}$
6: $y_{est} = \dfrac{n_t y_{est} + y_t}{n_t + 1}$
7: $n_t + +$
8: **if** $n_t \geq threshold\ value$ **then**
9: $save\,(x_{est}, y_{est})$ as a point feature

Algorithm 2 Hough Transform Algorithm for Straight Line

1: **for** all (x, y) **do**
2: **for** $\theta = 1^\circ \rightarrow 360^\circ$ **do**
3: $\rho = x\cos\theta + y\sin\theta$
4: Find$(\rho_{min}, \rho_{max}, \theta_{min}, \theta_{max})$
5: Make $A\,(\rho, \theta)$
6: Fill $A\,(\rho, \theta)$ with (ρ, θ) values
7: Vote $A\,(\rho, \theta)$
8: Get (ρ, θ) from $A\,(\rho, \theta)$ with highest votes

that, the experimental arrangement is explained in the fourth section. Subsequently, in the fifth section, the experimental results obtained through the application of TBF, Hough Transform, SONAR salient and the fusion algorithms are presented. Finally, concluding remarks are provided.

II. Feature Extraction Algorithms

In this section, commonly used feature extraction algorithms with SONAR data are introduced to provide a context for the combined algorithm developed in this work. The TBF, Hough Transform, and SONAR salient algorithms are commonly used for feature extractions. Following their exposition, the fusion algorithm is presented.

A. TBF Algorithm

The TBF algorithm can be used with SONAR data to find point features in an unknown environment [1]. SONAR data are represented as arcs in the use of this algorithm. The intersection of SONAR arcs from two different locations forms a candidate point feature. The candidate points are not considered features, until they have been identified from a sufficient number of locations to reach a threshold value. An illustration of the steps involved is provided in Algorithm 1. These steps are as follows: (1) loop over the SONAR scans from the current mobile vehicle position; (2) set the threshold counter value to zero; (3) loop over the SONAR scans from the previous locations except the current location; (4) represent each set of SONAR data as an arc. Find the intersection of the SONAR arcs for the current and previous locations. Identify intersection point as a candidate point feature; (5) for a successful step 4, update the x coordinate of the candidate point feature recursively; (6) for a successful step 4, update the y coordinate of the candidate point feature recursively; (7) for a successful step 4, increase the threshold counter value by 1; and (8) if the threshold counter value is greater than the chosen threshold value, accept the candidate point feature as an acceptable point feature and save this point feature in step 9. For further details about the TBF algorithm, a reader is referred to the work of Wijk and Christensen [15].

B. Hough Transform Alogrithm

The Hough transform can be used to detect different features such as edges, circles, ellipses, and non-standard shapes in an environment [16]. In this paper, Hough transform is used to detect line features such as planes. Tardós *et al.* [5] used the standard Hough transform to determine line features. In the Hough Transform algorithm, the line features are detected from the environment through a voting technique. The different steps involved, which are depicted in Algorithm 2, are as follows: (1) determine lines passing through a SONAR data point. Each SONAR data point can have many lines passing through it and to determine all of the different lines for a given SONAR data point, the SONAR (x, y) data is loaded into equation (1);

$$\rho = x\cos\theta + y\sin\theta \tag{1}$$

(2) for each SONAR (x, y) data point, vary the value of θ from 0° to 360° with an increment of 1°; (3) calculate ρ values for each θ to determine all of the different lines; (4) determine the maximum and minimum vales of ρ and θ; (5) for the range of ρ and θ values, create bins with equal spacing and assemble the matrix A that contains all the bins; (6) place the calculated ρ and θ values in the right bins of matrix A; (7) sum up the total number of objects in each bin and update matrix A; and (8) choose the bin with the highest tally (votes) from matrix A as the wining line. For further details on the Hough transform, one is referred to Hough [16], Baolong *et al.* [7], Tardós *et al.* [5], and Yap and Shelton [6]. The Hough transform can be applied to raw SONAR data for determining the line features in an environment. In a modification, Balong *et al.* [7] used TBF results and carried out Cluster Inhibiting Hough Transform (CIHT) to determine line features in an environment. They were the first to use a fusion algorithm with SONAR data and their fusion algorithm is known as Triangulation Hough based fusion. In this paper, the Standard Hough Transform is used on TBF results. In an earlier work of the authors, a comparative study of the TBF, Hough transform, and THF algorithms has been undertaken [13]. From this earlier work, it was found that the THF algorithm provided better results than the Hough Transform, when applied to raw SONAR data. Hence, the THF algorithm has been considered for the proposed feature extraction algorithm.

C. SONAR Salient Algorithm

The SONAR salient algorithm, which has been developed by Lee and Song [8] and Lee *et al.* [9], can be used to determine point and cylindrically shaped objects. Here, the SONAR salient algorithm is used to find point features from

Algorithm 3 Convex Saliency Circling or SONAR Salient Feature Extraction Algorithm Developed by Lee and Song [8], [9]

1: **for** $i = 1 \rightarrow m \times n$ **do**
2:   **if** $r_{min} < z_i < r_{max}$ **then**
3:     **for** $j = i + 1 \rightarrow m \times n$ **do**
4:       **if** $r_{min} < z_j < r_{max}$ **then**
5:         **for** $\phi_i = \theta_{i-min} \rightarrow \theta_{i-min} + \beta,\ \phi_i = \phi_i + \delta$ **do**
6:           FPA model { In: $(o_i, z_i, \theta_{i-min}), (o_j, z_j, \theta_{j-min}), \phi$; Out: (q_o, q_r)}
7:           **if** $p_{min} < q_r < p_{max}$ **then**
8:             **for** $k = j + 1 \rightarrow m \times n$ **do**
9:               **if** $|D - q_r - z_k| < \sigma$ **then**
10:                 $C = [C; q_o\ q_r]$

the environment by using three SONAR positions. The first two SONAR positions are used to find the point features (hypothetical circle), whereas the next SONAR position is used to validated the point feature. The steps that make up this algorithm are shown in Algorithm 3, and these steps are as follows: (1) loop over all SONAR data; (2) accept SONAR data between $r_{min} = 50$ cm and $r_{max} = 200$ cm; (3) loop over SONAR data that are different from the previous SONAR data; (4) similar to step 2; (5) represent each SONAR data set as an arc. A SONAR arc is divided into segments, with the number of them given by the equation $int(\frac{\beta}{\delta} + 1)$, where $int(\cdot)$ is the integer part operator that returns an integer. For example, a choice of $\beta = 30°$ and $\delta = 10°$ will result in 4 segments; (6) use footprint association model (FPA) of Lee and Song [8] and Lee *et al.* [9]. The inputs to the FPA are location, range, and the orientation of the SONAR sensor for two different SONAR data sets and the outputs are the hypothetical circles; (7) accept the hypothetical circle if the radius is between p_{min} and p_{max}; (8) loop over SONAR data that are different from the previous two SONAR readings data; and (9) if the gap between SONAR sensor to the center of the hypothetical circle minus radius of the hypothetical circle minus the range of the SONAR data from step 8 is less than the threshold value (σ) go to step 10 and save the hypothetical circle as an acceptable point feature.

