ISSN: 18158129 E-ISSN: 18151027
Volume 18, Issue 06, June, 2022

Application of SLAM & UWB for self-propelled robots in agricultural production and harvesting

Nguyen Le Dung[1,2*], Phan Huynh Lam[1,2]

Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Viet Nam[1]
Vietnam National University Ho Chi Minh City, Linh Trung Ward, Thu Duc District, Ho Chi Minh City, Vietnam[2]

Corresponding Author: 1,2*

ABSTRACT— The outbreak of the covid pandemic makes automation more and more promoted autonomous robots (AGVs) are increasingly essential. We see many self-propelled robots with different navigation technologies such as IMU, GPS, line tracking, RFID, Camera... In this paper, we introduce an application of SLAM and UWB so that the robot can be self-propelled in the production plant or harvesting farm. This is a new direction, and the world is also approaching to help self-propelled robots can be more perfect.

KEYWORDS: SLAM, UWB, AGV

1. INTRODUCTION

SLAM (simultaneous localization and mapping) is a model for building digital maps for self-propelled robots, most seen in factory delivery robots (AVGs) or civil vacuuming robots. SLAM is a system that uses digitized information obtained from sensors to reconstruct an external digital map by including environmental data into a map (2D or 3D). The robot or device can locate (localization) where it is the state in the map to automatically set up the path (path planning) in the current environment.

Automatic control of robotic equipment is divided into three main issues: localization, environment reproduction (mapping), and path planning.

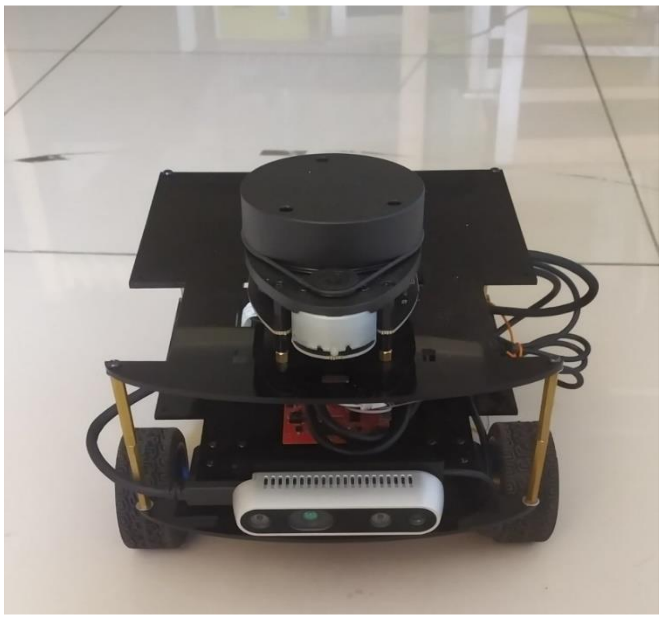


Figure 1. Self-propelled robot product with built-in sensors

The research team's self-propelled robot has fully integrated sensors such as a 3D camera, Lidar, IMU, encoder, UWB location, group towards the extended RTAB-Map. These sensors will help the robot walk more accurately and map out faster.

2. Materials and methods

Building digital maps

2.1 Some methods to build digital maps

Currently, there are many methods of implementing SLAM with open source code that is packaged into packages on ROS. Sensors used include laser-based SLAM and RGB-D Camera-based SLAM. Depending on the setting modes, some algorithms can support building 2D or 3D map models. In addition to increasing the accuracy, some algorithms also require more information about the robot's odometry.

a) GMapping [Grisetti et al., 2007] and TinySLAM [Steux and El Hamzaoui, 2010]: are two methods of using a particle filter to estimate the robot's trajectory. As long as there are enough estimated particles and the input odometry error is within the acceptable range, the particle filter represents the environment's characteristics. In particular, GMMapping is the default SLAM approach of ROS, which has a loop closures generation algorithm that has been widely used to construct a 2D mesh map of the environment from a 2D laser sensor. Once the map is created, it can be used with an adaptive Monte Carlo method [Fox et al., 1999] for positioning and navigation. [7].

b) Hector SLAM (proposed by Kohlbrecher in 2011): capable of creating 2D maps based on 2D lidar sensors with fast computation, saving computational resources. It has been proven capable of positioning with very high accuracy in practice. It is possible to combine more IMUs to build 3D maps. However, this is not a 3D SLAM approach that does not have loop closures detection algorithm, so it is impossible to recalibrate the map when the robot re-observes previously recorded landmarks. Another advantage of this algorithm is that it does not need information from other odometry, but in complex environments with many limitations in terms of visibility, relying solely on the laser sensor is not enough [7].

c) ETHZASL-ICP-Mapper4, based on libpointmatcher library [Pomerleau et al., 2013]: can be used to generate 2D mesh maps from 2D lidar and point_cloud collected from 2D or 3D lidar. But similar to Hector SLAM, this method does not have a loop closures detection algorithm, so the map cannot be edited. [7]

d) Karto SLAM (proposed by Vincent 2010), Lago SLAM5 (proposed by Carlone 2012), and Google Cartographer (2016): are graph-based SLAM lidar realization methods; they can build 2D grid maps based on graphs shown on the graph, which Google Cartographer can also build a 3D backpack map based on a 3D lidar sensor. During map construction, the algorithm builds submaps that meet the constraints in the graph. When loop closure is detected, the position in the submap is optimized and corrected for errors caused by sensor noise. Unlike Hector SLAM, this method can use additional odometry provided to increase estimation and map construction. [7]

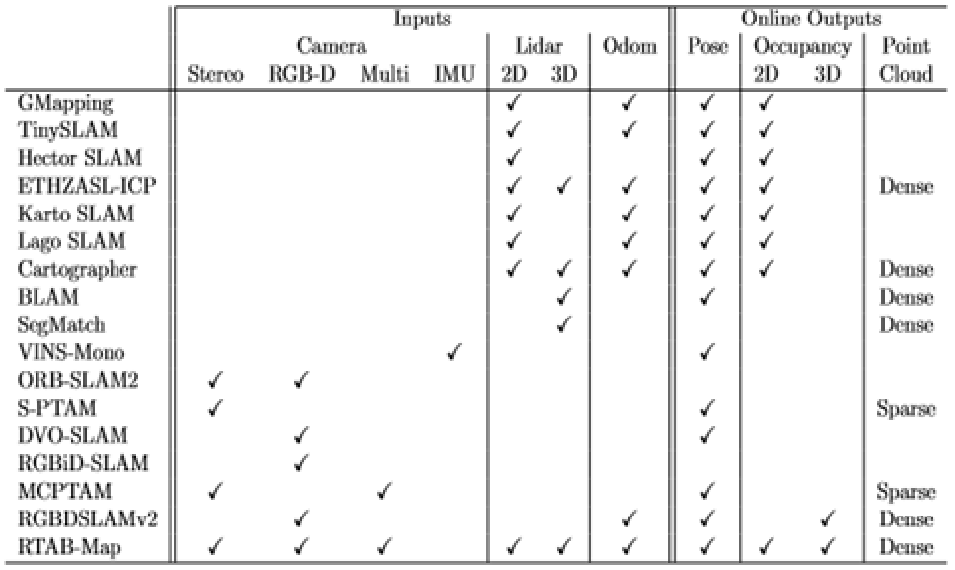
e) maplab [Schneider et al., 2018] and VINS-Mono [Yi et al., 2017]: published in recent years, they are visual graph-based SLAM systems. Using only the IMU and a camera, they can build a 3D map model. The maplab activity flow is divided into two steps: first, the data is collected through the camera and IMU, and then the map construction (including loop closure detection, graph optimization, multi-session, reconstruction). dense map setting) is done offline. The resulting 3D map can be used in positioning mode in later processes. In contrast to maplab, the mapping algorithm based on VINS-Mono can operate online, directly build maps and locate during operation. To keep processing time limited for large-scale environments, VINS-Mono limits the size of the graph, removing nodes without loop closures first, and then discarding others depending on density. of the graph. [7]

f) ORB-SLAM2 [Mur-Artal and Tard´os, 2017] and S-PTAM [Pire et al., 2017] are currently two of the state-of-the-art-based SLAM approaches Most modern imaging can be used with a depth camera. With ORB-SLAM2, it is possible to use an RGB-D camera as input to the algorithm, in addition, this method also uses DBoW2 to detect loop closure. [7]

g) RGBDSLAMv2 (Endres proposed in 2014): collect odometry information from sensors to estimate robot position. This algorithm can use multiple cameras at the same time, helping to create 3D mesh maps. [7]

Below is a comparison table of digital map construction algorithms

Table 1. Comparison of digital map construction methods



	Inputs							Online Outputs			
	Camera				Lidar		Odom	Pose	Occupancy		Point
	Stereo	RGB-D	Multi	IMU	2D	3D			2D	3D	Cloud
GMapping					✓		✓	✓	✓		
TinySLAM					✓		✓	✓	✓		
Hector SLAM					✓			✓	✓		
ETHZASL-ICP					✓	✓	✓	✓	✓		Dense
Karto SLAM					✓		✓	✓	✓		
Lago SLAM					✓		✓	✓	✓		
Cartographer					✓	✓	✓	✓	✓		Dense
BLAM						✓		✓			Dense
SegMatch						✓					Dense
VINS-Mono				✓				✓			
ORB-SLAM2	✓	✓									
S-PTAM	✓							✓			Sparse
DVO-SLAM		✓						✓			
RGBiD-SLAM		✓									
MCPTAM	✓		✓					✓			Sparse
RGBDSLAMv2		✓					✓	✓		✓	Dense
RTAB-Map	✓	✓	✓		✓	✓	✓	✓	✓	✓	Dense

2.2 Introducing the team's robot hardware

To be able to locate the robot quickly, the team uses UWB technology to replace the retrieval from 2D/3D digital maps. With this technology we can significantly reduce the processing time making the robot easy to operate for real-time applications and low-profile hardware.

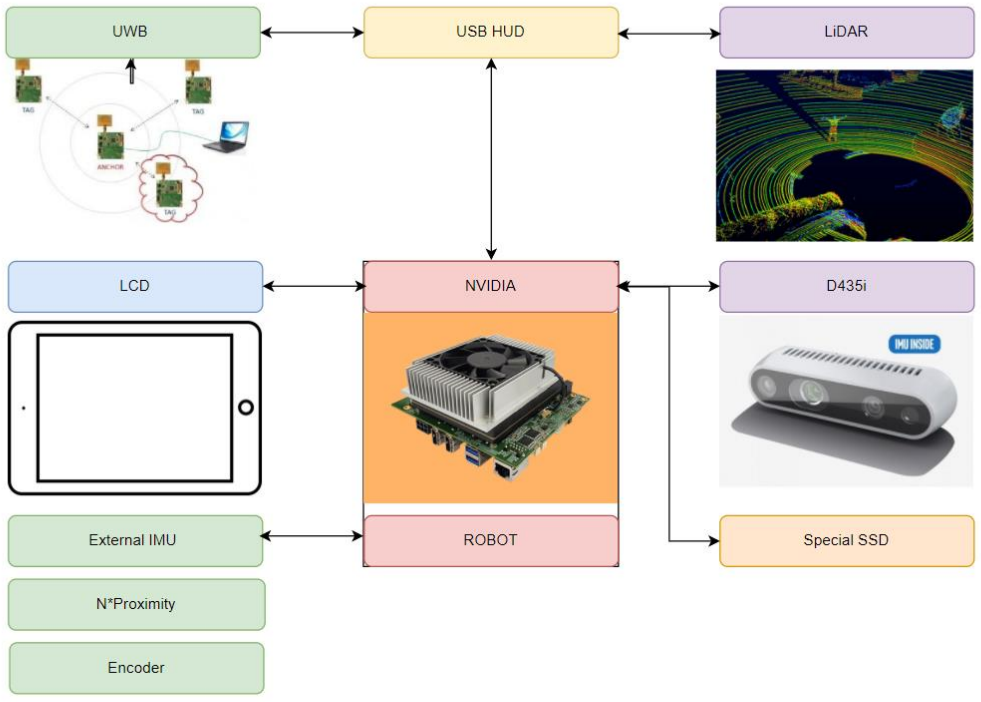


Figure 2. Robot block diagram

a) UWB sensor

The DWM1000 wireless transmission module uses the IEEE802.15.4-2011 UWB standard. These modules allow us to apply in projects to determine position or measure distance with high accuracy, error of about 10cm in real time (RTLS). The range is up to 290 meters and the transmission speed is up to 6.8 Mbit/s.

