UDACITY

‹ Return to "Robotics Software Engineer" in the classroom

DISCUSS ON STUDENT HUB

Go Chase It!

REVIEW

CODE REVIEW

HISTORY

Meets Specifications

Congratulations on completing this project. You passed all the requirements!
Keep up the great work!

Basic Requirements

The student submitted all required files specified in the criteria.

You submitted both your robot package and the ball chaser package. Great!

The student followed the same directory structure detailed in the project description section.

Both of your packages follow exactly the same directory structure as detailed in the project lesson. Good job.

Robot Design

Robot design requirements:

- Lidar and camera sensors.
- Gazebo plugins for the robot's differential drive, lidar, and camera.
- Housed inside the world
- Significant changes from the sample taught in the project lesson.
- Robot is stable when moving

Your robot meets all the requirements and is housed inside a world. That's great!

Gazebo World

Gazebo world requirements:

- Same as the world designed in the Build My World project or a new world that you design on the building editor for this project.
- Includes a white-colored ball

Your World includes a white ball that you will later chase, nice!

Ball Chasing

`drive_bot` requirements:

- A `ball_chaser/command_robot` service.
- Service accepts linear x and angular z velocities.
- Service publishes to the the wheel joints.
- Service returns the requested velocities.

`process_image` requirements:

- Subscribes to the robot's camera image.
- A function to analyze the image and determine the presence and position of a white ball.
- Requests a service to drive the robot towards a white ball (when present).

Both nodes are very well written and structured. They also both meets the purpose of driving the robot and processing images. Awesome job!

Note that the drive_bot node is not printing the velocity inputs given to the robot in the terminal. This is due to missing of following code line from the drive_bot code

```
ROS_INFO("DriveToTarget received- linear_x:%1.2f, angular_z:%1.2f",(float)req.linear_x, (float)req.angular_z);
```

It is important to print the velocities in the terminal because it helps us to understand the behavior of the robot properly. Moreover, printing necessary information in the terminal can many times help you to debug the project easily.

I am going to pass you in this one as I can see you have fulfilled the basic requirement of the rubric.

Launch Files

`world.launch` requirements:

- Launch the world (which includes a white ball).
- Launch the robot.

`ball_chaser.launch` requirements:

- Run the `drive_bot` C++ node.
- Run the `process_image` C++ node.

Both launch files are very well written and structures. They also both meets the purpose of launching your world and both C++ nodes that you wrote. You rock!

DOWNLOAD PROJECT

RETURN TO PATH

Rate this review