SLAM Project

1 Abstract

Mapping of unknown environments is one of the most challenging problem in robotics. In this project, A Simultaneous Localization and Mapping or SLAM algorithm is used to update the environment's map, Mapping, and keep tracking the location of the robot, Localization.

2 Introduction

2D mapping becomes unreliable with the increase of the complexity of environment types that robot will be in. GraphSLAM is a Simultaneous localization and mapping algorithm which uses sparse information matrices produced by generating a factor graph of observation interdependencies (two observations are related if they contain data about the same landmark) [1]. In this project, RTAP-MAP graphSLAM algorithm is used. RTAP-MAP is using a technique called loop closure to determine either the place has been seen before or not during the navigation.

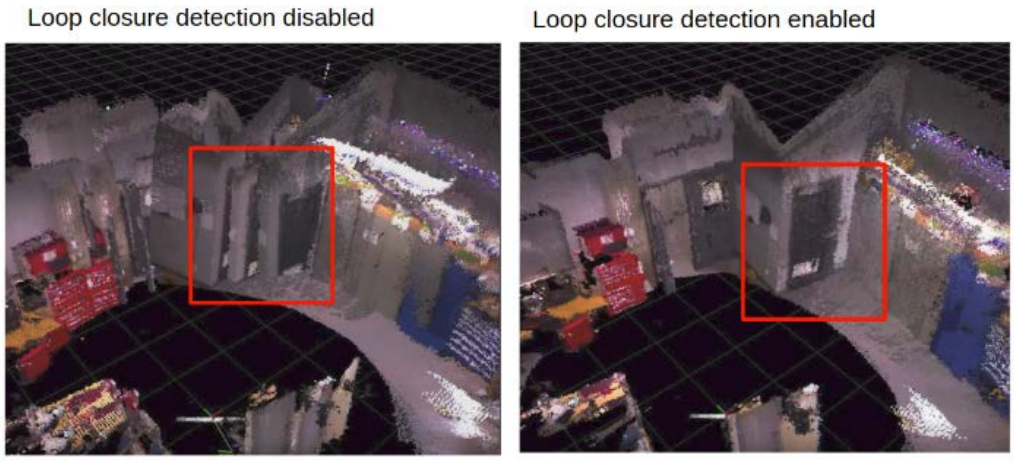
3 Background

SLAM is very active research area in robotics. It has many approaches to attack the mapping and localization problem.

EKF SLAM is a class of algorithms which utilizes the extended Kalman filter (EKF) for (SLAM). Typically, EKF SLAM algorithms are feature based, and use the maximum likelihood algorithm for data association. Associated with the EKF is the gaussian noise assumption, which significantly impairs EKF SLAM's ability to deal with uncertainty. With greater amount of uncertainty in the posterior, the linearization in the EKF fails [2].

