The 2017 World Congress on
Advances in Nano, Bio, Robotics and Energy (ANBRE17)
28 August - 1 September, 2017, Ilsan(Seoul), Korea

2D SLAM Solution for Low-Cost Mobile Robot based on Embedded Single Board Computer

*Seokju Lee[1)], Girma Tewolde[2)], Jongil Lim[3)] and Jaerock Kwon[4)]

[1), 2), 3), 4)] *Department of Electrical and Computer Engineering, Kettering University, MI, 48504, USA*

[1)] *lee7704@kettering.edu*, [2)] *gtewolde@kettering.edu*, [3)] *lim7722@kettering.edu*, [4)] *jkwon@kettering.edu*

ABSTRACT

This paper presents a low-cost mobile robot platform to solve the SLAM problem for indoor mobile robotics applications. It deals with the necessity of building a map of the environment while simultaneously determining the location of the robot within the map. In order to solve SLAM, this work uses an existing tool called GMapping, which is based on a RBPF (Rao Blackwellised Particle Filter) approach, provided by ROS. The GMapping tool offers laser-based SLAM for building a map. It was originally based on laser scan data and odometry information. In this work, we are utilizing the Microsoft Kinect Sensor, gyroscope, and wheel encoders on a mobile robot. The paper demonstrates the feasibility of the proposed approach of implementing SLAM on a small, light-weight and low-cost embedded single board computer instead of a more expensive full-fledged PC or laptop.

1. INTRODUCTION

Simultaneous localization and mapping is one of the challenging problems in mobile robot navigation that has attracted the interest of more and more researchers in the last decade. As of today, several approaches exist to address the problem with different levels of success. This problem is commonly abbreviated as SLAM. It is concerned with the robot's ability of acquiring a map of its environment and simultaneously localizing itself relative to this map. SLAM consists of having a system composed of a mobile robotic platform that offers movement measurement such as odometry, a processing unit exploring of given environment information, and one of several available SLAM algorithms. SLAM is one of the most difficult problems in

---

[1)] Graduate Student
[2)] Professor
[3)] Graduate Student
[4)] Professor

robotics to solve. The reason is that the mobile robot has to deal with challenges in getting accurate odometry and other sensor measurements, which are both error-prone and noisy; and create a correlation between them.

In this paper, we apply an existing GMapping tool for the implementation of the SLAM algorithm. The GMapping is one of the most well-established laser scanner-based SLAM algorithms that were proposed by (Grisetti 2006). This algorithm is based on a Rao-Blackwellised Particle Filter (RBPF) as an effective means to solve the SLAM problem and it can be easily used in the Robot Operating System (ROS). (Santos 2013) offers detailed overview, analysis, and performance evaluation of five typical 2D SLAM algorithms available in ROS. GMapping was shown to be robust in their experiments, when compared to other 2D SLAM approaches. Also, GMapping needs odometry information for position and heading direction estimation. Odometry can be improved by using the left and right wheel encoders equipped on the mobile robot, and the accuracy improved using a single axis gyroscope.

The open source robot operating system (ROS) is one of the most popular robotics frameworks today. It enables researchers to quickly and easily perform simulation and real world experiments with respect to mobile robotic research and provides a set of tools, libraries, drivers, and various system functionalities to help develop robotic applications and algorithms. A comprehensive overview of the ROS has been discussed by (Quigley 2009).

Numerous SLAM techniques have been developed by previous researchers utilizing different devices, including sonar sensor used in (Choi 2005), (Gambino 1996) and laser scanner used in (Chou 2013). However, sonar sensors have a major problem of high uncertainty of sensor measurements and false readings caused by multi-path phenomenon due to the specular reflection effects from the environment. Lasers are accurate but they are heavy and expensive, and have relatively higher power consumption. On the other hand, cameras are light, cheap, and can provide abundant environmental information. The introduction of the Microsoft Kinect Sensor and its effect as presented by (Zhang 2012) has allowed an extension of the available SLAM methods. (Henry 2012) proved that the Kinect Sensor can be a feasible alternative for 2D SLAM based on RGB-Depth mapping because it allows developers to take advantage of RGB and the depth information coming from the Kinect Sensor.

