Robotics (CPS 1/296) : Homework 4

Due: May 2, 2007

1 Detecting Hue

We will be doing SLAM in an environment populated with "landmarks" that are cylinders of known dimensions and distinctive hue. (We can't guarantee distinctive RGB values because of variations in illumination.) Write a routine that converts an image from RGB space to HSB or HSL space. Turn in your code.

2 Finding Landmarks

Write a routine that will search a scene for an object of known hue and then measure the width of this object in pixels. Test your routine by showing that it can identify objects with the same hue in different levels of illumination. (Note that hue is *not* invariant under changes in the type of lighting unless you do white balance adjustment.) Turn in some examples showing how this works.

3 Estimating the focal length and image sensor position of a camera

Devise a system for estimating the focal length and image sensor position of a webcam mounted on one of the robots. Describe your method and report your results.

4 Reporting Landmark Positions

Write a routine that will report landmark positions relative to the robot's position. Test this by placing some landmarks in the world, capturing images with the camera, and then comparing your results to measured distances.

5 Tracking the Robot

Place some landmarks in the environment and measure their positions relative to a starting position for your robot. Plan a path of about two meters for your robot where at least two landmarks will be visible at all times. Track your robot's position using either a particle filter or a Kalman filter. Plot your robot's position distribution at the end of its trajectory (either by plotting the particles, or by showing a one standard deviation ellipse) and indicate your estimate of the robot's position on the map based upon your own measurements.

Note that you will need your motion model and error model from the previous assignment.

6 SLAM

Implement SLAM using either FastSLAM or Kalman filter SLAM. Your program should take as input:

- A set of landmark hues
- A set of landmark widths
- A set of ground truth landmark positions (not used for SLAM, but used to graph your results)

Your program should produce an image as output which visualizes your distribution over landmark positions with ground truth overlaid.

You should turn in your code, along with an example of how it works.

Be prepared to provide a demonstration of your algorithm running with an actual robot, using a landmark arrangement that I will provide after your assignment is completed.

7 Extending SLAM (required for graduate students, optional for undergrads)

Evaluate an extension to the basic FastSLAM or Kalman Filter SLAM algorithm you implemented from the previous section. This can be an extension of your own creation, or one from the literature. Some possible choices would be FastSLAM 2.0, some method for automatically detecting landmarks, or some methods for handling ambiguity in the landmarks, but you shouldn't feel limited to these. (If you have a new idea, please come to talk with me about it well before you proceed, so I can give you some feedback.)

You should turn in the following:

- A description of the extension you are trying
- A description of the results you expect to get
- Your code
- An example of your code running
- An evaluation and analysis of how your implementation lived up to your expectations