Feature Chain Based Occupancy Grid SLAM for Robots Equipped with Sonar Sensors

Amit Kumar Pandey* K. Madhava Krishna* Henry Hexmoor+

* Robotics Research Center, International Institute of Information Technology, Hyderabad, India
+Computer Science Dept, Southern Illinois University, Carbondale, IL, 62901, USA
akpandey@research.iiit.ac.in mkrishna@iiit.ac.in hexmoor@cs.siu.edu

Abstract— **This paper presents a methodology for achieving SLAM onto the occupancy grid framework with the data only from sonar sensors. Sonar data are highly noisy and unpredictable. Sonar does not give the consistent readings for a point from two different positions, so the approaches which rely on correspondence reading matching will prone to fail without exhaustive mathematical calculations of sonar modeling and environment modeling. Also, if features are being use to localize then the robot needs to revisit those features exactly, to localize, which itself will not be accurate because robot will not be at the exact position from where that feature has been detected. Hence it will not get back those feature readings using sonar. Here we are presenting a hybrid approach based on feature chain. Instead of relying completely on feature mapping and point matching, it finds the links between features to localize. It will drastically reduce the need of revisiting a feature to localize and hence reducing the exploration overhead, while handling other issues of problems with point or feature matching. We map features onto Occupancy Grid (OG) framework taking advantage of its dense representation of the world. Combining features onto OG overcomes many of its limitations such as the independence assumption between cells and provides for better modeling of the sonar implicitly providing more accurate maps.**

I. Introduction

THIS paper provides a framework for Simultaneous Map Building and Exploration problem (SLAM) which is considered as chicken and egg problem in mobile robotics. Because for building an accurate map we need to know the exact position of the robot and to know that we need to localize the robot, but to localize the robot we need to have the map. During the process of map building, robot losses track of its position due to unavoidable odometric error. As shown in the fig. 1, which is actually the map of a straight corridor, build by a robot, the straight parallel walls are appearing as curved walls due to this problem. There are basically two broad approaches for SLAM. One is based on features matching and another is based on raw data matching. But these approaches are basically dependent on the assumption, that the input from the sensors are reliable and consistent. Generally a combination of sensors like laser and sonar are used or laser [1] & [2] or camera [3] are used. But very few authors baring [4.5] have addressed the SLAM

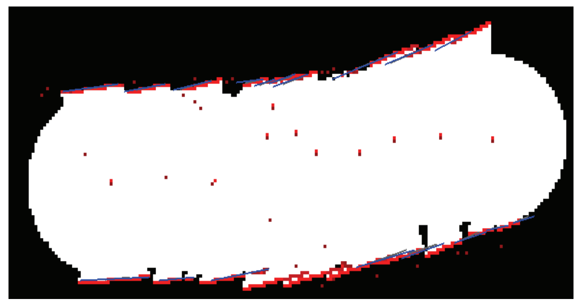

Fig. 1: Map of a straight corridor without local localization while building the map i.e. without SLAM. It is appearing as a curved corridor.

problem in indoor environment with only sonar sensors.. The reason is that, by using sonar sensors, the situation rapidly deteriorates since sonar readings are susceptible to high degree of uncertainty especially due to angular and radial errors along with specular reflection problems. In fact one cannot get a pair of consistent readings for a particular world location from two different positions.

As shown in the fig. 2 the world point P_w has been hit by two sonar beams from positions P_{t1} and P_{t2} of the robot at successive time instants. One may expect to obtain readings r_1 & r_2 as shown by solid lines. But practically one gets instead some unpredicted readings like r_{1extd} & r_{2extd}. So algorithms which will assume that those inconsistencies in the reading are due to robot's odometric error does not work

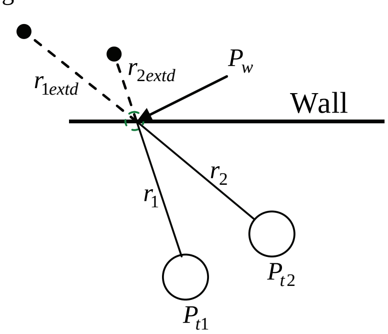

Fig. 2: From two position for sonar sensor instead of getting readings shown by solid line, we will get some unpredictable readings shown by extended dotted line, for a same world point Pw.

since it is very difficult to know whether the inconsistency in the reading is due to sonar error or due to error in robot pose for an environment. This problem is not seen however with the laser sensor.

