The Wayfarer modular navigation payload for intelligent robot infrastructure

Brian Yamauchi*
iRobot Research Group, iRobot Corporation, 63 South Avenue, Burlington, MA 01803-4903

ABSTRACT

We are currently developing autonomous urban navigation capabilities for the iRobot PackBot. The TARDEC-funded Wayfarer Project is developing a modular navigation payload that incorporates LIDAR, vision, FLIR, and inertial navigation sensors. This payload can be attached to any PackBot and will provide the robot with the capability to perform fully-autonomous urban reconnaissance missions. These capabilities will enable the PackBot Wayfarer to scout unknown territory and return maps along with video and FLIR image sequences. The Wayfarer navigation payload includes software components for obstacle avoidance, perimeter and street following, and map-building. The obstacle avoidance system enables the PackBot to avoid collisions with a wide range of obstacles in both outdoor and indoor environments. This system combines 360-degree planar LIDAR range data with 3D obstacle detection using stereo vision using a Scaled Vector Field Histogram algorithm. We use a real-time Hough transform to detect linear features in the range data that correspond to building walls and street orientations. We use the LIDAR range data to build an occupancy grid map of the robot's surroundings in real-time. Data from the range sensors, obstacle avoidance, and the Hough transform are transmitted via UDP over wireless Ethernet to an OpenGL-based OCU that displays this information graphically and in real-time.

Keywords: Robotics, navigation, payload, urban warfare, reconnaissance, autonomy

1. INTRODUCTION

iRobot Corporation is a world leader in developing mobile robots for combat operations. Our man-portable PackBot Scout robots[5] have been used to explore Taliban caves in Afghanistan and insurgent strongholds in Najaf. Our PackBot Explosive Ordinance Disposal (EOD) robots are being used every day to help U.S. soldiers disarm improvised explosive devices (IEDs) in Baghdad and other Iraqi cities.

The PackBot Scout and PackBot EOD represent the first generation of teleoperated robots that are providing real value to soldiers on the battlefield. At iRobot Research, we are developing the next generation of autonomous battlefield robots. The second-generation PackBot Wayfarer will be capable of performing fully autonomous reconnaissance missions in urban terrain.

For the Wayfarer Project, funded by the U.S. Army Tank-automotive and Armaments Research, Development, and Engineering Center (TARDEC), we are developing a modular navigation payload that will provide urban navigation capabilities for the PackBot or any other robot using the PackBot payload interface. The Wayfarer Navigation Payload will be a key component of the Aware infrastructure that we are developing for autonomous, intelligent robots.

Teleoperated robots like the PackBot have the potential to reduce the risk to Army warfighters in urban environments, but they are limited by both radio range and the need for a full-time operator. In urban environments, buildings and radio interference may substantially reduce the operational range of teleoperated robots. In addition, the need for a robot operator to devote full attention to the control of the robot (and possibly another soldier to cover the operator) increases the manpower demands associated with robotic reconnaissance.

The Wayfarer Project is an applied research effort that is tightly focused on developing robust techniques for autonomous reconnaissance in urban environments. Instead of attempting to solve the general navigation problem, we

* yamauchi@irobot.com

are developing specific reconnaissance behaviors that will be useful to Army warfighters in the near term. These behaviors include:

- **Perimeter Reconnaissance:** Move around the perimeter of a building complex and return video/FLIR images and map data from all sides of the complex.
- **Route Reconnaissance:** Move forward along a road for a specified distance and return to the starting point with video and FLIR image data and well as a map of the terrain.
- **Street Reconnaissance:** Follow a route specified using GPS coordinates of intersections and bearings of selected streets, and return with video/FLIR images and map data.

Our work is unique in taking a narrow, but deep, focus on the urban reconnaissance task. Instead of developing research-oriented navigation systems that have general capabilities, but limited reliability, we are developing a prototype UGV that demonstrates robust urban reconnaissance capabilities in realistic environments. Our plan is to complete the PackBot Wayfarer prototype by September 2005, and to demonstrate this prototype's reconnaissance capabilities in an urban test environment. Following successful completion of this project, our goal is to transition this technology into field-deployable UGVs as part of the iRobot PackBot product line. Our objective is to provide production UGVs with this technology to front-line Army warfighters in the 3-to-5 year time frame.