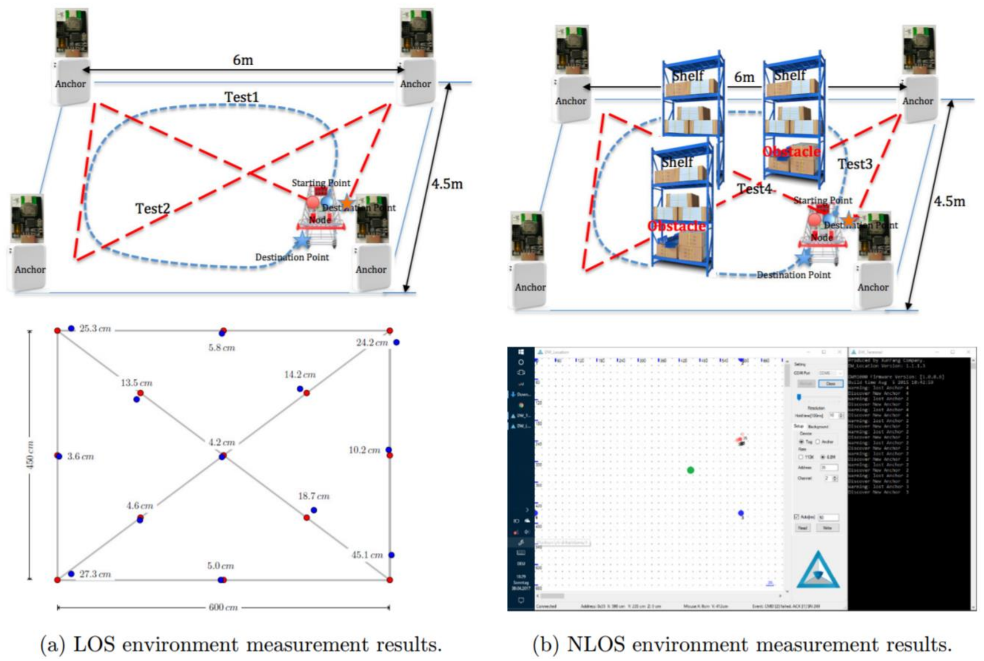


Figure 3. LOS and NLOS environment measurement results. [1]

b) Lidar sensor

ISSN: 18158129 E-ISSN: 18151027
Volume 18, Issue 06, June, 2022

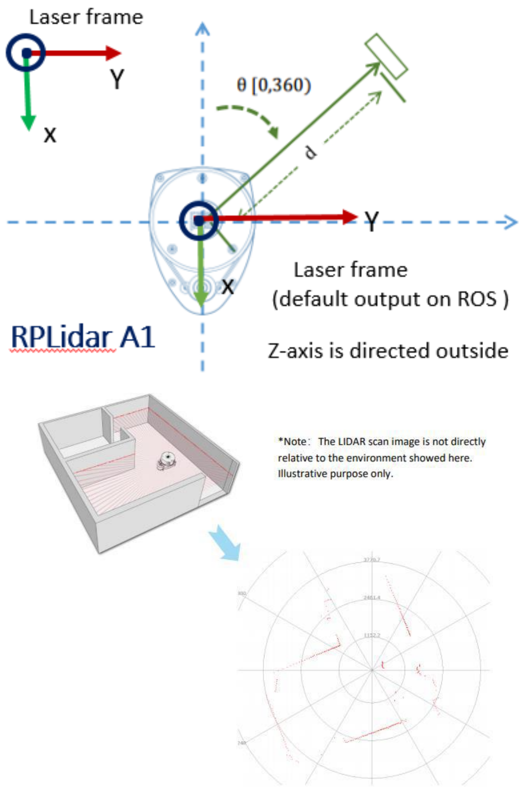

Laser frame
Y
x
θ [0,360)
d
Y
x
Laser frame
(default output on ROS)
RPLidar A1
Z-axis is directed outside
*Note: The LIDAR scan image is not directly relative to the environment showed here. Illustrative purpose only.

