Mobile Robotics in the Long Term
–Exploring the Fourth Dimension–

David Austin, Luke Fletcher, Alex Zelinsky

Robotic Systems Laboratory,
Research School of Information Sciences and Engineering,
Australian National University, ACT 0200, Australia.
david, luke, alex@syseng.anu.edu.au
www.syseng.anu.edu.au/rsl

Abstract

This paper explores the issues involved in deployment of mobile robots in real-world situations and presents solutions and approaches under development at the Australian National University. For deployment of mobile robots outside of the laboratory, long-term operation is required. Hence, we have developed an automatic recharging system. In addition, a web-based teleoperation system is used to provide missions to test the long-term reliability of the robot. The final aspect of real-world operation that is explored here is operations in dynamic environments. To date, researchers have assumed static environments for mapping and localisation. Here we propose methods to avoid this restriction.

1 Introduction

Mobile robotics is today reaching the point where deployment into real-world situations seems possible. Enabling technologies such as path planning, localisation and obstacle avoidance have all been proven in laboratory situations. However, long-term experiments with mobile robots are still quite rare. This paper describes the mobile robot and the framework for long-term experiments that is being established at the Australian National University. In addition, we discuss the common assumption made, that the environment of the mobile robot is static, and propose approaches to remove this restrictions.

To date, only a few researchers have considered the problems of long-term mobile robotics. The robot Xavier [1] has been in operation for a number of years, available for teleoperation on the web. However, Xavier possesses no automatic recharging system and therefore is only available some of the time[1]. Since the batteries are charged manually there is some supervision of Xavier's operation. Two generations of museum tour guides developed by Burgard *et al.* [2] and Thrun *et al.* [3, 4] both demonstrated longer term operations, online for a total of 32 hours 18 minutes and 94 hours and 23 minutes, respectively. Again, these systems required manual battery charging and so had a degree of supervision. In Japan, Yuta and Hada [5, 6] have started a project to develop an automatic recharging system but this system has yet to be demonstrated with long-term experiments. Our goal is to develop a system that can run 24 hours a day, 7 days a week for up to a year with no supervision. Once this goal has been achieved, we will know that mobile robotics has reached maturity.

A second, important consideration for the deployment of mobile robots in the real world is that they must be able to deal with dynamic environments. There are two factors that are generally neglected when considering the environment of the robot. The first is that there are truly dynamic objects, such as people, moving around at the same time as the robot is building its map and later when it is using the map to determine its position. Methods are needed which detect motion of objects in the environment and reject sensor data from those areas. Secondly, the environment of the robot will change over time as furniture is moved, doors are opened and closed, etc. While robots can successfully localise by treating any changes in the environment as noise (see for example, Thrun *et al.* [7, 8]), this is sub-optimal.

[1] 2pm to 4pm on every second weekday

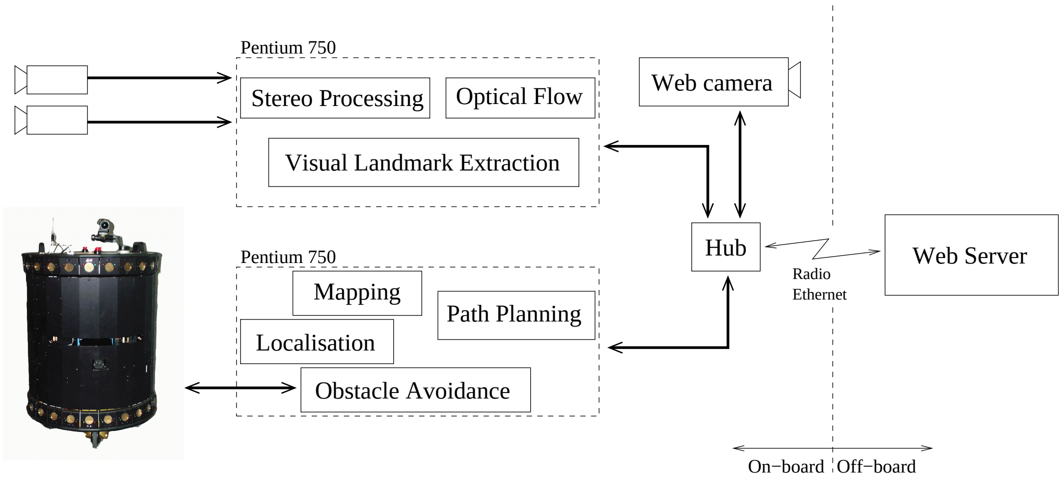

Figure 1: Overview of the hardware and software components that are used for the mobile robot

Clearly, techniques are required for localisation in dynamic environment and to maintain the map of the robot over time.

