# The SLAM Algorithm with RTAB-Map

Arda Kaya

***Abstract*— In this paper mainly used algorithms such as Grid-based FastSLAM and GraphSLAM will be discussed and one of the GraphSLAM method which is RtabMap will be demonstrated. In these demonstrations a differential controlled robot with RGB-d and Hokuyo sensor, will be simultaneously mapping its environment and localizing itself. The robot will be tested in two different simulated environments in Gazebo.**

## I. INTRODUCTION

Localisation is very important in robotic projects as most of the operations builds on this aspect. When the robot uses already known map to estimate its own position it is a localisation problem. However when the robot needs to operate in unknown environment, it needs to build its own map , therefore it can adapt new environment simultaneously and decides then actuates. In Robotics this process is called Mapping. In this paper, there are two environments mapped by same robot in simulation environment by using SLAM algorithm with RTAP-Map .

## II. BACKGROUND

The SLAM algorithm estimates robots poses and simultaneously creates a map of its environment at the same time. If we seperate the SLAM algorithm into two subsystems, the first part does the estimation of robot's pose, then the second part uses these estimations to create 2d or 3d map and these two subsystems repeats as a loop. One of the biggest advantages of the SLAM algorithm according to the Kalman and the Monte Carlo algorithms(MCL) is the SLAM algorithm uses all previous estimations while making new estimation. The Kalman and the MCL algorithms uses only last estimation to make new estimation for position. The most common used SLAM algorithms are: the Grid-based FastSLAM and the GraphSLAM.

## III. GRID-BASED FASTSLAM

FastSLAM uses custom Particle Filter approach to solve the full SLAM problem with known correspondences. It estimates a posterior over the trajectory using a particle filter approach. And FastSLAM uses low dimensional Extended Kalman Filter to solve independent features of the map which are modelled with local Gaussian. However, this landmark based solution does not work in this papers simulation scenarios which has no landmarks. Grid-based FastSLAM uses occupancy grid maps instead of predefined landmarks, therefore it works in unknown environments. Grid-based extended version of FastSLAM algorithm firstly estimates the robots trajectory by using the Monte Carlo Localisation. Secondly, it estimates the map by assuming known poses and using the occupancy grid mapping algorithm.

## IV. GRAPHSLAM

GraphSLAM is a SLAM algorithm that solves full SLAM problem. Algorithm covers the entire path and map, instead of most recent pose and map. This algorithm simultaneously creates a graph that has nodes as a reference of landmarks and robots location. There are flexible connections between these nodes which represents measurements and algorithm tries to find the best configuration of these nodes to minimise the error. In this paper, RTAB-Map will be used. Real-Time Appearance-Based Mapping(RTAB-Map) is a RGB-D Graph-Based SLAM method . You can see the outlines of this method in Fig-1.

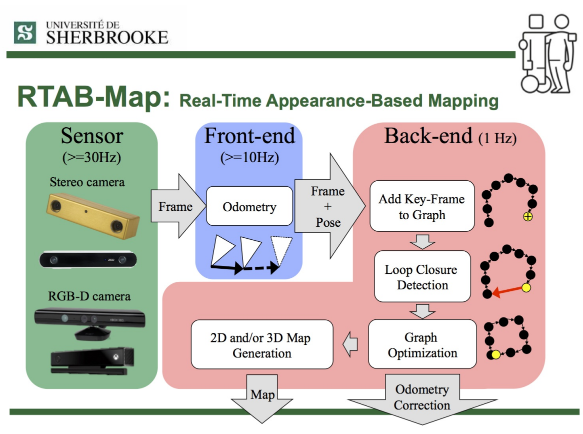


Fig. 1. RTAB-Map Diagram

## V. SIMULATION

Gazebo is used as a simulator. The first map called kitchen in simulation environment is provided by Udacity. You can download from this link. The second map's initial version was Gazebo's preset called Cafe. The model is updated by adding some tables and an human model. You can download cafe maps world file from this link.