2. WAYFARER HARDWARE

The Wayfarer Payload includes sensor and processing hardware to support autonomous navigation on the PackBot. The Wayfarer sensors include a SICK LD OEM 360-degree planar LIDAR, a Point Grey Bumblebee stereo vision system, a Crossbow six-axis fiber optic Inertial Measurement Unit (IMU), and an Indigo Omega FLIR camera. The PackBot's organic 700 MHz Pentium III CPU communicates with the SICK LIDAR and Crossbow IMU over RS-232 serial interface ports.

The Wayfarer Payload also includes a Plug-N-Run 700 MHz Pentium III processor that is dedicated to stereo vision processing. The Plug-N-Run captures images from the Bumblebee camera over a Firewire interface and communicates with the PackBot CPU via an Ethernet interface.

Figure 1: PackBot with Wayfarer navigation payload

Figure 1 shows a PackBot with the Wayfarer Payload. The Bumblebee stereo vision cameras are mounted at the front of the robot. The SICK LIDAR is mounted behind the Bumblebee cameras so that the laser beam plane passes directly over the cameras. Behind the SICK LIDAR is the Crossbow IMU. At the rear of the robot is the Plug-N-Run vision processor stack.

3. SCALED VECTOR FIELD HISTOGRAM (SVFH)

To enable Wayfarer to avoid obstacles in cluttered environments, we have developed a new obstacle avoidance algorithm that uses a Scaled Vector Field Histogram (SVFH). This algorithm is an extension of the Vector Field Histogram (VFH) techniques developed by Borenstein and Koren[2].

In the standard VFH technique, an occupancy grid is created, and a polar histogram of the obstacle locations is created, relative to the robot's current location. Individual occupancy cells are mapped to a corresponding wedge or "sector" of space in the polar histogram. Each sector corresponds to a histogram bin, and the value for each bin is equally to the sum of all the occupancy grid cell values within the sector.

The polar histogram bin values mapped to their bearings relative to the robot's heading. A bin value threshold is used to determine whether the bearing corresponding to a specific bin is open or blocked. If the bin value is under this threshold, the corresponding direction is considered clear. If the bin value meets or exceeds this threshold, the corresponding direction is considered blocked. Once the VFH has determined which headings are open and which are blocked, the robot then picks the heading closest to its desired heading toward its target/waypoint and moves in that direction.

We developed an extension of the VFH algorithm that we call the Scaled Vector Field Histogram (SVFH). The SVFH is similar to the VFH, except that the occupancy values are spread across neighboring bins. An obstacle that may be easily avoided at long range may require more drastic avoidance maneuvers at short range, and this is reflected in the bin values of the SVFH.

The extent of the spread is given by:

$$\theta = k / r$$

where k is the spread factor (0.4 in the current SVFH), r is the range reading, and θ is the spread angle in radians. For example: if k = 0.4 and r = 1 meter, then the spread angle is 0.4 radians (23 degrees). So a range reading at 1 meter for a bearing of 45 degrees will increment the bins from 45 – 23 = 22 degrees to 45 + 23 = 68 degrees. For a range reading of 0.5 degrees, the spread angle would be 0.8 radians (46 degrees), so a range reading at 0.5 meters will increment the bins from 45 – 46 = -1 degrees to 45 + 46 = 91 degrees. In this way, the SVFH causes the robot to turn more sharply to avoid nearby obstacles than to avoid more distant obstacles.

We have fully implemented the SVFH algorithm on the PackBot Wayfarers using 360-degree range data from the SICK LD OEM laser rangefinder. The SICK LD OEM provides a 360-degree range scan with 2 degree resolution at 5 Hz. The range values from each scan are used to compute a new SVFH. The SICK LD OEM provides range data out to 12 meters, but currently only range values out to 2 meters are used to compute the SVFH. This limit will be extended for use at higher speeds in outdoor environments.

Figure 2 shows the SVFH bins when the robot arrives at a hallway intersection. This image is taken directly from the real-time OCU display. Bright vectors representing SVFH bin values are superimposed over the background laser range data. The length of each vector is proportional to the value of the bin associated with the corresponding direction. Long vectors correspond to a large number of nearby range readings within the bin sector. Short vectors correspond to a small number of range readings near the limit of the range window (2 meters). If no vector is present in a given direction, this means that no obstacles are within the range window in that direction.

Figure 3 shows the corresponding clear directions at the intersection. Bright vectors point toward clear directions. If no vector is present in a given direction, this means that this direction is blocked. The SVFH detects all four of the open passages meeting at the intersection. Wider passages allow a wider range of orientations for the robot, while narrower passages allow a more limited range of orientations.