In consideration of these research results, we decided to use Kinect Sensor for vision and depth information, odometery from wheel encoders and a gyroscope, and particle filter based 2D SLAM algorithm to generate occupancy grid based environment map and locate the position of mobile robot on this map at the same time. Also, in this work, an embedded single board computer (Odroid-XU3 Lite) with the ARM-based quad core processor is used to run ROS on UbuntuARM 12.04 operating system and the Hydro version of ROS installed on UbuntuARM. Although Hydro version is currently not latest version, it is still most suitable for the implementation of SLAM using ROS-compatible mobile robot and Smartphone application developed by OSRF (Open Source Robotics Foundation) that supports the development, distribution, and adoption of open source software for use in robotics research. The embedded single board computer used in this work is responsible for high-level task processing, such as SLAM algorithm by using depth information from the Kinect Sensor and odometry estimation from the robot wheel encoders. The Kinect Sensor mounted facing the front of the mobile robot is connected to the Odroid-XU3 Lite board using USB interface. Android-based application is used on a Smartphone to wirelessly visualize the process of SLAM and to remotely control the mobile robot.

The ultimate goal in this paper is to present the low-cost and light-weight mobile robot design to solve the SLAM problem and demonstrate that it is a viable approach for practical mobile robotic application.

The reminder of this paper proceeds as follows. Related literature is presented in section 2. Section 3 discusses the whole system configuration for SLAM. The experimental setup and result are described in section 4. In section 5, we conclude with a summary and a discussion on future extension of this work.

2. RELATED WORKS

SLAM has been an interesting topic in mobile robotics for the last several years. Some related research relevant to our work is discussed in this section. In the field of mobile robotics, solutions for the problem of SLAM rely on probabilistic frameworks to cope with sources of errors such as sensor measurement noise and uncertainty in the measurement and estimation process.

As one of probability estimation theories, Particle Filter (PF) has received much attention as an approach for the SLAM problem. However, the PF family of algorithms usually requires a large number of particles to accomplish satisfactory results, which results in high computational complexity. Also, Particle Filters tend to have a particle depletion problem in the resampling process as reported in (Kwak 2007). This problem eliminates particles with low weights while performing SLAM for a mobile robot. This loss of information makes the algorithm accuracy to degenerate. An adaptive resampling technique has been developed in (Grisetti 2007), which alleviates the particle depletion problem by maintaining a reasonable variety of particles, and achieves interesting performance with a low number of particles, e.g. 30 particles, which translates into low computational requirements.

The papers (Zaman 2011), (Pajaziti 2014), (Ruensuk 2012) use ROS-based control system on ROS compatible robot and Kinect Sensor for localization, mapping, and navigation. In their experiments, a laptop computer is used to run ROS on Ubuntu operating system. On the other hand, we use a small embedded single board computer instead of a laptop computer. Single board computers with powerful computation and energy-efficient hardware have been becoming gradually lighter and smaller, and therefore potentially offer cost-effective solutions to many practical problems.

(Paola 2010) proposed the development of mobile robot system for surveillance in indoor environment. They also presented SLAM solution based on laser sensor, odometry information and RFID technology. With the use of ROS, ROS-enabled robot and Kinect Sensor, anyone can easily experience SLAM solution without requiring deep understanding of the actual SLAM algorithms or additional expensive hardware or sensor device. The development of low-cost mobile robot platform for SLAM through some hardware modifications to replace laptop computer and the utilization of SLAM algorithm available on ROS has been presented by (C. Hammer 2012).

3. IMPLEMENTATION DETAILES

For practical implementation, we use ROS-compatible mobile robot based on iRobot Create that is equipped with an embedded single board computer along with Wi-Fi module as well as Kinect Sensor. (T. Isaacs 2011) offers detailed information about the iRobot Create. An Input data for SLAM algorithm is provide by Kinect Sensor and odometry sensors from wheel encoders and gyroscope. Fig. 1 shows the overall hardware configuration mounted on the mobile robot used as the test-bed for the practical experiments.

