A Sensory Input System for Autonomous Mobile Robots

J. Patrick Bixler
David P. Miller

TR 87-14

A SENSORY INPUT SYSTEM FOR AUTONOMOUS MOBILE ROBOTS

J. Patrick Bixler
Department of Computer Science
Virginia Tech
Blacksburg, VA 24061

David P. Miller
Department of Computer Science
Virginia Tech
Blacksburg, VA 24061

ABSTRACT

In order to accomplish navigation in an unfamiliar world a robot must be able to build and update its own world map continuously and in real time. This paper proposes a sensory input system based on the fusion of simple low-resolution vision with directed high-resolution sonar. The basic idea is to use a simple vision system to locate the direction in which an obstacle lies, and then use an ultra-sonic rangefinder to determine the depth of the object and to gain clues about its shape. By fusing two simple systems we attempt to exploit the strengths of each while maintaining an acceptable computational cost. An idealized example is given and we discuss the possibilities and some of the problems.

INTRODUCTION

Given an accurate map of the world, an autonomous mobile robot can plan a navigation route between any two points in its domain. One method for accomplishing this task is to use an incremental route-planning strategy [21]. Using the most current version of its map, the robot computes the next few waypoints to traverse. Then, as it moves, the robot senses the surroundings and verifies or updates its world map. Repeating this process, the robot eventually achieves its destination. Although the question of which is the best planning paradigm remains very much open, all such systems require substantial knowledge about the surrounding world. This research focuses on the problem of creating a reasonably accurate wire-frame map of an unfamiliar world.

Because the robot is assumed to be in unfamiliar surroundings, the navigation system must be able to obtain information about the physical world from the robot itself in real time. This implies some kind of sensory data processing system, capable of both collecting and interpreting information, be mounted on the robot platform. A number of different sensory systems could be used including touch sensors, rangefinders, and vision systems [17,18]. Tactile sensors are apparently not useful in this situation because of their short range and limited targetability. Such sensors necessarily interfere with the surrounding world and would also severely limit the speed of the robot. Ultrasonic rangefinders also have limited targetability, but have a significant range and do not interfere with surrounding objects. Although substantial time is required to re-target these sensors [15], they represent an excellent method for measuring the distance to an object. Computer vision has been studied extensively and is now sufficiently developed to be of some practical use. Low level vision operations such as image enhancement, feature detection, and limited pattern recognition can be implemented in hardware making real time systems possible. While automatic interpretation of images falls far short of what humans are capable of, at least the mechanics of visual systems are reasonably well understood [11].

Whatever sensory system is used must be capable of determining, and then processing, that information which is relevant to the current goals of the robot. For example, recognizing which portion of the image represents objects that are sufficiently in the distance or not in the path of travel can allow time to be spent analyzing more immediate obstacles. The system must also be able to differentiate quickly between a shadow, which can be passed through and therefore ignored, and actual objects which cannot. Obviously, because the robot is moving, the time to detect obstacles must be short enough to allow the robot to change course. The system should be able to provide a boundary within which the object is absolutely certain to lie and a probability map, such as in [19], on the chances of actually encountering an obstacle.

There are also several other issues common to sensing and object recognition that are not necessarily of immediate concern in this application. For example, objects do not have to be described in great detail. In most cases, only a rough estimate of shape, or the projection onto the plane of the floor, and a fairly accurate estimate of location is needed. Secondly, because the robot is moving, the system need not be restricted to a single view. Rather, it can give a partial or preliminary description first and then complete and verify that description based on subsequent views [4]. It might also be possible to process some information off line. That is, whenever the robot comes to a situation that is relatively simple and unchanging (a long straight hallway, for example) some effort could be directed toward refining its total map based on previous stored views of other parts of the world.

Sensory Input System for Mobile Robots

The robot platform used for this research is Real World Interface's *Vectrobot* robot base. The robot is modified with a two degree of freedom pan and tilt platform upon which is mounted a high-resolution sonar, see Figure 1. A low resolution (128 by 128 pixels) RAM camera is also mounted on top of the robot on the single degree of freedom rotary platform. The camera will therefore be able to pan the horizon in a complete circle, and the sonar can be pointed at any small area within the camera's field of view.