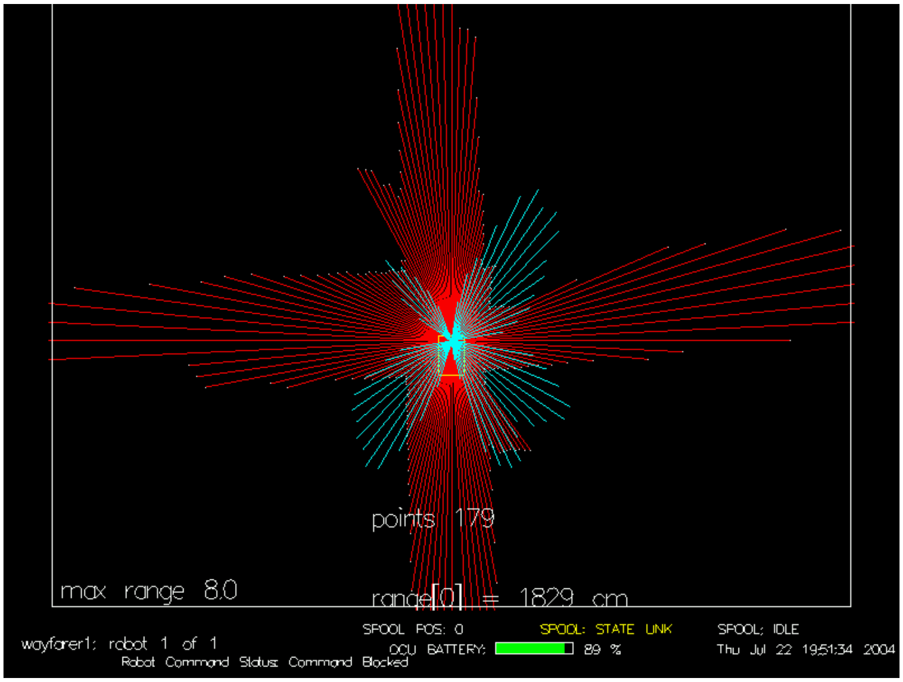

Figure 2: SVFH bins at intersection

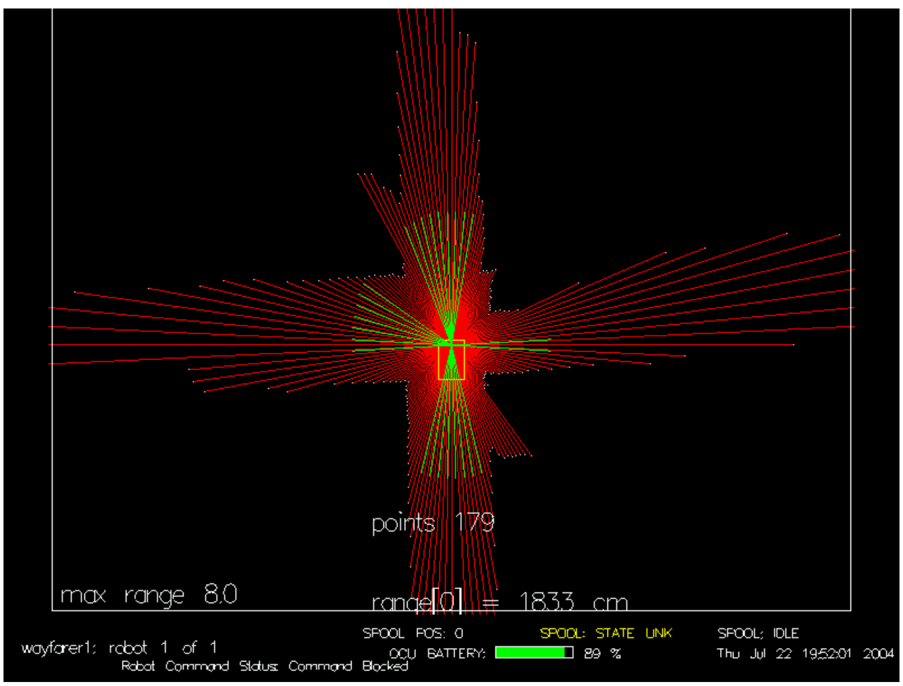

Figure 3: SVFH clear directions at intersection

4. OUTDOOR OBSTACLE AVOIDANCE

The Wayfarer obstacle avoidance system combines planar range data from the SICK LD OEM with 3D range data from the Point Grey Bumblebee vision system using the SVFH algorithm. We have tested this system both indoors and outdoors in a wide range of urban and natural settings.