D. Proposed Algorithm Combination

In the new fusion algorithm, the authors use the TBF, Hough Transform, and THF algorithms in parallel with the SONAR salient algorithm. The data are then combined by using the point-in-polygon detection (PIP) to determine which features are within the boundary of the environment. PIP is used to classify points as being inside or outside of a given polygon. For further details on PIP, the reader is refereed to Hormann and Agathos [17]. The proposed fusion algorithm scheme is shown in Figure 1. Since the some of the steps can be carried out in parallel, there can be a dramatic decrease in computational time. The different steps are as follows: (1) input raw SONAR data to the fusion algorithm; (2) apply limit constraints to the SONAR data and accept only SONAR

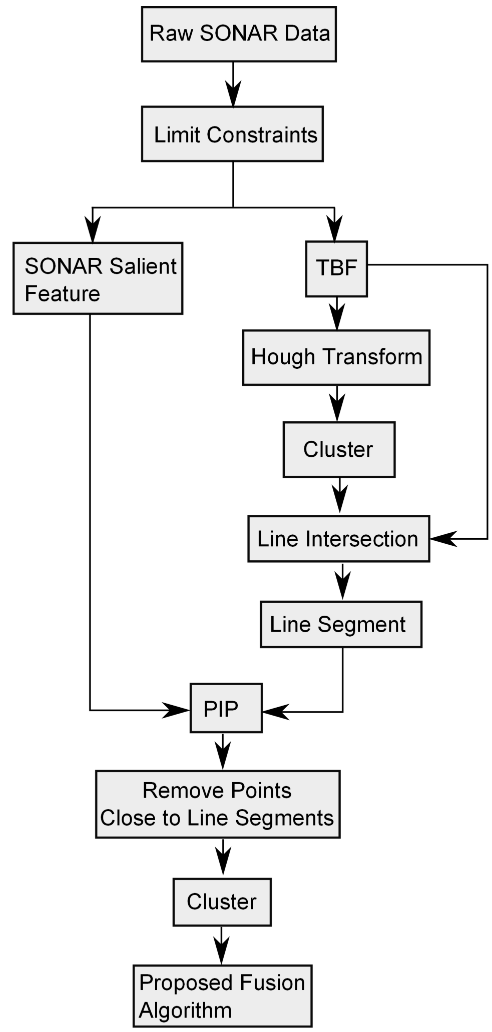


Fig. 1. Proposed algorithm fusion [14].

range data between 50 cm and 200 cm. The SONAR sensor (model HRLV-MAXSONAR-EZ) cannot be used detect objects (targets) that are closer than 30 cm; so, values lower than 30 cm are reported as 30 cm. Also, for this sensor, it is recommended to avoid data below 50 cm because of multiple reflections that occur when the SONAR sensor is close to an object. In addition, for distances above 200 cm, since the SONAR beam becomes large, the sensor cannot be used to detect objects above this range; (3) process the limit constraint results with the TBF algorithm as well as the SONAR salient feature algorithm; (4) process the TBF results with the Hough transform and refer to the outcome as THF results; (5) process the THF results with a k-mean clustering algorithm; (6) find the cluster lines intersection; (7) integrate the intersection points and TBF results to find line segments and determine the boundaries of the environment; (8) combine the environment boundary information with SONAR salient points by using PIP to determine the interior points; (9) remove the interior points close to the environment boundary if the minimum distance between the point and the line segments is less than the threshold value; (10) cluster the interior results by using a k-mean clustering algorithm; and (11) complete the process with the clustered interior results and the line segment results.

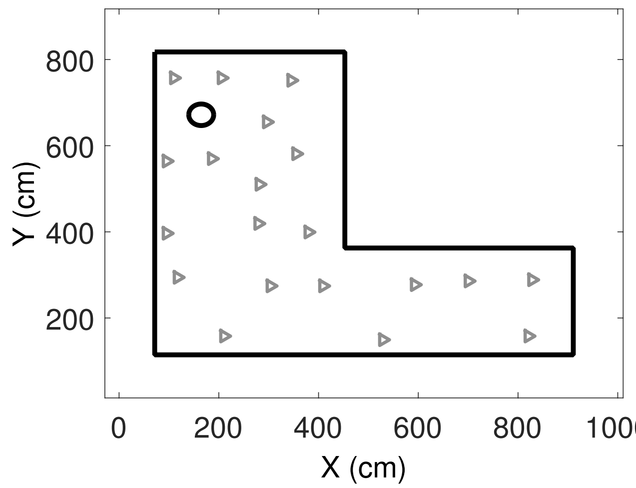


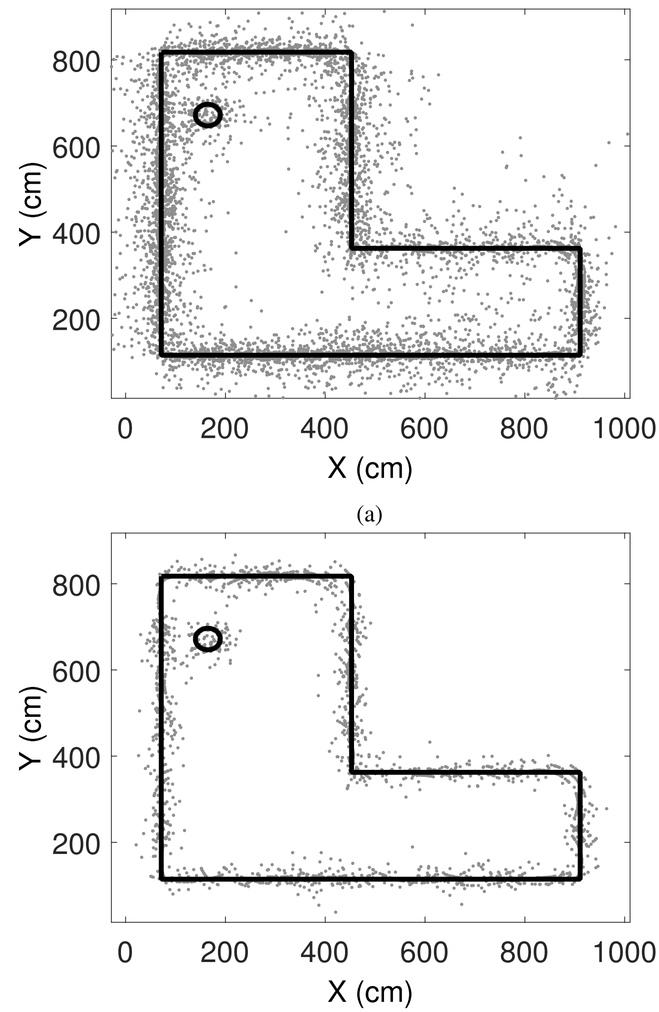
Fig. 2. L-Shaped Environment. The triangles represent different mobile sensor platform positions. The circular landmark inside the environment has a radius of 25 cm.

