Proc. 2002 Australasian Conference on Robotics and Automation Auckland, 27-29 November 2002

Autonomous Recharging for Mobile Robotics

Kirill Kouzoubov and David Austin

Robotic Systems Laboratory,
Research School of Information Sciences and Engineering,
Australian National University, ACT 0200, Australia.
{kirill | david}@syseng.anu.edu.au
robot.anu.edu.au

Abstract

To be fully autonomous, a robot should either have an indefinite power supply or should be able to renew it's energy by some means. In the case of indoor mobile robots this usually means recharging the batteries. This paper describes the recharging station and docking algorithms that we have developed and presents the results achieved. The docking station is a simple design, requiring only minor hardware modifications on the robot side, and the total cost is under $100. The docking algorithm uses a particle filter to extract the recharging station location from the laser range sensor data. The system was implemented on the XR4000 robot platform and, demonstrates that a simple approach can achieve high rates of success.

1 Introduction

Robots can achieve a lot more if they have more time to do it, unfortunately robots need energy to live. Indoor mobile robots commonly used for research in robotics [Borenstein *et al.*, 1996], usually use rechargeable batteries as the power sources. Batteries however, last for only a few hours, and so need to be recharged. Automating the process of recharging allows long-term autonomous operation without the need of human supervision and intervention.

There have been a number of successful attempts in implementing a recharging capability on autonomous mobile robots. Arguably, the first autonomous mobile robots capable of recharging were "tortoises" developed by [Walter, 1963], mid last century. More recently [Hada and Yuta, 1999] have presented the results of the week-long repetitive docking experiment. They have used infrared sensors and reflective tape on the floor as target to position robot for docking. The batteries were recharged from the external charger via a purpose-built DC connector. Recently, an interesting docking mechanism was reported that allows for high angular and displacement error during docking process[Silverman *et al.*, 2002]. A combination of vision and laser beacons are used to perform autonomous recharging of a Pioneer robot. Previously, a recharging system has also been implemented in our lab[Oh *et al.*, 2000]. It uses laser and infrared sensors to dock the ANU XR4000 robot to a standard IEC connector mounted on the recharging station. The ANU station is unlike many prior recharging systems in that it requires minimal modification to the robot. A second advantage of the ANU station is the low cost: less than $100. However, millimeter precision is required to plug an IEC connector in to the socket. The XR4000 robot is capable of performing sideways moves with millimeter accuracy, and the algorithm developed by Seungjun Oh[Oh *et al.*, 2000] used this feature of the robot to perform docking successfully. However due to the firmware bug in the robot's controller, the algorithm would stop working (small movements cease working) 24 hours after rebooting the robot. Here we describe the new recharging system, which requires less positioning accuracy.

The present recharging system takes a simplified approach using only one sensor and standard plugs and sockets. Use of standard components reduces construction and maintenance costs. It also helps to maintain safety of the system (another side-effect of the on-board recharging power supply is that the mains of 240V is required at the docking plug). The design of the docking mechanism we use is fairly simple, reducing maintenance times, it takes about twenty minutes for a person with low to moderate technical skill level to change over plugs and sockets (something that has to be done from time to time in any system).

The docking algorithm we have developed uses a particle filter to find and track the recharging station location and orientation. This allows multiple hypotheses about the recharging station location/orientation, so the robot can change its mind at any time if it was

Figure 1: The ANU Recharging Station

wrong at the start, and correct itself. Detecting failure early on is important since it allows us to abort safely before getting too close to the obstacle perceived as recharging station, and restart sooner reducing average docking time. Even when using just one sensor for target tracking we were able to achieve smooth docking path. The robot simply follows the arc from its start position towards the recharging station, although rotation on the spot is required at the very start.

The docking process consists of several steps:

1. **Locate the recharging station.**
2. **Plan Path.** The path is an arc starting at the current location and ending 10 cm in front of the plug, such that tangent at the end point is parallel to the plug.
3. **Initial Rotate.** Rotate on the spot, so that robot is on the tangent to the path planned.
4. **Follow the Arc.** Move along the arc, constantly reassessing your location relative to the recharging station using output of the particle filter.
5. **Docking.** If safe move straight forward with a slow velocity (10 mm/s).