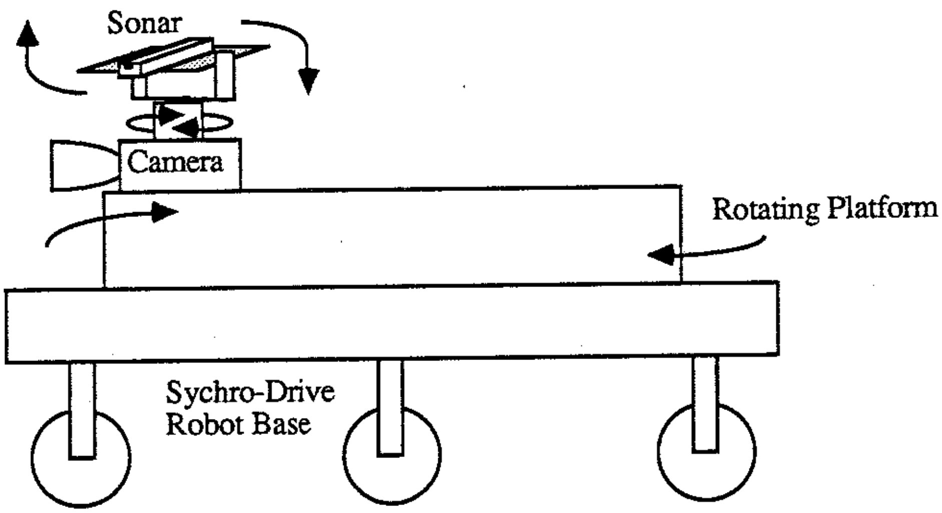

Figure 1. Sensor system mounted on robot platform.

THE VISION SYSTEM

A total reliance on vision for detecting obstacles can suffer from high computational expense and difficulties interpreting the scene. A single view of a scene rarely leads to a single consistent interpretation [6,10,23]. For example, when one object partially obscures another, there are certain edges for which it is impossible to tell whether they belong to one or both of the objects. Basically, some cue about depth is needed. Such a cue could be obtained from the use of two cameras making up a stereo vision system similar to the human visual system [12]. If one assumes that the robot is always capable of moving at least a small amount at any time, a single camera can supply the two required views [9,20]. In addition to having to solve the correspondence problem, however, stereo vision systems typically require substantial computation to locate objects accurately.

The approach taken here is to make the vision system be responsible only for giving a fairly accurate estimate of the direction in which the object lies. Once the object has been detected, the rangefinder can be directed to a small sample of points on or near the object. The depth information thus obtained can then be used to compute or confirm the exact location. As an example, suppose the robot is to cross the floor of a room on which a collection of obstacles has been randomly placed. Since the robot travels on the floor, the logical thing to do is to keep an eye on the floor and map the boundaries of clear floor space. This implies that the vision system be capable of quickly locating the edges that represent the intersection of each object with the floor. Once the robot knows the exact location of these edges it can essentially build polygonal columns around the obstacles and subsequently plan its route.

To this end the vision system will be required to produce an edge image of the current scene. There are a substantial number of edge detection algorithms described in the literature with many of them surveyed in [1] and [7]. Many are tuned to be most effective on a specific type of scene and it may well be that the sensory system maintain a catalog of techniques and attempt to use the most appropriate one for the current environment. In most cases one of the simpler gradient schemes, such as the cross difference or a Sobel operator will be the best choice.

Because of real time constraints and the fact that we need only find edges, a low-resolution (128 by 128 pixels) camera will be used. If we are able to localize the area of interest, however, a higher resolution image could be used. In any case, once the edge pixels have been identified they must be linked together to form actual edge segments. Here again there is a choice of techniques, such as a local similarity test, a global Hough transform, or perhaps a fast tracking algorithm. The edges that represent the intersection of the floor with an obstable are the most important ones and must be found first. We will also investigate the feasibility of simply growing the region of the floor in order to detect these edges quickly.