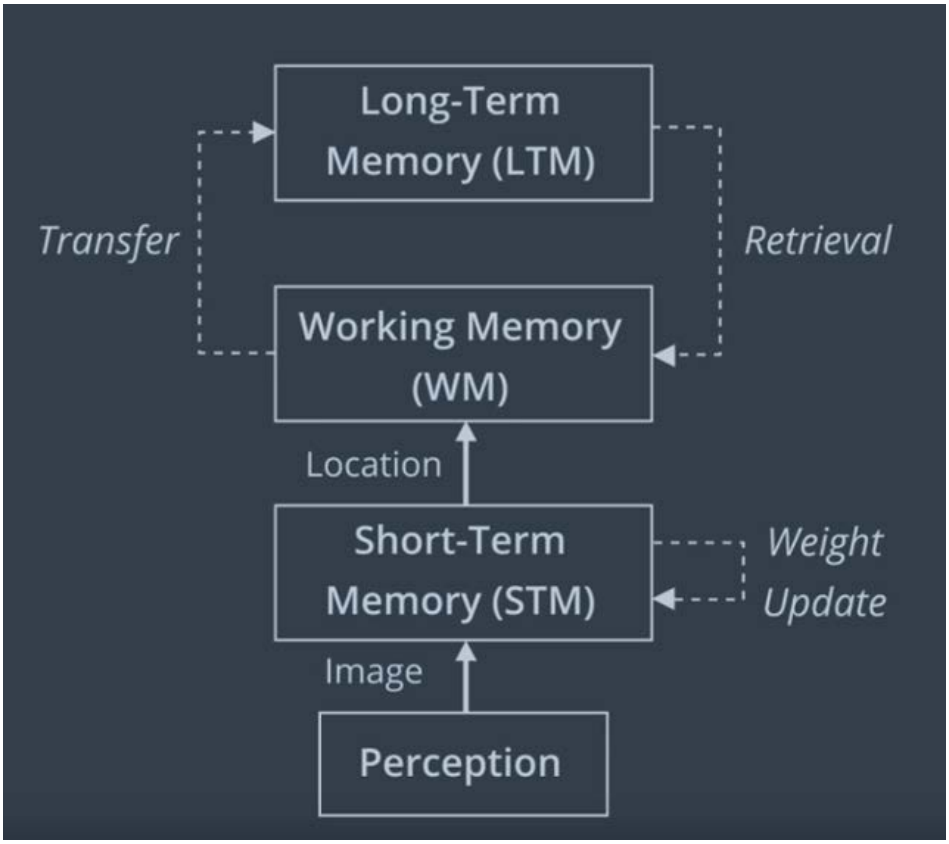
FastSLAM is a SLAM algorithm that integrates particle filters and extended Kalman filters. More specifically, correlations in the uncertainty among different map features arise only through robot pose uncertainty [3].

. RTAP-MAP is a GraphSLAM algorithm is using a technique called loop closure to determine either the place has been seen before or not during the navigation.



Loop closure is using a bag-of-words approach to determine the features of the place and determine by comparison these features with previous one either the robot have seen this place before or not. Because of the increase of overall captured images and features, a special memory management technique is used.

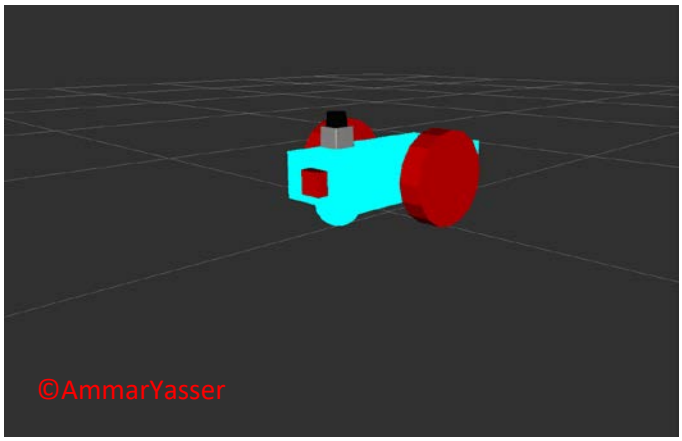
RTAB-Map uses a memory management technique to limit the number of locations considered as candidates during loop closure detection. This technique is a key feature of RTAB-Map and allows for loop closure to be done in real time. The overall strategy is to keep the most recent and frequently observed locations in the robot's **Working Memory (WM)**, and transfer the others into **Long-Term Memory (LTM)**.