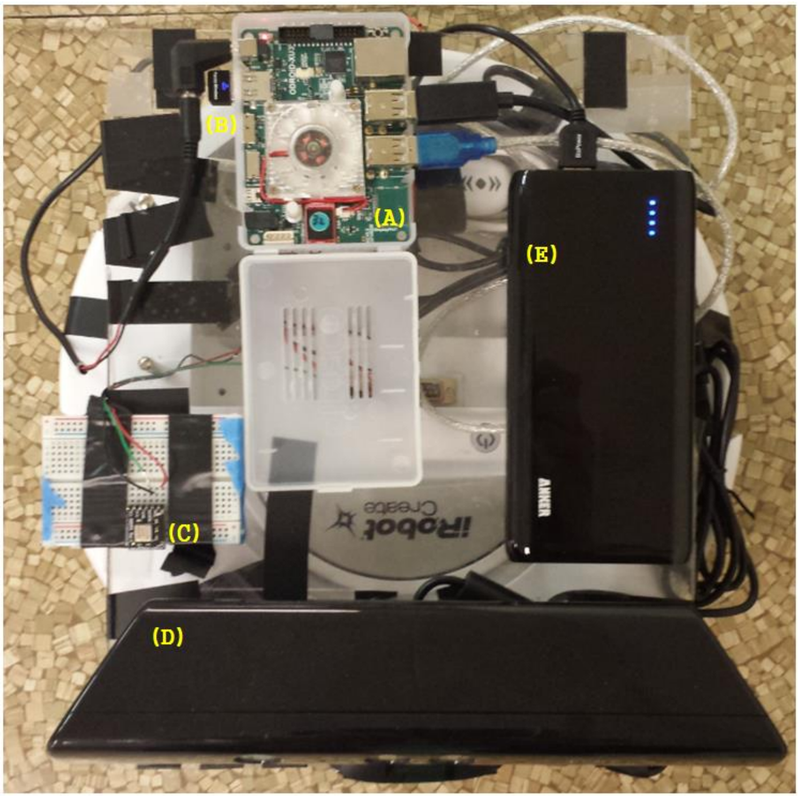


Fig. 1 The proposed iRobot create based mobile robot platform with (A) Odroid-Xu3 Lite board, (B) Wi-Fi module, (C) Gyro sensor, (D) Kinect Sensor, and (E) Battery for the embedded single board computer.

3.1 Hardware Design

This section presents the hardware design of the major components of the low-cost and light-weight mobile robot.

3.1.1 Odroid-Xu3 Lite and Wi-Fi Module

We evaluated several embedded single board computers that could run the entire Linux operating system for ease of setup and flexibility in software installation. Most of the available choices are primarily ARM processor based boards. Our choice is Odroid-Xu3 Lite computing device, which is powered by Samsung Exynos 5422 Octa-Core processor (1.8GHz and 1.3 GHz quad core CPUs) implemented on energy-efficient hardware and features 2GB of low-power DDR3 memory, eMMC 5.0 flash-storage and microSD slot with Linux or Android, support for OpenGL ES 2.0 graphics, USB 3.0 and multiple USB 2.0 ports, HDMI, and a 10/100Mbps Ethernet card etc. The cost of the embedded single board computer alone is $99 with the relevant accessories (a cooling fan, micro HDMI cable, and 5V / 4A power supply unit). A high-capacity external rechargeable battery, Astro E6, as shown in Fig. 1 (E) is used to provide 5V and 4A required to the single board computer. The external battery manufactured by Anker has enough capacity, 20800mAh, to run the board for more than 3 hours in our implementation of SLAM. Wi-Fi module, as shown in Fig. 1 (B) is connected to USB port on the embedded single board computer. Also, it provides the single board computer with a wireless connectivity.

3.1.2 Odroid-Xu3 Lite and Wi-Fi Module

The Microsoft Kinect Sensor, as shown in Fig. 1 (D) has become popular because of its low cost and its capability to provide RGB images and depth information simultaneously both running at a frame rate of 30 fps. Each frame of the depth information is made up of pixels that contain the distance from the camera to objects. The depth information is just an array of bytes, where each pixel is represented with two bytes (16 bits). The depth stream of Kinect is an 11-bit value allowing for 2048 different depth sensitivity levels. The range of Kinect’s depth sensor can be adjusted to either near or default range mode. According to the user guide of Microsoft Kinect Sensor, the operating range of the Kinect’s depth sensor is between 80 cm to 400 cm in default mode. The power cable of the Kinect Sensor needs to be modified to accept a 12V input from the power supply of the mobile robot. However, the mobile robot supplies unregulated power its battery. This can be easily solved by using a simple regulator circuit to produce a regulated 12V output. The website for the ROS wiki presents step by step instructions for modifying the iRobot Create to power a Kinect.

3.1.3 Gyroscope Sensor

Fig. 1(C) shows the sensor breakout board with an ADXRS610 single-axis gyro that can measure the robot's yaw rates up to 300 degrees/s and is connected to an analog input pin on the mobile robot to give sensor information. The gyro sensor allows the mobile robot to know the angular speed to calculate its orientation or to maintain its balance. The website for the DFRobot wiki presents a detailed specification of the gyro sensor used in this work. The mobile robot will create the map based on its position given by the gyro and wheel encoder sensor.

