

## Continuous Localization Using Evidence Grids \*

Alan C. Schultz and William Adams  
Navy Center for Applied Research in Artificial Intelligence  
Naval Research Laboratory  
Washington, DC 20375-5337, U.S.A.

### Abstract

*Evidence grids provide a uniform representation for fusing temporally and spatially distinct sensor readings. However, the use of evidence grids requires that the robot be localized within its environment. Odometry errors typically accumulate over time, making localization estimates degrade, and introducing significant errors into evidence grids as they are built. We have addressed this problem by developing a method for "continuous localization", in which the robot corrects its localization estimates incrementally and on the fly. Assuming the mobile robot has a map of its environment represented as an evidence grid, localization is achieved by building a series of "local perception grids" based on localized sensor readings and the current odometry, and then registering the local and global grids. The registration produces an offset which is used to correct the odometry. Results are given on the effectiveness of this method, and quantify the improvement of continuous localization over dead reckoning. We also compare different techniques for matching evidence grids and for searching registration offsets.*

### 1 Introduction

For mobile robots to perform autonomously in dynamic environments, they need to have the ability to determine their location in their environment. Previous techniques for localization have looked at learning and recognizing landmarks in the environment, either as geometric representations or as a representation of sensor readings. In this study, the robot does not need to rely on the presence of specific landmarks, but instead uses the entire local environment of the robot to determine its location.

An important issue in localization is how often to relocalize the robot in its environment. Many existing techniques only occasionally relocalize when either

\*This work was sponsored by the Office of Naval Research.

an error in position is detected or after an unacceptable level of positional error has accumulated. Many techniques attempt to model the actual error bounds. Here we propose to use *continuous localization* (CL). In CL, the robot is continuously relocalized making regular small corrections instead of occasionally making large corrections. The benefit is that the error is known to be small, and fast correction techniques can be used.

Other important issues are the sensors used for localization and the representation used to capture the sensed data. A wide variety of representations and sensors have been used, but in most cases, the data has come from a single sensor type, such as stereo optic vision. In this work, an *evidence grid* representation has been used [3, 8, 9]. Evidence grids provide a uniform representation for fusing temporally and spatially distinct sensor readings. All robot sensors can contribute to the task of localization, and the system is robust in the face of sensor failures and noise in individual sensor readings.

In this work, we have an a priori *long-term map*<sup>1</sup> which is an evidence grid representation of the region (room). The robot builds *short-term perception* maps of its immediate environment. These maps are of a short duration, and typically contain only very small amounts of positional or rotational error. These short term maps are then used to position the robot within the long-term map via a registration process, the offset of which is used to correct the robot's current odometry.

In this paper, we show that this technique is capable of eliminating accumulated odometry errors with a resulting constant translational error on the order of five inches, or approximately the size of an evidence grid cell. We also compare different techniques for matching evidence grids and for searching for registration offsets.

In Section 2, we briefly describe our representation

<sup>1</sup>Current work is examining how to simultaneously learn maps while using them to stay localized[12].

for long-term and short-term perception maps, and describe how these maps may be registered. In Section 3, the method for CL is described. The robotic platform used in our experiments is presented in Section 4. We give the results of several experiments that demonstrate the technique’s ability in Section 5. In Section 6 we describe related work, and we give a conclusion in Section 7.

## 2 Representation

With evidence grid representations a volume is divided into cells. Each cell contains a real value in the range  $(-1, 1)$  that represents the amount of evidence that a cell is occupied (1) or unoccupied (-1), or indicates that there is not enough information to determine the occupancy of the cell<sup>2</sup>. Cells are updated from sensor readings that are filtered through a stochastic sensor model that determines the sensor reading’s influence on each cell, based on the pose (position and orientation) of the sensor at the time of the reading. After each sensor reading, all relevant cells are updated using the new evidence from the sensor. Several techniques have been used to update the evidence in the evidence grid representation including Bayesian techniques [8, 3], and Dempster-Shafer techniques [9]. In the work reported here, Bayesian updating is used.

Although evidence grids may represent a three-dimensional space, our initial results examine a single horizontal layer of the evidence grid that is located at the height of the sensors. In Section 7, we will discuss the use of all horizontal layers of the evidence grid.

