Map My World Project using SLAM in ROS

Koki Mitsunami

Abstract—This paper describes mapping a world with SLAM (Simultaneous Localization and Mapping) algorithm and generation of a new environment for the algorithm by developing a ROS package. original map is generated in Gazebo, that would allow the robot to map itself in a particular map. RTAB-Map (Real-Time Appearance-Based Mapping) is used as a SLAM algorithm to develop a robot which can map environments in both 2D and 3D.

Index Terms—Robotics, Udacity, ROS, Robot Operating System, SLAM, RTAB-Map.

1 Introduction

In this project, ROS package for SLAM is developed to create a 2D occupancy grid and 3D octomap from a provided simulated environment and. Furthermore, newly simulated environment is also generated to map as well. These environments are successfully mapped using the RTAB-Map (Real-Time Appearance-Based Mapping) algorithm. The source for this project can be found here: https://github.com/mitsunami/RoboND-SLAM-Project

2 Background

A robot must construct a map of the environment, while simultaneously localizing itself relative to this map. This is a challenging task since neither the map nor the robot's pose are provided. With noise in the robot's motions and measurements, the map and the robot's pose will be uncertain. And the errors in the robot's pose estimates and map will be correlated. For robots to be useful to society, they must be able to move in environments that they have never seen before. For instance, one of SLAM's application is the robot vacuum cleaner. Such a vacuum comes equipped with sensors that help it map the room and estimate its poses. Other applications of SLAM include self-driving vehicles, be on the roads, an underground mines, or aerial surveillance, and even robots mapping the surface of Mars.

SLAM algorithms generally is categorized into five.

- Extended Kalman Filter SLAM (EKF)
- Sparse Extended Information Filter (SEIF)
- Extended Information Filter (EIF)
- FastSLAM
- GraphSLAM

In those, FastSLAM and GraphSLAM are two most common approaches. FastSLAM uses the particle filter approach with the low dimensional extended Kalman filter to solve the SLAM problem. GraphSLAM uses constraints to represent relationships between robot poses and the environment, and then tries to resolve all of these constraints to create the most likely map given the data. In this project, GraphSLAM is selected to implement SLAM application by choosing RTAB-Map package in ROS.

3 Scene and Robot Configuration

In this section, two simulated scenes and robot configuration are described. The two scenes are created as Gazebo world. One is a provided by the project, Kitchen and Dining world. Two is newly created for this project.

3.1 Kitchen and Dining Scene

As described above, Kitchen and Dining world is provided by the project. The layout of the world is shown below.

Fig. 1. Kitchen and Dining

3.2 Cafe Scene

Cafe world is created in Gazebo. The appearance of the world is shown below. The cafe world is larger than Kitchen and Dining world, which means that that makes SLAM more difficult. In the cafe world, several objects are located for clues of mapping. For instance, textured objects such as fountains, pipe trees and bricks are recognizable because they have more feature points while other texture-less objects like tables have less features.

3.3 Robot Configuration

This subsection explains robot configuration. The robot used is shown below.

The robot is consists of a 2D laser (hokuyo), a RGB-D camera, two wheels, rounded body. Its size is similar to

Fig. 2. My Cafe World

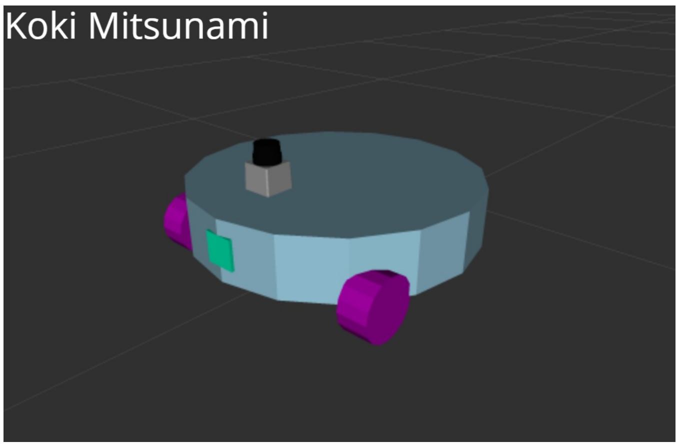

Fig. 3. Robot Appearance

a vacuum cleaner robot. And the wheels are controlled to proceed in the environments. The 2D laser is attached on top of the robot, and the RGB-D camera is mounted in front of the robot. The robot frames are shown below. The camera optical frame is rotated by 90 degrees to achieve correct mapping results.