3.2 Software Design

In this section, the software design techniques for the implementation of our proposed approach SLAM on a small, light-weight and low-cost embedded single board computer are described.

3.2.1 System Configuration of ROS

Fig. 2 illustrates the overall architecture for performing SLAM in ROS. The embedded single board computer is used to run the GMapping software by interfacing with the Kinect Sensor and mobile robot through USB interface. Also, it communicates with Smartphone equipped with an Android app, which is utilized for controlling the robot's movement and to visualize the progress of building a map.

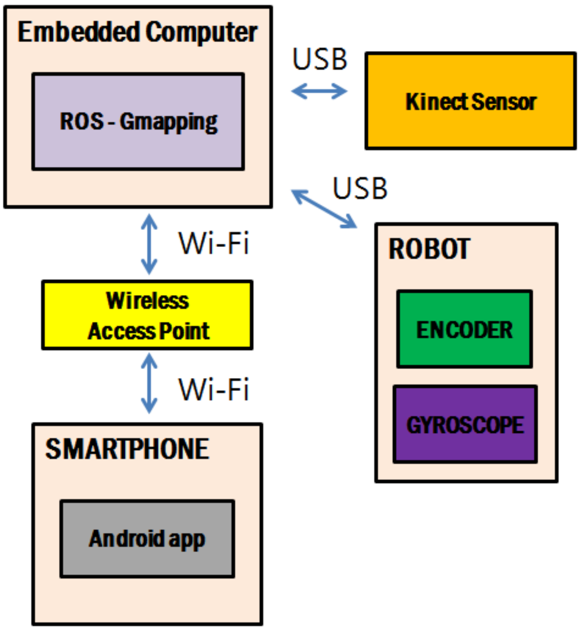


Fig. 2 Overall architecture of our system

The GMapping tool available through ROS requires several input data to function. The first of these is a stream of depth information, essentially composed of a series of distance measurements from the Kinect Sensor to objects it detects and the second in a set of odometery information required in the approximation of where the robot has traveled with respect to its previous position. ROS provides many software packages and tools that can easily access the robotic devices. One of them is Rviz for 3D visualization, which allows visualizing all kind of information regarding the robot and its environment. In this work, it is installed and used for visualizing the results of SLAM.

3.2.2 Implementation of SLAM using GMapping

ROS software is distributed in packages, which generally contain nodes. GMapping and turtlebot packages are used to implement SLAM. Fig. 3 shows the overall architecture of SLAM implementation. We look through some important nodes in this section. The turtlebot_node takes the sensor data from the mobile robot based on the iRobot create base and computes the odometry as the mobile robot moves. The gyro sensor is accessed through the analog input data and it is transmitted to the turtlebot_node. Then, the robot_pose_ekf node, which uses an Extended Kalman Filter to combine measurements from wheel encoder and gyro sensors, receives the gyro and odom data from the turtlebot_node to compute more accurate sensor data that is odom_combined.

The openni_camera package is installed in ROS to access the depth images of the Kinect Sensor. The received depth image is converted to equivalent laser scan data by using depthimage_to_laserscan node to provide the laser scan data to slam_gmapping and amcl nodes. The slam_gmapping node creates a map, usually in occupancy grid form and amcl node is responsible for determining the robot's position in the map it created. Both steps depend on each other.

AMCL (Adaptive Monte Carlo Localization) is a probabilistic localization method for a mobile robot moving in the 2D environment. After generating the map with GMapping, we use it for more stable localization with AMCL. The amcl node integrates knowledge of the map as well as laser scans and odometry to estimate the robot's position. This is done by matching laser scan data provided by Kinect Sensor with the map's wall, and using the quality of the match at different locations to infer probabilities for the actual position. To accomplish this, AMCL implements the adaptive Monte Carlo localization approach, which uses a particle filter to track the pose of a mobile robot within a known map.