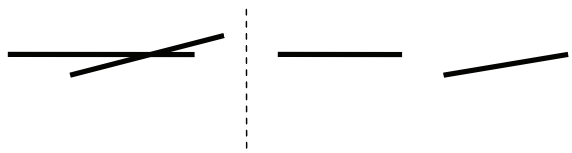

Fig. 3a Fig. 3b

Fig. 3a, 3b: For two instants even from nearly same position we can get two different wall segment features which can be overlapping but need not to be the same (fig. 3a) or non-overlapping (fig. 3b), for sonar sensor.

In feature based SLAM mapping once again there are problems with sonar sensors. The features which we can extract from sonar sensor are wall segments, corners and edges [6]. Each of such features has a sparse representation and a robot scan from two proximal places does not guarantee matching between features.

If we assume for a corridor like simple environment, with only array of sonar sensors, wall segments will be the feature which can be detect. And as shown in fig. 3a, in the best case it can detect two different wall segments with some overlapping parts and in worst case we will get non overlapping segments as shown in fig 3b. Hence not only is association of features between successive scans difficult, the usual image registration algorithms that try to solve for the rotational and translational displacement between successive scans do not work since overlap between features is minimal. While these problems do not exist with laser range finders and vision based sensors, sonar is still an attractive proposition due to its low cost.

In order to overcome the above problems and to achieve SLAM with array of sonar sensor only, in this paper we are presenting a novel approach of SLAM based on ***feature chains***.

II. Background Work

This paper is extension work of the papers [6] for including SLAM for robots having collection of data with array of sonar sensors. In [6], we have presented a novel approach to get a safe and more accurate map based on feature detection and mapping onto occupancy grid framework. For the continuity, the basics of the approach will be briefly discussed.

In general the reading from sonar is reliable only when at least one ray of the sonar beam hits normally to the surface, to which it is sensing. As shown in fig. 4 with different colors, three different sonar beams are hitting the walls with different axis angles, but the distance returned by all the three beams will be the perpendicular distance shown by the middle solid line, because only that ray will return back to the sonar sensor. And thus we will get a set of same reading, within a threshold, for all the sonar beams which have been fired within the range of beam width $\Delta\omega$ of one sonar beam. This region is called the ***Region of Constant Depth (RCD)***. And these RCDs are the most reliable readings in the case of sonar sensors.

There are special patterns of RCDs for walls and corners for a 360 degree scan by sonar sensors. In [6] we have handled this issue and able to extract the features like walls and corners and build the accurate and safe map of the environment. In [6], a detailed description of a Bayesian framework based approach for mapping is presented. Here we will assume that features like wall segments and corners have been extracted in each 360 degree scan by an array of sonar sensors, and we will directly use this information for extending the work of [6] for achieving SLAM.

III. Methodology

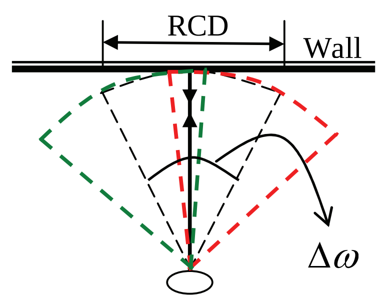

Fig. 4.: Concept of RCD. For all the three sonar beams shown as different colors, the perpendicular reading shown as middle solid arrowed line will be returned to the sonar sensor.

The overall approach has been shown in fig. 5. We will discuss each block based on their relevance in the current context.

A. *Sensing Environment*

After finding the approximate odometric errors in rotation and translation for self calibration, which is a one time experimental process, the robot starts sensing the environment. It makes a 360 degree scan of the environment. Instead of updating the map instantly with each reading, all the raw readings from all the sensors are being stored for further analysis and processing.

B. *Self Calibration Processing*

Since the data have been taken from various sonar sensors oriented at different angles and also with each degree of rotation some rotational errors are being introduced, we need to process the raw readings to filter out redundant data and to maintain inter sensor reading consistency. Rotational error calibration parameters are also used to correct the angle at which the reading is taken.

