Lab 4

Gyro Sensor Guided Dead Reckoning

Objective

In this experiment, students will use ROS, Arduino, MPU6050 gyro sensor and the iRobot Roomba Create to achieve the following goals:

1. Understand the concept of gyro sensor and its usage
2. Implement ROS nodes to publish and subscribe to the gyro sensor measurement using Arduino
3. Get familiar with Arduino programming

Reference Materials

1. MPU-6050 tutorials:
   • http://playground.arduino.cc/Main/MPU-6050.
   • http://diyhacking.com/arduino-mpu-6050-imu-sensor-tutorial/
2. Arduino tutorials
   • https://www.arduino.cc/en/Tutorial/HomePage
3. Rosserial tutorial
   • http://wiki.ros.org/rosserial_arduino/Tutorials

Lab Procedure

This lab consists of the following three parts:

Part 1: Run gyro sensor on Arduino board

1. Install Arduino IDE in your Raspberry Pi (R-pi).
   • Log in to your R-pi by running the command "slogin -Y pi@10.14.1.23x"
   • run "sudo apt-get update"
   • Then run "sudo apt-get install arduino"
   • run "rm .Xauthority"
   • Now, run "arduino" in order to bring up an IDE. Close the IDE. You should have the folder "~/sketchbook" in your R-pi system.

2. Connect your gyro sensor with Arduino board.

3. Run the gyro sensor on Arduino board.
• Download Jeff Rowberg's MPU 6050 interface SW:
  "https://github.com/jrowberg/i2cdevlib" into your Ubuntu VB system. Do "git clone
  https://github.com/jrowberg/i2cdevlib" in your folder in R-pi. Change directory to
  i2cdevlib. Then, do "git checkout c4e3d83"
• Among the downloaded SW, copy the two folders: "Arduino/MPU6050" and
  "Arduino/I2Cdev" into "~/sketchbook/libraries/" in your R-pi system.
• Choose File > Examples > MPU6050 > MPU6050_DMP6 in order to load an example
code.
• Upload the example code on the Arduino. Check on the performance of your sensor.

Part 2: Run gyro sensor in ROS environment

1. Try the following tutorial to get familiar with rosserial:
   http://wiki.ros.org/rosserial_arduino/Tutorials/Hello%20World
2. Create a node to publish roll, pitch and yaw angles of the gyro sensor (Hint: Use the
   example code you tested in Part 1-3)
3. Repeat the square trajectory of Lab2 and record both the odometry measurement and the
gyro sensor measurement. You will need to modify your program to subscribe to the gyro
sensor measurement.
   • Plot the both sensor readings (odometry and gyro) in your report. Discuss the results in
     the report. You can use MATLAB "plot" function for drawing your data.

Part 3: Do the square navigation of Lab2 using the gyro sensor

1. Use the gyro sensor to estimate the orientation of the robot. That is, trust only x and y
   position of the odometry measurement.
   • Plot the trajectory of odometry measurement. Also, plot the trajectory guided by the
     gyro sensor. Make sure to add these plots and discuss your results in the report.

Lab grading

Your grade is based upon three criteria: 1) Pre-lab, 2) Program test, and 3) Post-lab report with
emphasis upon the performance of your program and post-lab report. For this reason, your program
must be checked by TA before submitting your report.