Figure 4. Lidar sensor

Slamtec's A1 lidar sensor has been supported with an open source library for many platforms so that users can install and read information from the sensor. In particular, the sensor supports ROS with the rplidar_ros package, reads data from the sensor and publishes a topic /scan containing the message < sensor_msgs/LaserScan >

However, this is just raw data, to increase the reliability we give the data through a Speckle Filter. This filter will remove single points whose distance exceeds a set interval from other points and has a low frequency of occurrence. Speckle Filter is supported along with many different filters in the laser_filter package of ROS and can be configured with multiple filter stages in series. The package will subscribe to the /scan topic and publish the /scan_filtered topic, this topic will be used for the SLAM and Navigation model.

c) 3D Camera

Similar to the lidar sensor A1 Intel also built a realsense camera support package for ROS called librealsense. To use the full functionality of the D435i, including IMU, point_cloud, and depth_image, we need to reconfigure the launch file for the camera.

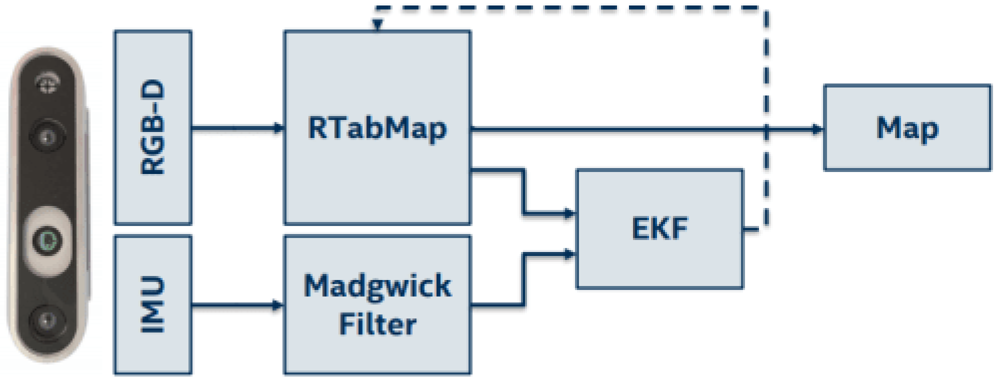


Figure 5. 3D Camera D435i with 3D Map model

3. Experimental results

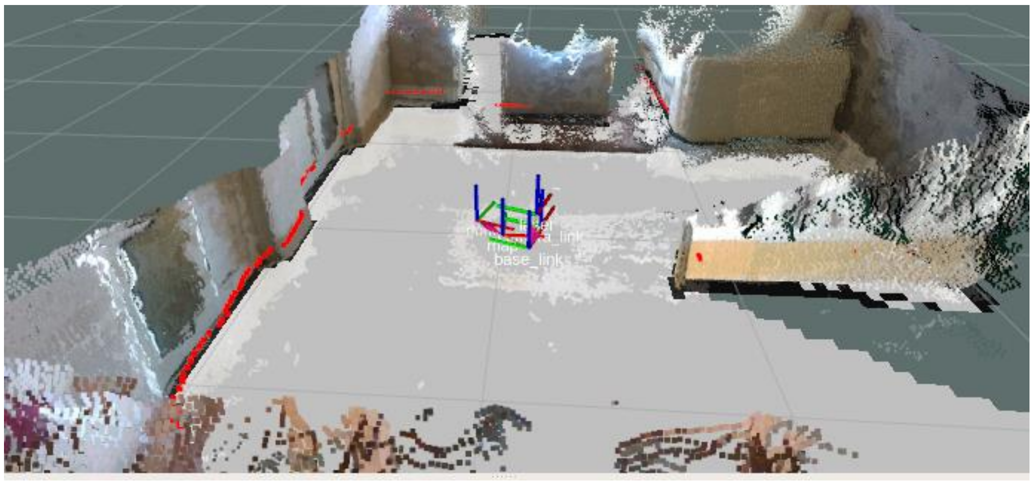


Figure 6. Experimental results of 3D map construction

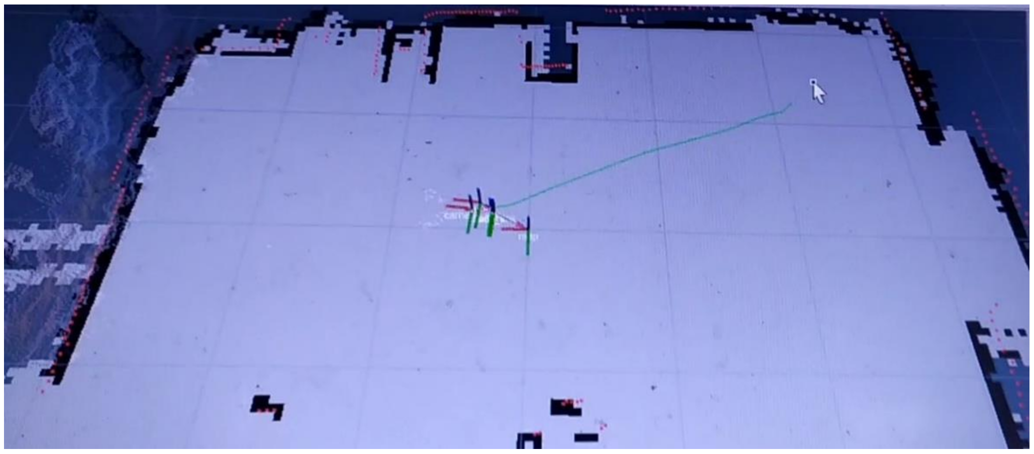


Figure 7. Experimental results of 2D map construction

4. Conclusion

By combining multi-sensors and UWB technology, the 3D mapping and positioning performance becomes powerful and fast; We overcome the disadvantages of the RTAB-MAP method and navigation requires a large number of computational resources, the speed of vehicle position estimation is slow in large spaces and is less characteristic due to the computational power of the board—insufficient processing response. Now to build

3D digital maps, we don't need too high a hardware configuration to deploy the system, even with just a basic embedded system.
• Positioning: thanks to the combination of lidar sensor, depth camera, the UWB robot is capable of locating its own position in the map even in the case of poor characteristic images or no image data.