Evidence grids have the advantage that they can perform sensor fusion, that is, they can combine the results from different sensors into the same representation. Evidence grids can also be updated in real time, allowing them to be successfully used for CL. Also, evidence grids are responsive to slow environmental changes; changes in the environment will be updated in the representation with additional sensor readings.

Many researchers currently use evidence grid representations in mobile robotics. One problem of using this representation of space is that the updating of the map with sensor readings requires that the current position and pose of the sensor be known. Unfortunately, odometry errors typically accumulate over

<sup>2</sup>In the Bayesian method, a value of 0 indicates that being occupied or being empty are equally likely, and this value is generally used as the *priors*. In the Dempster-Shafer method, not having enough evidence can be explicitly modeled.

time, making localization estimates degrade, and introducing significant errors into evidence grids as they are built. We have addressed this problem by developing the CL technique, in which the robot corrects its localization estimates incrementally and on the fly.

### 2.1 Long-term maps

A *long-term map* is an evidence grid representation of the environment that is built from many sensor readings, over a long time period in that region of space. In this paper, the evidence grids are produced in advance. Typically, each evidence grid will represent approximately one “room” in the environment. All sensor data contributes to this map, and this is the map that is used by other robotic processes, such as navigation and path planning. Fig. 1 shows an evidence grid of the robotic laboratory at NCARAI. The white space represents cells that have great evidence of the cell *not* being occupied (free space), the larger the circle, the greater the evidence of the cell being occupied. The medium size circles (like in the outer parts of Fig. 1) indicate cells where no evidence exists.

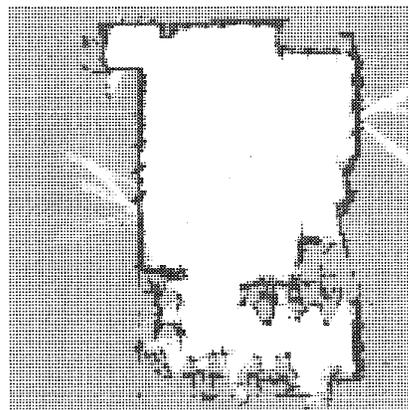


Figure 1: Long-term map of laboratory

### 2.2 Short-term perception maps

A short-term (or local) perception map represents the immediate temporal and spatial environment of the robot as an evidence grid. Only very recent sensor readings of the robot contribute to the local perception map. Several local perception maps of the robot’s environment may exist at the same time, each with a different amount of sensor data contributing to the “maturity” of that map. A short-term perception

map is considered mature when it has reached a limit in the amount of positional error that may have accumulated in the map. After a map has matured, it is used for correction of positional error, and then it is discarded.

Fig. 2 shows a short-term perception map recorded by the robot while it was in the upper, left hand corner of the room in Fig. 1. The circles and white space have the same meaning as in the previous figure. Note that objects are present in the short-term map that were not present when the long-term map was created.

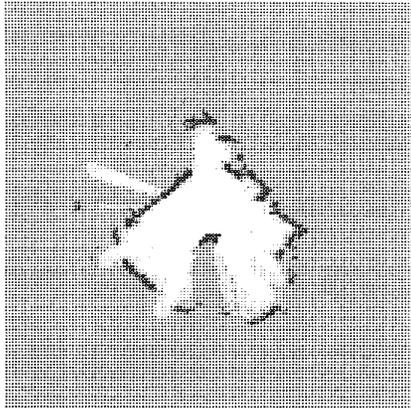


Figure 2: A short-term perception map

### 3 Continuous Localization

CL exploits the fact that the robot's odometric error usually increases gradually over time, except in extreme cases such as when the robot hits an obstacle, etc. By performing relocalization often, less effort is required to correct the error in odometry.

Fig. 3 shows a diagram of the CL process. Short-term perception maps are generated at regular intervals and several are maintained in memory. At the beginning of each interval, a new short-term perception map is created. During the time interval, new sensor data are fed to the new map and the previous maps still in memory. At the end of the interval, the oldest (most mature) short-term map is used to perform the registration against the long-term map and then discarded. The number of short-term maps that exist simultaneously and the amount of data that is entered into each map are runtime parameters of the system.

The registration of the short-term map to the long-term evidence grid produces an offset in both trans-

lation and rotation between the two. This offset, required to make the short-term map align with the long-term-map, is the same offset required to align the robot with the world, and is directly applied to the robot odometry (taking into account any robot motion since the registration was performed). All robot processes then use this new odometry.

