RTAB SLAM Project

Muthanna A. Attyah

Abstract—RTAB-Map is considered to be the best solution for SLAM to develop robots that can map environments in 3D. These considerations come from RTAB-Map's speed, memory management, and its software tools that are available for wide range of environments. In this project **rtabmap_ros** package, which is a ROS wrapper (API) for interacting with RTAB-Map will be configured in ROS on a simulated robot with RGBD camera and laser scanner. The algorithm will be used to SLAM in two different simulated worlds to generate a 2-D and 3-D occupancy maps for both worlds.

Index Terms—Robot, IEEEtran, Udacity, Localization, SLAM, RTAB Map, Octomap, ROS

1 Introduction

RTAB-MAP (Real-Time Appearance-Based Mapping) is a RGB-D Graph-Based SLAM approach based on an incremental appearance-based loop closure detector. The loop closure detector uses a bag-of-words approach to determinate how likely a new image comes from a previous location or a new location. When a loop closure hypothesis is accepted, a new constraint is added to the maps graph, then a graph optimizer minimizes the errors in the map. A memory management approach is used to limit the number of locations used for loop closure detection and graph optimization, so that real-time constraints on large-scale environments are always respected. RTAB-Map can be used alone with a hand-held Kinect or stereo camera for 6DoF RGB-D mapping, or on a robot equipped with a laser range-finder for 3DoF mapping. RTAB-Map tools are available for multiple environments including ROS, Ubuntu, MacOS, Windows, Android, Raspberry Pi, and Docker. [1]

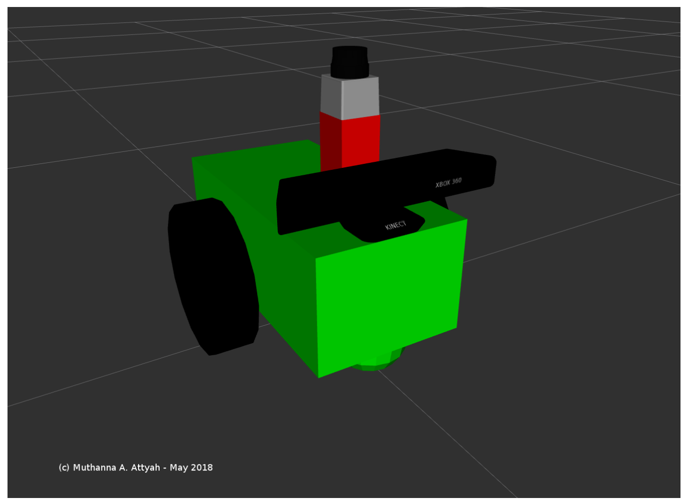

Fig. 1. robot

2 Background

Explain the importance of both mapping two and three dimensional space. The student provides a sufficient background into the scope of the problem / technologically while also identifying some of the current challenges in robot mapping and why the problem domain is an important piece of robotics. They further discuss and compare mapping algorithms

3 Model Configuration

Scene and robot configuration - Explain how your personal Gazebo world was created and what is the layout of it. Justify your choice of robot parameters, sensor location, and how you decided to configure your package structure. Scene and robot configuration - Student explains how the gazebo world was created by providing an overview of the layout of items in his/her customized Gazebo world. Student also describes the robot's parameters, sensor features, and reasoning on the package structur

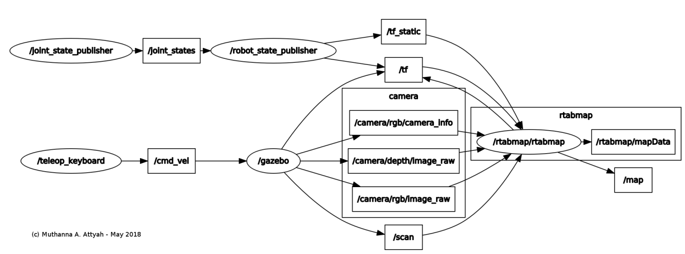

Fig. 2. ROS Graph

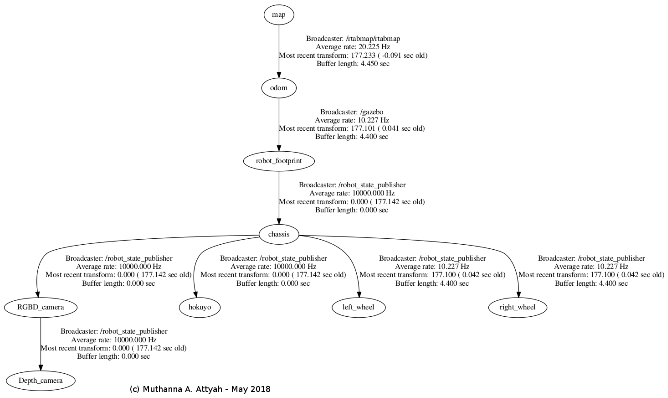

Fig. 3. TF Frames Map

3.0.1 *Gazebo plug-ins*

3.0.2 *Parameters*

4 World Creation

4.1 Supplied Gazebo World

Fig. 4. Kitchen World

4.2 Personal Gazebo World

Fig. 5. Personal World

5 Results

Results - Show the results of both occupancy grid and 3D map. The student should include the images for mapping process, final map (2D/3D) for both Gazebo worlds.