THE SONAR SYSTEM

In an ideal, static situation the robot should be able to make up a depth map of the world by carefully scanning in all directions with some sort of rangefinder [8]. Unfortunately, this approach is impractical or inefficient for several reasons. First, such scans take time. The rangefinder must be scanned in a wide arc centered about the current heading. If a complete map is to be made, then several scans at varying elevations are necessary. In addition to the time required simply to move the sensor, taking the readings themselves requires noticeable time. The problem is severe when one considers the fact that the robot is supposed to be moving, and is compounded if the environment is dynamic. Finally, both ultrasonic and laser rangefinders suffer problems of range limitations and reflections; relying on range information exclusivley is seldom sufficiently accurate.

Once an edge has been detected by the vision system, however, the sonar can be used to obtain depth information at a small sample of points along that edge. The sonar that we use consists of a narrow beam, high frequency transducer/receiver. The high frequency (compared with the 25kHz Polaroid sonar rangefinders often used in robotics research) has several advantages: greater accuracy, narrow beam (no side lobes), lower incident angle reflection, and a slightly reduced range that will tend to reduce echos. The sonar is mounted as close to the camera as possible in order to minimize the error in targeting. Once the x- and y-coordinates of a point in the image plane are known, the sonar can be fired in that direction. The resulting distance measurements then determine the location of the point in world coordinates.

To increase the confidence that an edge in the image actually represents a straight edge in the scene, several points along that edge will be measured. If the resulting world points are collinear, then it would be reasonable to assume that the edge was formed by the intesection of two planes. If not, the edge can be approximated by a sequence of vectors to some abritrarily prescribed tolerance. For edges that are known to be on the floor, the default assumption will be that the intersecting plane is a vertical wall. When several edges intersect in the image, test points along those edges will be further checked to see if they are coplanar. If so, the plane segment bounded by the edges under consideration will be added to the world map. Although this will necessarily lead to some errors, it is a reasonable first approximation. Subsequent views by the sensory system can be used to produce increasingly accurate mappings.

A variant of this approach to mapping three-dimensional space is to use the above technique to produce the first approximation. Then determine the location of some point in the interior of that polygon and triangulate the polygon based on that point. Successive refinements would eventually lead to a fairly accurate polyhedral representation of the robot's domain.

The primary role of the sonar is to determine the distance to a given point in the robot's world. The return signal, however, contains information that may be of value. If the sonar's return signal is sampled at a high enough rate and the curve is normalized, then some comparison and interpretation may be possible. As an example, consider the sample two-dimensional obstacles and their corresponding return signals shown in Figure 2. The smooth continuous nature of the first curve indicates that the obstacle is probably also smooth and possibly flat. The other two curves have corners in them corresponding in some way to the corners in the obstacles. Note that the second part of curve (c) is lower than the first part. This corresponds to the protruding corner in the obstacle. In curve (b) the second part is higher than the first part indicating that the obstacle is probably concave. Although it is not clear exactly how to interpret the curves, there is at least the potential for distinguishing some obstacles based on the shapes of their curves [16].