The avoidance system was capable of detecting walls (indoor and outdoor), doors, furniture, cars, trucks, trees, bushes, rocks, stairs, metal railings, and chain-link fences. Both the LIDAR and the stereo vision system are positioned so they can detect obstacles that the PackBot is not able of climbing. Lower obstacles such as curbs, which the PackBot can climb, are not included in the obstacle avoidance map. This allows the avoidance system to lead the robot over climbable obstacles while avoiding unclimbable obstacles at the same time.

The only unclimbable obstacles that the system failed to detect were a glass door (transparent to both vision and laser) and a telephone pole guide wire mounted at approximately 45 degrees to vertical (narrow cross-section in planar LIDAR, no vertical edges for stereo matching). A sonar sensor would detect both of these types of obstacles (glass and narrow metal wires). While these obstacles are sufficiently rare that we have no current plans to add sonar to Wayfarer, the combination of LIDAR, stereo vision, and sonar would provide the capability to detect virtually all of the obstacles a UGV might encounter in an urban environment.

Figure 4: PackBot Wayfarer steers down 180-degree handicap ramp

Figure 4 shows a PackBot Wayfarer steering down a handicap ramp. This ramp involves a 180-degree change of orientation with two sharp 90-degree turns. The ramp is bounded on all sides by a metal railing that is detected by both the LIDAR and the stereo vision system. The PackBot Wayfarer is able to navigate down and up this ramp completely autonomously, with no operator intervention.

The combination of LIDAR and stereo vision allows the PackBot Wayfarer to avoid difficult obstacles such as parked vehicles. The LIDAR can clearly see the vehicle wheels, but the chassis is above the plane of the laser. Fortunately, the stereo vision can see the vehicle body, and the obstacle avoidance system can use the fused data to reliably avoid collisions with parked vehicles. We experimented with a variety of vehicles (small cars, SUVs, pickup trucks) and vehicle colors (black, white, silver, red) and the PackBot Wayfarer was able to avoid all of them. It is possible that larger vehicles with much higher ground clearance might not be detected, but it is likely that the PackBot would be able to drive under these vehicles.

Figure 5: PackBot Wayfarer steers through narrow gap between bushes and car

Figure 5 shows the PackBot Wayfarer steering through a narrow gap between two complex obstacles – a row of bushes on the left and a black car on the right. The PackBot is able to climb the curb at the end of the gap, while still using its obstacle avoidance to avoid contact with the front of the parked car. As it climbs the curb, the LIDAR and the vision system are angled upward, but they continue to detect obstacles to the side of the robot (such as the car). The ground behind the robot is detected as an obstacle, but this only prevents the robot from turning around.

Figure 6: PackBot Wayfarer avoiding urban obstacles

Figure 6 shows the PackBot Wayfarer avoiding a variety of obstacles commonly found in urban environments. These obstacles include building walls, a metal ladder, a wooden staircase, a large plastic traffic marker, and a stack of concrete barriers. These obstacles have a wide range of sizes, shapes, materials, and colors. The robot is able to reliably detect and avoid all of these obstacles, while not perceiving other environment details (such as the markings on the ground) as false obstacles.

Figure 7: PackBot Wayfarer avoiding trees

Figure 7 shows the PackBot Wayfarer avoiding trees in a lightly wooded area. While the goal of the Wayfarer Project is to develop a navigation system for urban environments, our experiments show that the Wayfarer obstacle avoidance system will likely be useful in wilderness environments as well.

4. PERIMETER FOLLOWING

We use a real-time Hough transform to find the lines in the range data that correspond to building walls. The Hough transform[1] is a computer vision technique that works by transforming image point coordinates into votes in the parameter space of possible lines. Each point corresponds to a vote for all of the lines that pass through that point. By finding the strongest points in the parameter space, the Hough transform can determine the parameterized equations for the strongest lines in the image. Previously, Schiele and Crowley have applied the Hough transform to finding indoor walls in occupancy grid maps[4]. However, our research effort is unique in applying this technique to outdoor urban navigation.