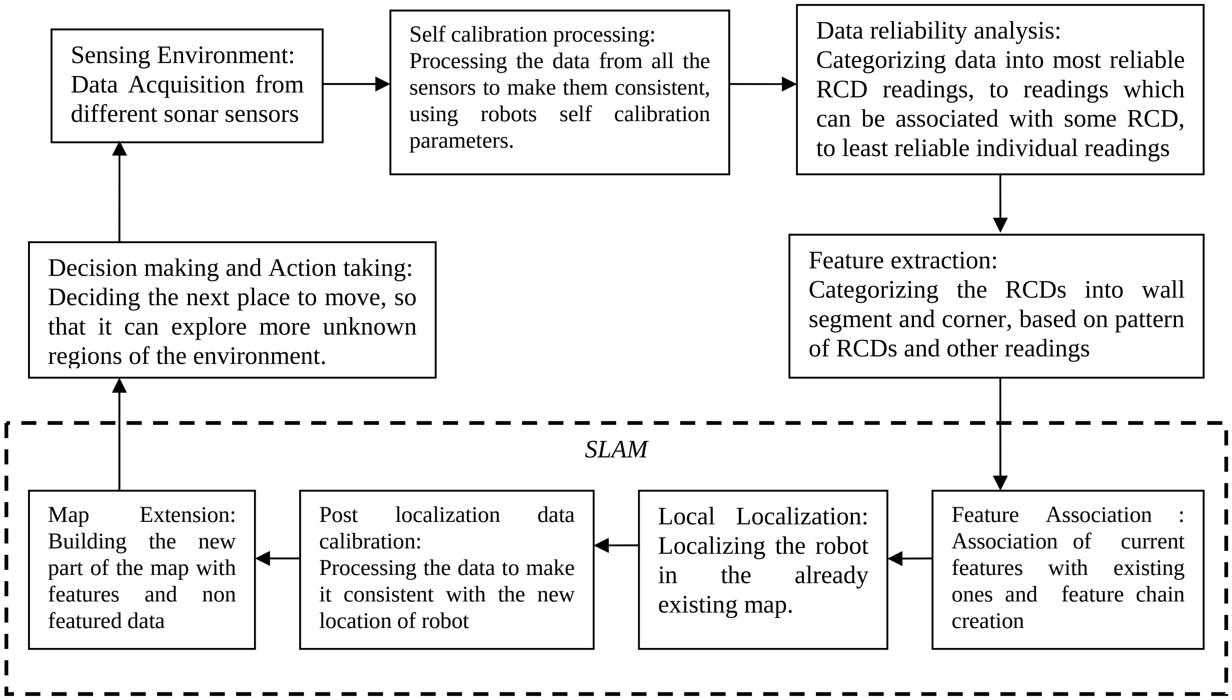

Fig. 5. The Diagram of the overall System with SLAM

C. Data Reliability Analysis

This basically categorizes the data by finding pattern in them and dividing into different clusters based on their reliability. As already mentioned RCDs are the most reliable data (but not 100% accurate), so first RCDs are found. Then those readings are found which can be associated with any of the RCDs using some association parameters. Thus basically the readings are divided into (i) RCDs, (ii) those which can be associated with some RCD (iii) non associated readings which are least reliable. They are used in different ways in building the map. For detail see [6].

D. Feature Extraction

Here we extract features based on the nature of sonar range pattern in RCD [6].

E. Feature Association

Since SLAM relies heavily on feature association, it is one of the crucial modules. We determine whether the current extracted feature is the part of or extension of any of the existing features or not. The most important feature is wall segment since they are based on RCD data whose angle and position are most reliable.

For the first 360 degree scan, for each wall segment feature found, a new feature chain is created and is made ***root*** of the chain.

For all the successive scans, features are being associated by analyzing various criteria. For associating wall features which are overlapping, a mean distance *d1* is calculated, which is the distances from both end points and middle point of one wall onto another wall. Also the angle *a1* between two wall segments is calculated. If they are within some threshold value, then both are associated. For non overlapping walls one more criteria is tested, the distance *d2* between the nearest end points of both the walls. If this is also within some threshold value only then both features are being associated. In our case we have taken *d1* as 20 cm, *d2* as 50 cm and *a1* as 20 degrees.

Once a current feature has been associated, that feature is inserted into that chain, which contains the already existing feature to which that feature has been associated. If the current feature can not be associated with any of the existing features then a new chain is being created with ***root*** as current feature.

F. Local Localization

After feature association has been done, to correct the localization error, we have defined two criteria for calculating orientation and translational error.