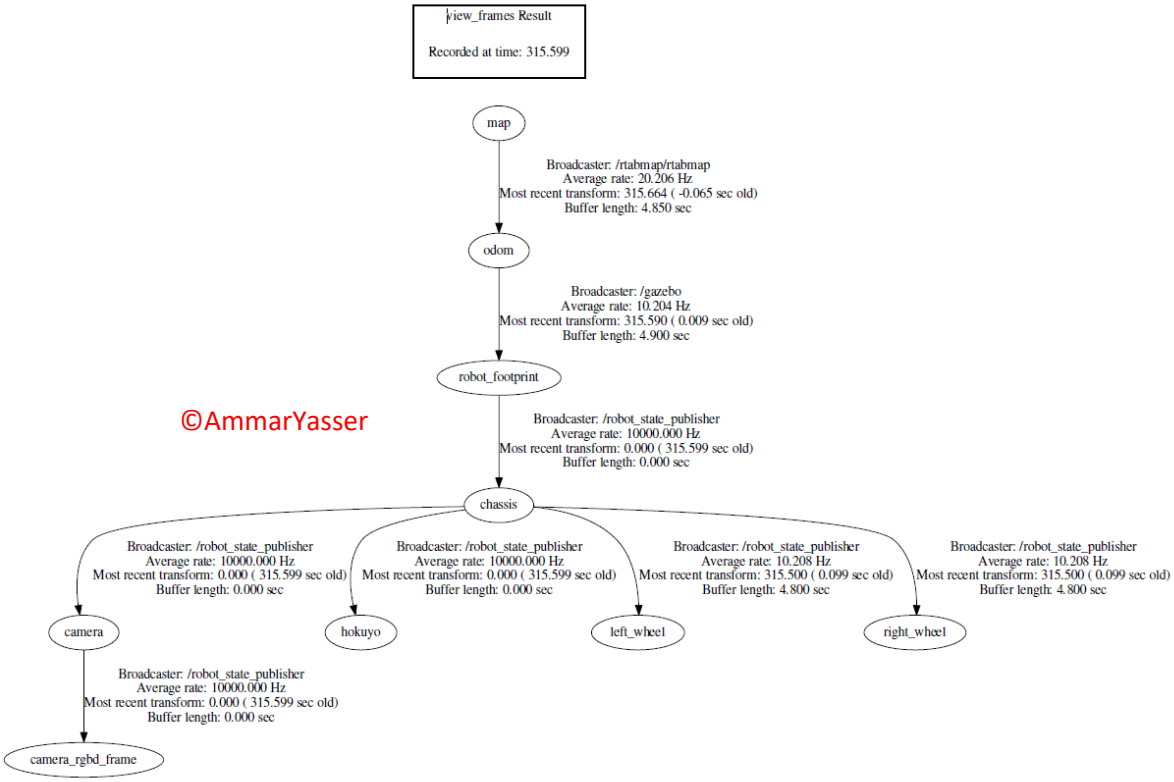


4 Robot Model

The robot is built in a URDF file and integrated to the gazebo file to simulate it physically.
The robot mainly has two differential wheels and a drive roller. The robot has two integrated sensors:
a. 2D Laser scanner (Hokuyo).
b. RGB-D Camera.

the following is TF frames





4.1 Ros Packages

The project uses mainly four packages

1- Gazebo: Used for the simulation of the robot and environment

2- RViz: Visualization of the map and sensors outputs

3- Rtabmap_ros: For SLAM algorithm

4- Teleop: for controlling robot movement

Every package has been launched separately using following launch files:

-world,launch (Gazebo and some times RViz)

-rviz.launch (RViz)

-teleop.launch (Teleop)

-mapping (Rtabmap_ros and map visualization)

4.2 Gazebo plugins

For the special types of the camera, laser and the robot wheel, the following plugins are added to gazebo