The registration process involves a search in the space of offsets in translation and rotation that minimizes the error in the match between the short-term and long-term maps. Since we expect the odometry error to be small, we restrict the registration search to be between  $\pm 6$  inches in translation and  $\pm 2^\circ$  in angle. (These values can also be changed as runtime parameters.) This restricted search space allows the search to be completed quickly, specifically before the interval expires and the next registration is attempted.

In Section 5.1, we will present an experiment that shows the effectiveness of CL. In Section 5.2, the effectiveness of several search and match functions are examined in detail. As will be shown, the technique is robust to several methods of both search and match functions. In the next section, we will describe the robot platform.

### 4 Robot and sensor platform

A Nomadic Technologies Nomad 200 robot is used in the following experiments. The robot uses a three-wheel synchronized steering system. The robot is controlled by an on-board Pentium-based computer that is running the Linux operating system. The robot also has a radio ethernet, allowing processes to run concurrently on other workstations on the network.

Although the Nomad 200 has other sensors, only two types of sensors are used for CL in these experiments. A set of 16 sonar sensors are evenly spaced around the robot approximately 28 inches above the floor, each with a half-cone of 11.5 degrees and a range from six inches to ten feet. The robot also has a triangulation-based structured light range finder. This system returns 482 range data in a 15 degree arc parallel to and 31 inches above the floor. The range of this system, as configured, is 12 feet.

For use with the evidence grids, each sensor has a sensor model which determines how each cell in the evidence grid is updated based on the sensor position and datum returned. For the sonar sensor model, grid cells in an arc at the sensed range receive a higher evidence of being occupied, while cells between the sensor and the sensed distance receive reduced evidence of being occupied. Since the sonars are more likely

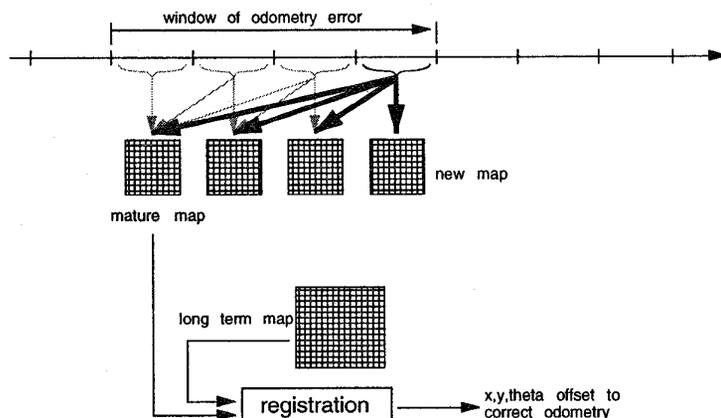


Figure 3: Continuous localization

to detect an object near its axis, cells closer to the sensor's axis receive larger adjustments than cells far from the axis. More information on the sonar sensor model is available in [8]. The sensor model for the structured light range finder provides strong evidence at the cell where the range datum lies, but makes no adjustment to any intermediate cells.

Sonar sensors can provide only coarse evidence of occupied space due to their wide field, but they are very effective at determining empty space, as an object anywhere within that space would likely have resulted in a shorter sensed range. The structured light range finder has the opposite properties. It can sense occupied space at a high resolution, but its horizontal, 2-D nature prevents it from sensing objects above or below the structured light plane. It therefore cannot be used with any confidence to rule the intervening space as empty.

## 5 Experiments

Several experiments were performed to determine the effectiveness of CL at reducing odometric error, and to determine which of several match functions and search functions yield better results.

### 5.1 Effectiveness of CL

The first experiment was conducted in a room measuring roughly 26 feet by 30 feet, open in the center with bookcases, desks, and chairs around the edges of the room. The robot was commanded to follow a square path near the center of the room, 8 feet on each side, by traveling to each corner's coordinates

in turn. CL ran independently of the motion process, maintaining 4 short-term perception maps and relocalizing approximately every 8 feet (each mature short-term map contained sensor data gathered during the most recent 32 ft of travel). The registration search method used was *center-of-mass* with the *binary* match function (described in detail in section 5.2).

Ten runs were made, with each run consisting of 80 laps around the square, a distance of 2560 feet (approximately 2 hours duration). The distance between the robot's odometric position and its true position was computed at the same corner for each lap. This measure includes rotational error, as motion causes error in orientation to be reflected as an error in position.