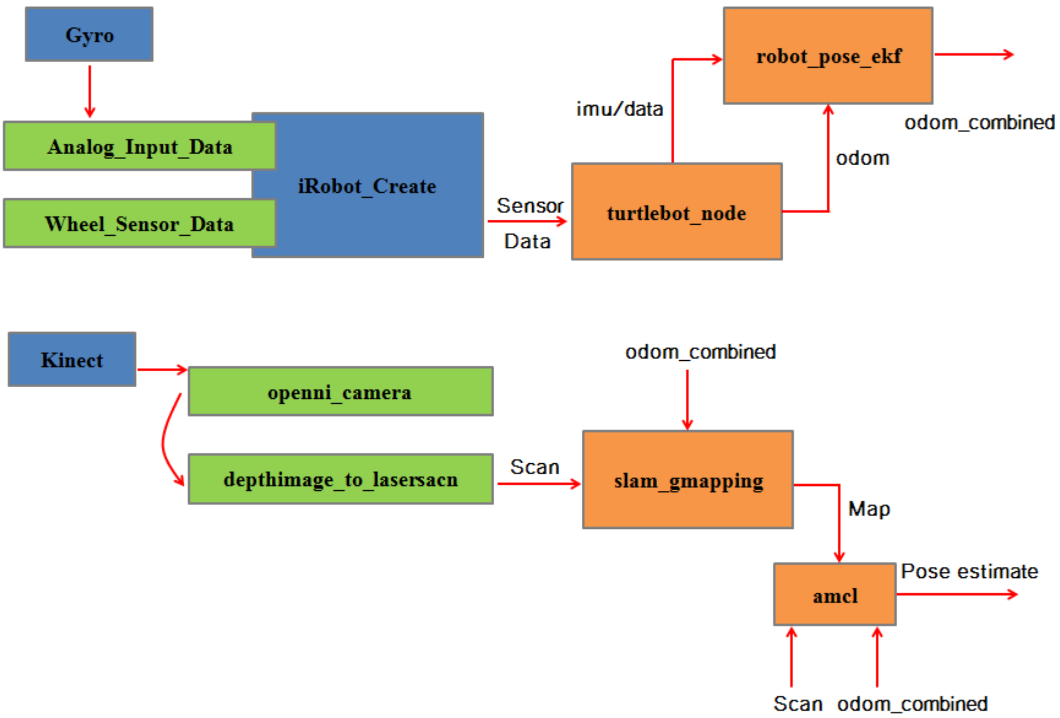


Fig. 3 Graphical representation of SLAM architecture

3.3.3 Smartphone Application

As shown in Fig. 4, a mobile application called "Make A Map" of Hydro version, available from the google play app store, is utilized for ROS-enabled robot to demonstrate a simple control scenario for mobile robot SLAM in this work. This application is very useful because it views actual environment, progress of building a map, and position and heading of the mobile robot as well as battery information of a mobile robot. Also, it can control a mobile robot and save a map when building a map is completed.

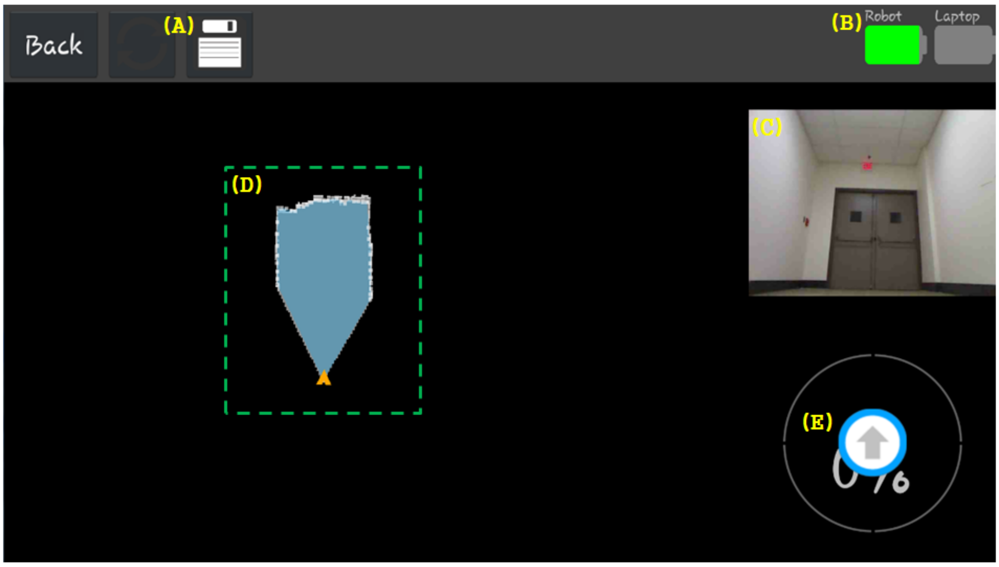


Fig. 4 The picture of the "Make A Map" application for Android
(A) Button to save the map (B) Battery information of the mobile robot
(C) Photo of Real-world environment
(D) Progress of building a map, and (E) Button to control a robot