This paper presents our initial steps towards the development of a reliable mobile robot system which can operate autonomously for long periods in real-world environments. Section 2 presents an overview of the hardware and software components of our robot. In Section 3, we discuss the problems of batteries, power management and recharging. Section 4 outlines our web-based teleoperation system which will be used to provide tasks for the mobile robot over a long time. In Section 5, we discuss the static environment assumption and its removal, including methods for detecting motion and proposals for maintaining maps. Finally, we conclude with a discussion in Section 6

2 System Overview

The hardware and software systems required to build and run a mobile robot are extremely varied and complex. Clearly, we, the robotics community, must develop methods to manage this complexity or we will be unable to create reliable and scalable robot systems. Experimentation on real-world robot systems is required to gain the experience and insight to address the issue of complexity.

The system developed for our mobile robot is shown in Figure 1. The robot contains two on-board computers, both 200MHz Pentium CPUs. The lower CPU runs the navigation software, consisting of localisation, path planning and obstacle avoidance modules. At this stage, we are using a navigation system kindly supplied by Sebastian Thrun though we are working on our own system[9]. In addition, the mapping and map maintenance (discussed in Section 5 below) runs on the lower computer. The upper CPU has video capture cards and is used to process the incoming video data.

3 Batteries and Recharging

One of the important aspects for autonomous operation of a mobile robot is its power source. Most mobile robots today use batteries to provide power both for motion and for computation. Batteries can only store a finite amount of energy. Therefore, the management of the energy supply of the robot is a crucial function for long-term operations. The robot must monitor the state of the batteries and periodically recharge them. There are many possible ways to implement recharging. Figure 2 shows the recharging station that has been built at the ANU. The recharging system for our robot has been implemented as follows: at the top there is an infrared beacon which is used to locate the recharger from a distance; once the robot is fairly close to the recharger, the robot servos using the laser scanner and the reference grid in the middle of the recharging station; finally, the robot docks with the power plug just below the grid (see Figure 2).

The robot is equipped with current and voltage measurement circuitry which can be used to estimate the amount of charge remaining in the batteries and hence the amount of time before the robot must dock and recharge. At present, we are having problems with the battery charging circuit which cause the batteries to be unevenly charged