III. Algorithm Simulations and Results

In this section, as a precursor to the experiments to follow, an L-Shaped closed environment and another closed environment are used to study the new fusion algorithm by using simulation data that were generated by using MATLAB software. Appropriate uncertainties are included with the raw SONAR data. The range and angular uncertainties for the chosen "HRLV-MAXSONAR-EZ" SONAR are 0.1 cm and 22.5°, respectively. These values are obtained from the sensor data sheet. The variance of the range of SONAR sensor is equal to $(0.1\text{ cm})^2$, and the variance of the bearing of SONAR sensor is equal to $(22.5°)^2$. Uncertainties (noise) were introduced to the range and bearing readings of the SONAR sensor simulations, in order to make the simulations more realistic. For example, if the SONAR range reading is 50 cm and range uncertainty is 0.1 cm, the new range value will be 50 cm $+ \; randn() * 0.1$ cm, where $randn()$ is a MATLAB function that produces values from a normally distributed distribution with a mean of 0 and a variance of 1. The SONAR bearing readings are similarly treated. Uncertainties were not introduced to the SONAR sensor location, since, the authors wanted to test the performance of the proposed feature extraction algorithm for a perfect localization of the mobile vehicle.

A. Environment I: L-Shaped Environment

The L-Shaped environment shown in Figure 2, consists of 6 corners, 6 line segments, and one circular interior feature. A full 360° SONAR scan with 1° increments was performed at each of the 20 different locations inside this environment. The mobile vehicle comes to a complete stop before a full SONAR scan is performed. This is done, since the motor which rotates the SONAR array moves relatively slowly. In the envisaged future application of the oil tank floor inspections, for similar reasons, it is expected that the mobile vehicle will also stop for SONAR and other sensor measurements. In general, as more locations are visited by the mobile vehicle, better results are obtained with the proposed algorithm fusion, at the expense of increased computational time. Also, the visited locations need to be appropriately separated (i.e., SONAR readings should not be grouped in one small area and so on). In an earlier work [13], the mobile vehicle visited 15 locations in a similar environment, with the visited locations being close to one corner only. In this prior work, the corner could be detected, but no other features in the environment could be identified. Based on the prior experiments, it was ascertained that a separation distance in the range of 30 cm to 100 cm worked out well for the visit locations of the considered mobile vehicle platform. A separation distance below 30 cm increased the overall number of visits of the mobile vehicle and unnecessarily increased the computational time. On the other hand, a separation distance above 100 cm could result in the loss of important environmental features, as the acceptable SONAR range data is between 50 cm and 200 cm. Through experiments and simulations for the considered sensors and mobile platform, it was found that separation distances between 30 cm and 100 cm provided good results. The raw SONAR data are shown in Figure 3a. First, the raw SONAR data are processed with the



(a)

(b)

Fig. 3. The actual L-shaped environment is overlaid on top of the SONAR data for comparison purposes. After the application of limit constraints, data below 50 cm and above 200 cm have been removed. (a) Raw SONAR data. (b) Results after application of limit constraints.

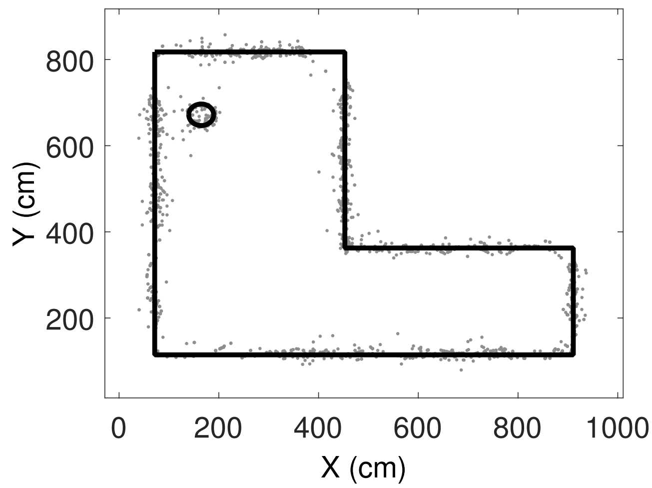
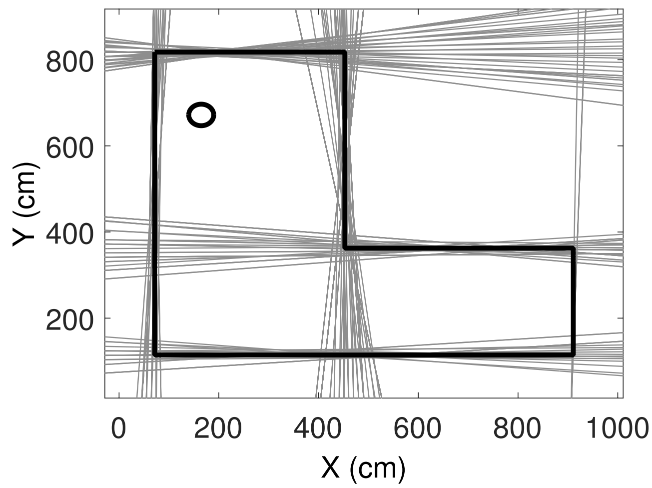
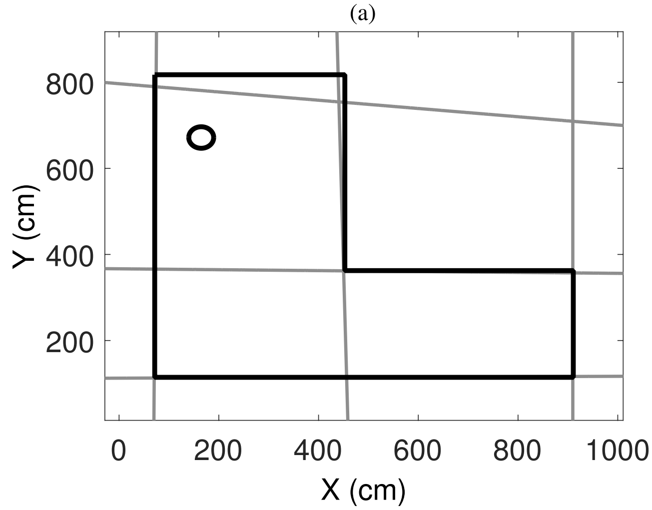


Fig. 4. TBF results for the L-Shaped environment with $n_t > 0$.



(a)



(b)

Fig. 5. THF results for $n_t > 0$ for the L-Shaped environment. The number of bins and lines used are 200 and 140, respectively. The intersections of the clustered lines have been calculated. (a) THF results. (b) K-mean clustering results.

limit constraint algorithm. Data below 50 cm and above 200 cm have been removed, and the results of this processing are shown in Figure 3b.

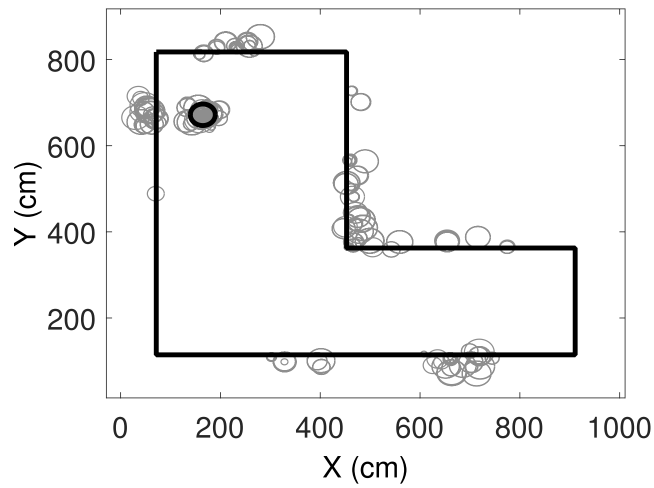


Fig. 6. SONAR salient feature extraction result for the L-Shaped environment for $p_{min} = 3$ cm and $p_{max} = 30$ cm, before classification.