The results are displayed in Fig. 4 as an average across all ten runs. The robot's nonlocalized pose (simple dead-reckoning) steadily drifted, growing without bound. The localized curve shows that continuous localization was able to keep the robot's pose error at a constant level, averaging 5.35 inches (136 mm) and with a standard deviation of 2.08 inches (53 mm) across all points of all runs.

### 5.2 Search and Match Functions

The second set of experiments were run to determine the best of several search routines and matching functions that could be used to register the long-term and short-term perception maps.

In order to describe the search routines, it is useful to first describe the search space in which they work. The search space is all possible poses within  $\pm 6$  inches in translation and  $\pm 2^\circ$  in rotation of the robot's current pose. This corresponds to a 3-D space

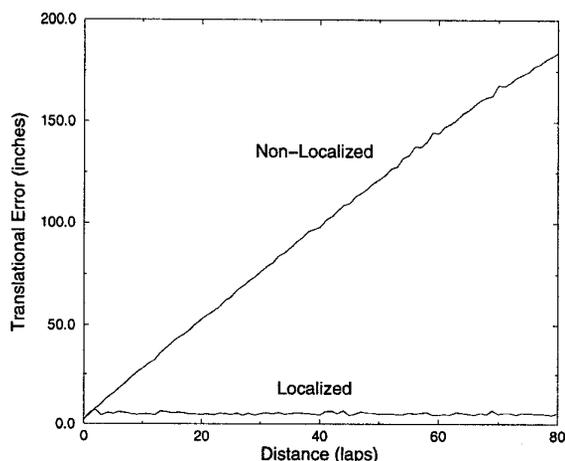


Figure 4: Experiment 1: Effect of continuous localization

with axes  $x$ ,  $y$  and  $\theta$ .

The two search routines tested were an *iterated hillclimber* and a *center-of-mass calculation*.

The iterated hillclimber search (designated in the text and graphs as  $H$ ) uses an initial resolution to divide the space into pose cells. The match between the short-term map and the long-term map is computed for the robot's pose and the center of the 26 immediately neighboring pose cells ( $3^3 - 1$ ). If a neighbor is found with a better match, then the process repeats using that pose cell as the center. If no neighbor is found to be better, then the hillclimber re-divides the space at double the resolution and repeats the process. The search stops when a predetermined resolution is reached. For the experiments reported here, an initial step size of 1.5 inches and 1.25 degrees was used, with a final resolution of 0.375 inches and 0.3125 degrees.

The "center-of-mass" search (designated in this paper as  $C$ ) similarly divides the search space into pose cells, but picks a random pose within each pose cell and uses those random poses to compute a set of match scores that are distributed throughout the search space. The match scores are normalized to the range  $[0,1]$ , raised to the fourth power to exaggerate the peak, and then a center-of mass calculation is performed for all cells. The exaggeration of the peak is necessary because the match score is typically very flat within the small search space, and without it the center-of-mass calculation would always pick a pose near the center of the search space (very close to the robot's current pose). The center-of-mass calculation

is preferable to simply choosing the pose cell with the maximum score because the sparse sampling of the space (one pose per pose cell) can create additional noise, and sampling at a higher resolution would be computationally prohibitive for real time operation.

The two match functions examined in this work are designated the *binary match* (referred to in this paper as  $B$ ), and the *product match* (referred to in this paper as  $P$ ). For both functions, the short-term map is aligned with the long-term map according to the pose from the pose cell the search is processing. The evidence from each grid cell of the short-term map is compared to the spatially-correspondent grid cell of the long-term map, and the score summed across all grid cells. Given the alignment for which the match score is to be computed, if  $C_L$  is the corresponding cell in the long-term map to the short-term map cell  $C_S$ , then we define the match score:

$$MatchScore = \sum_{all C_S} CellScore(C_{S_i}, C_{L_i})$$

For each match function, the cell scores are determined as follows. The binary match function ( $B$ ) compares the cells' evidence for simple agreement. It returns 1 if the cells agree occupied or agree empty, and returns 0 if they disagree or if either cell has no evidence (a value of 0).

$$CellScore(C_{S_i}, C_{L_i}) = \begin{cases} 1 & \text{if } 0 < C_{S_i}, 0 < C_{L_i} \\ 1 & \text{if } 0 > C_{S_i}, 0 > C_{L_i} \\ 0 & \text{otherwise} \end{cases}$$