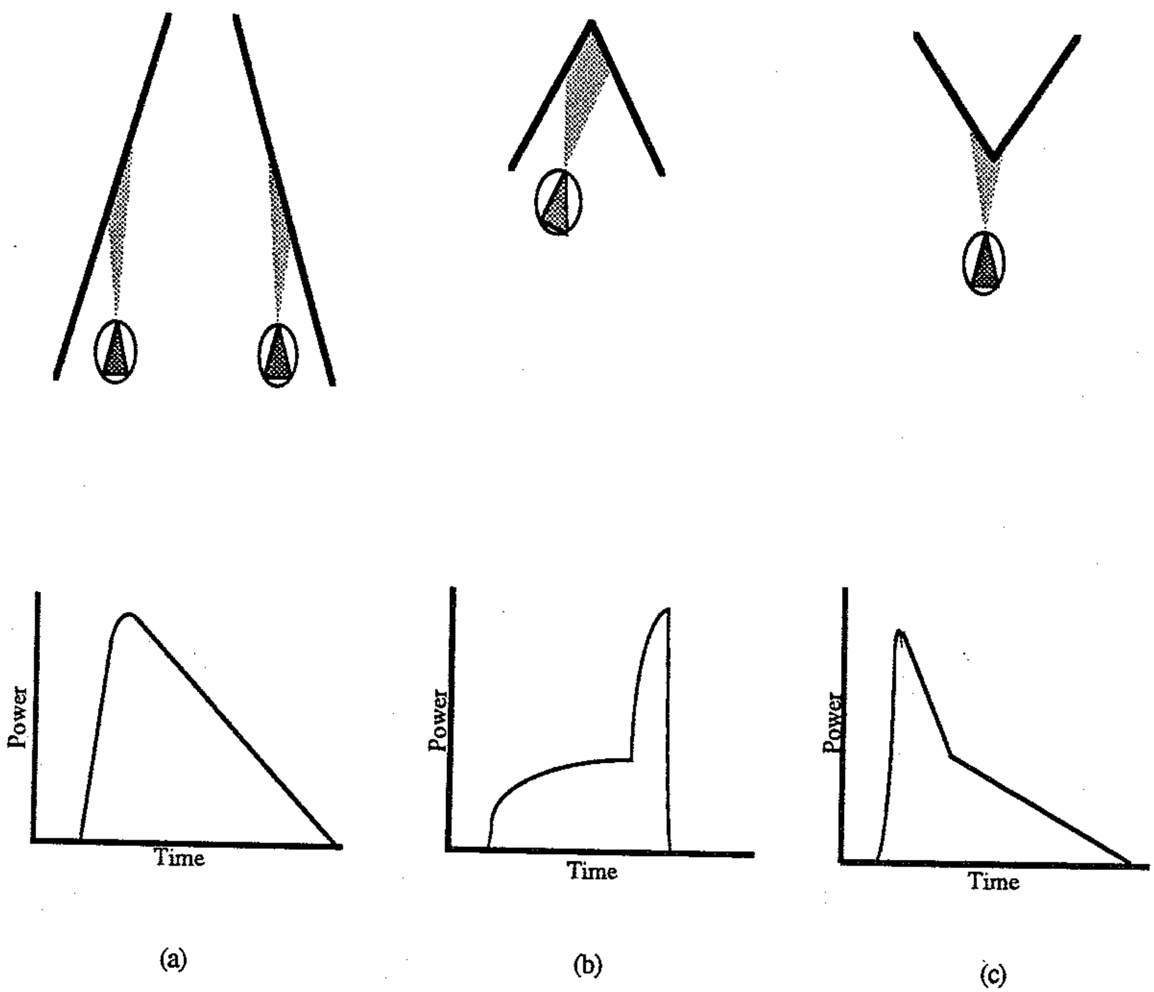

Figure 2. Sample 2-dimensional obstacles and return signals

An application of this is to distinguish edges caused by shadows from those representing actual objects. Relying solely on vision it is difficult, if possible at all, to make such distinctions. However, looking at the return signals improves the likelyhood of correctly disambiguating the two cases. In the case of the shadow edge, the return signal should be smooth, whereas an actual surface edge should return a signal with some shape in it, such as a bump or a step.

Once the edges have been located in world coordinates, a high level representation of the edges can be formed and from this the beginnings of a 3-dimensional wire-frame grid map of the world can be constructed [2]. A grid map will be used rather than a more detailed object map such as the Mercator [4,5] or Spam systems [13]. While a Mercator map provides more information, much of its strength derives from the sensory system providing object identification information, in addition to position information. The Mercator system uses information about an object's color, texture, and other physical properties in order to make inferences about hidden and partially obscured obstacles. While such information is unquestionably useful, it is at this point too expensive to gather and compute for guiding a real-time robot. Also, there currently is no reliable theory for representing an object map where objects can move, appear, and disappear through time. The grid map used here will be updated with every view. Because the approximate position of the robot will be known through dead reckoning, the search space for assimilating new information will be quite small. An object will be represented only as its approximate shape and last known position.

A SAMPLE SCENE

Figure 3 shows a simple scene consisting of a main hallway with several side hallways and a cube-shaped obstacle on the floor. The obstacle casts a shadow due to a light source in the hallway to the right. For the sake of illustration, we assume that the vision system produces an edge image that is a reasonable approximation of Figure 3. The robot is instructed to navigate to a position somewhere down the second hallway to the right.