We have interfaced the Hough transform line detector with a perimeter following behavior. The ***follow-perimeter*** behavior attempts to steer the robot so that it is parallel to the strongest line detected by the Hough transform. To prevent the robot from oscillating between two lines that are approximately the same strength, an accumulator array is used to integrate the strength of line orientations over time. For computational efficiency, all lines with the same orientation vote for the same orientation, regardless of the range from each line to the robot. Orientations are grouped into 5 degree bins for a total of 72 bins.

The value of accumulator bin a_i at time t is given by:

$$a_{i,t} = (1-\lambda)\, a_{i,t-1} + \lambda \sum_{\forall j:\, i\beta < \theta(H_j) < (i+1)\beta} v(H_j)$$

where $a_{i,t-1}$ is the accumulator bin value at the previous timestep, λ is the decay rate (between 0 and 1), H is the set of lines detected by the Hough transform, H_j is the jth line from this set, $v(Hj)$ is the number of points voting for this line, $\theta(Hj)$ is the orientation of the line, and β is the bin size.

Note that all orientations are in **world coordinates** not robot-relative coordinates. This is important for correct vote accumulation when the robot is turning. The accumulator is providing a running (exponential decay) tally of votes for particular **global orientations** in the world coordinate frame. Some error will accumulate due to odometry slip, but as long as the decay rate is set to a sufficiently high value, the contributions from older line hypotheses will decay rapidly enough so that “blurring” due to odometry drift will be minimized.

The *follow-perimeter* behavior outputs a desired absolute heading in world coordinates. This desired heading is passed to the SVFH obstacle avoidance system. The SVFH then selects the obstacle-free heading that is closest to the desired heading output by *follow-perimeter*. This allows the robot to reactively steer around obstacles that are located next to walls and then resume wall-following automatically.

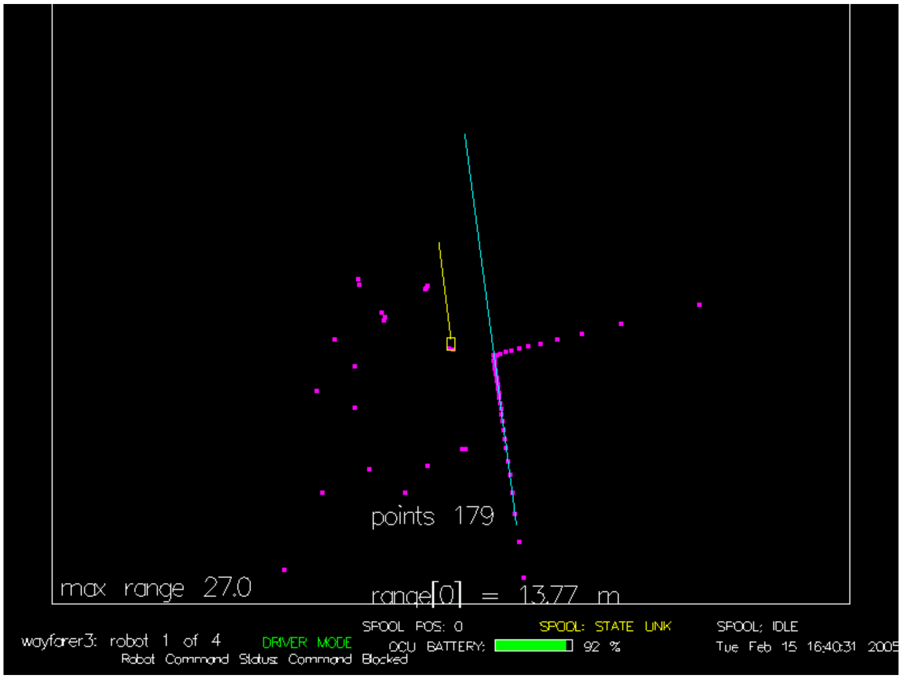

Figure 8: PackBot Wayfarer at building corner

We have begun tests of the perimeter following system in the urban environment outside the iRobot Corporation headquarters in Burlington, MA. Initial experiments show that the Hough transform works well to detect the exterior walls of the building. Figure 8 shows the laser scan from the PackBot Wayfarer as it passes a building corner. The dots indicate individual laser range readings. The left line indicates the position and orientation of the strongest line detected by the Hough transform, which corresponds to the building wall. The right line represents the desired accumulator heading of the perimeter following behavior, which is parallel to the wall. As the robot passes the corner, this heading rotates to match the adjacent (perpendicular) wall. The scattered range points to the left of the robot are range returns from trees.

