Map My World Robot Project for Udacity's Robotics Software Engineer Program

Leroy Friesenhahn

***Abstract*—The Map My World Robot projects evaluate the effectiveness in using SLAM (Simultaneous Localization and Mapping) technique in mapping the robot's environment. It was found that RTAB-Map may be the best solution for implementing SLAM due to it speed and memory requirements. The previous created robot is used to generate a 2D occupancy grid and 3D octomap of two environments.**

Project one will use a supplied environment. The previous project robot is upgraded sensors to supply the necessary messages for the RTAB-Map. The robot is teleoped around the room to generate the map.

Project two uses the same robot as in project one in a individual created environment. The same technique is used to generate the robot's customized environment map.

INTRODUCTION

A Robot new to an environment or in a changing environment must have a way of navigating its environment. Although maps can be generated for a given location, these maps can quickly become outdated dues to changes in the environment. This requires the robot to be able to generate its own map as it transverse it environment.

There are many different techniques for robot generating maps. This project will use a SLAM technique called RTAB-Map (Real Time Appearance Based Mapping). A robot is able to map its environment as it travels through it surroundings. The information obtained is used to allow the robot to generate a map to use in navigation.

This project will map a given and and created simulated environment to create a 2D occupancy map and a 3D octomap using the SLAM technique known as RTAB-Map. The goal of the project is to map the environment(s) in the least number of passes with at least 3 closures.

Rtabmapviz is used to provide a visual representation of the mapping in real time. The generated map is stored in a local database that can be viewed at a later time using rtabmap-database Viewer.

BACKGROUND/FORMATION

Acquiring information to generate a map is a difficult process. The challenges exist in the following areas:

Size - the larger the size of the environment in respect to the perceptual range of the robot, the larger the challenge in generating the map

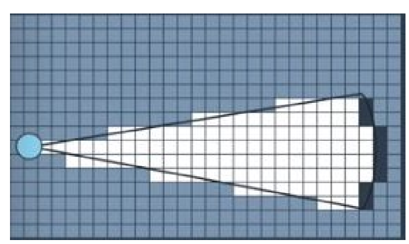
Noise – Sensor have noise. This adds to the complexity of generating a map

Perceptual ambiguity - Similar looking places visit by the robot at different times present the problem of distinguishing these locations

Cycles – Returning to a given points from multiples paths (not necessarily a straight line) present a problem for the robot to adjust. (odometry reading may differ greatly)

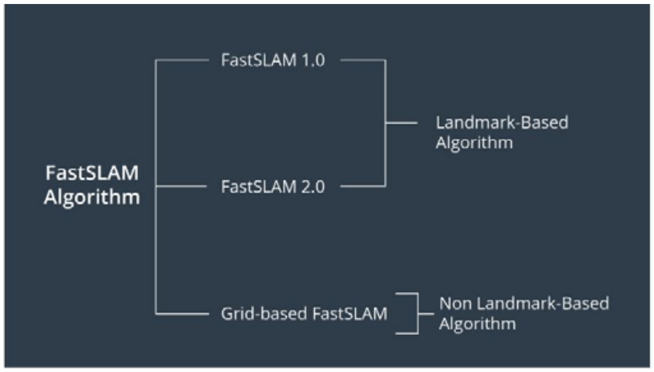
The Udacity Software Robotics program introduces the following mapping methods:

Occupancy Grid Mapping - Each grid cell of a Occupancy Grid Mapping is identified as Unknown/Undiscovered Zone, Free Zone, or Occupied.



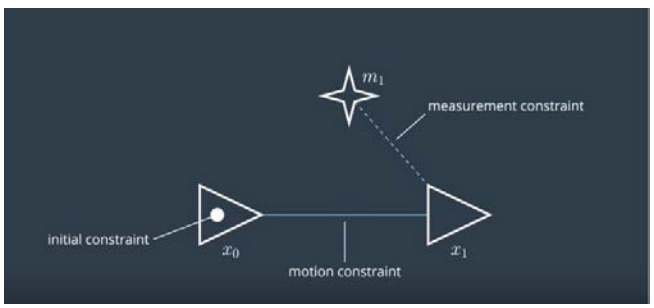
Field of a sensor used in Occupancy Grid Mapping

Grid based FastSLAM - FastSLAM(SLAM - Synchronized Location and Mapping) algorithm uses particle filter approach to solver the SLAM problem. Each particle keeps a guess of the robot trajectory. This reduces the mapping to known poses. With only known poses, it is limited to processing landmark based environments. The Grid Based FastSLAM algorithm incorporates SLAM with the MCL (Monte Carlo Localization) method and the Occupancy Grid Mapping. Incorporating Grid Based Mapping solves the problem of needing predefined landmarks.