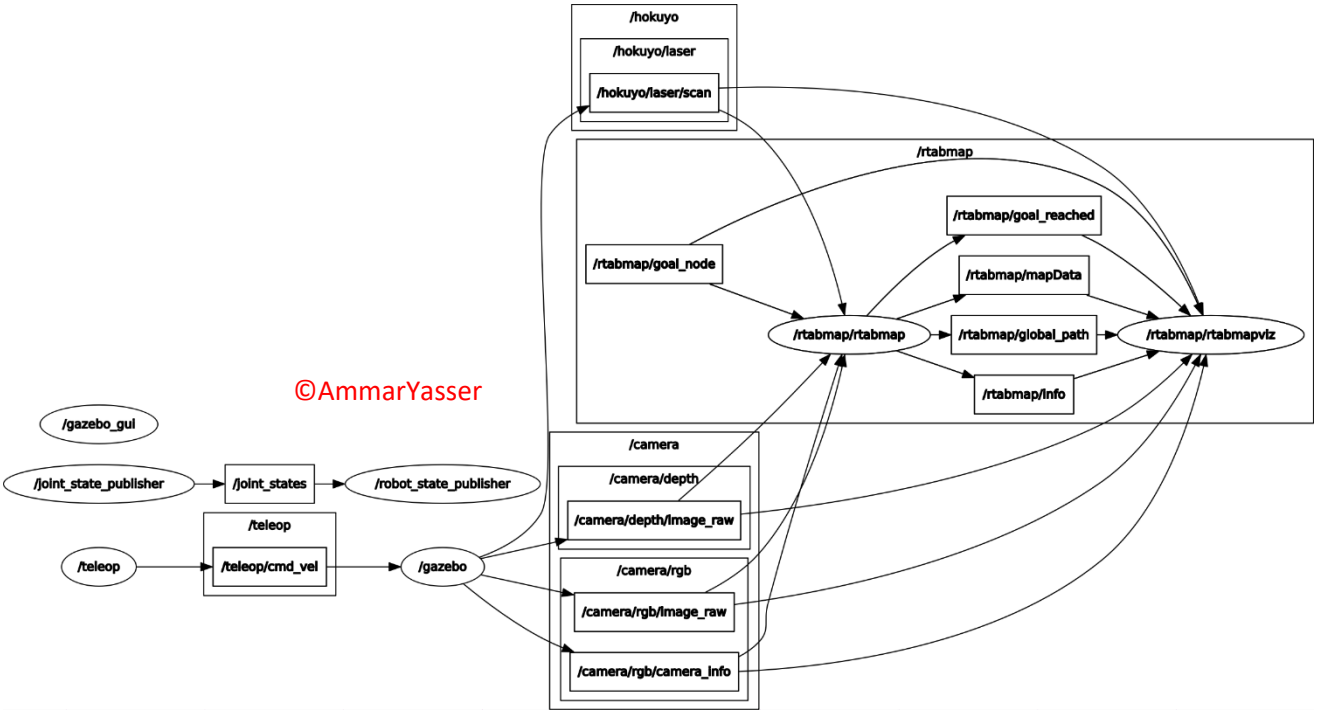
libgazebo_ros_diff_drive.so (for the wheels drivers)

libgazebo_ros_openni_kinect.so (for the camera)

libgazebo_ros_laser.so (for the laser scanner)