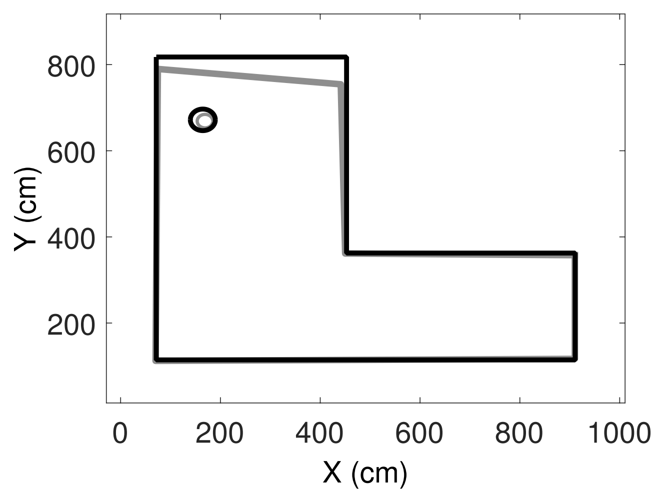


Fig. 7. Fusion algorithm results for chosen L-shaped environment.

Next, the limit constraint data are processed with the TBF algorithm and the threshold value has been arbitrarily selected to be 0. By selecting a low threshold value for the TBF algorithm, it was ensured that no important features were removed. In an earlier study of the authors [13], it was shown that by increasing the threshold to a value above 100, a considerable number of the important features were removed, whereas for a threshold below a value of 100, no important features were lost. This observation is consistent with the conclusion drawn by Wijk and Christensen [18]; they used a low threshold value to avoid the removal of important features. In this paper, the selected threshold values for the TBF algorithm were 0 for the simulations and 10 for the experiments; with a 360 SONAR scan at each location, no important features were found to be removed. The obtained TBF results are shown in Figure 4.

Next, the processed TBF data are sent for application of the Hough Transform. The resulting THF data are shown in Figure 5a. Subsequently, the THF data are clustered by using k-mean clustering as shown in Figure 5b. Also, the limit constraint data are sent to the SONAR salient algorithm. The SONAR salient algorithm results are shown in Figure 6. Next, the SONAR salient data are used in the PIP algorithm

TABLE I

CORNER COORDINATES FOR L-SHAPED ENVIRONMENT AND ABSOLUTE PERCENTAGE ERRORS BETWEEN ESTIMATES AND ACTUAL VALUES

Corner #	**Actual Data** X(cm), Y(cm)	**Simulation Results** X(cm), Y(cm)	**% Error** X(%), Y(%)
1	71.45, 817.64	74.62, 789.87	4.43, 3.40
2	452.74, 817.64	441.18, 754.27	2.55, 7.75
3	452.74, 362.47	449.98, 361.40	0.61, 0.30
4	910.85, 362.47	910.14, 357.59	0.08, 1.35
5	910.85, 114.46	910.16, 116.81	0.08, 2.05
6	71.45, 114.46	70.01, 112.23	2.02, 1.95

TABLE II

INTERIOR FEATURE DATA FOR L-SHAPED ENVIRONMENT AND ABSOLUTE PERCENTAGE ERRORS BETWEEN ESTIMATES AND ACTUAL VALUES

Interior Feature #	**Actual Data** X(cm), Y(cm), **Radius(cm)**	**Simulation Results** X(cm), Y(cm), **Radius(cm)**	**% Error** X(%), Y(%), **Radius%**
1	164.71, 671.75, 25	169.10, 668.76, 15.63	2.66, 0.44, 37.48

TABLE III

LINE DATA FOR L-SHAPED ENVIRONMENT AND ABSOLUTE PERCENTAGE ERRORS BETWEEN ESTIMATES AND ACTUAL VALUES

Line #	**Actual Data** ρ(cm), θ(degree)	**Simulation Results** ρ(cm), θ(degree)	**% Error** ρ(%), θ(%)
1	817.64, 90.00	793.17, 84.52	2.99, 6.09
2	452.74, 360.00	458.93, 361.39	1.37, 0.38
3	362.47, 90.00	366.64, 89.40	1.15, 0.67
4	910.85, 360.00	909.86, 360.00	0.11, 0.00
5	114.46, 90.00	112.53, 90.25	1.69, 0.27
6	71.45, 360.00	69.79, 359.68	2.32, 0.09

to sort the data into interior and exterior points. This can readily be implemented in MATLAB by using the function "inpolygon". Interior points near the boundary are removed if the distance between the point and the line segments is less than the chosen threshold value. Exterior points and interior points near a boundary are removed. Following that, the remaining interior points are clustered by using a k-mean clustering algorithm. The inner group cluster determines the interior features in the environment. Next, a line is drawn between intersection points if sufficient TBF data (85%) are present between the intersection points. Finally the line segments and the cluster of the inner group are put together. The results obtained with the new fusion algorithm are shown in Figure 7.

The simulation results are compared to the actual data for features such as corners, interior features, and lines in Tables I-III. First, the corners' data obtained from the simulations are compared with the actual data. With the fusion algorithm, the authors are able to detect all corners and the maximum percentage errors in X and Y coordinate values are found to be below 5% in X and 8% in Y as shown in Table I. Also, with the fusion algorithm, the authors are able to detect the interior feature. The percentage error between simulation and actual data for the center of the interior feature in X and Y is found to be below 3%. The radius error is about 38% as shown in Table II. This relatively high error in the interior feature radius is suspected to be due to the averaged data. The choice of a higher acceptable range might help alleviate this error. Finally, the errors made in line data from the simulations are found to be less than 3% and 7% in ρ and θ, respectively, as presented in Table III.

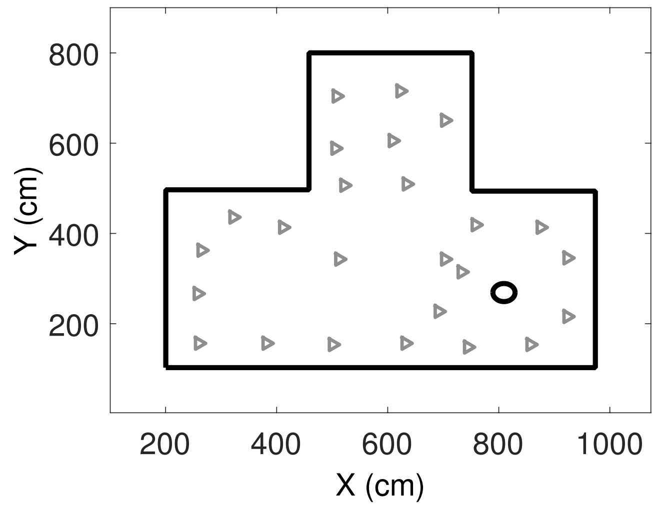


Fig. 8. Second closed environment. The triangles represent different mobile vehicle positions. The circular landmark inside the environment has a radius of 20 cm.