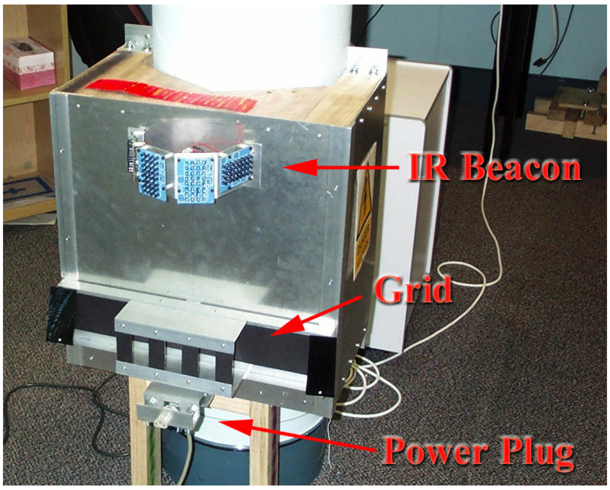

Figure 2: Recharging station

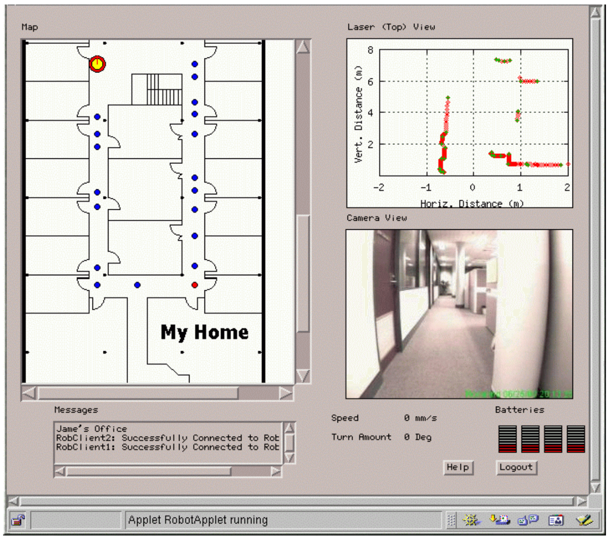

Figure 3: Web-based teleoperation interface

and so we are having problems estimating the remaining charge. However, we are making modifications to the charging circuit to remedy this problem. Also, we believe that the data gathered from longer term experiments will permit us to build a good model for the batteries.

4 Web-based Teleoperation

One of the requirements for long-term experimentation with a mobile robot is a task or series of tasks to conduct which will take considerable time. An option is to just generate random points within our building and send the robot to them. Instead, we have chosen to make the robot available on the web for remote users. Figure 3 shows the user interface that is presented. Remote users can click on a map of the building, sending the robot to those locations. For feedback, remote users are provided with position information, current laser scans and images from the web camera. The teleoperation system includes landmark-based localisation scheme which keeps track of the robot's position, a graph-based navigation system for finding paths to user goals and an obstacle avoidance module to avoid un-mapped obstacles.

For the moment, the teleoperation system is used simply to provide an ongoing set of tasks for the robot to complete. In the future we would like to investigate methods to allow remote users greater control and perhaps even the ability to write Java programs to control the robot.

5 Dynamic Environments

To date, almost no research has been conducted on the effects of dynamic environments on the localisation and navigation schemes that are necessary for operation of a mobile robot. Existing localisation schemes treat motion as noise and so are less robust than is possible [10, 11, 12]. Most mapping schemes ignore the problems of creating and maintaining a map in an environment which contains moving objects (e.g. [13]). An assumption of a static environment is almost universal. However, this assumption cannot be justified in the vast majority of mobile robot applications because there are humans moving about and furniture and other objects which also change position over time. This project will extend previous work in feature-based mapping [14, 15] to dynamic environments. This area is particularly important if mobile robots are to be used in the real world where laboratory assumptions do not hold.

5.1 Localisation in dynamic environments

People, the most common type of dynamic object in the indoor mobile robotics environment, disturb localisation schemes. To keep the problem tractable the static environment assumption remains a cornerstone of most localisation approaches. To localise in an environment with dynamic objects the objects must be detected and filtered from the sensor data. As mentioned in [7], dynamic objects introduce systematic rather than Gaussian noise into the sensor readings. In order for the Markov assumption to still hold researchers must either filter sensor readings to remove motion or augment the robot state to track the dynamic objects. Clearly, discarding sensor readings that

