COMPSYS 726 Project 1 Report

1. Literature review and justification

1.1 Literature review

For the robot to autonomously navigate by itself without human intervention, 3 things are needed which are mapping, localising, and path planning. While the more ambitious goal of autonomous exploration requires choosing and planning paths to the goals in order to automatically cover the environment to be mapped, SLAM puts aside the complexity of the planning step during map building and assumes the robot is guided in prefixed trajectories or by teleoperation. [1] SLAM is the process by which a mobile robot can build a map of an environment and at the same time use this map to compute its own location. [2] Although there are many SLAM methods, after researching both 2D and 3D slam, I decided to use one of the 3D slam methods due to two reasons. Firstly, some methods can perform loop closure. If loop closure is used, as the system moves through space and builds a model of its environment, it will continue to accumulate measurement errors and sensor drift, which will be reflected in the map being generated. [4] Secondly, re-localisation can be performed. Re-localization occurs when a system loses tracking (or is initialised in a new environment), and needs to estimate its location based on currently observable features. [4] If the system is able to match the features it observes against the available map, it will localise itself to the corresponding pose in the map, and continue the SLAM process. [4] I used RTAB-Map, a 3D SLAM approach that will be covered in more detail in the following section.

1.2 SLAM method justification

As mentioned above, in this project, I used Real-Time Appearance-Based Mapping (RTAB-Map) that uses RGB-D (colour and depth) sensors for mapping the environment in real-time and localising. My main motivation for choosing this algorithm was to enable navigation in dynamic environments with moving obstacles. Although I was not able to complete the dynamic environment task within the project's due date, I plan to explore it further after the project. There were other factors that led me to choose this SLAM method in addition to tasks in a dynamic environment.

First of all, RTAB-Map offers loop closure detection, which is useful in both static and dynamic environments. The loop closure occurs when the system recognises that it is revisiting a previously mapped area, and connects previously unconnected parts of the map into a loop, correcting the accumulated errors in the map and improving the accuracy of the map. [3] Also, loop closure detection helps to reduce the cumulative error of the robot's estimated pose, effectively increases the accuracy of the results, and reduces the drift over time. [5] [6] Secondly, it uses a memory management approach, which ensures the satisfaction of real-time constraints independently of the scale of the mapped environment. [7] Moreover, RTAM- Map offers precise odometry estimation by combining the LiDAR data, cameras, and other exteroceptive sensors through sensor fusion. [8] This enables more precise and reliable mapping in static environments where many sensor modalities can complement each other's advantages and disadvantages. Researchers have also compared various SLAM methods and found that RTAB map can be considered as one of the best methods for odometry in indoor environments [9], due to its strong localization capabilities, precise motion control, and real-time performance.

2. Project description

2.1 Project Design

The RTAB-Map package is used for this project. (http://wiki.ros.org/rtabmap_ros) RTAB-Map includes an autonomous navigation function using RGBD SLAM, eliminating the need for additional navigation tools like AMCL to safely navigate the robot from one location to another without collisions. In this project, I designed a new simulation world in Gazebo that consisted of various obstacles and walls. Below figures are Gazebo simulation world and maps produced by RTAB-Map. RTAB-Map generates 3D point clouds of the environment and a 2D occupancy grid map,which were designed to construct an environment using a probabilistic representation of spatial information [4] for navigation.

