Contents

1 Useful Resources

Robot modelling in urdf is rather tedious. Here are a couple of guides which I have found useful.

- URDF/xacro guide
- Guide for gazebo plugins (used to simulate sensors/actuators in gazebo)
- Sonar Plugin Guide

2 Installation

1. Install the joint state publisher:

```
$ sudo apt install
    ros-melodic-joint-state-publisher-gui
```

2. Next, git clone realsense package into your catkin folder and run catkinmake.

```
$ git clone
    https://github.com/pal-robotics/realsense_gazebo_plugin.git
```

3. Enter the following command into the terminal to setup gazebo environment variables

```
$ echo "source /usr/share/gazebo/setup.sh" >> ~/.bashrc
$ source ~/.bashrc
```

3 Package Overview

3.1 urdf

The urdf folder is used to store (.urdf.xacro) and (.gazebo.xacro) files.

- (.urdf.xacro) files describe the transformations between the various links of the robot, along with other variables such as intertia.

- (.gazebo.xacro) files would describe the kinematic properties of the robot such as velocity limits, coefficient of friction etc.

3.1.1 Intel Realsense D435 camera

- The (_d435.gazebo.xacro) file is used to describe the various kinematic properties of the depth camera.

- The (_d435.urdf.xacro) file is used to describe the various frames of the depth camera.

  The (<camera>_bottom_screw_frame) is used as the main reference point for the depth camera. It is centered about the bottom tripod mount of the depth camera. Thus, to define the position of the depth camera on the robot, we would have to provide the transformation from base link to the bottom screw frame link

3.2 meshes

The meshes folder is used to store the various mesh files of the robot(.stl .dae). The mesh files are used to visualize various components of the robot(sensors, chassis) and to enable accurate collisions within gazebo.

3.3 rviz

The rviz folder is used to store the various rviz configs.

3.4 launch

- The launch folder is used to store roslaunch files. Launch files titled simulate_<robotname>.launch are used to launch the gazebo simulation environment and rviz.

- Launch files titled view_<robot>.launch are used to view the robots urdf model in rviz.

4 Mesh Manipulation

4.1 Reducing the quality of a mesh

In most cases, the generated meshes are too large for gazebo to render at a high framerate. Thus, we would have to reduce the resolution of our meshes.

1. Open meshlab and import the mesh (file > import mesh)

2. Click on Filters > Remeshing, Simplification and Reconstruction > Quadratic edge collapse decimation

3. Select a lower number of (target number of faces) and click apply.

4. click on file > export mesh as > (enter mesh name and select *.stl* file type)

5 Gazebo World from 2D Map

5.1 Setup & Installation

1. Ensure that the map server is installed on your system.

   ```
   $ sudo apt install ros-melodic-map-server
   ```

2. Clone the *map2gazebo* repository into your catkin/src folder.

3. Navigate to the *map2gazebo* package in your catkin/src folder and run the following commands.

```
$ pip install --user trimesh
$ pip install --user numpy
$ pip install --user pycollada
$ pip install --user scipy
$ pip install --user networkx
```

4. Lastly, navigate to your catkin_ws folder and run catkin_make

5.2 Generation of stl file

1. launch roscore

2. Navigate to the folder containing the *pgm* and *yaml* file for the map that you would want to convert and run the following command to launch the map server.

```
$ rosrun map_server map_server <map name>.yaml
```

3. Then, run the following command to generate the *map.stl* at a given directory. (replace */home/roald/Desktop/generatedmaps* with your own path)

```
roslaunch map2gazebo map2gazebo.launch
    export_dir:=/home/roald/Desktop/generatedmaps
```

5.3 Importing the mesh into Gazebo

test test