The Hough transform works well to detect the building orientation even when the beam does not directly intersect the wall. Figure 9 shows the PackBot Wayfarer following the raised landscape behind iRobot HQ. In this example, the laser plane intersects the grass and dirt of the landscape instead of the building wall, so the returned points are not perfectly aligned. Despite this, the Hough transform is able to accurately determine the heading of the building perimeter.

Even where there are multiple wall segments, the Hough transform works effectively to find the line with the greatest number of points (Figure 10), and the robot proceeds to follow this line.

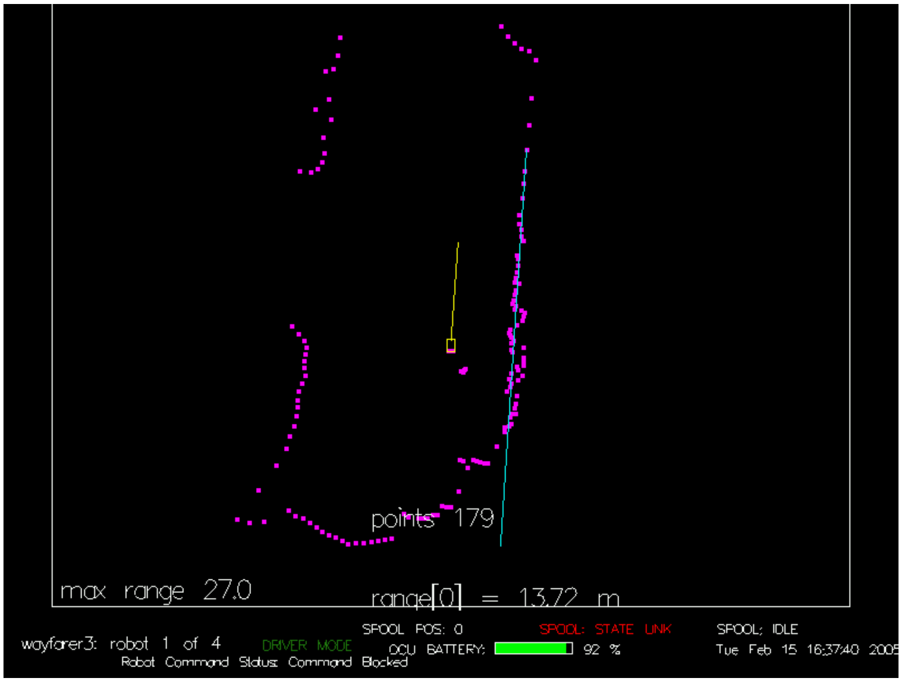

Figure 9: PackBot Wayfarer following landscape behind building

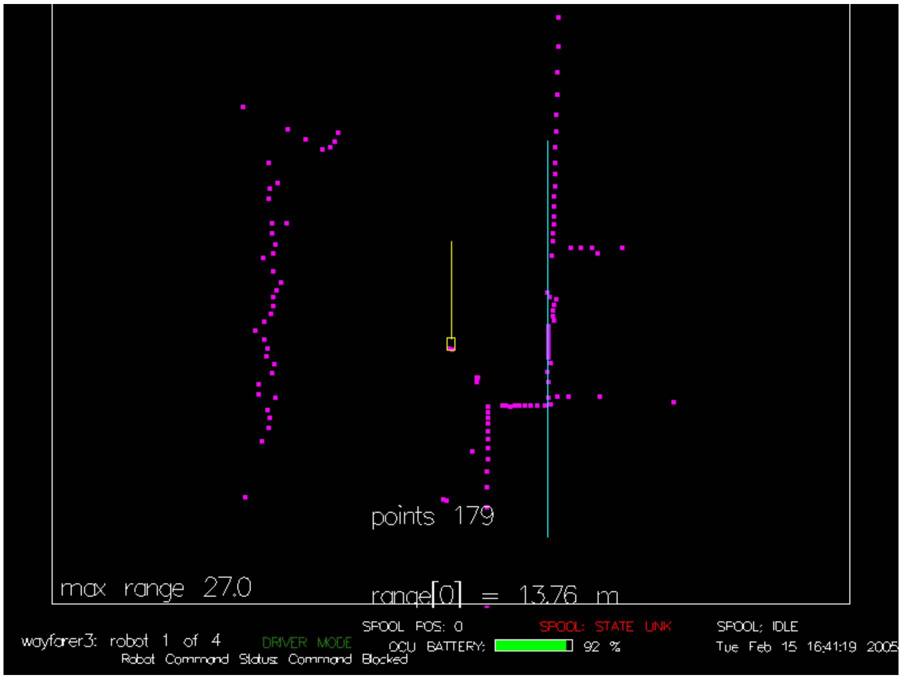

Figure 10: PackBot Wayfarer following strongest line

The one problem that we encountered was that when the robot was tilted so that it was not parallel to the ground, the laser plane would intersect the ground. In some cases, this created a "false positive" potential line that could confuse the perimeter following behavior. To deal with this problem, we developed a range filter that uses the sensor data from the PackBot's pan/tilt sensor to project the laser points into 3D. Then, the points in 3D that are located below the robot (relative to the gravity vector) are removed from the laser scan before this scan is passed to the Hough transform. When the robot is tilted, the laser plane will intersect the ground at some point below the robot (assuming the robot is not directly adjacent to the ground), and so these points will have a negative z-coordinate value relative to the robot. In simple urban terrain, we can just ignore these points. In more complex terrain, we may need to explicitly avoid these points.

We have successfully tested an initial version of this scan filter in indoor environments. We will soon test this filter in combination with obstacle avoidance and exploration outdoors and determine whether it increases perimeter following performance.

5. MAPPING

The Wayfarer mapping system is now operational and fully integrated with the Wayfarer obstacle avoidance and perimeter following systems. The mapper runs on the robot CPU and receives laser range data and position data via Aware. The mapper can be used with either teleoperated or autonomous control. As the robot moves through the environment, the mapper uses the range and position data to construct an occupancy grid[3] map of the world.