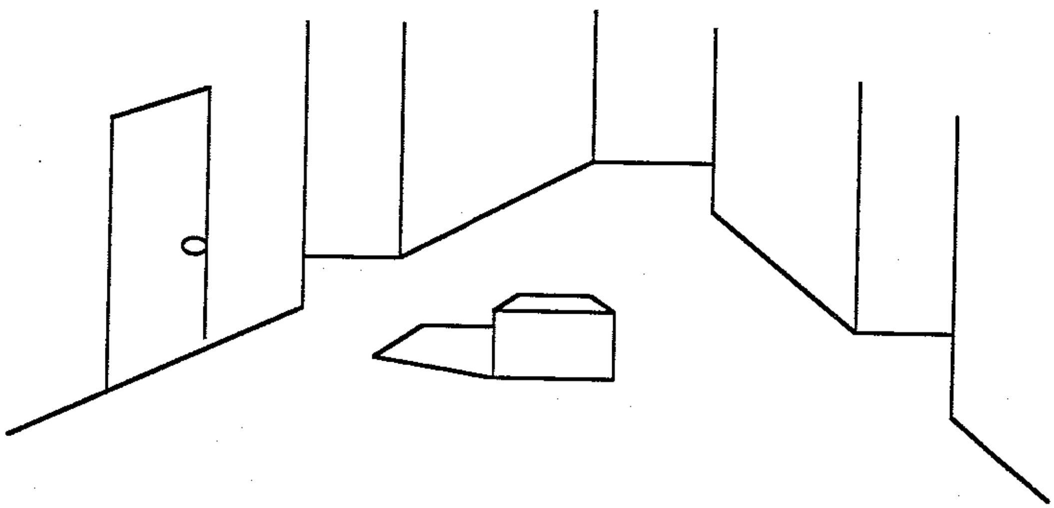

Figure 3. A sample hallway scene with obstacle.

The edges that lie in the floor and are in the immediate vicinity of the robot are measured first and the default assumption that they represent vertical walls is verified. Next, the edges that are associated with the shadow are considered and are all found to be flush with the floor. This indicates that they do not represent part of the obstacle and they can essentially be ignored. The edges of the door frame to the left are then evaluated and found to be in the plane of the left wall, also indicating no obstacle. The robot now knows what clearance is available to either side of the obstacle and can plan its route. As it proceeds around the obstacle it continues to refine its map of the obstacle and verifies the exact location of the side hallways.

CONCLUSIONS

Several similar projects have previously been reported in the literature. Some have relied primarily on vision-based sensory systems [18,24], while others have tried to use only rangefinders [3,14,19]. Our approach differs in that it has each sensor perform only a very simple task, and attempts to produce the necessary information through the combination of the two.

To be sure, there are still many difficulties that must be overcome. The quality of the edge detection efforts is a primary concern. Detecting edges in a restricted domain with good lighting and where all obstacles are made up of simple polyhedra may be manageable, but it may be quite difficult in a more natural setting with irregular, curved shapes and multiple shadows. Shadows may also conceal an obstacle altogether: if the vision system does not detect an edge, the sonar will not know to look for an object.

Even though the sonar will be required to scan relatively few points, those points may not be near each other. The time needed simply to retarget the sonar may be significant, and the accuracy with which the sonar can be aimed must be very carefully controlled. A certain minimum error due to the fact that the camera and the transducer are offset slightly must be able to be tolerated.

In order to get any useful information from the curve of the return signal, the rate at which the signal is sampled must be sufficiently high. In some cases the signal will not be spread out enough to return much more that a short blip. Even when the curves are well formed and reasonably long they are by no means unique. It is easy to exhibit examples of two distinct obstacles that produce the same reflection. This ambiguity is further aggravated by secondary reflections and by the uncertainty about the reflectance properties of the obstacles' surface.

All computations must be performed quickly enough to allow the robot to update its world map at a reasonable rate, and of course, all computations must be adjusted to allow for the motion of the robot itself. One final difficulty is the correlation between what the robot thinks should be there and what it actually sees. When the robot takes a subsequent view and finds an edge in a different than expected location, it must be able to determine if it has found a new edge or if it should correct the location of a previous edge. The problem of matching polyhedra from different views is extremely difficult even when the parameters are well-known [4].

We have performed a number of simulations of the sonar on 2-dimensional obstacles and are in the process of developing a 3-dimensional version of the simulator. The actual sonar device has been designed and is in the final stages of being built. Concurrent experiments are planned involving both the device and the simulator, and preliminary work on testing various edge detection algorithms on low-resolution images is also underway.