First orientation error is being found out. For this average angle difference *Avg_ang_diff* for all the associated features from the ***root feature*** of the chain is being found out. Then the robot is virtually placed in different orientation in the range of (*-Avg_ang_diff* to *+Avg_ang_diff*) in step of 1 degree from the current assumed orientation of the robot. The corresponding features are also being oriented to make them consistent with respect to the new virtual orientation of the robot. For each virtual orientation the new average angle difference is being calculated. Then the angle which is minimizing the average angle difference is considered as the actual angle of the robot.

For finding the translational error, robot is virtually placed in each cell in a window of size N x N with the orientation already found above. Also accordingly the features are shifted to make them consistent with respect to new virtual position of the robot. Then the distance *d1* explained in subsection *II E* is calculated. Then the position which minimizes the distance *d1* is considered as the actual position of the robot. Thus the corrected position and orientation of the robot is obtained.

G. Post Localization Data Calibration

Since a more accurate position of the robot has been found after localization, we have to change various parameters of all the RCDs, and associated and non associated data of the current scan, so that they will be consistent with the new location of the robot. For the wall segment the a,b,c parameters of the line equations are re-calculated. For other data point their new (x,y) co-ordinate are being re-calculated.

H. Map Extension

Having obtained a more accurate position of robot and accordingly modified features and readings parameters, the map gets updated by integrating the features onto occupancy grid as done in [6]. Thus a more accurate map closer to ground truth is obtained.

I. Decision Making and Action Taking

Once the map has been extended and updated, robot has to decide where to move next, to explore more unknown regions. For this frontier based exploration is used. Frontier cells are those cells which are empty and also at the boundary of unknown cells. Based on clusters of such cells, frontier regions are found. Then robot plans a path to reach to that region. And then again the whole process starts.

IV. Experimental Results and Analysis

We have tested our algorithm in different real environments. We are using Amigobot equipped with 8 sonar sensors fixed at different angles. Fig 6a and fig. 6b show two portions of a corridor as it bends around a corner, that is not at right angles, to the right shown by an arrow in 6a.

Figure 7 shows the map built, using the present method of SLAM for the environment shown in fig. 6a & 6b. The path of the robot has been shown in green line. Figure 8 shows the map obtained by robot of the same environment but without robot's position being corrected. The enhancement is visible. In fig. 8 the robot's position and orientation becomes more inaccurate after some scans resulting in a map that is shifted and tilted. Using the present approach however the map in figure 7 that is closer to ground truth results.

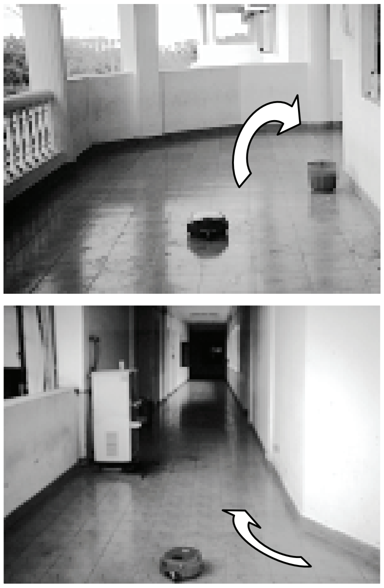

Figure 6a,6b: 6a portion of the corridor. 6b: Another portion of the same corridor. Note that wall at turn are not at 90 degree angle.

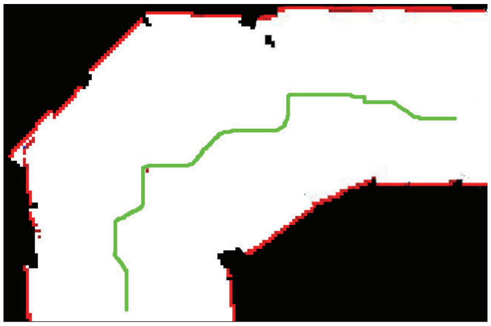

Fig. 7.: Map obtained by present method using current approach of SLAM for environment shown in fig. 6.

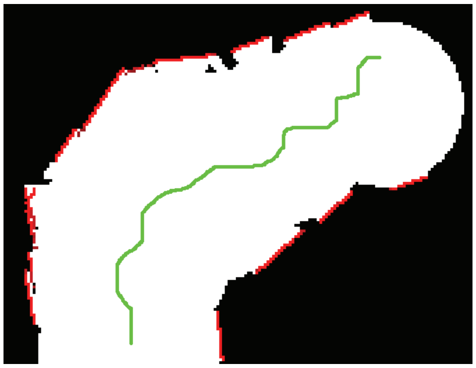

Fig. 8.: Map obtained of the same environment shown in fig. 6 without SLAM.