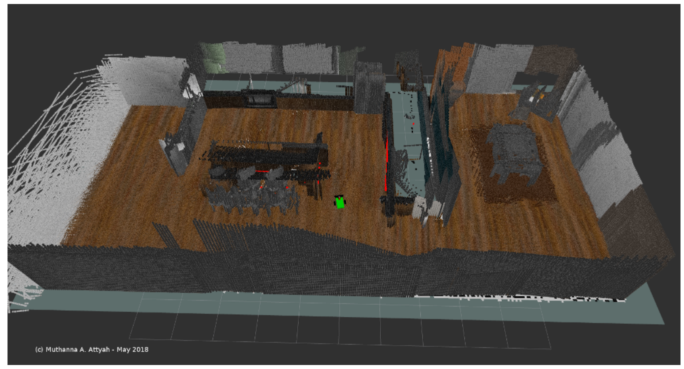

Fig. 6. Kitchen World 3D Map

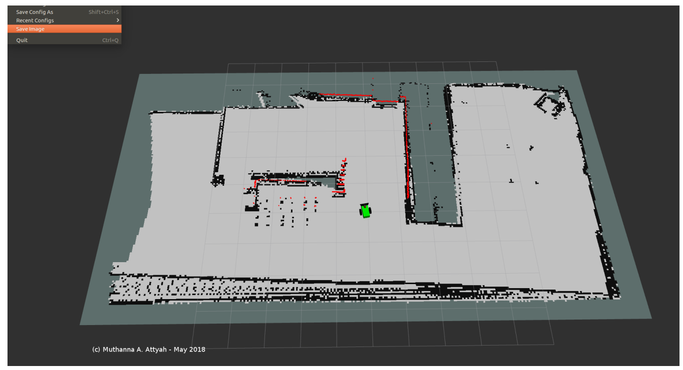

Fig. 7. Kitchen 2D Map

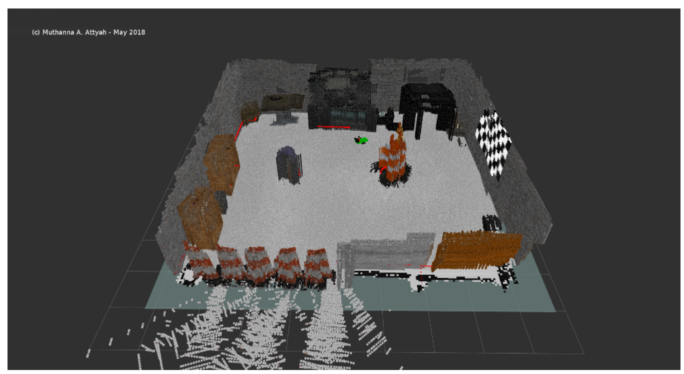

Fig. 8. Personal World 3D Map

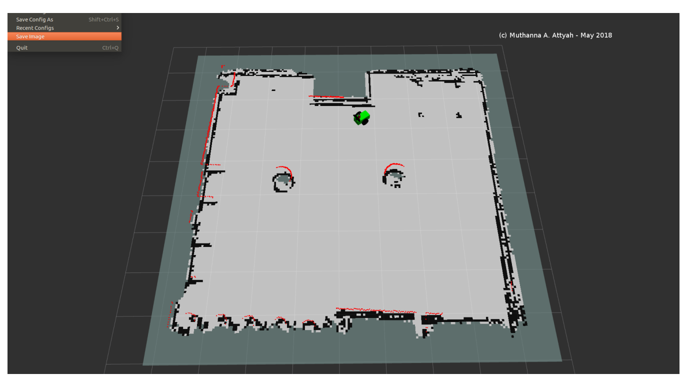

Fig. 9. Personal 2D Map

6 Discussion

What went well, what went wrong. Reflect upon the results of your robot's performance, and the performance of mapping in both worlds. Justify your answers with facts. The student explains how the procedure went and methodologies to improve it. The student should compare and contrast the performance of RTAB Mapping in different worlds.

7 Future work

Student discusses future desires with RTAB-Map. Talk about any robots and environment they applied this too. "Future Work - The student can discuss how they would like to leverage this tool in robotics. The student identifies other areas where mapping could be done and for what reason. Such as simulated room or physical place.

Optional - The student applies RTAB Mapping to a real robot."

Suggestions to Make Your Project Stand Out!

Stand out submissions should have a unique environment with a robot that is able to map correctly. Furthermore the application to real hardware would be a huge win!

References

[1] M. Labb and F. Michaud, *Real-Time Appearance-Based Mapping GitHub Repo*. Introlab.