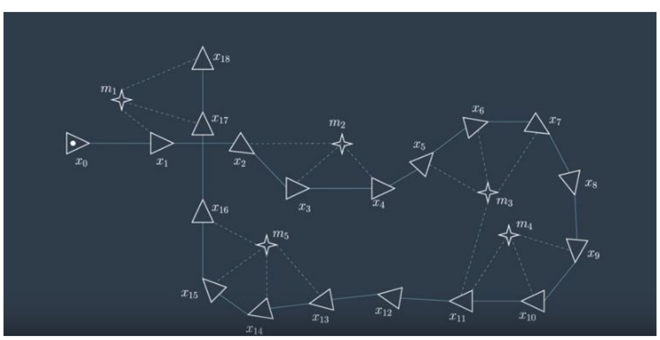
Grid based FastSLAM advantage for FastSLAM

Graph-SLAM - Graph-SLAM uses a graph based technique consisting of poses, features, motion constraints(between two poses) and measurement constraints(connection between a feature and a pose) to generate a environment map. Graph-SLAM represents poses and features as shown.



GraphSLAM poses, features, motion constraint and measurement constraint

As the robot moves and explores its environment, the map is updated.



Updated GraphSLAM map

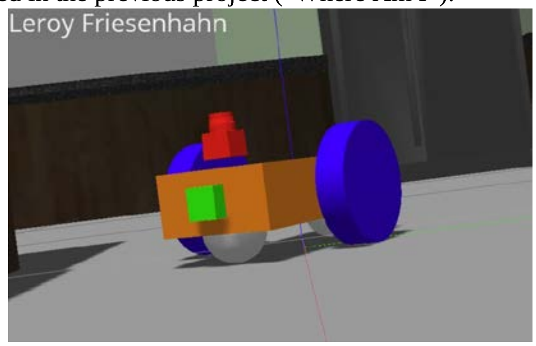
RTAB-MAP - RTAB-MAP (Real Time Appearance Based) mapping based on the GraphSLAM method. Using vision sensors, data is collected to localize the robot and map the environment. A concept called loop closure is used to determine if the robot has seen this location previously. Loop closure with Visual Bag of Words provides optimization for the process.

Loop closure detection uses working memory to limit the number of images being searched. Working and long term memory is used to decrease the search time.

RTAB-Map produces 2d occupancy grid map, 3doctomap, or 3D point cloud.

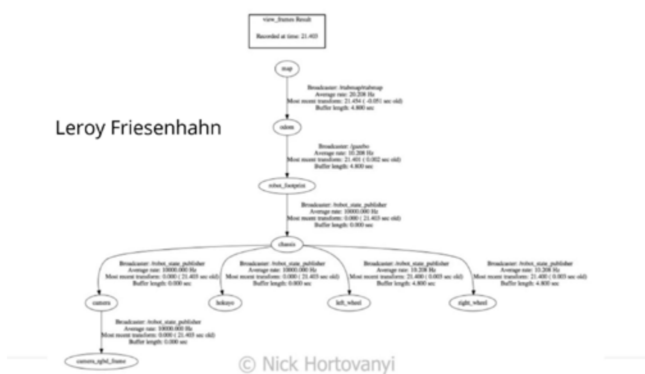
ROBOT CONFIGURATION

THE robot used for this project was based on the robot used in the previous project (“Where Am I”).



Map My World Robot

A few modifications were made to accommodate the mapping data need for this project. The main upgrade was replacing the camera with an RGB-D sensor. The RGB-D sensor measures depth which is need in the RTAB-Map method. A joint was added to assist in rotating the RGB-D sensor 180 degrees. The new joint compensated for the problem of RGB-D point cloud pointing up.



Visulization of the robot frame

SCENE(S)

THE simulated robot environment(s) consisted of a supplied one - kitchen_dining.world and a customized

Kitchen_dining.world

one - customized_world.world.

.

Customized created world

The customized world was created using gazebo. A café model was selected. Additional tables, fountain, fire hydrant, and mailboxes were added to the scene.

.

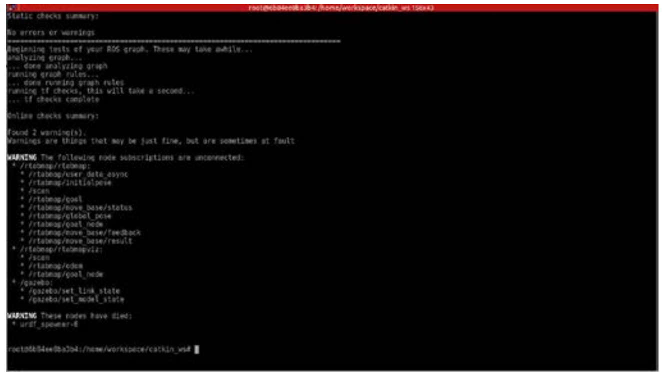
RESULTS

There were no results from this project. The program was unable to map the environment at this time. After more than 40+ hours and no support from Udacity, the project was halted until additional assistance can be obtained.