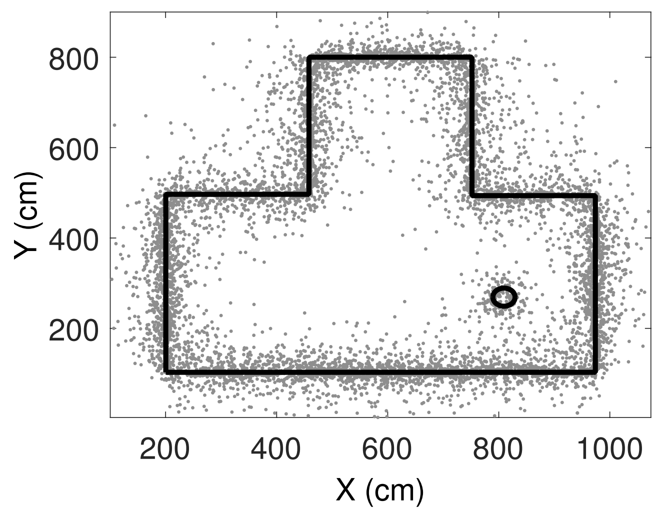


Fig. 9. Raw SONAR data, the actual closed environment is overlaid on top of the SONAR data for comparison purposes.

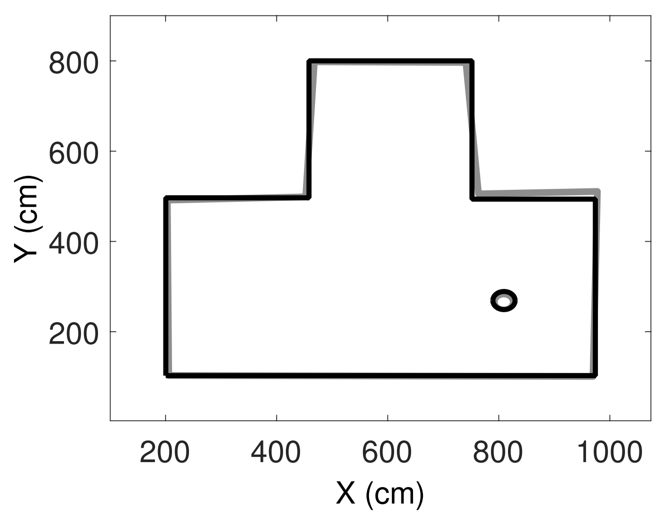


Fig. 10. Fusion algorithm results for second environment.

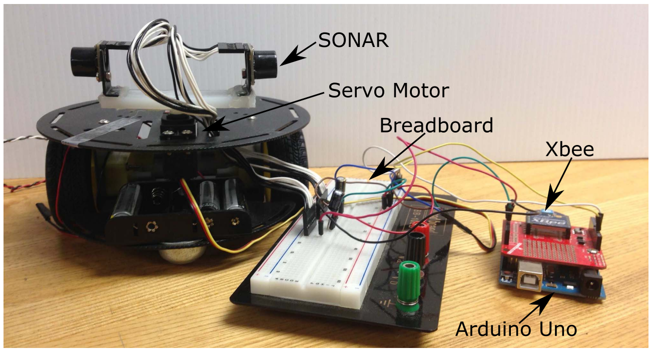


Fig. 11. Mobile vehicle used in experiments.

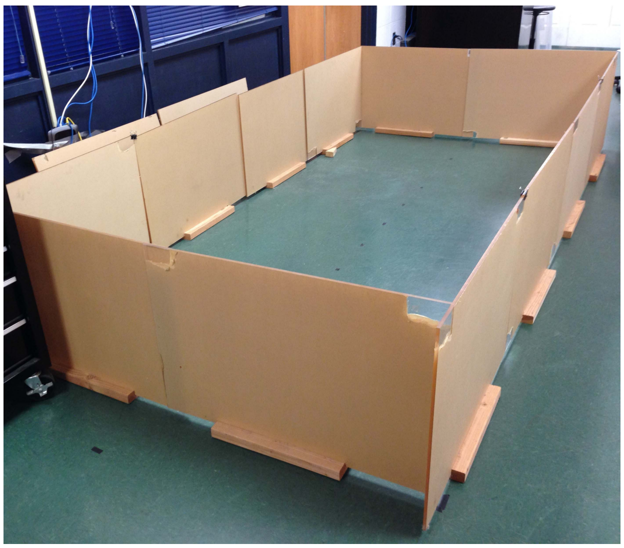


Fig. 12. Experimental arrangement with no interior features [13].

Fig. 13. Experimental arrangement with two interior features [14].

B. Environment II

The second closed environment consists of 8 corners, 8 line segments, and one circular interior feature. In this environment, the mobile vehicle was taken to the 25 different locations shown in Figure 8 and a full SONAR scan was performed

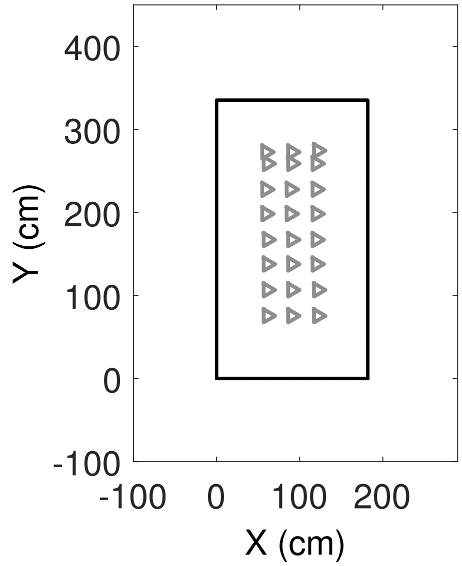


Fig. 14. Different locations of mobile platform in the environment without interior features. Triangles are used to mark the locations.