2 Recharging Station

The recharging station consists of the power plug and the target for the robots laser sensor (see Figure 1). Our XR4000 has an on-board charging circuitry which is connected to the AC 240V power supply by a standard IEC plug, as typically used to power computers. IEC plugs are quite good in that they are proven to be safe and cheap and readily available, however millimeter precision is required to plug an IEC connector in to the socket. As discussed above, our XR4000 robot is capable of millimeter accuracy but due to the firmware bug in the robot's controller small movements cease working after approximately 24 hours of operation. To overcome this problem we decided to develop a docking system which does not rely on the sideways moves and only uses a car-like model for motion, with one exception: on-the-spot rotation is allowed. Initially we were hopeful to achieve millimeter accuracy, but we failed in that and had to redesign the plug.

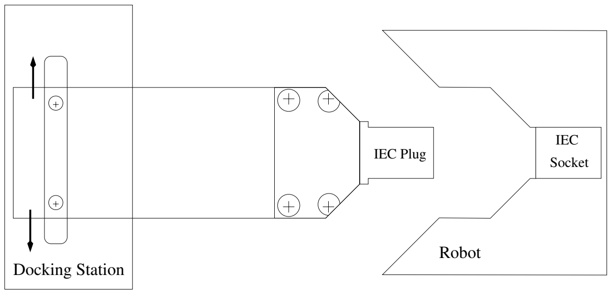


Figure 2: Recharging plug and socket. Compliant motions are shown with arrows.

Our solution is as follows: if the robot cannot position itself accurately enough relative to the plug, make the plug slide into the right spot instead. So we mounted the connector on a compliant platform (see Figure 2), which can move sideways $10mm$ left and right from the centre position and change it's orientation by a few degrees. The plug is also spring-loaded and can be pushed towards the recharging station by $20mm$, this also helps to reduce the impact during docking. This connector design allows for slight misalignment of the robot during the docking, increasing the reliability. When the robot is docking with the recharging station the forces from any misalignment move the plug to the correct position by sliding it left or right from the centre.

The decision to use a compliant platform rather than designing a new type of the connector was mainly due to safety concerns. As there is high voltage involved the plug must be safe for people. It seemed that designing a platform that can be aligned during docking was an easier and safer solution in our case. We also investigated moving the recharging power supply off-board, but the tight integration of the power supply to the main power distribution board made this too difficult. It would have required re-engineering a significant part of the main power distribution board.

The target we use to identify the recharging station is a grid which generates a highly distinct pattern when "viewed" by the laser range sensor. We have reused the grid developed by Seungjun Oh [Oh *et al.*, 2000] for the previous ANU recharging system. It consists of the backplate, $50cm$ in length, and four $3cm$ wide strips positioned $8cm$ in front of the backplate. The strips are equally spaced with $3cm$ gaps between them (see Figure 3). The central gap is located right above the centre position of the plug. All strips and the backplate have matte surface to reduce specular reflection and hence improve range sensor accuracy.

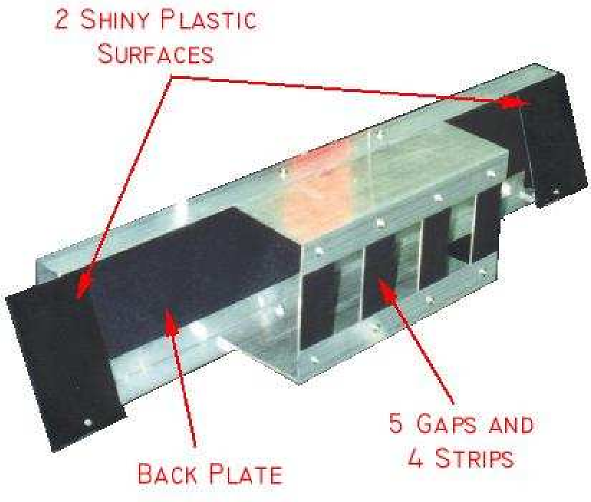


Figure 3: The recharging grid.

3 Position Tracking

We use laser range sensor data fed to the particle filter algorithm [Liu and Chen, 1999] to constantly reassess the robot's position and orientation relative to the recharging station. Laser data arrives on average three times a second, which is a fairly low frequency, so odometry readings are used in between the laser scans. Odometry information is available at $10Hz$ on our system. Particle filters have been commonly used in mobile robot localisation recently [Jensfelt *et al.*, 2000b; 2000a; Fox *et al.*, 1999; 1998] and proved to be particularly robust.