4. RESULT

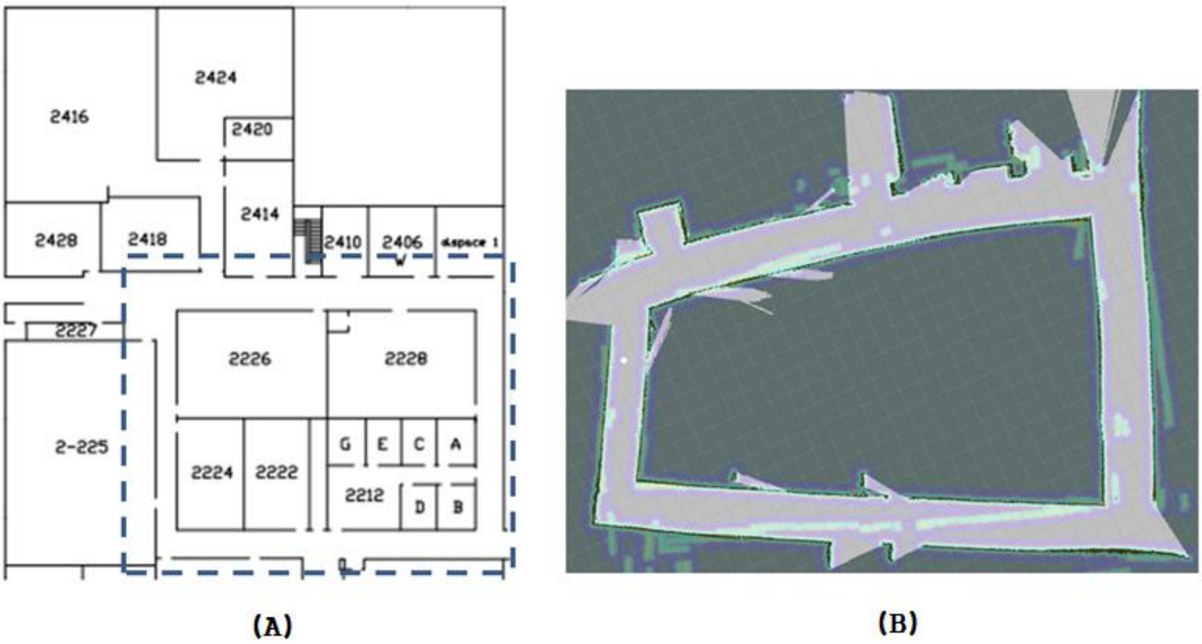


Fig. 5 The map of the hallways of Chemical Engineering Department at Kettering University. All measurements are made based on the wall references.

In order to demonstrate our approach, experimental tests were carried out. Before the SLAM is performed, odometry and gyro calibration of mobile robot is required to improve the odometry accuracy. In the first experiment, we used a laptop computer. The first testing actual environment for the SLAM is shown in Fig. 5(A). Fig. 5(B) shows the final maps obtained during the SLAM operation for the hallways of the Chemical Engineering department at Kettering University. The embedded single board computer is used in the second experiment. A 32G eMMC flash memory installed with UbuntuArm operating system is mounted on the single board computer. This single board computer based test is implemented to demonstrate the feasibility of our approach. The actual environment for the SLAM for the second test is shown in Fig. 7. Fig. 6 shows a progress of building a map. The test algorithm has been run with 80 particles and it generates occupancy grids as the final output for a metric map. Fig. 7(B) shows the map made from the mobile robot. The 2D map of the environment is built using GMapping based on the Rao-Blackwellised Particle Filter with inputs from the odometry and gyro sensors and the depth information from Kinect Sensor.

The experimental results show that the proposed approach in this work successfully maps the hallway, though the map created has some misalignments due to odometry errors. The ultimate goal of the experiment is to verify the fulfillment of the existing SLAM algorithm using a laptop computer and demonstrate the feasibility of the proposed approach of implementing SLAM on a small, light-weight and low-cost embedded single board computer