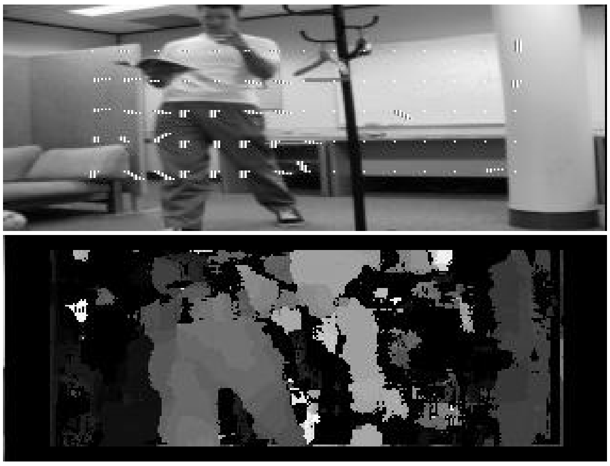

Figure 4: Example flow and depth-map images

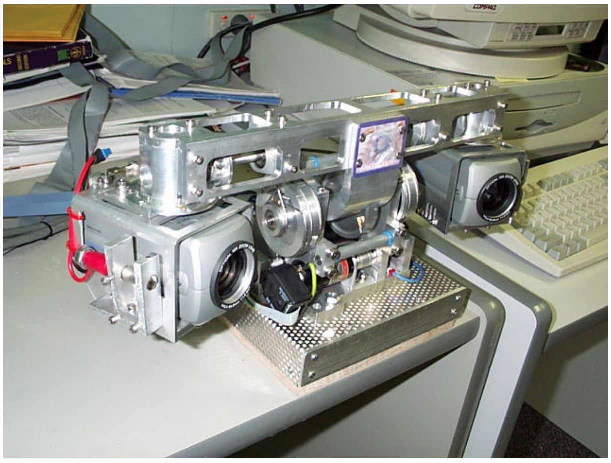

Figure 5: CeDAR active vision system

come from dynamic objects in the environment is by far the simpler approach [7].

Our approach similar, using stereo vision we want to: detect the presence of dynamic obstacles, estimate their trajectory and mask their data from other sensors used for localisation. A depth-map and 3D flow field will be generated in real time to detect the dynamic obstacles. The depth-map and 3D flow field are created following the method of [16]. This method exploits computational redundancy, cache optimisation and the Intel processor MMX instructions to produce a real-time 3D flow field. Figure 4 shows the real-time flow field and depth-map of a dynamic object. The 3D flow field effectively highlights motion in the scene that is not due to egomotion of the robot. Detecting dynamic objects becomes a process of segmenting the 3D flow field into regions where the magnitude is large. The trajectory is estimated using the average of the flow vectors associated with the object. We will combine the stereo vision detection of motion with our previous systems for [11, 12, 17] localisation to improve their robustness.

In future the disparity map and 3D flow field may also be used for reactive path planning. The trajectory of people walking about the robot can be estimated and then a path can be planned to swerve around the persons if possible. Person following experiments are also planned as CeDAR, an active stereo camera platform will soon replace the fixed stereo pair for the robot. CeDAR is shown in Figure 5 and described in [18].

5.2 Mapping of semi-static environments

Semi-static environments are an important class to consider because there are immediate applications in industrial situations such as warehouses, where there are less people but the environment does change slightly over time. This is an important issue for map maintenance. For example, many maps slowly become inaccurate as the furniture is moved. In a typical room a significant number of landmarks lie on the furniture and so are frequently moved. To deal with the slow changes in the environment, techniques are required for map maintenance. Present techniques gradually decay the certainty of moved objects and create new objects in the new position [19, 13]. Note, however, that this takes quite some time as a considerable number of measurements must be aggregated. Also, this treatment of semi-static objects is undesirable because it does not remember anything further about the object. Intelligent techniques are needed which can extract more information from the motion of semi-static objects to more rapidly detect when the object moves.

We will build on previous work in map making and extend it to semi-static environments. Our approach is to develop feature-based maps (grid-based maps require considerable memory and, given that we want to study the map over long periods of time, storage is a big issue) and we formulate the mapping process as a minimisation problem, in which an explicit error function is minimised [14, 15]. The feature-based mapping problem is much too complex to determine the optimal solution, the global minimum of the error function. Instead, sub-optimal solutions must be sought. We define two sets of operations which: a) integrate new sensor readings into the existing map and b) clean up the resulting map. When a

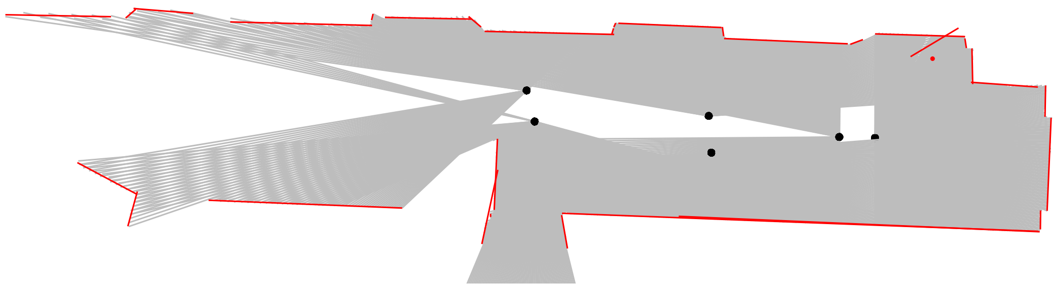

Figure 6: Map built using the optimisation technique [15]

new sensor reading is taken, each of the additive operations is tried and the one which minimises the error function is selected. After the integration of a sensor measurement, the cleaning operations are used to remove some of the errors caused by real-world data. Map optimisation proceeds by finding the best operation to apply at each step.

