
CSCE574 – Robotics

Spring 2014 – Project 2

Assigned: February 3

Due: February 23 ~~February 21~~ ~~February 18~~

The purpose of this assignment is to introduce you a few new elements of ROS—namely launch files, bag files, and rviz—and to give you some practice working with sensor data from a real robot.

You may do this assignment on your own or in pairs.

Getting Ready

This assignment relies on the material from Chapters 6, 7, and 9 of *AGITR*. Chapters 4, 5, and 10 may also be useful, but are less crucial for this assignment. Two additional short PDFs, dealing with rviz and `sensor_msgs/LaserScan`, are available on the course website.

The Task

The course website contains a collection bag files recorded by a robot equipped with a laser range finder. When each of these bags was recorded, the robot and the nearby obstacles remained motionless. However, in each case, one or more *people* walked past the robot. Your job is write a program to *track* the people that the robot sees, and *count* the number of unique people that appear.

- ✓ Create a new package called `count_people_lastname` or `count_people_lastname1_lastname2`.
- ✓ Within that package, create a launch file called `count.launch`, which starts both your program and `roslaunch play`.
 - This launch file should use an argument called `bag`. The argument will be set on the `roslaunch` command line to the name of the bag file to play. This file is assumed to be in your package directory. For example, after copying the example bags to the `src/count_people_okane` directory, your instructor tests his program using this command (not necessarily from that directory):

```
roslaunch -screen count_people_okane count.launch bag:=example1.bag
```

This format is required for full credit to ensure a consistent interface.
 - The launch file should be configured to terminate all of the other nodes when `roslaunch play` completes.
- ✓ Several times per second, your program should publish a message of type `sensor_msgs/PointCloud` on the topic `person_locations`. This message should contain one point for each visible person.
 - The header field of this message type includes a field called `frame_id`, which you should set to `laser`.
 - The `channels` field of this message should be left blank.
- ✓ Each time a *new* person appears, your program should publish a message of type `geometry_msgs/Point32` on the topic `person_appearances`.

CSCE574 – Project 2 Cover Sheet

Name(s): _____

ROS usage (25):

- Submitted file contains a well-formed ROS package.
- Package is named correctly.
- Package dependencies are correct.
- Package is configured correctly to build executable.
- Launch file starts rosbag play.
- Launch file starts other nodes.
- Launch file accepts argument correctly.
- Launch file determines after rosbag play is done.

Correctness (65):

- Detects people.
- No false positives.
- No false negatives.
- Robust to noisy data.
- Computed positions match real locations.
- Person count is correct.

Report (10):

- Report is complete and clear.
- Required sections exist under readily identifiable headings.
- Report is free of typos and grammatical errors.

Other comments:

Total:
