RoboND Project: Map My World Robot

Milton Wong

Abstract—In this project, we are exploring the mapping capabilities of RTAB-MAP using the URDF robot model in previous project with newly equipped depth camera and necessary configurations. Then, we apply RTAB-MAP for building the map of the provided environment (kitchen-dining.world) and explore loop closure on the resulting map. Then, simulated environment is built in Gazebo and RTAB-MAP is applied for another mapping.

Index Terms—Graph-SLAM, Mapping, Loop closure.

1 INTRODUCTION

IN previous project (Where Am I), weve solved the problem of determining the robot localization on a given map by using the odometry and sensors data. But what if robot doesnt have an given map and the robot needs to operate in an unknown environment? The mapping process is needed in this scenario.

Mapping is a process of building the map of an environment through the measurement from robots sensors. In this project, the mapping is constructed by *Real-Time Appearance Based Mapping* (RTAB-MAP) [1] algorithm, which is a RGB-D camera based Graph-Based SLAM [2] approach that using appearance based loop closure detector for correcting accumulated errors due to the errors in sensor/odometry measurements.

RTAB-MAP algorithm is fast enough for real-time performance, due to the special memory management approach, by using the limited number of locations for loop closure detection and graph optimization.

2 BACKGROUND

SLAM is a challenging task to solve, as map is needed for localization and the Robots poses (result of localization) is need for mapping. The problem is more challenge than localization and mapping isolated since neither the map nor the robot poses are usually provided.

There are several algorithm to tackle this problem. One seminal work is Fast-SLAM [3], which is particle-filtering approach. The second one is Graph-SLAM, which is a SLAM algorithm that solves the full SLAM problem. This means that the algorithm recovers the entire path and map, instead of just the most recent pose and map. This difference allows it to consider dependencies between current and previous poses.

One advantage of Graph-SLAM is the reduced need for significant on board processing capability. Another advantage that can be immediately appreciated is Graph-SLAM improved accuracy over Fast-SLAM.

In this project, RTAB-Map together with sensor information gathered from a Kinect-like RGB-D camera and Hokuyo laser are applied to obtain a 3-D point cloud and 2-D occupancy grid map for two simulated environments (the given kitchen scene and customized scene).

Real-Time Appearance-Based Mapping (RTAB-MAP) is an Open Source 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 visited 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.

3 MODEL CONFIGURATION

For RTAB-MAP algorithm, we need to provide a depth camera capabilities for our robot, thus we equipped it with OpenNI Kinnect Gazebo Plugin ($libgazebo\_ros\_openni\_kinect.so$).

In order to get the working $laser/scan$ topic for the $kitchen\_dining$ environment we also added the second $laser/scan$ by converting depth raw image into laser scan topic with the $depthimage\_to\_laserscan$ plugin.

The resulting TF frames for the robot are the following: figure

The robot model used in this project is based on base robot used in the previous project. The robot model chassis is made by a rectangular box of 0.4x0.2x0.1 meter with a cylinder on the front. The cylinder radius is 0.1 meters and 0.1 of length. Two wheels with a radius of 0.1 meters are located at both sides of the chassis, in the front side while a passive back caster is at the back.

Then , the robot is equipped with a laser range finder, as the previous model, and with a newly sensor such as RGB-D camera. The laser range finder was moved to the top of the robots neck. The RGB-D camera is located at the front of the chassis. The details of the TF of the robot is shown at Fig. 1.

4 WORLD CONSTRUCTION

Two worlds are used in this project. One is the provided one by Udacity called Kitchen dining world (Fig. 2) and the other one is the customized world.

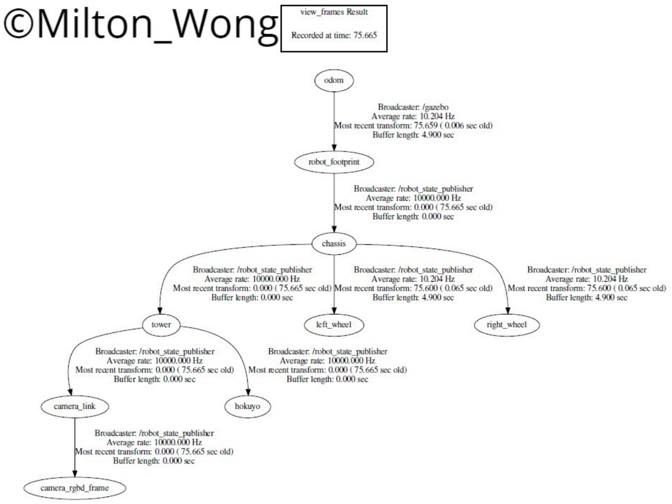

Fig. 1. Robot link summary diagram.