Fig. 9.: Corridor environment having two straight parallel wall.

Fig. 9 shows another corridor environment having parallel walls. Figure 10 is the map obtained by present method. Figure 11 is map without the SLAM approach. Once again the difference is obvious. In fig. 11 robot loses track of its position and hence the straight corridor becomes a curve, whereas in fig. 10, it able to localize by using current approach, so the straightness of the parallel walls has been maintained.

Fig. 12 shows side by side comparison of both the maps, with a dotted rectangle overlaid on them. The shift of the walls from the sides of the rectangle is more in right image which is without SLAM and the shift in left image is negligible.

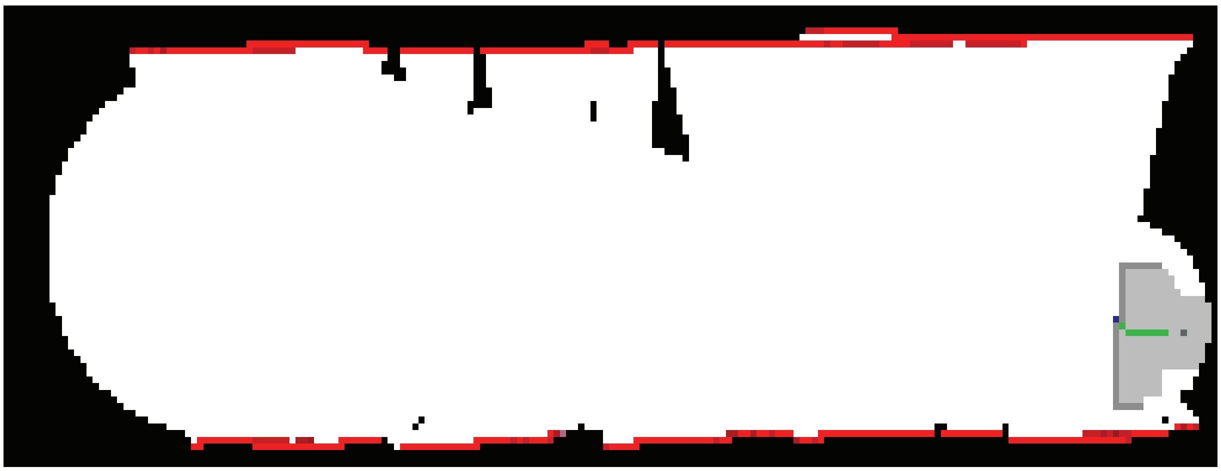

Fig. 10.:Map of the corridor environment shown in fig. 9 using present approach of SLAM.