The product match function ( $P$ ) determines the degree of agreement, taking the product of the cells' actual evidence, a value between -1 (empty) and 1 (occupied). Cells in agreement produce a score in the range  $(0, 1]$ , depending on the confidence of their individual evidence. Cells in disagreement produce a score in the range  $[-1, 0)$ , and if either cell has no evidence, a score of 0 is produced.

$$CellScore(C_{S_i}, C_{L_i}) = C_{S_i} C_{L_i}$$

Early work with the CL method revealed that the search space had large regions in which many registration poses resulted in the same match scores. This effect was suspected of causing the hillclimber to give up early due to the inability to find a better neighbor in the search space, resulting in a non-optimal choice of pose. To counter this problem, interpolation (designated with a  $I$  in the following text and graphs) can be performed on the long-term grid cells, such that the center of each grid cell retains its original evidence, but other locations within that grid cell have

evidence values bilinearly interpolated with neighboring grid cells. When the search routine aligns the center of the short-term map cells with the long-term map cells, the interpolated evidence value is used for computing the match score. Small variations in pose (map alignment) can thus yield differing corresponding long-term map cell values and thus differing overall match scores.

To evaluate the various combinations, the same environment was used as in the first experiment, with the robot following the same square path and with pose error being measured at the same corner. Eight trials were conducted, with each trial being a unique combination of search routine, match function and interpolation. (In the following figures and discussion, each trial is designated by the combination of letters *H, C, B, P, I* indicating which of the above techniques are being used. For each trial, 5 runs were made (except CP, CPI which had 10 runs). Each run consisted of 40 measured points (40 laps), with the pose error measured as before.

Shown in Fig. 5 is the average pose error across all runs for each trial. Error bars indicate a 95% confidence interval. As a group, the center-of-mass combinations were significantly better ( $p = .01$ ) than those using the hillclimber. In all cases, the binary and product match functions performed equivalently. Being of roughly comparable computational cost, we have chosen the product match function (*B*).

The CP and CPI combinations did not have significantly different performance, nor did interpolation have any consistent effect overall. Interpolation's smoothing of the search space appears unnecessary when used with the center-of-mass search, which performs its own smoothing during the averaging process inherent to it. Since interpolation incurs additional computational cost without providing any additional benefit, the CP combination was selected for future work.

## 6 Related work

In [11], Yamauchi uses evidence grids to perform occasional localization by matching evidence grids. In that study, evidence grids are created for each specific "place" along the robots path. When the robot revisited a specific place, it created a new evidence grid to match against the evidence grid for that location to correct its position.

An alternate search method by Lu [7] looks promising although it is intended for free-form scans without the use of evidence grids, and the effect of using it on

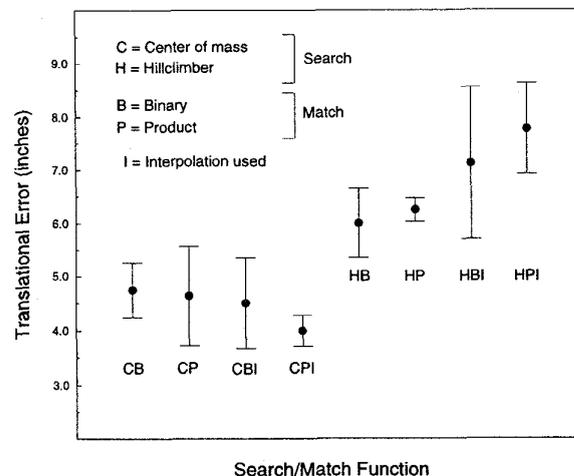


Figure 5: Experiment 2: Different search and match functions compared

the artificially rasterized data of evidence grids is an open question.

In an approach similar to that presented here, Schiele and Crowley [10] compared grid matching to other localization methods that included detecting and matching edge segments in the evidence grids. Their work did not give quantitative results on matching evidence grids, nor did it examine various methods for matching or searching for poses. The work presented here seeks to determine the sensitivity of grid matching to changes in some of its fundamental parameters and determine suitable values for them.