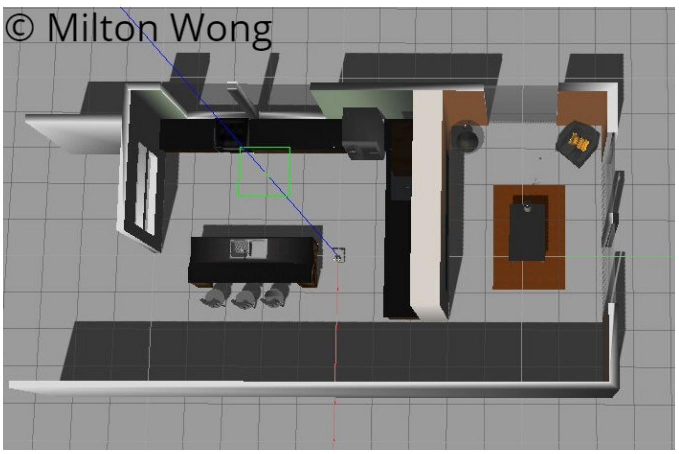

Fig. 2. Kitchen world environment.

The customized world is shown in Fig. 3. It applies one template available in Gazebo as base environment. The chairs, tables and walls are added to enrich the space . Moreover, pedestrians are also appended to increase the complexity of the scene.

5 Results

In order to simulate the project, it was necessary to launch world, mapping and Rviz files, and run the teleop script.

5.1 Kitchen dining world

Fig. 4 shows the 3D mapping results of SLAM. It was recorded by Rviz and saved in a database using RTAB-Map package. The robot was able to correctly map the environment, rebuilding almost everything with a great level of details. Several loop closures were performed during the simulation.

From the 2D map shown in Fig. 5, we can see where the kitchen, chairs and tables as well as the sofa in the corner of room.

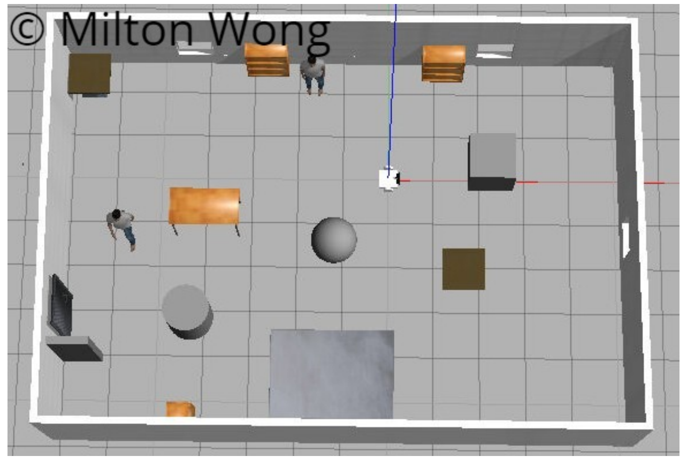

Fig. 3. Customized world environment.

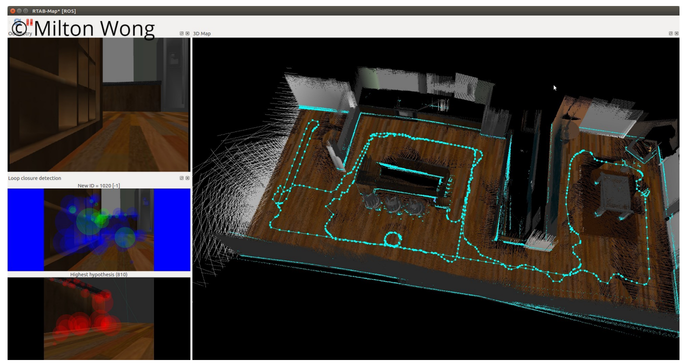

Fig. 4. 3D mapping results of RTAB-MAP for kitchen world environment.

5.2 Customized world

The results of SLAM using the customized map was successfully mapped using RTAB-MAP. Similar to kitchen dining case, 3D map was created with great details. The 2D map showed in Fig. 7 reflects the successful result.

5.3 Discussion

The performance of the robots mapping for each environment have a few similarities. Both were capable of clearly mapping 2D map boundaries, except for the unbounded section of the Kitchen Dining area. The missing piece unmapped on the right side is likely due to that side there are

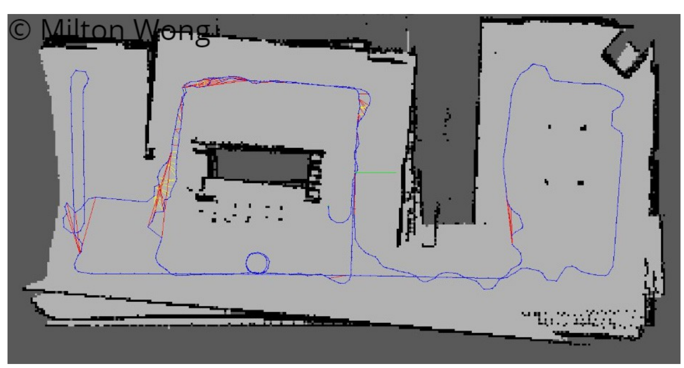

Fig. 5. 2D grid map for kitchen world environment.