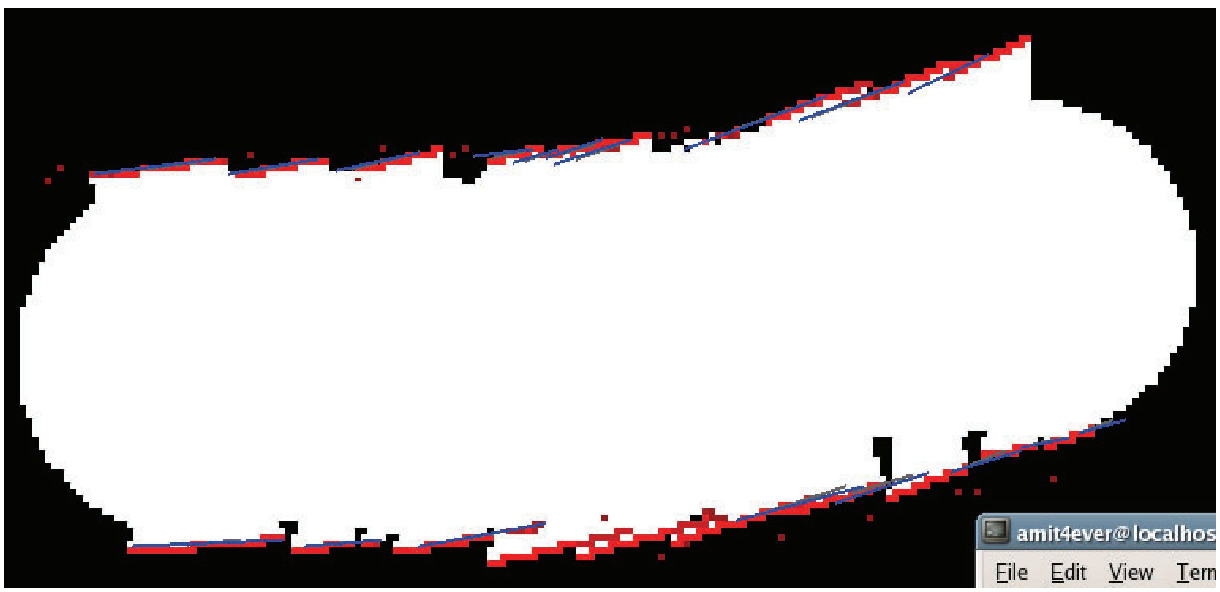

Fig. 11.:Map of the corridor environment shown in fig. 9 without SLAM.

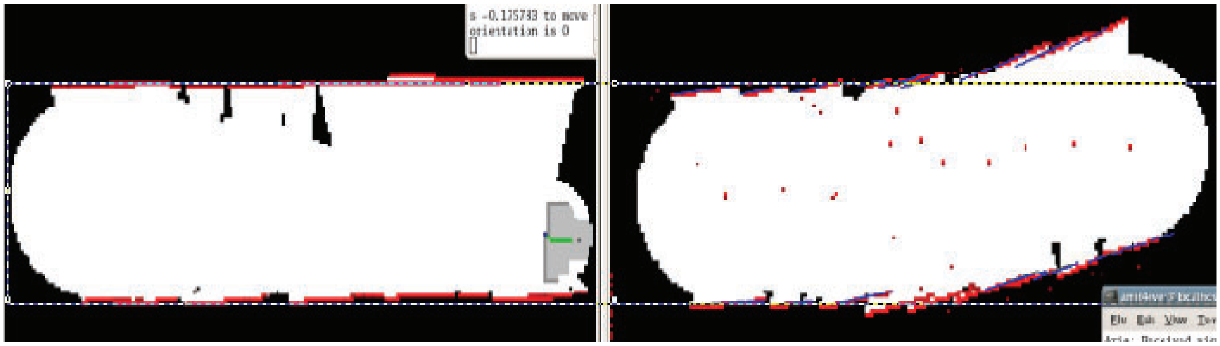

Fig. 12.: Comparison of both the maps obtained with SLAM (left) and without SLAM (right) , with an overlaid dotted rectangle resembling actual environment.

V. CONCLUSION

We have presented a ***feature chain*** based approach for SLAM by using data from sonar sensors only. The readings from sonar sensors are unpredictable after some minimum angle of incident of the sonar beam onto the obstacle. The presented approach is able to handle the various issues with sonar data and successfully able to localize and build more the accurate map. One of the benefits is that the robot does not need to re-visit the previously detected features each time it has to localize. It can use the information of earlier features to localize at current position even when those features have not been detected in present scan, by maintaining and using the ***feature chain***. The results are showing the performance gain of current approach.

REFERENCES

[1] Jens-Steffen Gutmann, Christian Schlegel, "AMOS: Comparision of Scan Matching Approaches for Self Localization in Indoor Environment,"1996.

[2] Cyrill Stachniss, Dirk Hahnel, "Exploration with Active Loop Closing for FastSLAM," IROS, 2004.

[3] Li Maohai, Hong Bingrong, "Novel Mobile Robot Simultaneous Localization and Mapping Using Rao-Blackwellised Particle Filter," International Journal of Advance Robotics System, 2006.

[4] Juan D. Tardos, Jose Neira, Paul M. Newman, John J Leonard, "Robust Mapping and Localization in Indoor Environment Using Sonar Data," International Journal of Robotics Research, 2002.

[5] Olle Wijk, Henrik I.Christensen, "Triangulation – Based Fusion of Sonar Data with Application in Robot Pose Tracking," IEEE Transactions on Robotics and Automation,2000.

[6] Amit Kumar Pandey, K. Madhava Krishna, Mainak Nath "Feature Based Occupancy Grid Maps for Sonar Based Safe mapping," IJCAI , 2007, to be published.