4.3 Nodes and Topics

The following rqt_graph shows the relation between the nodes

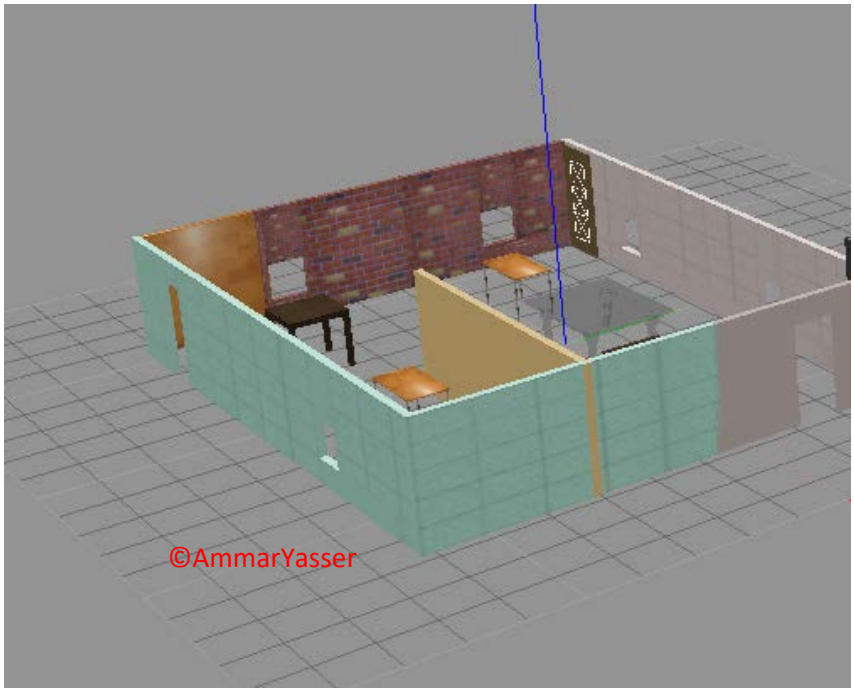


4.4 worlds

3 worlds are tested for the project,

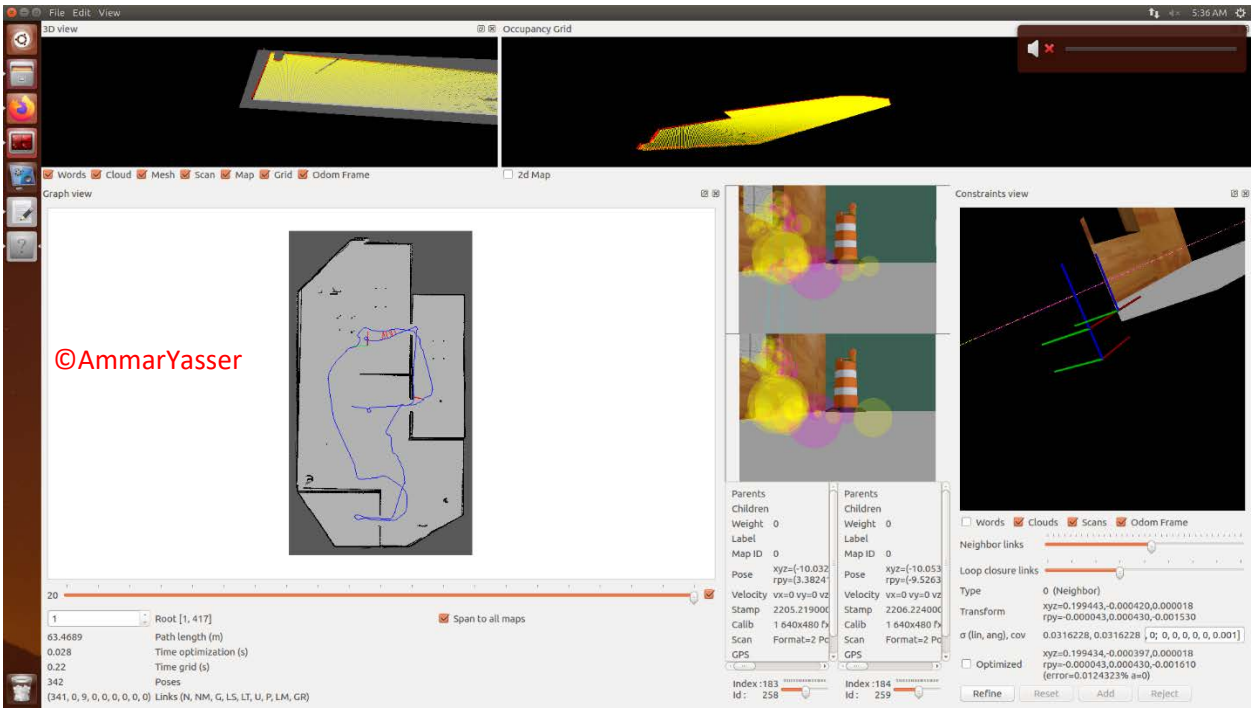
1- Kitchen world (given but have some visualization problems)