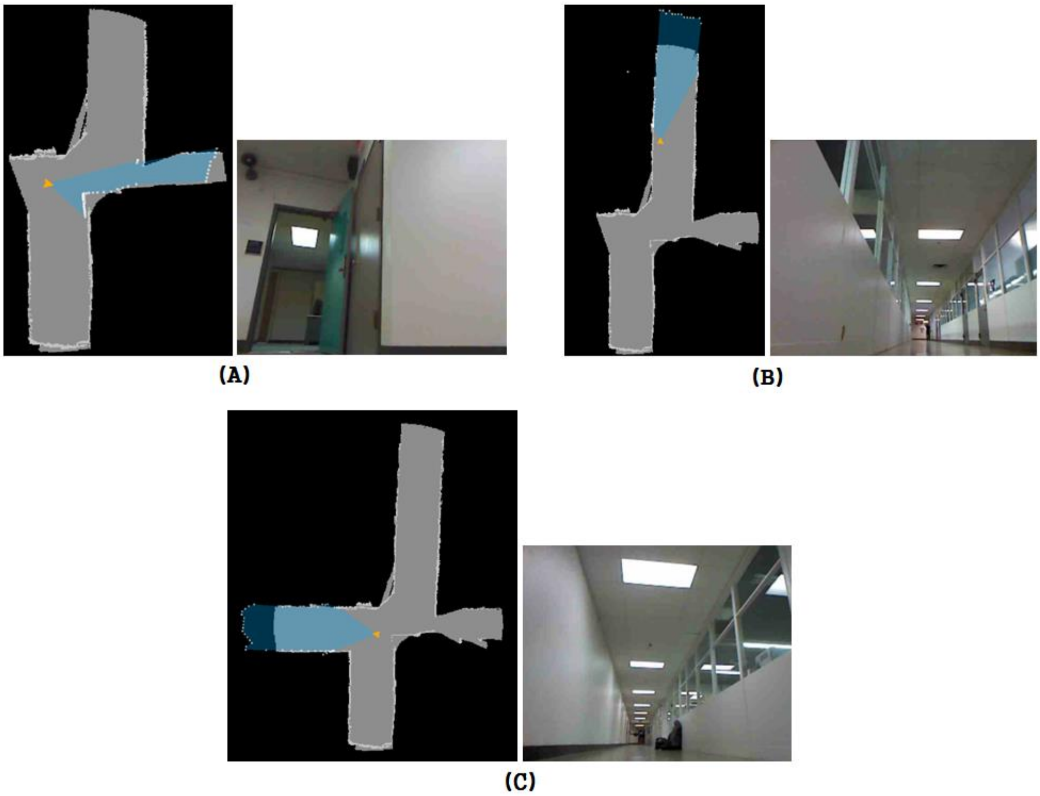
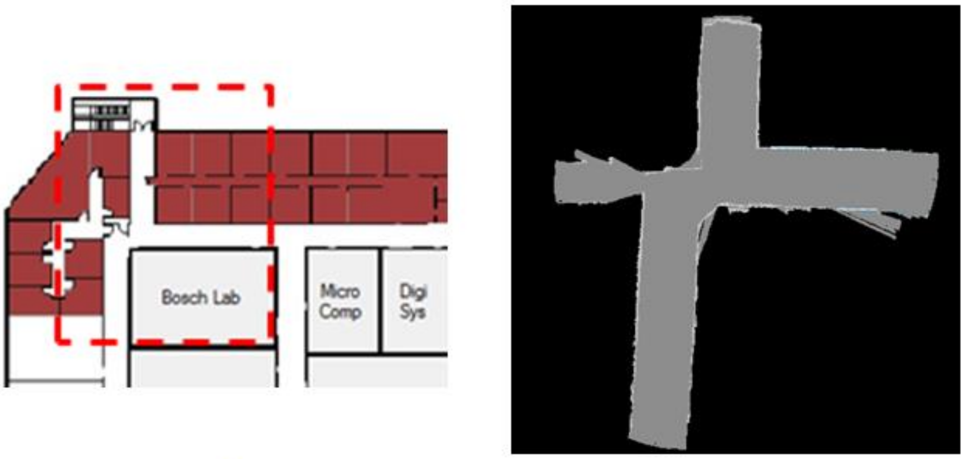


Fig. 6 The progress of building a map



(A) (B)

Fig. 7 The map of the corridors in the ECE lab.
All measurements are made based on the wall references.

5. CONCLUSIONS

We have presented the design for a low-cost mobile robot platform for building a map. The GMapping tool used in this work is based on a Rao-Blackwellised Particle Filter to track the robot trajectories for the SLAM approach, using an adaptive resampling technique. In order to use the GMapping algorithm, ROS enabled robot that provides odometry information collected from wheel encoders and gyroscope and the depth information collected from the Kinect Sensor are used to build up a 2D model of the environment. The Robot Operating System (ROS) was installed on an embedded single board computer to run the GMapping software provided by ROS. Each component was chosen based on low-cost, reasonable performance, and low-energy consumption factors. The technique was demonstrated to be effective for building a map while localizing the robot at the same time.

