ESE650 Project 4: SLAM

Code due date: **3/22/2018 at 1:20pm** on Canvas, <pennkeyID>_project4.zip
Report due date: **3/22/2018 at 11:59pm** on Canvas, <pennkeyID>_project4.pdf

In this project, you will implement the structure of mapping and localization in an indoor environment using information from an IMU and range sensors. You will integrate the IMU orientation and odometry information from a walking humanoid with a 2D laser range scanner (LIDAR) in order to build a 2D occupancy grid map of the walls and obstacles in the environment. After this, you will integrate additional camera and depth imagery from a Kinect One sensor to build a textured map. You will first implement your 2D localization and mapping system before moving on to the RGBD part. Training sets of odometry, inertial, and range measurements from a THOR-OP humanoid robot will be provided for this project.

Training Data Download: Now available at https://upenn.box.com/v/ese650-p418-train
https://upenn.box.com/v/ese650-p418-train-rgb
Test Data Release: 3/22/2018 1:30pm, https://upenn.box.com/v/ese650-p418-test
https://upenn.box.com/v/ese650-p418-test-rgb

:These data sets contain timestamped sensor values, corresponding to the raw sensor readings. Download these files and be sure you can load and interpret the file formats. You can find the 'docs/config_slam.pdf' file that explains about the data.

Upload: on Canvas

(1) Code (due 3/22/2018 1:20pm, <pennkeyID>_project4.zip)
: Do include the resulting map images but do not include data when you submit.

(2) Write-up (due 3/22/2018 11:59pm, <pennkeyID>_project4.pdf)
: Write a project report including the following sections: Introduction, Problem Formulation, Technical Approach, Results and Discussion.
: Make sure your result includes (a) SLAM image, (b) Textured Maps of training and test data set (If you have video files, please include them!)

1. SLAM: First, you should run your robot using pure walking odometry measurements and yaw gyro readings. Make a 2D map using this data before correcting using range readings. You should then be able to provide a visualization of the motion of the robot within a 2D map. Up to this point, you will have done the first phase of this project.

Next, you will need to simultaneously localize the robot pose, and construct the surrounding 2D map using a pose filter and occupancy grid algorithm. Cpp files will be provided, along with python binding instructions, to help you experiment with 2D LIDAR scan matching.

2. TEXTURED MAP: The second portion of this project will be to integrate the RGB and depth images from the Kinect to build a textured visualization of the environment. You may work with dense point clouds, or with a sparser set of tracked features for this part. You will do this with pure odometry, and with your full SLAM routine.

3. TESTING. You should then make sure that your program can take input sensor readings from unknown environment. We will release the test data in the morning in order to give some time to run your program with test data. You should be able to show your results with proper visualization from an unknown test dataset.

4. Help functions for python are provided. Please see files in the 'docs' directory for data description and additional implementation details.