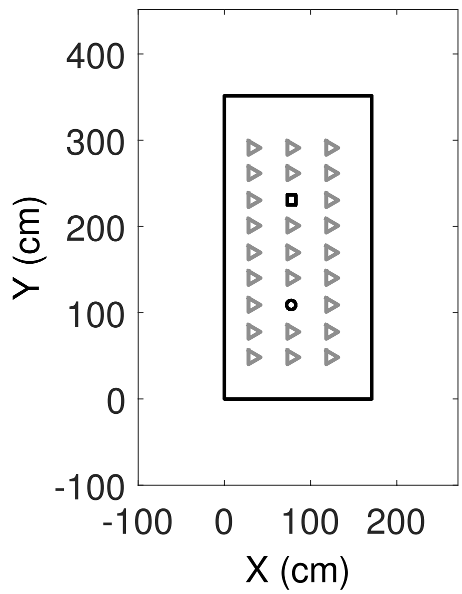
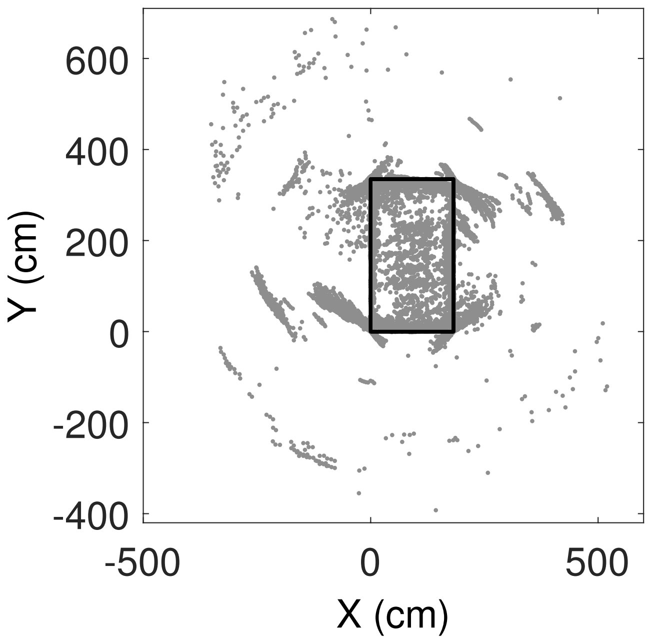


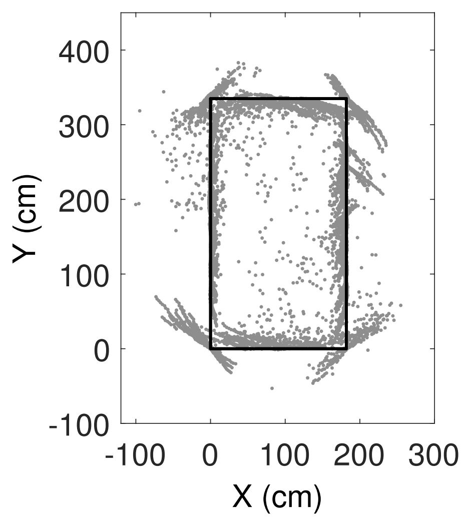
Fig. 15. Different locations of mobile platform in the environment with 2 interior features. Triangles are used to mark the locations, and the interior features are outlined with a square shape and a circle shape.

at each location with 1° angle increments. Feature extraction studies similar to those carried out for the first environment were performed. The associated raw SONAR data for this case are shown in Figure 9. The results obtained with the fusion algorithm are shown in Figure 10.

Next, the comparisons made between the estimated features and the actual data value for the second environment considered are presented. First, the simulation results for the corners are compared to the actual corner locations and the maximum percentage errors in the X location is found to be below 3% and in the Y location is found to be below 4%. When the interior feature results from the simulations are compared with the actual values for the interior feature, it is found that the maximum percentage error for the interior feature's center is



(a)



(b)

Fig. 16. The actual environment with no interior features is placed over the SONAR data for comparison purpose only. For the limit constraint data, data below 50 cm and above 200 cm have been removed [13]. (a) Raw SONAR data for experimental environment without interior features. (b) Results after application of limit constraints.

below 1% for the X value and below 2% for the Y value. The error in the radius estimation is found to be below 30%. Finally, the simulation results for the lines are compared to the actual line values. The maximum percentage errors for the lines is found to be below 7% in ρ and below 2% in θ.

IV. Experimental Arrangement

The mobile vehicle is composed of two SONAR sensors, servo motor, breadboard, XBee and Arduino accessories, as shown in Figure 11. The two SONAR sensors are attached to a servo motor and oriented to face opposite directions. The servo motor can rotate 180° in 1° increments, and with this arrangement a full 360° scan can be made. The data are transferred to a computer by using the wireless connection (XBee) for post-processing. The mobile vehicle was tested in two different environments, as shown in Figure 12 and 13. There are no interior features in the first closed environment. In the first environment, the mobile vehicle is moved to 24 different locations, and a full scan is performed at each location. The different vehicle locations are shown in Figure 14, where each true position of the vehicle has been marked by a triangle. In the second environment, the mobile vehicle is moved to 25 different locations and a full scan is performed at each location. Two interior features are included in the second environment. In Figure 15, each true position of the mobile vehicle has been marked by a triangle and the interior features are also outlined.

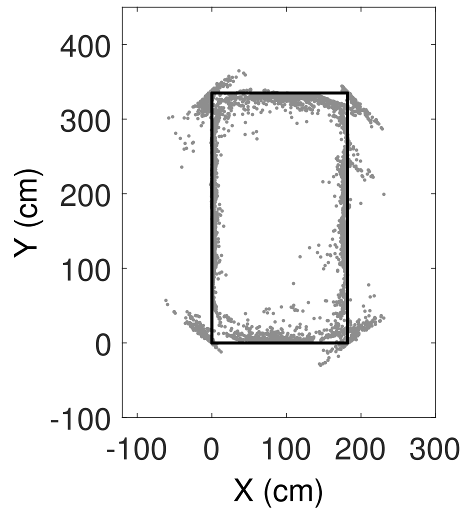


Fig. 17. TBF results $n_t > 10$ for environment with no interior features

V. Experimental Results

In this section, the feature extraction results for the two considered environments are presented. SONAR data are processed in both cases by using the new fusion algorithm presented in this paper. First, the environment with no interior feature is considered. The raw SONAR data is shown in Figure 16a. As in the simulations, first, the raw SONAR data is processed by using the limit constraint algorithm, and data below 50 cm and above 200 cm are removed. The outcome is shown in Figure 16b. In the next step, the limit constraint results are processed by using the TBF algorithm, for which the value of 10 has been selected as the threshold. For studies on the effects of this threshold value, the reader is referred to the authors' previous work [13]. The TBF result is shown in Figure 17. Subsequently, the processed TBF data is input to the Hough Transform, and the resulting THF data is shown in Figure 18a. Next, the THF data are clustered by using k-mean clustering and the line intersections are calculated as shown in Figure 18b. The limit constraint results are also processed by using the SONAR salient algorithm. The obtained results of this processing are shown in Figure 19. Next, the SONAR salient results are separated into interior

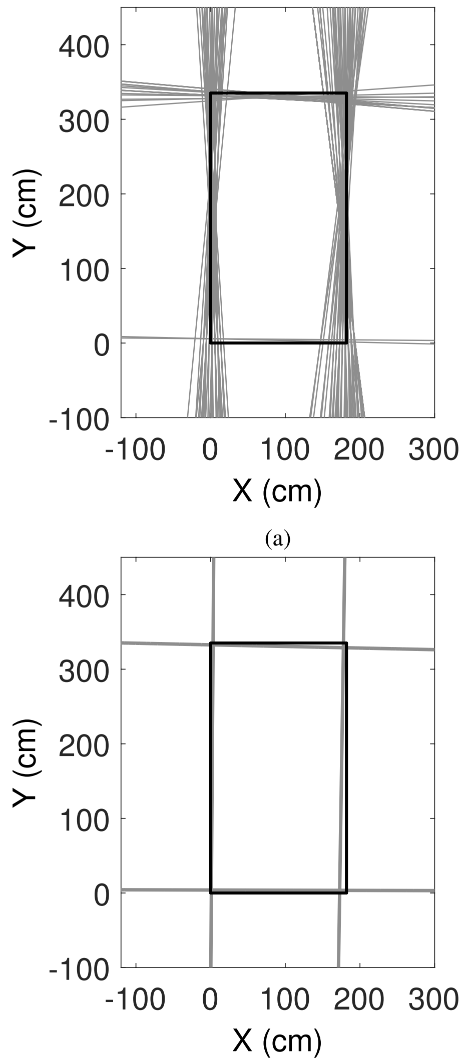


Fig. 18. THF result for $n_t > 10$ for the environment with no interior features. The number of bins and lines used were 200 and 50, respectively. The intersections of the clustered lines have been calculated. (a) THF results. (b) Kmeans clustering results.