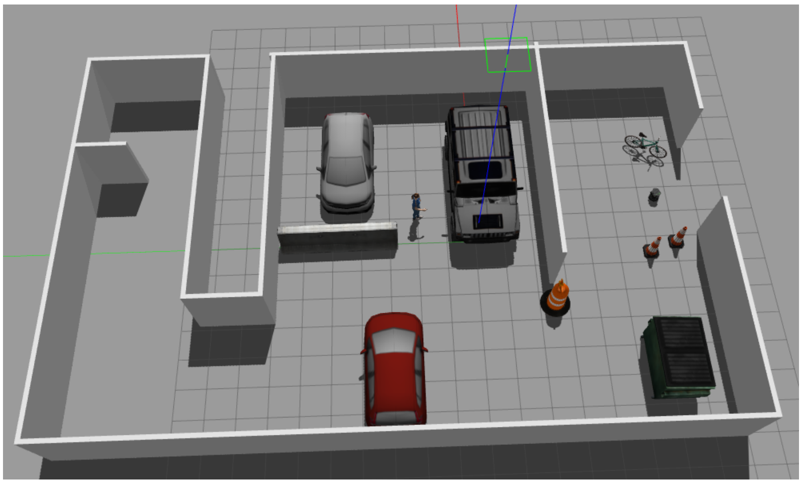

Fig 1. Gazebo simulation world

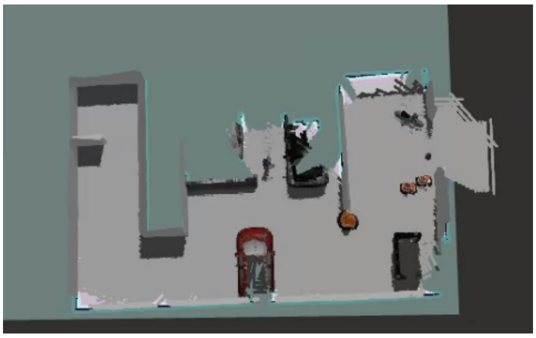

Fig 2. 3D map

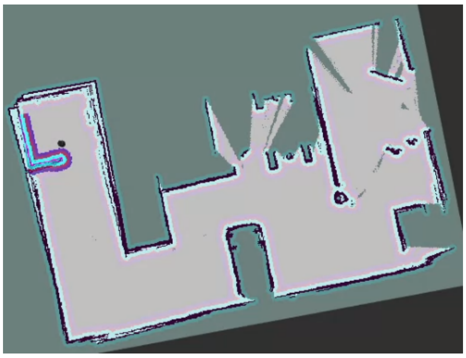

Fig 3. 2D map

LaserScan data is used to read LIDAR information and detect the distance between the robot and obstacles. For the image displayed in RViz and camera, the topic camera/rgb/image_raw is used. The map is set to the topic /map, the Global map is set to /move_base/global_costmap/costmap, and the Local map is set to /move_base/local_costmap/costmap. RTAB-Map performs 3D mapping with the cloud set to /rtabmap/mapData. For more detail, the figure below which is rqt_graph shows the currently running nodes and their connections (topics) between them.

