

Project

University of Wisconsin - La Crosse

TurtleBot Control with Pygame

Task: Create a program that enables the user to control a Turtlebot in a Gazebo environment using Pygame. The program should display a sprite on the Pygame screen that represents the Turtlebot's movement in Gazebo.

Instructions:

1. Create a Turtlebot3 in the Gazebo environment using the `'turtlebot3_world.launch'` world file.
2. Create a Pygame window that displays the Turtlebot sprite in the center of the screen.
3. Use the Pygame keyboard module to detect when the user presses the `'w'`, `'a'`, `'s'`, and `'d'` keys.
4. When the user presses `'w'`, the Turtlebot should move forward in Gazebo, and the Turtlebot sprite on the Pygame screen should move up at a certain speed. When the user presses `'s'`, the Turtlebot should move backward in Gazebo, and the Turtlebot sprite on the Pygame screen should move down at the same speed. When the user presses `'a'`, the Turtlebot should turn left in Gazebo, and the Turtlebot sprite on the Pygame screen should turn left at a constant angular speed. When the user presses `'d'`, the Turtlebot should turn right in Gazebo, and the Turtlebot sprite on the Pygame screen should turn right at the same constant angular speed. In addition, when the user release the corresponding key, the turtle should stop moving or turning.
5. To map the Turtlebot's movement in Gazebo to the Turtlebot sprite's movement on the Pygame screen, you will need to determine a ratio that maps the distance moved by the Turtlebot in Gazebo to the pixels moved by the Turtlebot sprite on the Pygame screen.
6. Make sure to update the position of the Turtlebot sprite on the Pygame screen every time the user presses a key.
7. Test the program by running the Turtlebot in Gazebo and moving it around using the Pygame controls.

Points distribution:

- Setting up the Turtlebot in Gazebo and creating ROS commands: 15 points
- Creating the Pygame interface: 15 points
- Implementing movement controls for the Turtlebot in the Pygame interface: 25 points
- Implementing movement indication for the Turtlebot sprite on the Pygame screen: 45 points

To complete this task, students will need to be familiar with ROS, Gazebo, and Pygame. They will need to be able to create a ROS node that initializes and control the Turtlebot, as well as a Pygame program that displays the Turtlebot sprite and responds to user input. They will also need to develop a mapping ratio that maps the Turtlebot's movements in Gazebo to the movements of the sprite in the Pygame window.