LOW-COST RANGE SENSING FOR LAPTOP ROBOTS

Alan Davidson, Ben Tribelhorn, Tommy Leung, and Zach Dodds
Computer Science Department
Harvey Mudd College
1250 N. Dartmouth Ave
Claremont, CA, 91711
email: {adavidso, btribelh, tleung, dodds}@cs.hmc.edu

ABSTRACT
This paper details the design, creation, and deployment of two low-cost range sensing devices for mobile robot platforms. The first is a set of sonars individually mounted on servo motors. These sensors' angular coverage allows for effective obstacle avoidance, but does not provide precise surface placement due to sonar's well-understood conical response window. An inexpensive laser range finder based on a web camera and servo-mounted laser pointer provides a much more precise, but narrower, proximity detector. Taken together, this sensor suite cost less than $300, but demonstrated its considerable capabilities for reactive and deliberative spatial reasoning at the 2005 AAAI mobile robot scavenger hunt competition.

KEY WORDS
sonar, laser, rangefinder, vision, robotic sensors, low-cost robotics

1 Introduction

Laptop-based robots offer powerful platforms for research and education at a price and maintainance level much lower than systems now in common use. Advantages of a laptop robot include a gentle learning curve in interacting with the software, the ability for students to program and use their own computers, and the capability to investigate topics on a physical agent that are less easily supported on other low-cost platforms. Also, *any* software, including OpenCV, Matlab, and other specialized packages, can be used onboard.

We have been among the first to put laptop robots to a public test. The venue for this was the AAAI 2005 Scavenger Hunt Competition. [2] The scavenger hunt is a showcase for robotics researchers to demonstrate the abilities of their robots to find and identify objects in a dynamic environment. Objects, such as a beach ball, orange cones, a stuffed dinosaur, and bowls of different colors, were scattered around a hotel hallway. Autonomous agents attempted to report the object locations and to deliver the beach ball to a specified destination. The Harvey Mudd College team, composed of Mr. Tribelhorn, Mr. Davidson, and Dr. Dodds as well as Julian Mason and Susanna Ricco, both won this competition and received a Technical Innovation award for "Overall Excellence in a Fully Autonomous System." [4]

Range sensors were crucial to our entry's design. We implemented our own sensors at a fraction of the price of commercial models. This paper details the design of our $200 sonar ring with individually panning units, as well as a $60 laser rangefinding device constructed from off-the-shelf components.

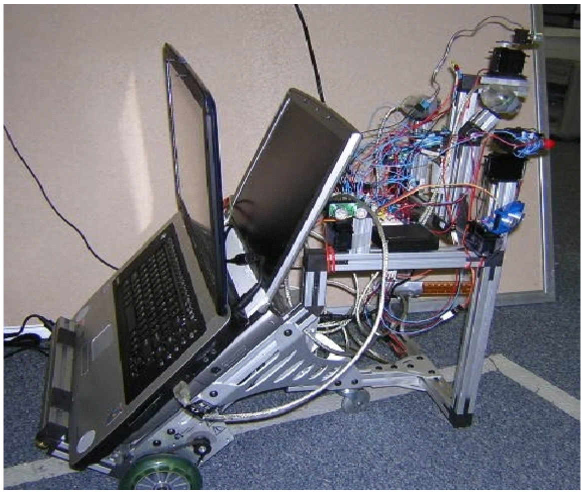

Figure 1. Our robot used an array of servo-mounted sonar sensors and a laser-pointer-based range finder. The chassis is an Evolution ER1; two laptops provide its computational engine. Without the laptops the platform totals $600.

2 Background: Comparing Range Sensors

Table 1 compares the prices and capabilities of several range-sensing devices.

Certainly the premium that commercial laser rangers and sonar rings command are worth it in applications in which speed and resolution are crucial. For example, in DARPA's Grand Challenge, several teams are creating entries based primarily on SICK laser data. [9, 3] However, cost-constrained applications, for which our designs are more suitable, represent a fast-growing subset of robotics research and education. This is particularly true of re-

Name	Range (cm)	Sensor layout	Cost	Reference
Denning sonar ring	300 cm	24 @ 15° intervals	\$18000	`www.southcom.com.au/~robot`
SICK Laser Range Finder	800 cm	180 @ 1° intervals	\$5000	`www.sickusa.com/sickhome.htm`
Pioneer sonar ring	300 cm	8 @ 22° intervals	\$2000	`www.activmedia.com`
IDEC IR ranging unit	50 cm	1	\$250	`www.idec.com/usa`
Devantech sonar unit	300 cm	1	\$30	`www.robot-electronics.co.uk`
Sharp GP2D12 IR unit	75 cm	1	\$12	`www.acroname.com`

Table 1. Several commercially available range-finding sensor systems, their maximum range (the minimum is about 10 cm for each), the layout of the range data obtained, and an online reference. The premium for complete systems with high bandwidth is clear. Educational and other cost-sensitive applications must rely on piecing together sensor suites from inexpensive parts. This paper offers two complementary designs that trade off proximity-sensing precision and coverage.

searchers that seek to apply robots as prototypes or proofs-of-concept within the context of other fields, such as computer vision, AI, collaborative autonomous agents, data-collection, etc.