3.1 Initialisation

We start with no prior knowledge of where the recharging station might be in the environment. So we attempt to locate it by generating a large number of hypotheses. First, we extract lines from the laser scan using the hough transform [Risse, 1989]. The more data points support the line, the higher ranking that line has. We then use the 20 best lines to generate initial hypothesis. We go along those lines in $5mm$ steps and place three hypotheses on every step: one perpendicular to the line and two others at 89.5° and 90.5°. Obviously, the initialisation requires that the robot is facing the recharging station. In addition, this initialisation step assumes that the robot is reasonably close ($\approx 1 - 2m$) to the recharging station so that the grid forms a significant line in the hough transform data. The localisation methods that we use [Jensfelt and Christensen, 2001; Jensfelt *et al.*, 2000a] have no difficulty in achieving this level of accuracy.

After generating hypotheses we run the particle filter for a period (the next 10 laser data sets), before choosing the particle with highest weight as a potential location for the recharging station. This delay allows the particle filter to converge to the grid. Convergence is necessary because the hough transform used for initialisation matches any line-like structure, not just the grid of the recharging station.

3.2 Particle Filter

We have chosen to use a particle filter algorithm because it allows multiple hypotheses about the location and orientation of the recharging station. This is a desirable quality since it assists in developing a more robust system. The robot can change its opinion about the actual location of the recharging station at any time and correct itself.

The particle filter algorithm consists of three steps:

1. **Predict:** Guess possible solutions (distribute particles).
2. **Evaluate:** Assign weights to particles.
3. **Resample:** Redistribute particles based on weights assigned in previous step.

In the case of the recharger, each particle represents estimate of the location and orientation of the grid in the global coordinate frame. The robot's location within the global frame is maintained by the motor controllers. Due to inaccuracy of odometry sensors the reference point of the global frame drifts in the environment over time. So even though the grid itself does not move, it appears to the robot that motion has occurred.

The prediction stage of the particle filter deals with the odometric drift, it relies on the odometry readings and a kinematic model of the vehicle (including drift) to reassess particle locations when the robot moves. After the prediction phase, we evaluate each "guess" by assigning a weight to each particle. This is done by the means of the evaluation function (described in the next section).

Once evaluation is complete it is time to prune away bad guesses. This is done by the means of resampling. We pick particles randomly, the more weight the particle has, the more likely it is to be picked for the next round. Each particle can be picked more than once, in which case several copies of the same particle will be present. Effectively, by resampling we replace weak particles with more copies of strong ones. Of course, since the process is random, weak particles can still survive, but not for long, and if they do not regain strength they will eventually be pruned away.

So now we have done all steps of the particle filter, and hopefully end up with the better set of particles than we had before. The evolution of the particles in the particle filter is shown in Figure 5 in Section 3.4 below. The remaining step is the most problematic for particle filters: to determine a single estimate for the true location of the grid. One obvious solution would be to use best particle as grid true location, but this approach is prone to noise. Here, we decided to use weighted centroid of the particles in the neighborhood of the strongest particle instead. This gives stability, and gives a better estimate of the actual location of the recharging station.

3.3 Evaluation Function

The evaluation function gives us a measure of how good our guess about grid location and orientation is, given the set of data points. We use laser range sensor data as input to the evaluation function. The evaluation function uses only a subset of the data points: only points in the neighborhood ($0.5m$ radius) of the particle are used, others have no effect on the output of the function.

The evaluation function consists of three parts:

1. **Matching Points Reward** Every data point in the neighborhood of the grid which is close enough ($\leq 20mm$) to one of the line segments of the grid is considered to be a matching point and adds towards the reward part, the closer the point the better.

2. **Non-Matching Point Penalty** Data points in the neighborhood of the grid which do not match contribute towards the non-matching point penalty. We use a sigmoid function to calculate the penalty, if there are only a few non-matching points the penalty is low, if there are more of them penalty increases rapidly.

3. **Non-Matching Segment Penalty** Segments of the grid that do not have data points to support them add to the non-matching segment penalty. For that purpose, we treat the front and back rows separately. The penalty is really severe if one of the rows is not matched, so that flat surfaces like walls have low ranking. In other words, if all data points that match the grid only match the back segment, or only match at the front segments, we say it's highly unlikely that there is a grid there.