Note that casting the mapping process as an optimisation problem directly results in the ability to correct past mistakes. Consider a case where a sonar reading is assigned to a line segment in the map and later information reveals that this data point has been assigned to the wrong map feature. As part of the optimisation process, a cleaning operation can then be applied which takes the outlying sonar reading and assigns it to the correct object in the map (assuming that the set of operations is correctly chosen [14]). In this manner, if a decision is made which, in the light of new data, turns out to be false, the cleaning operations can correct the mistake.

Figure 6 presents results of the map building process from six laser scans taken in a corridor. This figure demonstrates the promise of this method, it is able to build a good map of the environment from a small amount of sensor data. Rejection of the smaller line segments will further improve the quality of the map. This map contains 45 line segments and so requires little memory to store, an important consideration when we are studying the map over time.

A particular problem for robots, and particularly in computer vision, is to associate each of the sensor values with an object in the environment. For vision, this means dividing the picture into sections that are associated with the various objects in the picture. For a human, who understands the way the objects in the picture behave, this problem is fairly simple. However, a robot does not have our extensive knowledge and experience. Modeling of the motion of objects over time can provide an automated solution to this problem, endowing the robot with experience. For example, Dar *et al.* [20] present a preliminary approach to recovering the behaviour of objects from image sequences and Beetz *et al.* [21] present a semi-automatic method for acquiring all of the objects within designated area of the environment. However, both of these methods require a considerable amount of information be provided manually. Robots must obtain this information automatically if they are to be successful. By modeling semi-static objects, this project will develop fully automated methods for object acquisition.

5.3 Mapping of dynamic environments

To the best of our knowledge, all existing mapping techniques assume that the environment contains no moving or dynamic objects. This is a strong assumption which cannot be justified outside the laboratory. Existing mapping techniques treat data arising from moving objects as noise and, hence, the map takes longer to build because more sensor measurements must be aggregated. Also, dynamic objects are not made available for path planning and so obstacle avoidance methods [22] must be used.

Detection of dynamic objects requires high bandwidth sensing and so vision must be used. Vision is a promising sensor which provides rich information and is ideally suited to dynamic environments. Our method, based on the work of Kagami *et al.* [23, 16] provide real-time depth and motion information and demonstrate the possibilities of this sensor in dynamic and unstructured environments.

6 Discussion

Our goals is to develop a complete mobile robot system which is capable of reliable operation 24 hours a day, 7 days a week for months on end without supervision. We believe that mobile robotics is today reaching the level where deployment in real-world situations seems possible. However, there are two issues remaining: 1) there has been little study of the long-term robustness of existing techniques and 2) there has been almost no consideration of the fact that the environment of a mobile robot is dynamic, not static as is commonly assumed.

Long-term experiments are crucial to test the reliability of existing methods and to demonstrate that mobile robotics has reached some level of maturity. Our robot will be responding to commands from web users over a period of many months and any failures that occur will allow us to develop a more reliable and robust system.

In addition to testing the reliability of a mobile robot system, this project allows us to address one of the significant remaining issues for localisation and mapping. Over longer periods of time it is impossible to consider the environment as static and localisation and mapping must both explicitly address the changes that occur over time. We will build on our earlier works in localisation and mapping and test the approaches outlined in Section 5 above. Another interesting aspect of this project is that the robot will have the ability to study its environment over time, hopefully learning more. If we consider humans, it is clear that a considerable amount of time is spent studying the nature of the world and learning the nature of a wide variety of objects in the environment. Perhaps this experiment will provide suitable data for robots to similarly learn about their environment.

Acknowledgements

The authors would like to acknowledge and thank Colin Thomsen for developing the teleoperation system and Seungjun Oh for developing the recharging system.

References

[1] R. G. Simmons, R. Goodwin, K. Z. Haigh, S. Koenig, J. O'Sullivan, and M. M. Veloso, "Xavier: Experience with a layered robot architecture." To appear in the ACM magazine Intelligence, edited by Manuela M. Veloso.