## VI. ROBOT

Robot model is a differential wheeled robot. Its chassis size is 40, 20, 10 cm and wheels radius are 10 cm. RGB-D sensor is positioned in front of chassis. And the Hokuyo sensor is positioned on top of the chassis. You can download the Robot models xacro file from this link. You can find more details about the Robot in fig.2 and there is a high resolution version of view-frames result in this link.

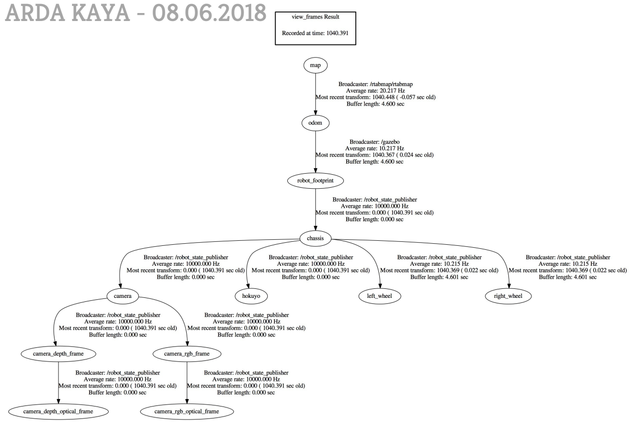


Fig. 2. TF Frame of the Robot

# VII. Results

## A. *Provided World*

When the scene initialized robot starts mapping 2d and 3d. It is controlled by teleop package. You can see the result in fig.3 and fig.4.

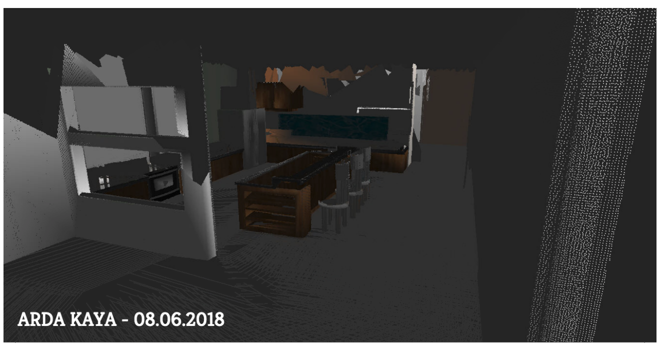


Fig. 3. Generated Point Cloud from Provided World

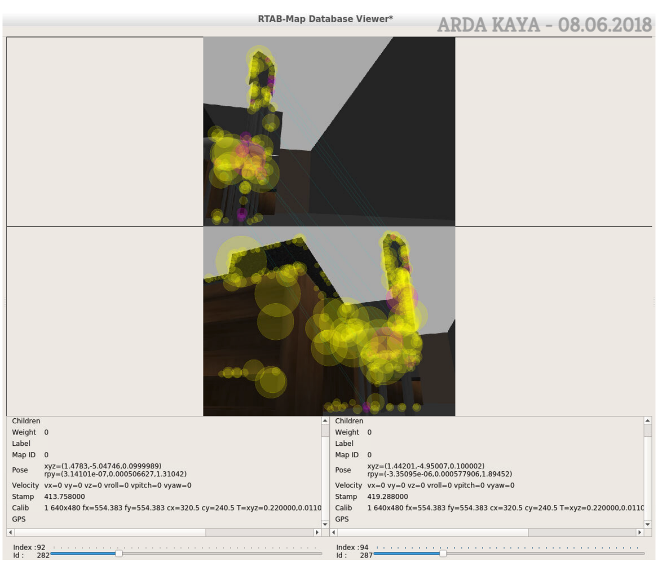


Fig. 4. RTAB-Map Database Viewer Result from Provided World

## B. *Cafe World*

When the scene initialized robot starts mapping 2d and 3d. It is controlled by teleop package. You can see the result in fig.5 and fig.6.


Fig. 5. Mapping Progress in Rviz from Cafe World


Fig. 6. Generated Point Cloud from Cafe World

# VIII. Discussion

In both environments, when the robot does circular repeating motion mapping error becomes higher. Therefore, the robot's path while mapping kept more straight instead of circular repeating motion.

# IX. Future Work

For further development , RTAB-Map will be adapted to a drone in simulation environment.