CS Capstone Proposal

Tim Sizemore & Andriy Rudyk

Dr. Holliday

CS496

24 January, 2014

Purpose

This semester we plan to focus on implementing artificial intelligence into the tour website framework. Using hazard detection, we plan for the robot to be able to catch user input that would potentially hinder the performance of it and relay that some message that would communicate to the user that the robot cannot proceed with the given command. If time permits, we would also like to include some routine that allow the robot to navigate a room without a controller potentially leading to unmanned virtual tours. Specifically, we have a sonar sensor for this semester and the same camera from last semester. Using these two components, we hope to achieve the previously stated objective.

By using the sonar sensor, we should be able to get a basic idea of our surroundings by just measuring distance to the nearest object of our forward facing side. Using that data, the robot can determine if there is a door or a wall; realistically, any stationary can be detected by the sonar sensor. The limitations of sonar are obvious. We will not be able to differentiate sudden drops, such as stairs, from a hallway. The robot going down stairs, as opposed to a hallway, would be devastating. While the sonar sensor can do basic object detection, it is the camera that will be capable of advanced hazard detection.

With the camera, we will be able to not only use the sonar sensor to determine distance, but also use the camera functionality to find potential hazards. We hope this combination of sensors will allow us to successfully navigate hazards and create routines where the robot will autonomously traverse a room. The camera will use either openCV's Raspberry Pi plugins or ROS which is Robot Operating System. Our choice will depend on which framework is lighter as it needs to be ran on the Pi.

Testing both the sonar and the camera detection features will require a lot of setup. The

sonar testing, luckily, is pretty straight forward. Using the already tested keyboard controller for the robot, we are able to control its movements around a 3D space; therefore, we can move the robot around and output the sensors data to determine if it is correctly determining objects distance through a console. Though it is mostly trial and error, this is definitely the easiest approach to testing hardware. As for the camera, we know that it can transmit video but not until we have a better grasp on the frameworks mentioned above will we be able to design appropriate tests, but it will most likely be trial and error just like the sonar sensor.

Testing autonomous routines is somewhat trickier. Letting the robot loose in a controlled environment first is ideal but putting the robot on any potentially disastrous course is nerve wracking. Incidentally, California actually has standards for testing autonomous vehicles in response to Google's driverless car. While a lot of the information is insurances jargon and cost-benefit breakdowns, there are some interesting snippets about closed road courses and multiple remote termination switches. Fortunately for us, we can chase the robot down if it goes GLaDOS. Other than allowing the robot to see stationary objects in its path and then turning in a random direction and moving again, we have no concrete ideas on hazard detection testing.

This semester we are really focusing all on artificial intelligence features. While only using a minute portion of what AI has to offer, we hope to create autonomous routines that aid in the original idea from last semester. By interpreting the command the robot receives, it can avoid potentially hazardous situations while giving the user a seamless experience of the touring area has to offer. We still have to tweak some of the website features but afterwards it is off to make WALL-E.

All our code will be version controlled through GitHub, just like last semester. Our project can be followed at https://github.com/arudyk/PiBot where we will have releases pushed through after every major development.

Schedule of Completion

- *4 February, 2014*
    - Cleaned key aspects from last semester such as the web socket, the video stream through a browser, and the overall design of the robot.
- *18 February, 2014*
    - Finished all major parts of last semesters guidelines.
- *4 March, 2014*
    - Started the underlying framework and setup needed for object detection using sonar and video.
- *18 March, 2014*
    - Finished the sonar object detection and video object recognition.
- *1 April, 2014*
    - Using the previous week's accomplishments, successfully implemented hazard detection. This encompasses recognizing stairs, doors, or anything else that hinders the robot's movements.
- *15 April, 2014*
    - By building on the hazard detection framework, successfully implemented a simple hazard deterrence routine.
- *29 April, 2014*
    - Implemented hazard deterrence within the the tours system completed at the beginning of the semester.