Many localization techniques rely on structures in the environment that can serve as landmarks, for example, vertical structures such as door posts and poles [2], large planes [5], or other geometric beacons [6]. Using specific landmarks often requires the robot to perform special maneuvers in order to locate or recognize these landmarks [1]. In our work, such maneuvers are unnecessary. Because our method uses all available sensor data without the requirement of specific features in the environment, the robot can localize itself transparently while carrying out its assigned task.

## 7 Conclusions/continued work

Many match functions and search methods are possible and we have only shown the performance of a few combinations. However, the method of continuous localization presented here has been shown to be robust

to the registration search method and match function. We have achieved an average pose error equivalent to the size of the evidence grid cell, and do not believe that other combinations would produce significantly better performance. Because evidence grids can fuse sensor readings, other combinations of sensors can be used. In future studies, we are interested in how other combinations of sensors will perform.

One problem others have noted with evidence grids is the inability to handle dynamic environments. We are looking at a method of updating the long-term map with the short-term map in order to track slow changes in the room, such as moved furniture, doors opened or closed, and blocked passages. Initial results indicate that the long-term map can be adaptive to changing environments, and still allow CL to work well [4].

A second problem occurs when there is a sudden large change in the robot's odometry, such as a hard collision that allows the wheels to slip. We are investigating techniques that will allow the CL algorithm to detect these conditions, and expand the range of the search in pose space.

We believe that the CL method will be robust to the underlying grid representation. We will demonstrate the CL method using a Dempster-Shafer version of evidence grids [9].

Continuing work also includes the integration of CL with frontier-based exploration [12] in order to map the room while remaining localized, eliminating the a priori map requirement and providing an accurate, learned long-term map. Initial results indicate that accurate maps of the room can be simultaneously learned and used for continuous localization.

## References

- [1] Bauer, R., "Active manoeuvres for supporting the localisation process of an autonomous mobile robot," *Robotics and Autonomous Systems* 16: Elsevier, pp 39-46, 1995.
- [2] Chenavier, F. and Crowley, J., "Position estimation for a mobile robot using vision and odometry," *proc. IEEE Int. Conf. on Robotics and Automation*, Nice, France: IEEE, pp 2588-2592, 1992.
- [3] Elfes, A. E., "Multi-source spatial data fusion using bayesian reasoning," *Data Fusion in Robotics and Machine Intelligence*, (eds. M. A. Abidi and R. C. Gonzalez), Academic Press, pp 137-163, 1992.
- [4] Graves, K., Adams, W., and Schultz, A., "Continuous Localization in Changing Environments," *proc. of the IEEE Int. Symp. on Computational Intelligence in Robotics and Automation*, IEEE, Monterey, CA, July, 1997.
- [5] Horn, J. and Schmidt, M., "Continuous localization of a mobile robot based on 3D-laser-range-data, predicted sensor images, and dead-reckoning," *Robotics and Autonomous Systems* 14: Elsevier, pp 99-118, 1995.
- [6] Leonard, J., Durrant-Whyte, H., and Cox, I., "Dynamic map building for an Autonomous Mobile Robot," *The International Journal of Robotics Research* 11: IEEE, pp 286-298, 1992.
- [7] Lu, F., and Milios, E., "Robot pose estimation in unknown environments by matching 2-D Range scans," *proc. IEEE Computer Vision and Pattern Recognition*, Seattle: IEEE, pp 935-938, 1994.
- [8] Moravec, H. P., "Sensor fusion in evidence grids for mobile robots," *AI Magazine*, pp 61-74, 1988.
- [9] Hughes, K., and Murphy, R.R., "Ultrasonic Robot Localization using Dempster-Shafer Theory", *SPIE Stochastic Methods in Signal Processing, Image Processing, and Computer Vision*, invited session on Applications for Vision and Robotics, San Diego, CA, July 19-24, 1992.
- [10] Schiele, B., Crowley, J., "A comparison of position estimation techniques using occupancy grids," *Robotics and Autonomous Systems* 12: Elsevier, pp 163-171, 1994.
- [11] Yamauchi, B., "Mobile robot localization in dynamic environments using dead reckoning and evidence grids," *proc. IEEE Int. Conf. on Robotics and Automation*, New York, NY: IEEE, pp 1401-1406, 1996.
- [12] Yamauchi, B., Schultz, A., and Adams, W., "Mobile Robot Exploration and Map-Building with Continuous Localization," *proc. IEEE Int. Conf. on Robotics and Automation*, Leuven, Belgium: IEEE, 1998.