3 Designing a $200 Panning Sonar Ring

Our sonar ring is composed of five Devantech SRF04 units, shown in Figure 3. [1] These retail for $25 each. Each sonar is mounted on a Maxx MX-400 servo motor, which can be purchased for about $12 each. [7] We manipulate these with a Pontech SV203 servo controller board, as shown in Figure 2. [8] This board connects to a USB-serial adapter allowing it to interface with most laptops. We communicate with the board by sending it strings of text over a serial port.

Figure 2. The Servo Controller Board. The pins to the right control servo motors, while the pins at left measure analog voltages.

As shown in Figure 4, the SRF04 has four connective pins: power, ground, input, and output. The input required is a short 10μs pulse trigger, after which the sonar sends out a burst of sound and sets the output wire high. When the sonar receives the echo, it sets the output wire low again. Therefore, the time that the echo pulse spends high is directly proportional to the distance that the echo traveled. If no echo is received, the output wire is set low after 18ms. The sonar can take a new reading every 36ms, so it operates at around 28Hz.

Since our robot is controlled by a laptop, we needed to transfer this reading onto that computer. Thus, we sought to convert the sonar's TTL signals into a protocol accessible by one of the laptop's ports.

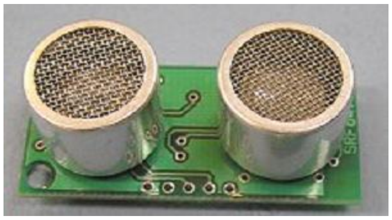

Figure 3. The transmitter and receiver of the Devantech SRF04.

One convenient solution to this problem lies, surprisingly, in the servo controller board. We were able to use the servo controller pins as the input to the sonar unit. This causes the sonar to constantly ping, so that the sonar's output wire produces a pulse train whose high-voltage timespan is proportional to object distance.

The Pontech controller board also has five pins which can read analog voltages between zero and five volts. We constructed an RC circuit to convert the pulse train to a constant, average voltage which would be readable by the controller board. The circuit we use is a low-pass filter, diagrammed in Figure 5. The circuit filters out the high frequency component of the signal (the constant switching from low to high voltage), leaving the average voltage in the pulse train. This resulting voltage is proportional to the amount of time the pulse train is high. The equation that

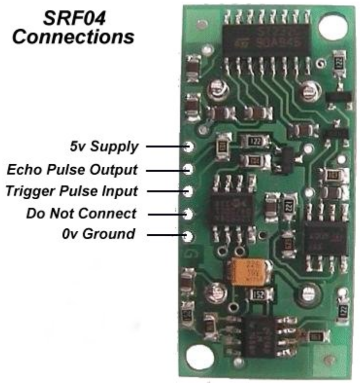

Figure 4. The connections to the SRF04. One of the reasons for our design's simplicity is that the sonar's input requirements are similar to those for a standard servo motor. We transform the time-based output signal into a voltage level with an RC circuit.

governs this behavior is

$$2\pi F \cdot X < \frac{1}{RC} < \frac{2\pi}{F} \cdot Y$$

where R is the value of the resistor, C is the value of the capacitor, X is the rate of change of the sonar readings (for us, this was the velocity of the robot), and Y is the frequency of the pulse train (for our sonar, this was 28Hz). F is a multiplicative constant to ensure that each quantity is significantly smaller/larger than the others. We set F to 10, which allows some play in the values to compensate for slightly different operating conditions and device characteristics.

Our circuit uses a 3.3μF capacitor and a $180\text{k}\Omega$ resistor. These values were chosen for performance based on the limitations above.

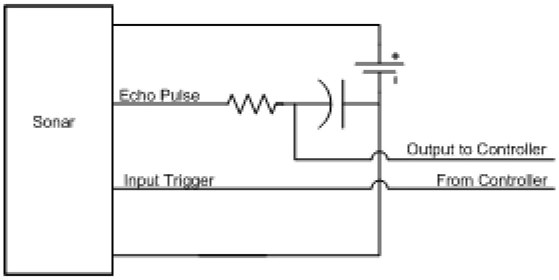

Figure 5. Our sonar circuit, which passes the pulse-width-modulated signal from Pontech's servo controller straight to the input trigger. The output line is passed through a low-pass filter in order to be read by the analog inputs of the same Pontech board.