REFERENCES

Giorgio Grisetti, Gian Diego Tipaldi, Cyrill Stachniss, Wolfram Burgard, and Daniele Nardi (2006), “Fast and Accurate SLAM with Rao-Blackwellized Particle Filters”, Robotics and Autonomous Systems, Vol. 55 (1), pp. 30-38.

Joao Machado Santos, David Portugal, and Rui P. Rocha (2013), “An Evaluation of 2D SLAM Techniques Available in Robot Operating System”, Proceedings of ‘13 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 1-6.

Morgan Quigley, Brian Gerkey, Ken Conley, Josh Faust, Tully Foote, Jeremy Leibs, Eric Berger, Rob Wheeler, and Andrew Ng (2009), “ROS: an open-source Robot Operating System”, Proceedings of ’09 IEEE International Conference on Robotics and Automation(ICRA), pp 5-10.

Jinwoo Choi, SungHwan Ahn, and Wan Kyun Chung (2005), “Robust sonar feature detection for the SLAM of mobile robot”, Proceedings of ’05 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 3415-3420.

Fabio Gambino , Giuseppe Oriolo , and Giovanni Ulivi (1996), “A Comparison of Three Uncertainty Calculus Techniques for Ultrasonic Map Building”, SPIE International Symposium on Aerospace/Defense Sensing and Control, Vol. 2761, pp. 249-260.

Yu-Shin Chou and Jing-Sin Liu (2013), “A Robotic Indoor 3D Mapping System Using a 2D Laser Range Finder Mounted on a Rotating Four-Bar Linkage of Mobile Platform”, International Journal of Advanced Robotic Systems, Vol. 10 (1), pp. 257-271.

Zhengyou Zhang (2012), “Microsoft Kinect Sensor and Its Effect”, IEEE Multimedia, Vol. 19(2), pp. 4-10.

Peter Henry, Michael Krainin, Evan Herbst, Xiaofeng Ren, and Dieter Fox (2012), “RGB-D Mapping: Using Kinect-Styles Depth Camera for Dense 3D Modeling of Indoor Environments”, International Journal of Robotics Research, Vol. 31(5), pp. 647-663.

Nosan Kwak, In-Kyu Kim, Heon-Cheol Lee, and Beom-Hee Lee (2007), “Analysis of Resampling Process for the Particle Depletion Problem in FastSLAM”, Proceedings of ‘07 IEEE International Conference on Robots & Human Interactive Communication (RO-MAN), pp. 200-205.

Giorgio Grisetti, Cyrill Stachniss, Slawomir Grzonka, and Wolfram Burgard (2007), “Improved techniques for grid mapping with Rao-Blackwellized particle filters”, IEEE Transactions on Robotics, Vol. 23(1), pp. 34-46.

Safdar Zaman, Wolfgang Slany, and Gerald Steinbauer (2011), “ROS-based Mapping, Localization and Autonomous Navigation using a Pioneer 3-DX Robot and their Relevant Issues”, Proceedings of ‘11 Saudi International Electronics, Communications and Photonics Conference (SIECPC), pp. 1-5.

Arbnor Pajaziti and Petrit Avdullahu (2014), “SLAM – Map Building and Navigation via ROS”, International Journal of Intelligent Systems and Applications in Engineering (IJISAE), Vol. 2 (4), pp 71-75.

Mintra Ruensuk (2012), “Kinect-Enabled Autonomous Office Assistant Robot”, Master’s thesis at the department of computer science of Asian Institute of Technology School of Engineering and Technology in Thailand, pp 1-34.

Donato Di Paola, Annalisa Milella, Grazia Cicirelli, and Arcangelo Distante (2010), “An Autonomous Mobile Robotic System for Surveillance of Indoor Environments”, International Journal of Advanced Robotic Systems, Vol. 7(1), pp 19-26.

Patrick C. Hammer, David J. Cappelleri, and Micheal M. Zavlanos (2012), “TortoiseBot: Low-cost ROS-Based Mobile 3D Mapping Platform”, Technical Report No. TR2-DC, Department of Mechanical Engineering, Stevens Institute of Technology.

Jason T. Isaacs, Daniel J. Klein and Joao P. Hespanha (2011) “A Guided Internship For High School Students Using iRobot Create”, International Federation of Automatic Control (IFAC) World Congress, Vol. 44(1), pp. 5195-5200.