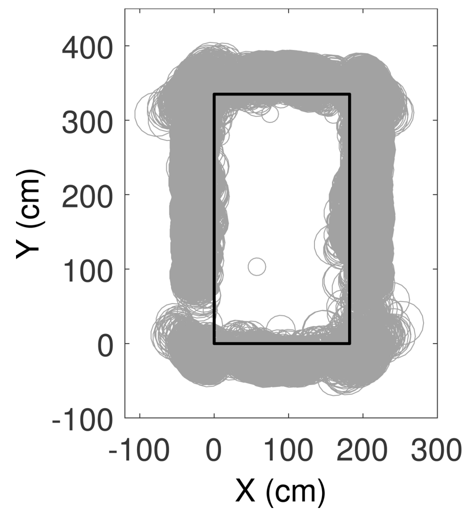


Fig. 19. SONAR salient feature extraction result for the environment with no interior feature, accepted radius range between 3 cm and 30 cm, before the classification.

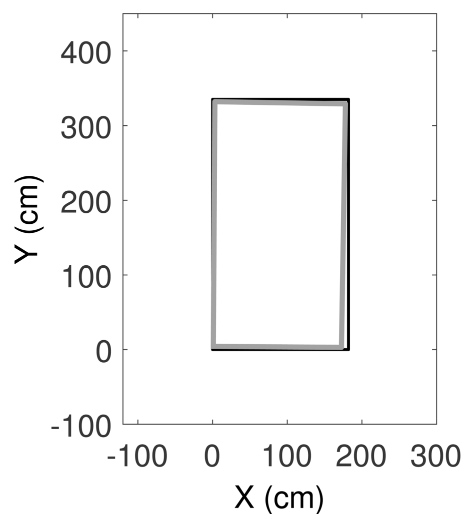


Fig. 20. Fusion algorithm results for the environment with no interior features.

and exterior groups by using PIP analysis. In this case, as there are no interior points, the k-means clustering is not performed. After that, by using the TBF data, lines are drawn between the intersection points. This is carried out if there are sufficient TBF data (70%) between the considered intersection points. Finally, the line segments and the clusters of the inner groups are put together and the resulting outcome of the fusion algorithm is shown in Figure 20. Through the studies, it was found that to improve the results of the fusion algorithm, the mobile vehicle positions needed to include a number of locations, which are not close to each other, as also previously noted in this paper. In addition, the raw SONAR data were noisy. With the aids of limit constraints and the TBF algorithm, the data quality was enhanced for processing with the Hough transform. Overall, the most significant features of the environment were extracted by using the fusion algorithm.

A comparison of the features extracted from the experimental data with the actual data is made in Tables IV and V.

TABLE IV
CORNER COORDINATES FOR THE ENVIRONMENT WITH NO INTERIOR FEATURES AND ABSOLUTE ERRORS BETWEEN ESTIMATES AND ACTUAL VALUES

Corner #	Actual Data X(cm), Y(cm)	Exp. Results X(cm), Y(cm)	|Actual-Exp.| X(cm), Y(cm)
1	$182.00, 0.00$	$172.59, 3.01$	$9.41, 3.01$
2	$0.00, 0.00$	$1.00, 4.27$	$1.00, 4.27$
3	$0.00, 335.10$	$3.07, 332.14$	$3.07, 2.96$
4	$182.00, 335.10$	$177.90, 329.22$	$4.10, 5.88$

It is found that with the new fusion algorithm, one can detect all corners and the maximum absolute errors in X and Y values are below 10 cm and 6 cm, respectively. For the line comparisons, it is found that the maximum absolute errors in ρ and θ are below 10 cm and 2°, respectively.

In another experimental study, the authors examined the performance of the fusion algorithm to extract features for the

TABLE V
LINE DATA FOR ENVIRONMENT WITH NO INTERIOR FEATURES AND ABSOLUTE ERRORS BETWEEN ESTIMATES AND ACTUAL VALUES

Line #	Actual Data ρ(cm), θ(degree)	Exp. Results ρ(cm), θ(degree)	|Actual-Exp.| ρ(cm), θ(degree)
1	$335.10, 90.00$	$332.62, 88.76$	$2.48, 1.24$
2	$182.00, 360.00$	$172.78, 359.12$	$9.22, 0.88$
3	$0.00, 90.00$	$3.97, 89.85$	$3.97, 0.15$
4	$0.00, 360.00$	$0.99, 359.60$	$0.99, 0.40$

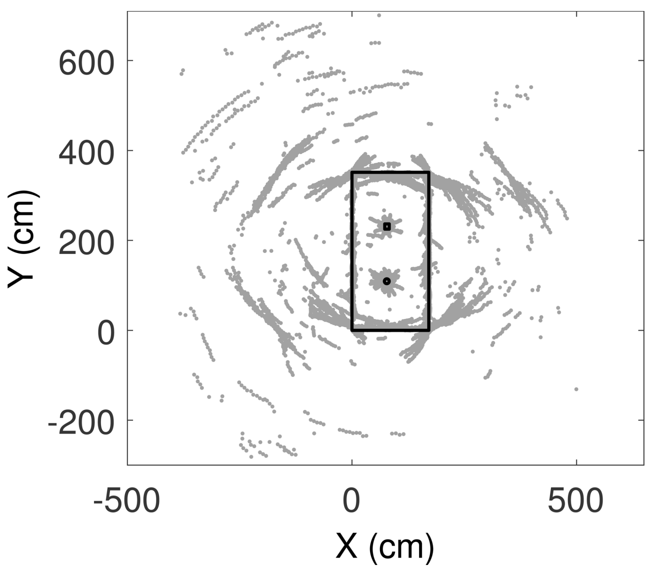


Fig. 21. Raw SONAR data, the actual environment with two interior features is overlaid on top of SONAR data for comparison purposes.