We communicate with the servo controller board over a serial port in Python. This turned out to be extremely easy because we simply send strings of text to it, and receive strings from it. In addition, the software allows control over the individual orientation of each sonar unit.

4 Designing a $60 Laser Rangefinder

In order to improve the resolution and accuracy of our robot's forward-looking range sensing, we also designed and built a custom laser rangefinder. It is composed of an off-the-shelf laser pointer ($10), two servo motors ($12 each), and a web camera ($25). The servos enable the laser to pan and tilt, as shown in Figure 6. This construction uses the camera and the laser to calculate distances in a manner similar to the way in which binocular vision functions.

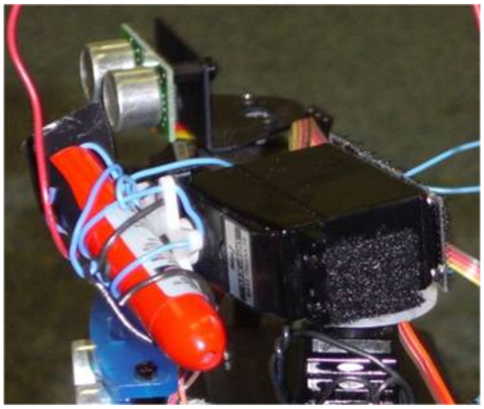

Figure 6. Our laser rangefinder, with a sonar unit pictured behind it. The laser is an off-the-shelf pointer mounted on pan/tilt servos. For our application, varying lighting conditions made it necessary to use only the panning capability of the system.

The camera's exposure is set very low in software, so that only very bright features are visible. The laser dot will appear only in a specific region of the image because we perform only a horizontal sweep with the laser pointer. Other sources of light such as floor reflections and lighting are therefore easy to remove from consideration. If external conditions permit, the laser pointer's ability to tilt would provide data on full planar patches, rather than the cross-sections we used.

We calculate the distance to a surface using triangulation of the three vectors depicted in Figure 7: the direction $\vec{I}$ from the camera to the surface (which is extracted from the camera pixel of the laser dot), the direction $\vec{S}$ of the laser, and its relative position to the camera $\vec{L}$. To obtain more accurate results, we placed the camera and the laser pointer on opposite sides of the robot, increasing the parallax between them. Using the camera lens as our origin, we measured the vector $\vec{L}$ from the camera to the laser pointer. We calculated the direction of $\vec{S}$ as a function of the pan and tilt positions from the servos. We found a roughly linear relationship between the servo positions and the laser's angle. The final vector $\vec{I}$ is found using the pixel in the

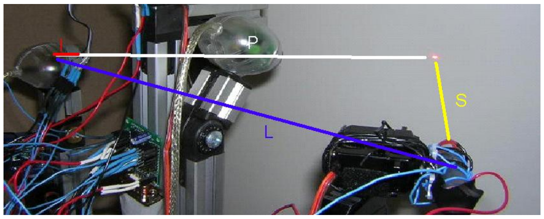

Figure 7. A diagram of the laser ranger and its associated geometry: $\vec{I}$ is the vector from the webcamera's center to the pixel illuminated by the laser dot, $\vec{P}$ is a coincident ray, stretched to the surface on which the laser dot is projected. $\vec{L}$ is the system baseline from camera to laser origin and $\vec{S}$ is the vector from laser origin to the surface whose distance is being measured.

camera image that we have identified as the laser dot. After correcting our image for lens distortion, we were able to treat our camera as an ideal pinhole camera to find the relationship between pixel position and the spacial direction of the pixel's ray $\vec{I}$.