• Navigation: the robot has the ability to plan a path and move to the designated point easily by digital map or through UWB positioning technology.
• Avoid obstacles: the robot has the ability to move around, avoiding obstacles on the map as well as new obstacles that have not been stored thanks to lidar and 3D cameras.
The team aims to integrate the fruit recognition part to complete a post-harvest robot in the future.

We acknowledge the support of time and facilities from Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, for this study.

5. Acknowledgment

We acknowledge the support of time and facilities from Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, for this study.

6. References

[1] Guohua Hu, Pascal Feldhaus, Yuwu Feng, Shengjie Wang, Juan Zheng, Huimin Duan, and Juanjuan Gu, Accuracy Improvement of Indoor Real-Time Location Tracking Algorithm for Smart Supermarket Based on Ultra-Wideband, International Journal of Pattern Recognition and Artificial IntelligenceVol. 33, No. 12, 2058004 (2019)

[2] Wiki ROS, "ROS/Concepts",[Online]. Available: http://wiki.ros.org/ROS/Concepts

[3] Wiki ROS, "rtabmap_ros",[Online]. Available: http://wiki.ros.org/rtabmap_ros

[4] Wiki ROS, "navigation," [Online]. Available: http://wiki.ros.org/navigation

[5] More robots, "blog," [Online]. Available: http://moorerobots.com/blog

[6] L. Joseph, Mastering ROS for Robotics Programming, 2015.

[7] Mathieu Labbé and Fran¸cois Michaud. RTAB-Map as an Open-Source Lidar and Visual SLAM Library for Large-Scale and Long-Term Online Operation, 2019.

[8] Intel Team, "Intel® RealSense™ Depth Camera D435i",[Online]. Available: https://www.intelrealsense.com/depth-camera-d435i/

[9] Intel Team. Intel® RealSenseTM Product Family D400 Series Datasheet

[10] Nvidia Team, "Jetson Nano",[Online]. Available: https://www.nvidia.com/enus/autonomous-machines/embedded-systems/jetson-nano/

[11] Vandad Imani, Keijo Haataja, Pekka Toivanen, Three Main Paradigms of Simultaneous Localization and Mapping (SLAM) Problem.

[12] SHANGHAI SLAMTEC CO., LTD, RPLIDAR A1 Low Cost 360 Degree Laser Range Scanner Introduction and Datasheet.

[13] Soonshin Han,ByoungSuk Choi, JangMyung Lee. A precise curved motion planning for a differential driving the mobile robot.

[14] Wiki ROS, “costmap_2d”,[Online]. Available: http://wiki.ros.org/costmap_2d