Each sub-part of the evaluation function lies in the range $[0, 1]$ and the total evaluation function is simply the product of these parts.

Although the laser range sensor is generally quite good, it sometimes generates false data points. This problem becomes more significant when robot gets close to the grid. It seems that when the laser sensor receives more than one reflection from the beam it returns the average distance reading. As a result false points appear at the jump edges in the gaps of the grid.

We deal with this problem by introducing imaginary segments in the grid matching algorithm (see Figure 4). These segments start at the edge of the front strip and continue to the backplate along the line connecting the edge of the strip and the laser sensor. Once these segments are added, they are treated by the evaluation function as if they were real. We only use the imaginary segments at the final stages of the docking, when close to the grid and the likelihood of false readings is higher.

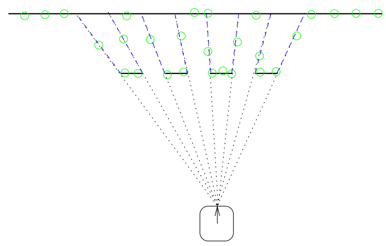


Figure 4: Imaginary Grid Segments

3.4 Particle Filter Evolution

Figure 5 demonstrates the evolution of particles. We start with large number of guesses (part a). After a few resamples particles start to condense (part b). Then we choose the best guess and start rotating towards it, there are still several clusters of particles (part c). Some time later the robot has moved towards the recharging station, and all particles are now clustered in one location, that of the recharging station.

4 Control

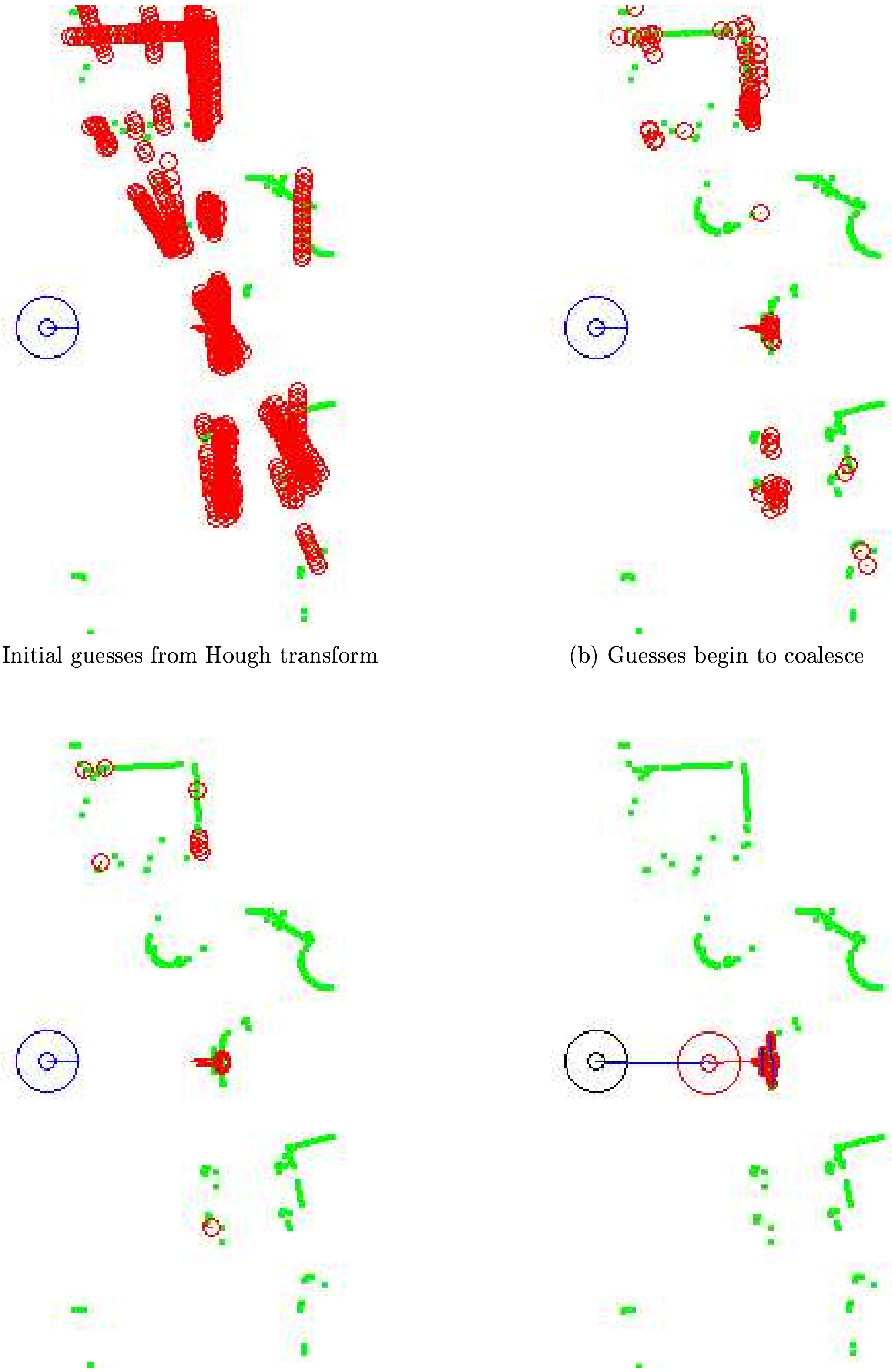
Once the first estimate of the grid position is available, a circular arc path is planned from the current location of the robot to the grid. The arc is selected so that it is perpendicular to the grid at the end (hence, the robot is moving parallel to the plug when docking).