We know that the laser dot is on the surface of a solid object and that it is the intersection of the line $\vec{S}$ through the laser pointer and the ray $\vec{I}$ extending out of the lit pixel. Thus, letting $\vec{P}$ represent the vector from the camera's center to the the laser dot on the external surface, we have

$$\begin{aligned} a\vec{I} &= \vec{P} \\ \vec{L} + b\vec{S} &= \vec{P} \end{aligned}$$

where a and b are scalars. We can now split the vectors into their Cartesian components and rephrase the system as

$$\begin{bmatrix} I_x & 0 & -1 & 0 & 0 \\ I_y & 0 & 0 & -1 & 0 \\ I_z & 0 & 0 & 0 & -1 \\ 0 & S_x & -1 & 0 & 0 \\ 0 & S_y & 0 & -1 & 0 \\ 0 & S_z & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} a \\ b \\ P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -L_x \\ -L_y \\ -L_z \end{bmatrix}$$

This system of equations is overspecified, and an exact solution is unlikely to exist because of modeling and/or roundoff error in the robot, camera, and servo motors. However, a least-squares best fit can be found by performing a psuedo-inverse operation. To do this, we multiply each side of the equation by the transpose of the large matrix, turning it into a square, invertible matrix. We can then multiply by the inverse of the new matrix to find the values of $\vec{P}$. That is to say, if M is the large matrix in the previous equation, then

$$\begin{bmatrix} a \\ b \\ P_x \\ P_y \\ P_z \end{bmatrix} = \left(M^{\mathrm{T}} M\right)^{-1} M^{\mathrm{T}} \begin{bmatrix} 0 \\ 0 \\ -L_x \\ -L_y \\ -L_z \end{bmatrix}$$

The resulting solution is the point which minimizes the distance between the vectors $a\vec{I}$ and $\vec{L} + b\vec{S}$.

The advantages of this setup are clear. This laser rangefinder is cheap and easy to assemble. It has higher resolution than sonar while remaining just as inexpensive. Finally, the method is software-based, making it easy to adapt to specific systems.

5 Results and Performance

We had good results with the sonar design. We were able to use the sonar ring for collision avoidance, and the sensors kept our platform from colliding with walls, other robots, tablecloths, and human legs reliably. Because of the poor angular precision of sonar, it is less well suited for localization, however. The laser shouldered this more demanding – and time-consuming – task.

Our laser rangefinder also worked impressively well. At ranges up to a meter it is accurate to roughly two centimeters, and at ranges up to four meters it is accurate within twenty centimeters. Beyond this, however, the laser dot becomes too faint to be identified by our camera.

At the scavenger hunt, our robot employed Monte Carlo Localization (MCL), utilizing both laser rangefinding and sonar data. [10] MCL provides a collection of probable positions for where the robot might be. We make the assumption that the robot is actually at the position of the most likely hypothesis. The accuracy of the range-data allowed the robot to localize to within a foot its actual position during the competition. [4]

6 Perspective

It is important to push the state-of-the-art across the full spectrum of costs, rather than just the highest-cost sensors which only a very elite group of institutions will have access to. We have shown that effective sonar rings can be built quite inexpensively. We have also shown that simple laser rangefinders are within almost any enthousiast's budget.

Future work will focus on improving common robotics algorithms by incorporating vision. In particlar, there is ongoing research in three dimensional mapping [11], using visual data to extend the capabilities of MCL, and improving robot odometery using densely approximated optical flow.

For additional information including source code, please visit `http://cs.hmc.edu/~dodds/AAAIsh`

7 Acknowledgements

We would like to thank Harvey Mudd College's Professor Spjut for his help in defining the contraints of our sonar circuit. This work was supported by funds provided by National Science Foundation DUE CCLI award #0411176 and by Harvey Mudd College.

References

[1] Acroname Inc. *The Devantech SRF04 Ultrasonic Range Finder*, September 2001.

[2] American Association for Artificial Intelligence. *http://palantir.swarthmore.edu/aaai05/*, 2005.

[3] Carnegie Mellon Red Team Robot Racing. *http://www.redteamracing.org/*, 2005.

[4] Alan Davidson, Mac Mason, Susanna Ricco, Ben Tribelhorn, and Zachary Dodds. Scavenging with a laptop robot. In *Proceedings of the Workshop on the 2005 AAAI Mobile Robot Competition*, Pittsburgh, PA, July 2005. AAAI Press: Melo Park, CA.

[5] Evolution Robotics. *http://www.evolution.com*, 2002.

[6] Intel Corporation. *Open Source Computer Vision Library: Reference Manual*, December 2000.

[7] Maxx Products International Servo Motors. *http://www.maxxprod.com/mpi/mpi-15.html*, 2005.

[8] Pontech. *SV203 Servo Motor Controller Board: User's Manual*, 1.20 edition, 1998.

[9] Stanford Racing Team. *http://www.stanfordracing.com/*, 2005.

[10] S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust monte carlo localization for mobile robots. *Artificial Intelligence*, 128(1-2):99–141, 2000.

[11] Kamil Wnuk, Faith Dang, and Zachary Dodds. Dense 3d mapping with monocular vision. In *Proceedings of the 2nd International Conference on Autonomous Robots and Agents (ICARA '04)*, December 2004.