NavBot: The Navigational Search-and-Rescue Robot

Matthew Marge, Ayman Sawas, Juan Carlos Liberato, Murtaza M. Karim, Manish Muttreja, Nader Alrawahi, and Brian Fink

Department of Computer Science, Stony Brook University, Stony Brook, NY 11794-4400
Department of Mechanical Engineering, Stony Brook University, Stony Brook, NY 11794-2300
Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794-2350
{mmarge, aysawas, jliberat, mukarim, mmuttrej, nalrawah, bfink}@ic.sunysb.edu

Abstract

The Stony Brook Robot Design Team has focused on two main areas of research in the creation of NavBot, our new robot created for the American Association of Artificial Intelligence's (AAAI) Scavenger Hunt Event: navigation and computer vision. The purpose is to create an intelligent machine that is able to navigate the conference floor for specific objects at the AAAI Conference in Pittsburgh, Pennsylvania.

To achieve the desirable speeds required in a rescue robot, NavBot utilizes high-rpm servo motors with a manually adjustable gear train. The two 12V servo motors enhance the maneuverability of the robot by providing minor adjustments in the robot's speed as it navigates an area. The manually adjustable gear trains allow NavBot to function in different environments. This is achieved by controlling the force and the speed required for maneuverability in the terrain. NavBot also utilizes a dual-track motion system to handle rough terrain. To enhance the power of NavBot, it was designed with a light aluminum skeleton. In addition to the large space provided for necessary electrical and computer equipment, the skeleton serves as a secure casing that prevents damage to vital parts.

Since objects in the AAAI Scavenger Hunt Event can be located at varying heights, NavBot is equipped with a linear vertical motion system. This system enables the robot's arm-like appendage to reach objects at heights of up to 2.5 feet. The gripping mechanism is also designed to be universal for many objects with varying shapes and sizes. This arm is implemented using two shafts placed close together to form a column. A belt, controlled by two opposing motion servo motors, placed on the top and bottom of the column, is spun around like a pulley system. Attached to the belt is small box-like cart with a front opening to enclose objects once they are picked up. The cart is designed to move in a vertical motion. The front opening doors are also implemented by low-torque servo motors.

A camera mount is positioned in close proximity to the gripping mechanism in order to provide NavBot with an excellent line of sight toward objects. Using two servo motors, its CMU Cam can pivot with 4 degrees of freedom. Together, the motors are designed to rotate the camera to approximately 140 degrees of freedom in the horizontal axis, and 70 degrees of freedom in the vertical axis.

NavBot's computer vision system requires both sensors and remote wireless communication. Multiple sonar sensors enable NavBot to detect objects in its path. As necessary, sensors will have different distances of detection. We chose sonar sensors because of their reliability, ease of interfacing with the microprocessor, and their flexibility with distance calculation. A Radio Frequency module establishes a wireless connection between the control module and the robot for tasks that require fine movements. We also require the RF module to implement task control for the robot. This way, we can switch tasks and load subroutines onto the robot before the start of each task.

Savage Innovation's OOPIC-R Microprocessor will act as NavBot's brain, controlling all hardware operations. Its compiler allows the usage of Java, Basic, and C to implement its numerous functionalities, which is necessary to create a strong autonomous system. A unique feature of the OOPIC-R is its virtual circuits, which allow multitasking, where objects within NavBot pass information to each other. The Navbot has four sonar sensors connected to the OOPIC, which permit better navigation. The OOPIC-R retrieves color and position information from its multiple CMU Cams, allowing NavBot to adapt to its environment.