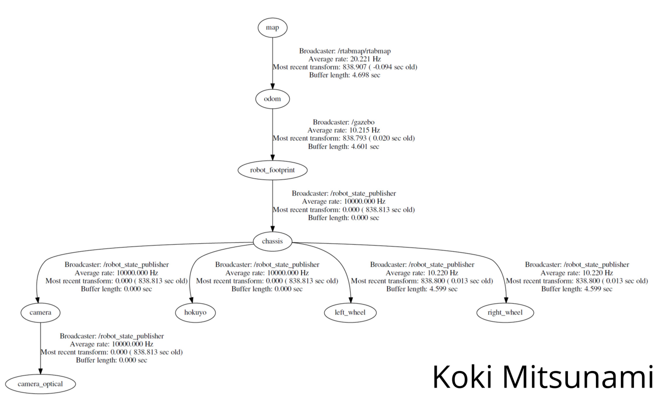

Fig. 4. Robot Configuration

4 Results

This section shows ROS package structure and the results of final map (2D/3D) for both Gazebo worlds.

Environment structure in ROS is shown below. The robot is controlled by teleop node which transmits commands by a keyboard. Then, images from the 2D laser and the RGB-D camera on the robot are used to execute SLAM processing in RTAB-Map package.

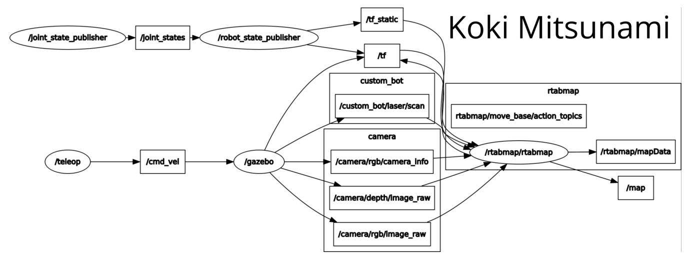

Fig. 5. ROS Package Structure

Desktop screenshot at start time of the simulation is shown below. Gazebo window, RViz window, RTAB-Map console and teleop console can be seen in the figure. The simulation is executed on a Jetson TX2 board.

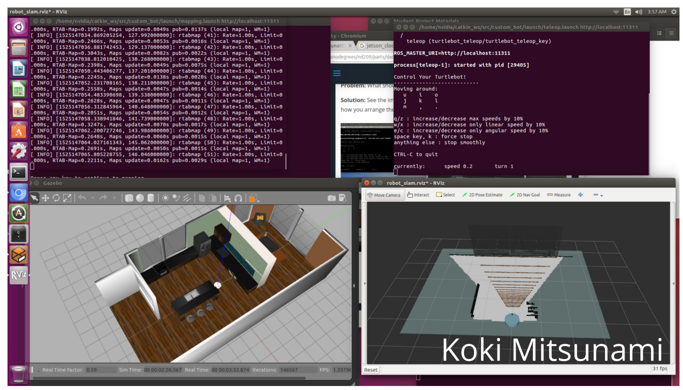

Fig. 6. Desktop Screenshot

4.1 Kitchen and Dining Scene

In Kitchen and Dining scene, the robot is controlled manually to move around the three times. The final 2D/3D map achieved are shown below.

