CAP 6671: Robotics Assignment: Due Apr 23rd

Dr. Gita Sukthankar
gitars@eecs.ucf.edu

This assignment is to give you a chance to implement some of the robotic algorithms we discussed in class. You are encouraged to work in teams of 2-3 people since it will make it possible for you to build a robot with more functionality.

Simulator Platforms

There are four options that you can use for implementing the project:

- physical robot: The Create robot and the Qwerk controller board. We will have a workshop next week and one the following week (time TBD) where you can borrow your robot and learn how to use it: The Java software for controlling the robot is available at: `http://www.terk.ri.cmu.edu/software/terk-source-code.php`
- Microsoft Robotics Studio and Visual Simulation Environment: `http://msdn.microsoft.com/en-us/robotics/aa731520.aspx` A sample project for this is available on the course web page. Programming for this option can be done in C# or the VPL.
- Robocup simulator This is a good option for those of you planning to do Robocup for your final project. `http://sserver.wiki.sourceforge.net/`
- Teambots A very simple Java-based simulator designed for Robocup. `http://www.teambots.org/`
- simulator/game of your own choice

By next Fri (Apr 10th), I'd like to receive email about your choice of platform and project partners.

Deliverables

To get credit for the assignment, you will have to turn in the following things:

- code must be checked into the class subversion repository (URL TBD)
- provide a sheet listing all the team members and references for all the code that you used in completing the project. You are allowed to use outside code to complete your project but it must be correctly referenced.
- video (optional)

Grading

Your must use a single robot platform; you can't choose to implement some functionality in multiple simulators. What we are looking for is a single highly capable robot rather than pieces of one robot scattered over multiple simulators.

For this assignment (10 pts) you must demonstrate a robotic path planner (algorithm of your choice). Two good choices are: 1) the A* code from the previous assignment 2) a potential field planner. You must demonstrate that with a map of the environment, your robot can avoid obstacles and reach a goal. You must demonstrate your path planning for multiple goal locations and two different obstacle layouts (which you should build yourself).

Extra points will be awarded for demonstrating the following:
1-2 pt: create a movie displaying your robot in action
1 pt: create a GUI showing the map of the environment
1 pt: implement a coverage algorithm for systematically searching an unknown area of a given size
1 pt: update the map when new sensor data is obtained (1 bonus point for doing this with the physical robot)
1-2 pt: create a teleoperation interface for driving your robot
1-2 pt: camera-based object instance recognition. Your robot should be able to detect an object and its location in the image.
1-3 pts: robot localization from sensor data To get credit for this you must submit a short writeup explaining how your localization algorithm works.
1-2 pt: any other cool functionality that you can think of adding to your robot along with a short writeup explaining it.

The assignment should be submitted by Wed Apr 22nd and we will have a demo session that evening (time and place TBD).

Remote Students

Teams composed entirely of remote students have the option of doing a movie submission or coordinating with Antoniya and having her demonstrate your work. You will get an extra 1 pt credit to acknowledge the fact that demonstrating your work remotely will be harder to do.