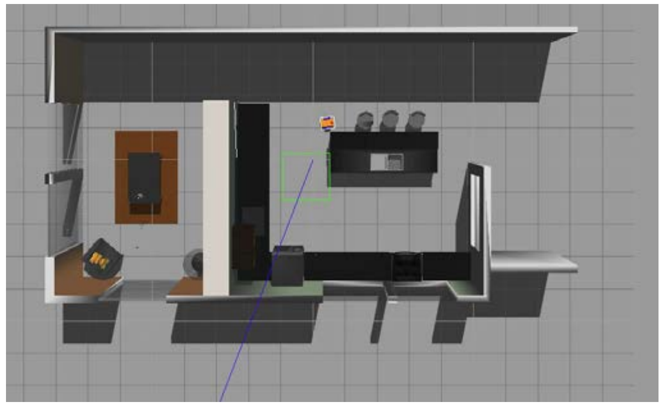
DISCUSSION

The program was completely developed and executed with no errors. The rtab_run procedure was executed and appeared to run correctly except for the mapping.launch procedure. The mapping.launch procedure included command to invoke the rtabmpviz procedure. Debugging reports were run without visual error conditions (at lease to the inexperienced individual.

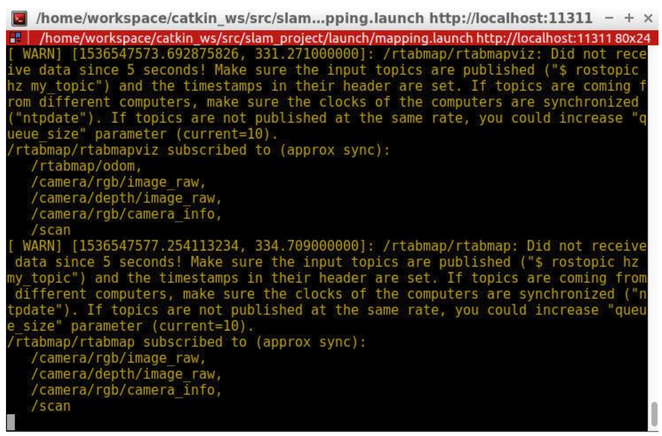
Output of rqt_graph



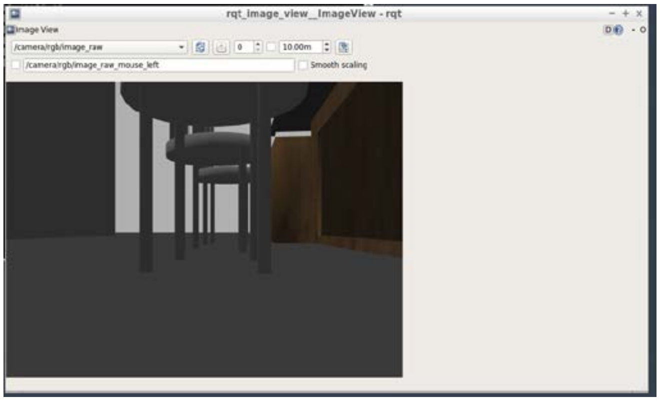
Output from roswtf



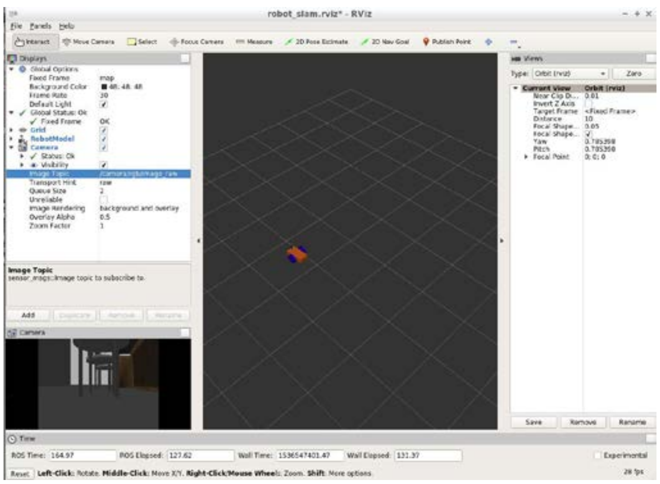
Output from gazebo



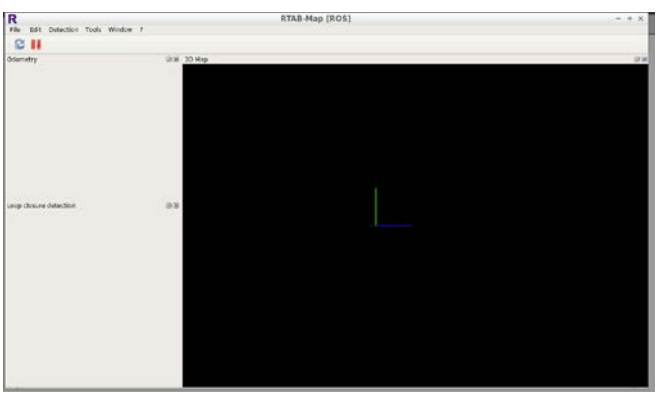
Output from the mapping procedure



Output from rqt_image_view



Rviz output – camera view lower left corner



RTAB-Map Output

Conclusion/Future Work

The project is close to being completed. A incorrect parameter, procedure is preventing the correct results. It is sincerely believed support from the reviewer could bring a successful conclusion to this project. Request for assistance has been unanswered and questions posted by other students have been unanswered for days, weeks and even months. With only 14 hours left on my Udacity's Workspace, addition help is need to successfully complete this course.