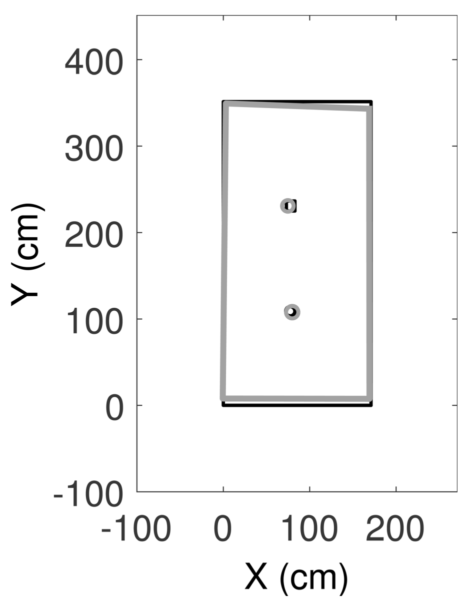


Fig. 22. Results of fusion algorithm for experimental environment with two interior features.

environment with two interior features, shown in Figure 13. Feature extraction studies similar to those carried out for the empty environment were performed. The raw SONAR data for this case is shown in Figure 21. The outcome of the new fusion algorithm is presented in Figure 22. It is seen that the new fusion algorithm is able to capture the most significant environment features, which are the corners, lines, and two interior features.

A comparative discussion between the features extracted by using the fusion algorithm and the actual values is provided next. For the experimental environment with interior features, with the new algorithm, the authors are able to detect all corners and the maximum absolute differences in X and Y values are found to be below 3 cm and 8 cm, respectively. For the line comparisons, the maximum absolute differences in ρ and θ are found to be below 8 cm and 3°, respectively. The maximum percentage error in determining an interior feature is below 5% for the X value, below 2% for the Y value, and below 35.00% for the radius value. As expected, in the SONAR salient algorithm processing, although the location of the square cross-sectioned feature was picked up, the square cross-sectioned feature was picked up as a feature with circular cross-section. In addition, there was a 34.17% error in the determined radius value. In future work, this will be examined for enhancement. One direction would be to implement a higher acceptable range.

VI. CONCLUDING REMARKS

Through numerical studies and experiments, the authors have examined algorithms for extracting features from SONAR data. To take advantage of some of the attractive features of existing algorithms, a fusion algorithm has been developed and shown to capture all of the significant features in the environments studied. The obtained features were found to be unique with no repeated characteristics or redundancy. Through the numerical studies, it was learned that the SONAR Salient algorithm can be effective in locating interior features but considerable errors do occur in the radius values of the determined features. The experimental results also support the effectiveness of the proposed fusion algorithm for extracting features in closed environments without and with interior features. The choice of a high enough number of locations and locations not close to each other were found to be important for enhancing feature extraction quality. In the future, the authors plan to use this fusion algorithm in an Extended Kalman Filter (EKF)-SLAM scheme. This integration is expected to help the EKF-SLAM in terms of speed, as only the pertinent features data result from the fusion algorithm.

ACKNOWLEDGMENT

The authors would like to thank the Abu Dhabi National Oil Company (ADNOC) for their support of this project.

REFERENCES

[1] O. Wijk, P. Jensfelt, and H. I. Christensen, "Triangulation based fusion of ultrasonic sensor data," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 4. May 1998, pp. 3419–3424.

[2] J. J. Leonard and H. F. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," *IEEE Trans. Robot. Autom.*, vol. 7, no. 3, pp. 376–382, Jun. 1991.

[3] J. Choi, S. Ahn, and W. K. Chung, "Robust sonar feature detection for the SLAM of mobile robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Aug. 2005, pp. 3415–3420.

[4] Z. Weiqin, "Sonar features extraction algorithm for a mobile robot," in *Proc. 3rd Int. Symp. Intell. Inf. Technol. Appl. (IITA)*, vol. 3. Nov. 2009, pp. 689–692.

[5] J. D. Tardós, J. Neira, P. M. Newman, and J. J. Leonard, "Robust mapping and localization in indoor environments using sonar data," *Int. J. Robot. Res.*, vol. 21, no. 4, pp. 311–330, 2002.

[6] T. N. Yap and C. R. Shelton, "SLAM in large indoor environments with low-cost, noisy, and sparse sonars," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2009, pp. 1395–1401.
[7] L. Baolong, H. Bo, W. Yongqing, Z. Xuan, and G. Lei, "Triangulation & Hough transform based fusion of sonar data for mobile robotics," in *Proc. 2nd IEEE Conf. Ind. Electron. Appl. (ICIEA)*, May 2007, pp. 1165–1170.
[8] S.-J. Lee and J.-B. Song, "A new sonar salient feature structure for EKF-based SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2010, pp. 5966–5971.
[9] S.-J. Lee, D.-W. Cho, and J.-B. Song, "Novel sonar salient feature structure for extended Kalman filter-based simultaneous localization and mapping of mobile robots," *Adv. Robot.*, vol. 26, nos. 8–9, pp. 1055–1074, 2012.
[10] L. Kleeman and R. Kuc, "An optimal sonar array for target localization and classification," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1994, pp. 3130–3135.
[11] S. Fazli and L. Kleeman, "Simultaneous landmark classification, localization and map building for an advanced sonar ring," *Robotica*, vol. 25, no. 3, pp. 283–296, May 2007.
[12] J. Steckel and H. Peremans, "BatSLAM: Simultaneous localization and mapping using biomimetic sonar," *PLoS One*, vol. 8, no. 1, p. e54076, 2013.
[13] H. Ismail and B. Balachandran, "A comparison of feature extraction algorithms based on sonar sensor data," in *Proc. ASME Int. Mech. Eng. Congr. Expo.*, 2013, pp. V04AT04A024-1–V04AT04A024-9.
[14] H. Ismail and B. Balachandran, "Feature extraction algorithm fusion for SONAR sensor data based environment mapping," in *Proc. ASME Int. Mech. Eng. Congr. Expo.*, 2014, pp. V04AT04A025-1–V04AT04A025-8.
[15] O. Wijk and H. I. Christensen, "Extraction of natural landmarks and localization using sonars," in *Proc. 6th Int. Symp. Intell. Robot. Syst.*, vol. 98. 1998, pp. 231–240.
[16] P. V. C. Hough, "Method and means for recognizing complex patterns," U.S. Patent 3 069 654, Dec. 18, 1962.
[17] K. Hormann and A. Agathos, "The point in polygon problem for arbitrary polygons," *Comput. Geometry*, vol. 20, no. 3, pp. 131–144, Nov. 2001.
[18] O. Wijk and H. I. Christensen, "Triangulation-based fusion of sonar data with application in robot pose tracking," *IEEE Trans. Robot. Autom.*, vol. 16, no. 6, pp. 740–752, Dec. 2000.

Hesham Ismail received the B.S. degree in mechanical engineering from the Petroleum Institute, Abu Dhabi, United Arab Emirates, and the M.S. degree in mechanical engineering from the University of Maryland, College Park, MD, where he is currently pursuing the Ph.D. degree in mechanical engineering. He is an ADNOC Fellow. His current research interests include simultaneous localization and mapping, feature extraction using SONAR sensors, and mobile vehicle dynamics.

Balakumar Balachandran (SM'11) received the B.Tech. degree in naval architecture from IIT Madras, India, and the M.S. and Ph.D. degrees in aerospace engineering and engineering mechanics from Virginia Tech, Blacksburg, VA. He is currently a Professor of Mechanical Engineering with the University of Maryland, where he has been since 1993. His research interests include nonlinear phenomena, dynamics and vibrations, and control. He is a fellow of the American Society of Mechanical Engineers and the American Institute of Aeronautics and Astronautics.