The robot drives along the arc with constant translational velocity of $30mm/s$ and its angular velocity is controlled by a combination of two PID controllers to steer the robot towards the recharging station. One controller is concerned with the radial displacement error, i.e. how far robot is from the arc, and the other one is concerned with the orientation error. Clearly, better control strategies could be employed. However, this approach was developed as a simple test and has proved quite effective in practice.

5 Docking Results

Twenty seven test runs from different initial locations and orientations have been performed. Figure 6 shows locations from which docking has been attempted and the results are summarised in the Table 1. Three attempts did not succeed, giving a success rate of 89 per cent.

Out of the three attempts that failed, one was detected before docking: the orientation error was too big (3.4°). In the other two cases robot docked too far to the left from the plug, and the connection was not established. In these cases failure can be detected by monitoring battery voltage levels, and corrected by retrying. In all cases, the robot successfully docked, sometimes automatically detecting failure the



(a) Initial guesses from Hough transform

(b) Guesses begin to coalesce

(c) Guesses converge (to the correct location), robot begins to rotate

(d) Following arc (current position shown, as well as final docking position)

Figure 5: Evolution of particle filter. *Dots indicate the laser readings, small circles show hypotheses about the recharging station location (particles of the particle filter). The large solid circle is the robot's location. The grey large circle in (d) is the planned final location of the robot.*

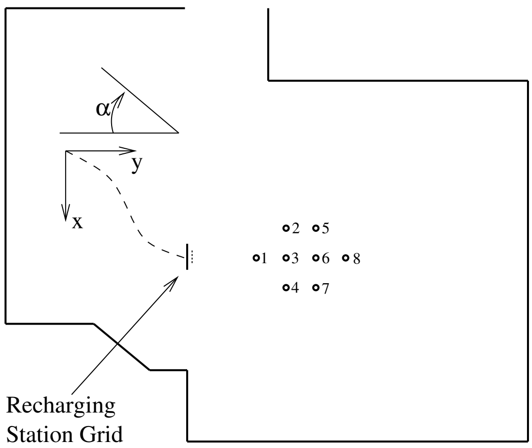


Figure 6: Test locations and coordinate definitions within the mobile robotics lab.

			Success Rate			
Location	X(m)	Y(m)	30°	0°	−30°	Av.T
1	0.0	1.0	1/1	1/1	1/1	73 s
2	-0.5	1.5	1/1	1/1	1/1	121 s
3	0.0	1.5	1/1	1/1	1/1	105 s
4	0.5	1.5	1/1	1/1	1/2	120 s
5	-0.5	2.0	1/2	1/1	1/1	135 s
6	0.0	2.0	1/1	1/1	1/1	128 s
7	0.5	2.0	1/1	1/1	1/1	140 s
8	0.0	2.5	1/2	1/1	1/1	155 s

Table 1: *Results of the test runs. Coordinates are as defined in Figure 6.*

first time and succeeding on the second attempt. The docking process is quite fast, taking less than two minutes on average. One positive side-effect of the new recharging system is that the process appears smoother and more purposeful to the lay observer. The old position-based recharging system appeared far more hesitant, stopping for measurement after every movement.