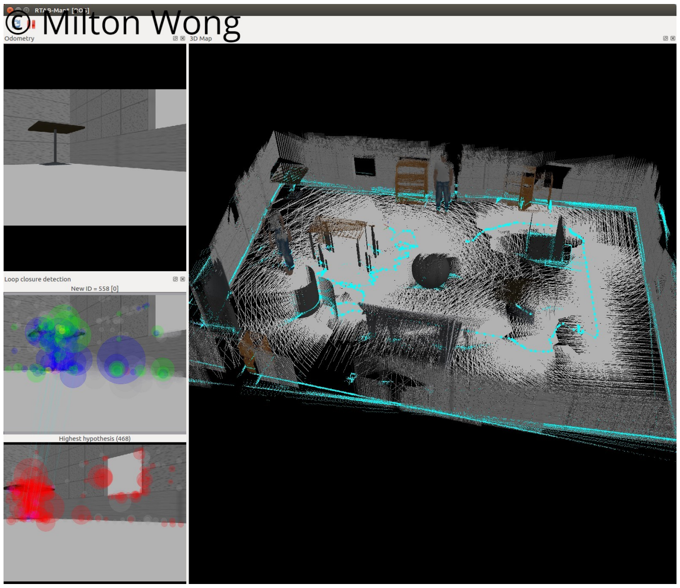

Fig. 6. 3D mapping results of RTAB-MAP for customized world environment.

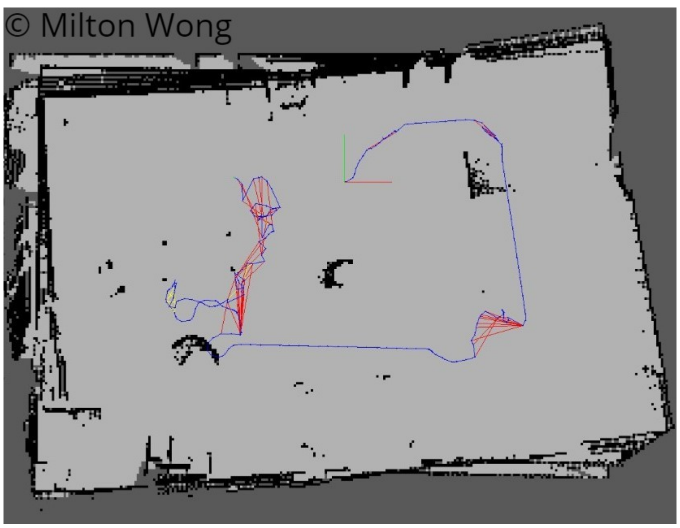

Fig. 7. 2D grid map for customized world environment.

windows which could affect the detection. Both worlds have been navigated via keyboard at very low velocity. Increasing the velocity produce mismatches in the 3d reconstruction. The results of the 3D maps are a slightly better in Kitchen dining than the customized world. More features in the provided map may be the reason of this difference.

6 Conclusion / Future Works

This paper has successfully demonstrated the use of appearance based GraphSLAM implemented with RTAB-Map in a selection of simulated environments. It is noted that performance may be improved by adding an additional RGBD camera in another orientation (e.g. rear-facing).

It should also be possible to extend the results to accomplish a number of tasks of interest. For example, RTAB-MAP can be extended to recognize a target object and navigate toward it, which is a classical mobile robotics task that would be applicable in a number of practical scenarios. Furthermore, it should also be possible to implement the demonstrated software stack in hardware (e.g. an actual mobile robot). This would be an interesting demonstration of GraphSLAM in the real world.

References

[1] M. Labbe and F. Michaud, "Appearance-based loop closure detection for online large-scale and long-term operation," *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 734–745, 2013.

[2] G. Grisetti, R. Kummerle, C. Stachniss, and W. Burgard, "A tutorial on graph-based slam," *IEEE Intelligent Transportation Systems Magazine*, vol. 2, no. 4, pp. 31–43, 2010.

[3] M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit, *et al.*, "Fastslam: A factored solution to the simultaneous localization and mapping problem," *Aaai/iaai*, vol. 593598, 2002.