[2] W. Burgard, A. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun, "The interactive museum tour-guide robot," in *Proc. of the National Conference on Artificial Intelligence*, 1998.

[3] S. Thrun, M. Bennewitz, W. Burgard, A. B. Cremers, F. Dellaert, D. Fox, D. Haehnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz, "Minerva: A second generation mobile tour-guide robot," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 1999.

[4] W. Burgard, A. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun, "Experiences with an interactive museum tour-guide robot," *Artificial Intelligence*, vol. 114, no. 1-2, 2000.

[5] S. Yuta and Y. Hada, "Long term activity of the autonomous robot - proposal of a bench-mark problem for the autonomy," in *Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'98)*, pp. 1871–1878, Oct. 1998.

[6] Y. Hada and S. 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)*, pp. 297–302, Oct. 1999.

[7] S. Thrun, M. Beetz, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schultz, "Probabilistic algorithms and the interactive museum tour-guide robot minerva," *Robot Research*, vol. 19, pp. 972–999, 2000.

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

[9] D. Austin, "Dave's robotic operating system." http://forde.anu.edu.au/dros.

[10] D. J. Austin and P. Jensfelt, "Using multiple gaussian hypotheses to represent probability distributions for mobile robot localization," in *Proceedings of IEEE Intl. Conf. on Robotics and Automation*, 2000.

[11] P. Jensfelt, O. Wijk, D. J. Austin, and M. Andersson, "Experiments on augmenting condensation for mobile robot localization." Accepted for IEEE Intl. Conf. on Robotics and Automation, 2000.

[12] P. Jensfelt, D. J. Austin, O. Wijk, and M. Andersson, "Feature based condensation for mobile robot localization." Accepted for IEEE Intl. Conf. on Robotics and Automation, 2000.

[13] S. Thrun, D. Fox, and W. Burgard, "Probabilistic mapping of an environment by a mobile robot," in *Proc. of IEEE Intl. Conf. on Robotics and Automation*, pp. 1546–1551, May 1998.

[14] D. J. Austin and B. J. McCarragher, "Geometric constraint identification and mapping for mobile robots." Journal of Robotics and Autonomous Systems, to appear, 2001.

[15] D. J. Austin, "Non-monotonic geometric mapping for mobile robots," in *Proceedings IAS-00, Venice*, pp. 628–635, July 2000.

[16] S. Kagami, K. Okada, M. Inaba, and H. Inoue, "Design and implementation of onbody real-time depthmap generation system," in *Proc. IEEE Int. Conf. on Robotics and Automation*, (California, USA), IEEE Computer Press, 2000.

[17] S. Thompson and A. Zelinsky, "Local positioning in topological maps using automatically selected landmarks." Submitted to 2001 IEEE/RSJ Conference on Intelligent Robots and Systems (IROS'01).

[18] O. Sutherland, H. Truong, S. Rougeaux, and A. Zelinsky, "Advancing active visions systems by improved control and design," in *Proc. Intl. Symposium of Experimental Reobotics (ISER 2000)*, December 2000.

[19] J. J. Leonard, H. F. Durrant-Whyte, and I. J. Cox, "Dynamic map bulding for an autonomous mobile robot," *International Journal of Robotics Research*, vol. 11, no. 4, pp. 286–298, 1992.

[20] T. Dar, L. Joskowicz, and E. Rivlin, "Understanding mechanisms: from images to behaviours," in *Int. Conf. on Robotics and Automation*, pp. 3552–3558, 1998.

[21] M. Beetz, M. Giesenschlag, R. Englert, E. Gulch, and A. B. Cremers, "Semi-automatic acquisition of symbolically-annotated 3d-models of office environments," in *Int. Conf. on Robotics and Automation*, pp. 2280–2285, 1999.

[22] J. Borenstein and Y. Koren, "Histogramic in-motion mapping for mobile robot obstacle avoidance," *Journal of Robotics and Automation*, vol. 7, no. 4, pp. 535–539, 1991.

[23] S. Kagami, K. Okada, M. Inaba, and H. Inoue, "Realtime 3d flow generation system," in *Proc. IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems*, (Taipei, Taiwan), IEEE Computer Press, 1999.