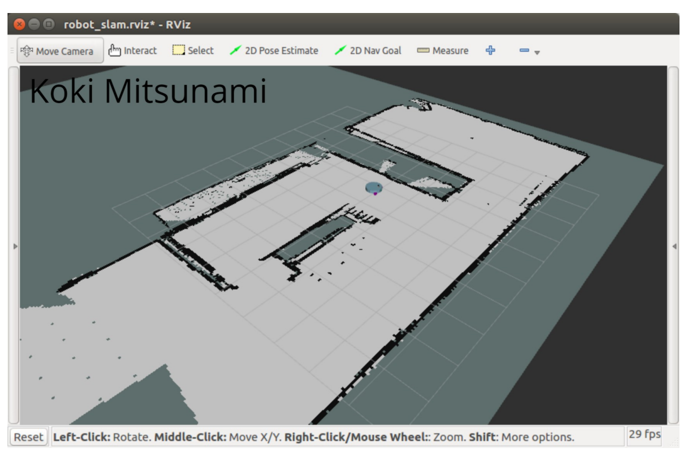

Fig. 7. Final 2D Map of Kitchen

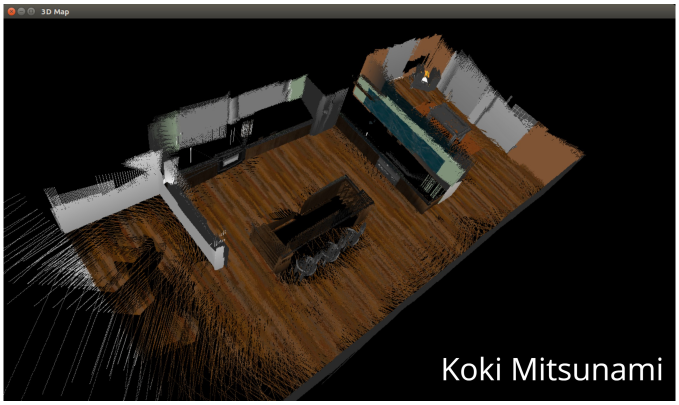

Fig. 8. Final 3D Map of Kitchen

4.2 Cafe Scene

Similar to previous description, the robot is controlled manually to move around the three times in Cafe scene. The final 2D/3D map achieved are shown below.

Fig. 9. Final 2D Map of Cafe

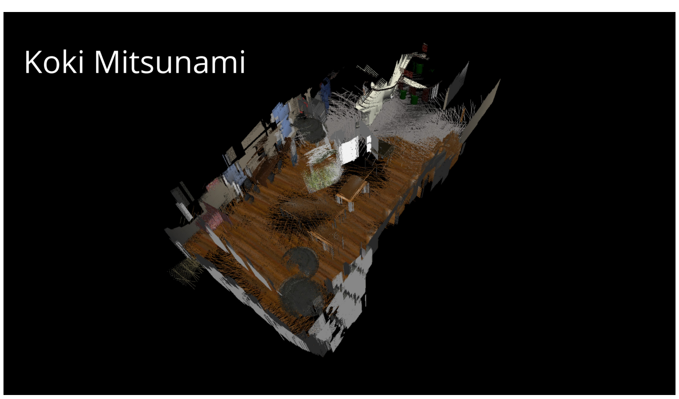

Fig. 10. Final 3D Map of Cafe

5 Discussion

There are some differences between the result of Kitchen and Dining world and Cafe world as seen in the previous section. Most critical point is that mapping accuracy in Cafe scene is worse due to some reasons. Causes considered are the room size and objects in the room. The room size of Cafe is much bigger than Kitchen and Dining. Therefore, the mapping becomes difficult because SLAM in a larger room needs more consistency in terms of feature matching.

Another possible cause is that instability of the robot trajectory. The robot in this project does not move straight ahead even when command sent is ”go straight”. The actual trajectory is shown. Instability movement can be seen in the figure. Loop closure processing in SLAM might be failed since the robot does not pass the same way due to the instability. That might due to the robot configuration such as wheels’ location or overall structure.

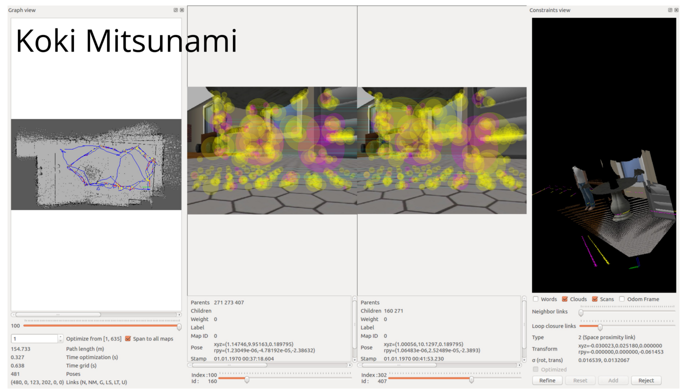

Fig. 11. Trajectory in Cafe

6 Future work

In this project, the robot can simultaneously localize and map worlds. However, the result of SLAM becomes poor depend on environments. To improve that, some countermeasure can be considered. One is that tuning SLAM parameters. SLAM has a lot of parameters to tune up. It is necessary to adopt appropriate parameters depend on the environment. For example, there are many ways to extract feature points while SURF is chosen as feature detector algorithm in this project. Another measure can be re-structure of the robot configuration. As described in the previous section, stability of the movement should be improved and there are countless options to expand the configuration. Than can be improved for a purpose of the robot.

Additionally, it is important for robots to autonomously adapt external environments. Hence, more cognitive ability or path planning and navigation should be added to robots as a next step.