Assignment 4: Vision
CS 393R: Robotics
Due Date: Friday, October 18, 2013

Your task: Write code to detect each of the six beacons while walking and scanning its head, and learn to take and use logs.

This assignment may be done individually or in pairs, with current teams sharing the robots. Please notify Jake and Professor Stone by Friday, October 11 if you decide to work on your own.

Beacon detection will rely on your own blob detection implementation based on this paper. You will use blob detection to form beacons, access the beacon objects in the world object block, and fill in the appropriate height, width, distance, and coordinate measurements.

You will find the vision window of the tool useful for debugging. When a beacon is detected and its appropriate world object is filled out (width, height, image center X and Y), the tool will draw the beacon over the image to indicate its perceived size and location.

A log is available on the lab machines under /usr/local/logs. If you've edited your WorldObject.h file then you won't be able to use the log directly, but you can substitute in the images for any log you take yourself so long as it has the same number of frames. You are encouraged to debug your initial implementation using logs with the "run core" option enabled - this will make the process faster and will minimize the amount of time you need to work directly on the robot.

Once your code is ready, take a log with your robot walking and scanning its head. The process for taking a log is found in the nao setup page under "How to Take a Log".

Checklist: (Scores will be out of 10)

[____] (1 point) Show that your code successfully detects all beacons in the provided logs.

[____] (1 point) Show that your code doesn't detecting anything that is a non-beacon as a beacon in provided logs. Non-beacons include: objects that look like beacons, sideways beacons, beacons stacked so that there are three consecutive colors, etc.

[____] (1 point) Show that your code successfully fills out beacon objects so that they display correctly in the vision window.

[____] (2 points) Show that your code successfully detects all beacons without detecting anything else as a beacon in a log you took sometime before the demo. The robot should have been walking and scanning its head when the log was taken, and at least two non-beacons must be present and correctly handled.

[____] (2 points) Take a log (or logs) during your demo. Then show me in view log that your code successfully detects all beacons without detecting anything else as a beacon. The robot will need to walk and scan its head while you take the log. I will bring some non-beacons to test with (in addition to the real beacons).

[____] (1 point) Show that your beacon distance estimates are accurate, i.e. within 5% of the actual range.

[____] (2 points) Clarity and quality of your memo. Email it - along with any code files you changed - to Jake and Peter by 2pm on October 18.

Extra Credit:
[____] (1 point) Implement an algorithm to determine if a beacon is partially observable. Display a message in the Log window of the UTNaoTool noting whether each beacon in each frame is partially observable.