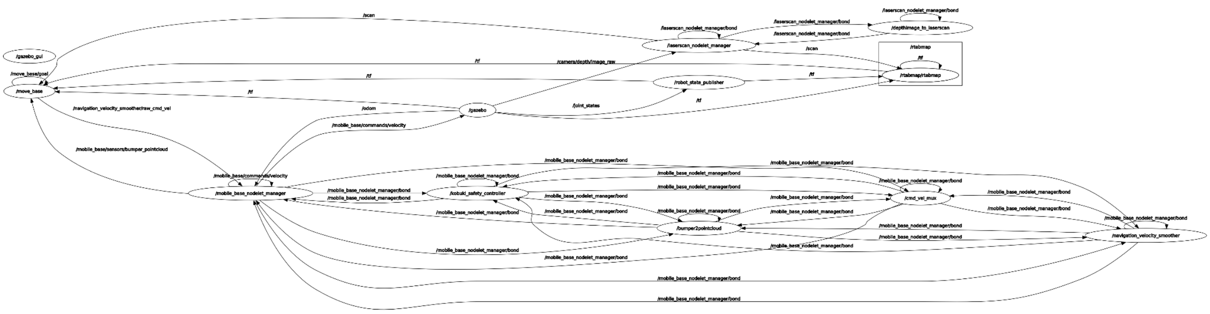

Fig 4. Rqt_graph

2.2 Evaluation and limitation

The approach I used for this project performs well when the mapping is accurate and without any distortions. However, I have observed some cases where the mapping process gets distorted, due to the robot jumping to another location or possibly the map moving. As a result, the robot took longer to figure out the correct path after detecting some obstacles. In my tests with a relatively less distorted map (similar to the demo video), the robot successfully navigated in 4 out of 5 trials without rotating in the same location to find the path after obstacle detection. However, in one instance, when the robot approached an obstacle, it seemed to struggle in determining the correct path to the destination point and rotated 2-3 times. I believe that the distortion (even if it's small) in the map has impacted the navigation performance. This was further supported by successful navigation in tests where the map was created for a shorter duration with less obstacles and thus had no distortion at all. In addition to RTAB-Map, I also experimented with gmapping, which is a famous grid-based FastSLAM and an accurate solution for 2D Mapping [10]. The key advantage of FastSLAM is that data association decisions can be made on a per-particle basis which leads to efficient and scalable estimation of the robot's pose and map. [11] However, I noticed that the map accuracy of gmapping was significantly lower compared to RTAB-Map in multiple attempts. The performance of the navigation system implemented by AMCL was directly impacted by the poor precision of the map produced by gmapping. The navigation performance was negatively affected and did not perform well.

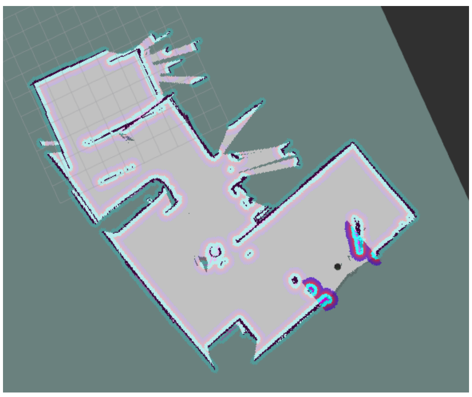

Fig 5. Map produced by gmapping

Since RTAB-Map is less robust and less computationally efficient [12], in handling large-scale areas with a lot of landmarks or features, RTAB-Map may run into difficulties. The computational demands for mapping and loop closure detection may grow as the map size does as well, which may have an effect on memory usage and real-time performance.

Reference

[1] Analysis of a RGB-D SLAM system using Real-Time Appearance-Based Mapping on the Kbot robot Jon Ander Ruiz 2022

[2] Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms, Hugh Durrant-Whyte, Fellow, IEEE, and Tim Bailey

[3] G. Grisetti, R. Kümmerle, C. Stachniss, and W. Burgard, “A tutorial on graph-based

slam,” IEEE Intelligent Transportation Systems Magazine, vol. 2, no. 4, pp. 31–43,

2010.

[4] 3D lidar SLAM: The Basics, Kudan, 2022

[5] Research of SLAM for static environment based on Kinect, Hong-Yan Chena and Ya-Jun Zhang, 2016

[6] Role of Deep Learning in Loop Closure Detection for Visual and Lidar SLAM: A survey, Saba Arshad and Gon-Woo Kim, 2021

[7] Memory Management for Real-Time Appearance-Based Loop Closure Detection, Mathieu Labb´e and Franc¸ois Michaud

[8] RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation: LABBÉ and MICHAUD

[9] Comparison of Various SLAM Systems for Mobile Robot in an Indoor Environment, Maksim Filipenko, Ilya Afanasyev, 2018

[10] Experimental Evaluation of ROS Compatible SLAM Algorithms for RGB-D Sensors, Bruno M. F. da Silva1, Rodrigo S. Xavier1, Tiago P. do Nascimento2and Luiz M. G. Gonçalves3,2017

[11] FastSLAM: An Efficient Solution to the Simultaneous Localization And Mapping Problem with Unknown Data Association, Sebastian Thrun1, Michael Montemerlo1, Daphne Koller1, Ben Wegbreit1, Juan Nieto2, and Eduardo Nebot2

[12] Robust and Scalable Visual Simultaneous Localization and Mapping in Indoor Environments using RGBD Cameras, Jan Frost, M.Sc., 2016