Figure 7 shows the results of 33 trials with the docking system integrated into the whole autonomous mobile robot system. There is a higher rate of failure because the robot localisation is often poor. Docking is still successful, even if automatic retries are required.

6 Conclusion

Our docking controller was designed to be a part of a larger system, and as a result we concentrated mainly on detecting failure, the recovery capabilities of the controller are rather limited, since they are delegated to higher levels of the overall system. Despite the simple design and implementation, the recharging system proved to be quite effective. A commendable success rate of 89% has been achieved from wide area in the neighbourhood of the recharging station. Good failure detection was observed, even when running recharger from tricky locations.

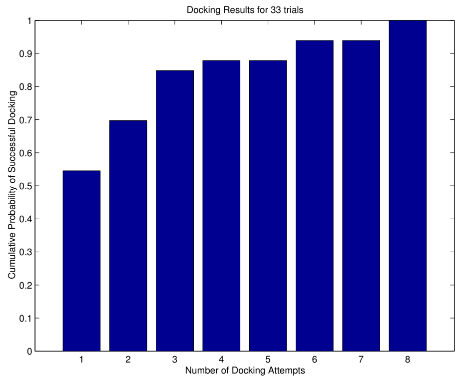


Figure 7: Cumulative probability of successful docking as a function of number of attempts when integrated with the whole robot software system.

This recharging system is not limited to the XR4000 robot platform. It can easily be implemented on other differential drive robotic platforms equipped with laser range sensor for a cost of $100.

References

[Borenstein *et al.*, 1996] J. Borenstein, H. R. Everett, and L. Feng. *Navigating Mobile Robots.* A K Peters, 1996.

[Fox *et al.*, 1998] D. Fox, W. Burgard, S. Thrun, and A.B. Cremers. Position estimation for mobile robots in dynamic environments. In *Proc. of the National Conference on Artificial Intelligence*, 1998.

[Fox *et al.*, 1999] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11, 1999.

[Hada and Yuta, 1999] Yasushi Hada and Shin'ichi Yuta. Robust navigation and battery re-charging system for long term activity of autonomous mobile robot. In *The 9th International Conference on Advanced Robotics('99 ICAR)*, pages 297–302, October 1999.

[Jensfelt and Christensen, 2001] Patric Jensfelt and Henrik I. Christensen. Pose tracking using laser scanning and minimalistic environmental models. *IEEE Transactions on Robotics and Automation*, 17(2):138–147, April 2001.

[Jensfelt *et al.*, 2000a] Patric Jensfelt, David J. Austin, Olle Wijk, and Magnus Andersson. Feature based condensation for mobile robot localization. In *IEEE Intl. Conf. on Robotics and Automation*, 2000.

[Jensfelt *et al.*, 2000b] Patric Jensfelt, Olle Wijk, David J. Austin, and Magnus Andersson. Experiments on augmenting condensation for mobile robot localization. In *IEEE Intl. Conf. on Robotics and Automation*, 2000.

[Liu and Chen, 1999] J. Liu and R. Chen. Sequential monte carlo methods for dynamic systems. *Journal of the American Statistical Association*, 94:590–599, 1999.

[Oh *et al.*, 2000] Seungjun Oh, Alexander Zelinsky, and Ken Taylor. Autonomous battery recharging for indoor mobile robots. In *Australian Conference on Robotics and Automation*, August 2000.

[Risse, 1989] T. Risse. Hough transform for line recognition. *Computer Vision and Image Processing*, pages 327–345, 1989.

[Silverman *et al.*, 2002] Milo C. Silverman, Dan Nies, Boyoon Jung, and Gaurav S. Sukhatme. Staying alive: A docking station for autonomous robot recharging. In *IEEE Intl. Conf. on Robotics and Automation*, 2002.

[Walter, 1963] W. Grey Walter. *The Living Brain.* W.W. Norton, New York, 1963.