REFERENCES

1. Abdou, I.E. "Quantatative design and evaluation of enhancement thresholding edge detectors." *IEEE Proceedings*, 67 pp 753-763, (1979).

2. Bixler, J.P. and J.P. Sanford. "Finding straight lines and curves in engineering line drawings." *Technical Report TR-87-12* , Virginia Tech Department of Computer Science, (1987).

3. Crowley, J.L. "Dynamic world modeling for an intelligent mobile robot using rotating ultra-sonic ranging device." *Proceedings of the 1985 International Conference on Robotics and Automation,* pp 128-135, IEEE, St. Louis, MO. (March 1985).

4. Davis, E. *Representing and acquiring geographic knowledge.* Morgan Kaufman, (1986).

5. Davis, E. "The Mercator representation of spatial knowledge." *Proc. IJCAI*, 8, pp 295-301, IJCAI (1983).

6. Guzman, A. "The decomposition of a visual scene into three dimensional bodies." *Proc. FJCC,* pp 291-301, Fall Joint Computer Conference, (1968).

7. Hildreth, E.C. "Edge detection." *Technical Report 858*, MIT AI Laboratory, (1985).

8. Letovsky, S. "Interpreting range data for a mobile robot." *Proceedings of the Fifth National Conference of the Canadian Society for Computational Studies of Intelligence*, pp 70-72, CSCSI, London, Ontario (May 1984).

9. Longuet-Higgins, H.C. and K. Prazdny. "The interpretation of a moving retinal image." *Proceedings of the Royal Sociecty of London*, pp 385-397, (1985).

10. Mackworth, A.K. "Interpreting pictures of polyhedral scenes." *Artificial Intelligence*, 4, pp 121-137, (1973).

11. Marr, D. Vision, W.H. Freeman, (1982).

12. Marr, D. and T. Poggio. "A computational theory of human stereo vision." *Proceedings of the Royal Society*, Series B, 204, pp 301-328, (1979).

13. McDermott, D.V. and E. Davis. "Planning routes through uncertain territory." *Artificial Intelligence*, 22, pp 107-156, (1984).

14. Miller, D.P. "A spatial representation system for mobile robots." *Proceedings of the 1985 International Conference on Robotics and Automation*, pp 122-127, IEEE, St. Louis, MO. (March 1985).

15. Miller, D.P. "Scheduling robot sensors for multisensory tasks." *Proceedings of the 1986 Robots West Conference*, SME, Long Beach, CA. (September 1986).

16. Miller, D.P. and J.P. Bixler. "A taxonomy of obstacles as seen by an ultrasonic rangefinder." *Proceedings of the 1987 IEEE Conference on Systems, Man, and Cybernetics,* IEEE, Alexandria, VA. (October 1987).

17. Moravec, H.P. "Obstacle avoidance and navigation in the real world by a seeing robot rover." PhD Thesis, Stanford University, (September 1980).

18. Moravec, H.P. "Visual mapping by a robot rover." *Proceedings of the International Joint Conference on Artificial Intelligence*, pp 598-600, IJCAI, (1979).

19. Moravec, H.P. and A. Elfes. "High resolution maps from wide angle sonar." *Proceedings of the 1985 International Conference on Robotics and Automation*, pp 116-121, IEEE, St. Louis, MO. (March 1985).

20. Roach, J.W. and J.K. Aggarwal. "Determining the movement of objects from a sequence of images." *IEEE Trans. on PAMI*, 6, pp 554-562, (1980).

21. Slack, M.G. and D.P. Miller. "Route planning in a four dimensional environment." *Proceedings of the 1987 Workshop on Tele-Robotics*, JPL, Pasadena, CA. (January 1987).

22. Ullman, S. *The interpretation of visual motion*, MIT Press, (1979).

23. Waltz, D. "Understanding line drawings of scenes with shadows." in *The Psychology of Computer Vision*, Patrick Winston, ed., McGraw-Hill, (1975).

24. Waxman, A.M., J. LeMoigne and B. Srinivasan. "Visual navigation of roadways." *Proceedings of the 1985 International Conference on Robotics and Automation*, pp 116-121, IEEE, St. Louis, MO. (March 1985).