The occupancy grid is a Cartesian grid in the world coordinate frame, divided into cells (20 cm x 20 cm in the current representation), with each cell storing the probability that the corresponding location in space is occupied. All cells are initialized to 0.5 occupancy probability.

Whenever the mapper receives a range reading, it uses two-dimensional ray tracing to trace the path from the sensor to the point corresponding to the reading. The mapper increases the probability that the cell containing range point is occupied and decreases the probability that all cells between the sensor and the range point are occupied. Over time, this allows the mapper to build a representation of open, occupied, and unknown space in the environment.

The mapper periodically transmits a compressed (thresholded, run-length encoded) version of the map region surrounding the robot via UDP to the OCU. The OpenGL-based map viewer runs on the OCU and updates the graphical view of the map in real-time, allowing the operator to see the map as it is being constructed. In teleoperated mode, the operator can use the map to guide exploration. In autonomous mode, the operator can watch as the robot explores the world and builds its map. If the robot travels beyond communications range, it will continue to explore and map, and when it returns within communication range, the operator will see that map that it has generated.

Figure 11 shows an indoor map of iRobot HQ generated by the Wayfarer mapping system. The black lines are spaced at one meter intervals. The PackBot Wayfarer constructed this map autonomously, using its perimeter following behavior to follow the walls to the left of the robot and using obstacle avoidance to prevent collisions. The robot was started at a position at the bottom of the map and was allowed to explore the environment autonomously with no operator intervention.

This map was constructed using only odometry for position estimation and shows some drift in the robot's heading estimate, especially when the robot turns around corners. We plan to experiment with a number of methods for reducing this error, including using the fiber-optic gyros on the IMU for heading estimate, using the onboard compass, and using scan matching with laser range readings to estimate the robot's rotation.