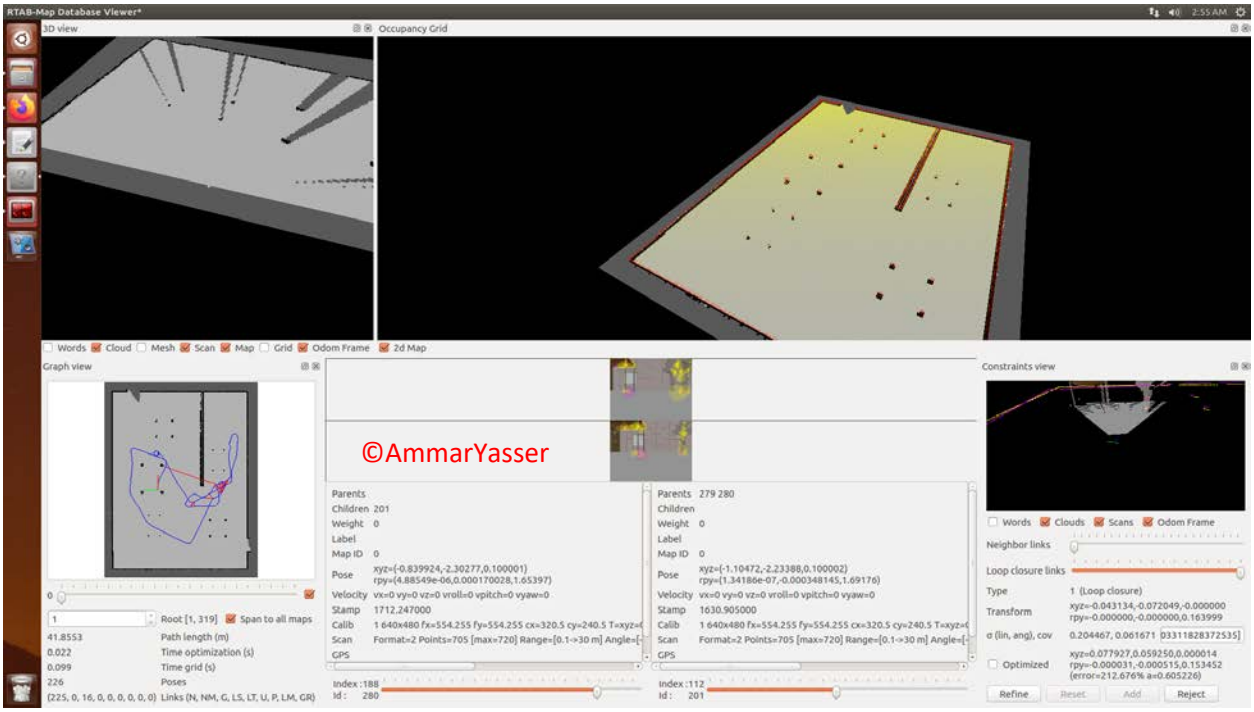
2 built worlds

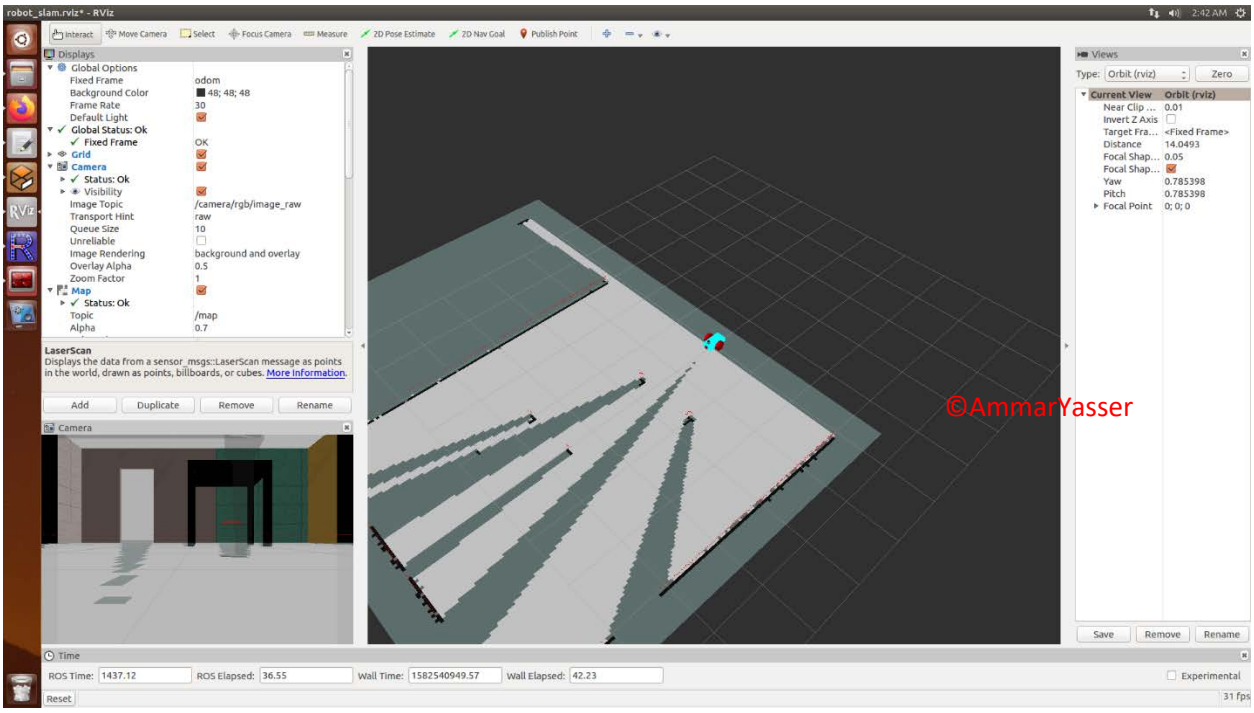


5 Results

The package results is very good for both maps and loop closures,



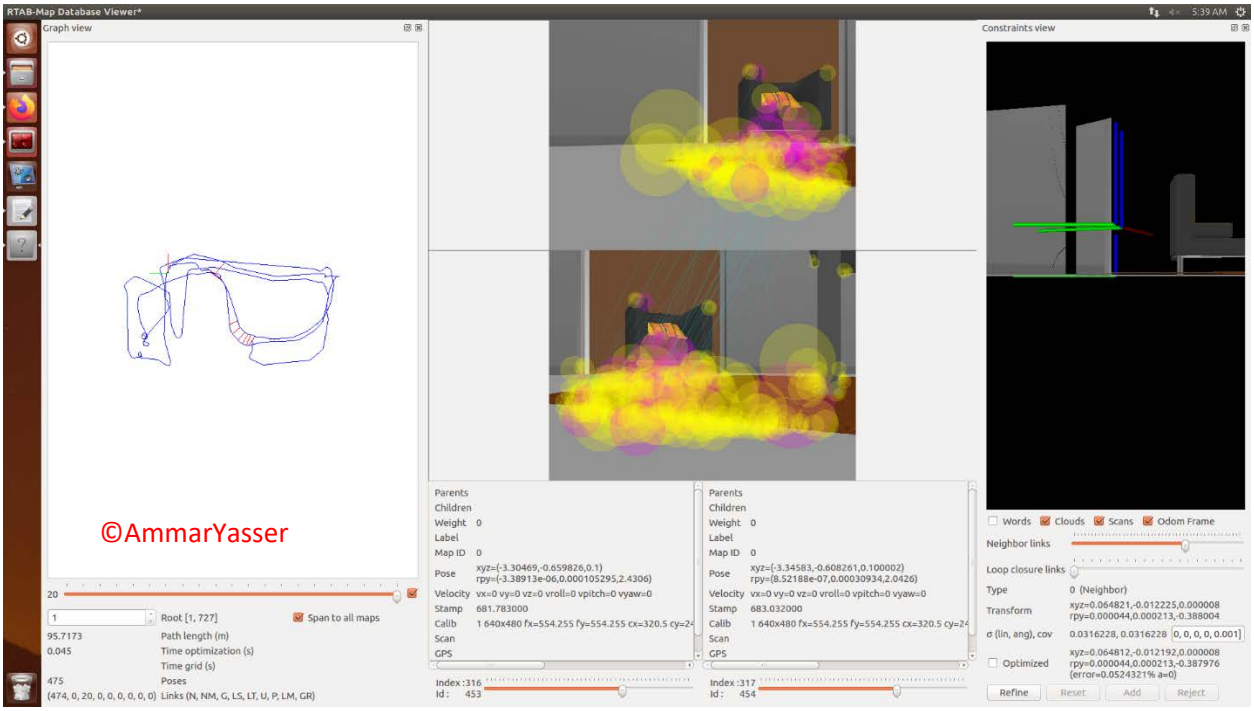




Some problems with kitchen world

The laser sensor doesn't interact with the environment

So, the map obstacles don't appear



6 Discussion

The overall results are very good. Including that Rtabmap-SALM algorithm is very powerful to attack the mapping problem in the 3D and 2D approaches.

RtabMap requires more computation than other but using a GPU accelerator or a Jetson TX2 platform could solve this problem.

7 Future work

- Try to Fix kitchen world
- Improve the performance of the Rtabmap using the parameters.
- In real world, deploy the algorithm on Jetson platform and test it with real world.
- Improvement with appearance of the robot model
- Exclude the laser scanner by the depth image of the camera, or mixing them to find the best measurements.