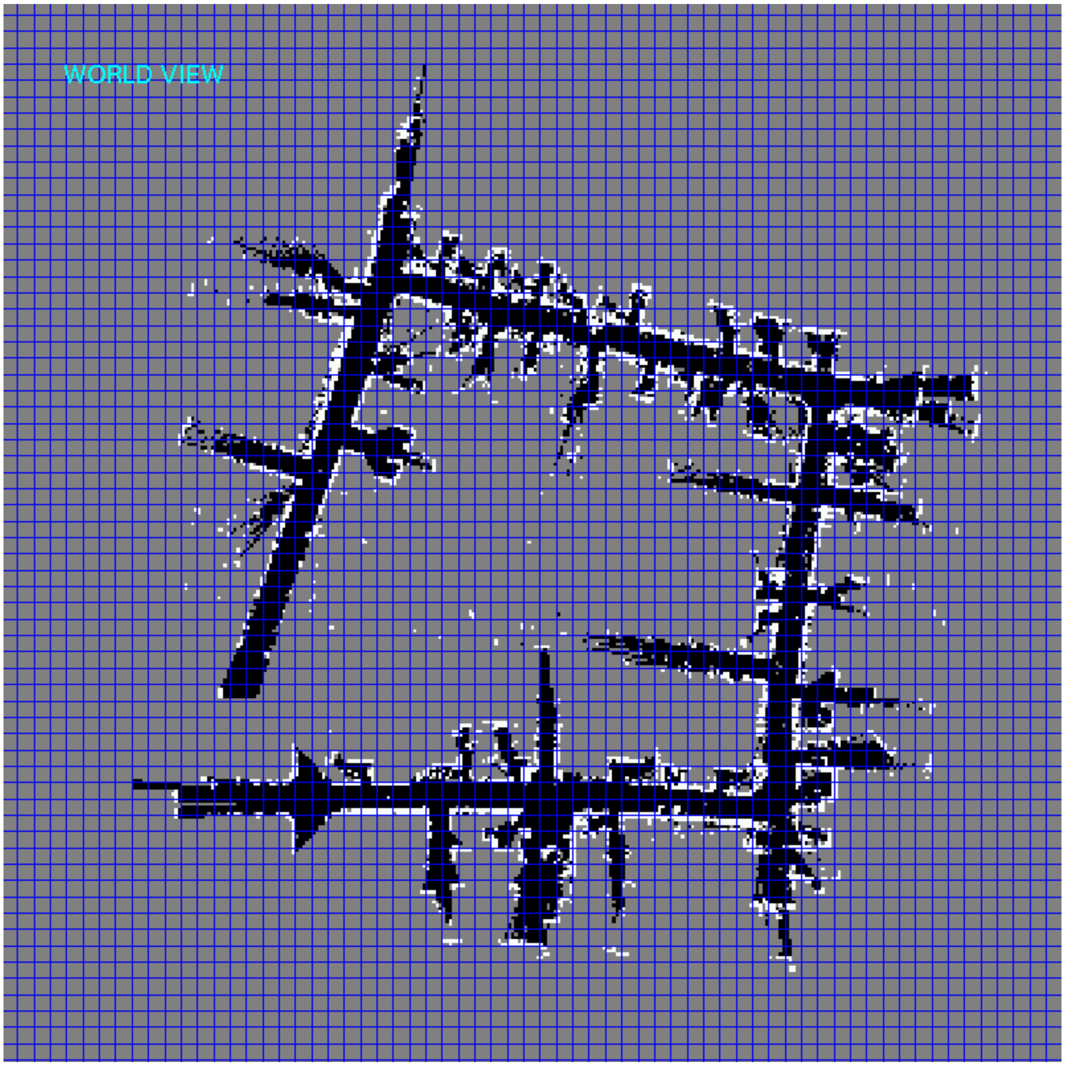

Figure 11: Indoor map generated by fully-autonomous Wayfarer mapping system

We will soon test the mapping system outdoors in combination with the perimeter following system to determine how effectively the robot can autonomously explore and map the exterior of buildings.

6. FUTURE WORK

In future work, we will develop a street following behavior and an intersection detection system, both using the Hough transform for detecting building and street orientations. We will test the fully-integrated Wayfarer system in urban environments to measure its ability to perform urban reconnaissance missions. Our plan is to have a fully-autonomous urban reconnaissance capability for the PackBot Wayfarer prototype by September 2005.

ACKNOWLEDGMENTS

This work was funded by the U.S. Army Tank-automotive and Armaments Command (TACOM) Tank-Automotive Research, Development, and Engineering Center (TARDEC) under contract DAAE07-03-L147.

REFERENCES

1. D. Ballard and C. Brown, *Computer Vision*. Upper Saddle River, New Jersey: Prentice-Hall, 1982.
2. J. Borenstein and Y. Koren, "Real-time obstacle avoidance for fast mobile robots." *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 19, No. 5, Sept./Oct. 1989, pp. 1179-1187.
3. H. Moravec and A. Elfes, “High resolution maps from wide angle sonar.” In *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*, pp. 116-121.
4. B. Schiele and J. Crowley, “A comparison of position estimation techniques using occupancy grids.” In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, May 1994
5. B. Yamauchi, “PackBot: A versatile platform for military robotics,” In *Unmanned Ground Vehicle Technology VI*, edited by G. Gerhart, C. Shoemaker, and D. Gage, Proceedings of SPIE, Volume 5422 (SPIE, Bellingham, WA, 2004), pp. 228-237.