Autonomous 2D Mapping of an Unknown Environment using Single 1D LIDAR and ROS

Arun T S
Department of Mechatronics Engineering
Vit Vellore, Tamil Nadu,632014
India

Prof. Krishna S
Department of Mechatronics Engineering
Vit Vellore, Tamil Nadu,632014
India

***Abstract*— Mapping is the process to represent the environment into other forms such as a sketch map or the other. Process mapping provides the solution of the problem of how a robot can automatically recognize the environment and how robots can provide information such as maps representations of the environment around the robot. This paper presents autonomous mobile robot with mapping capability with only using 1D single LIDAR and without odometry information of the mobile robot, addressing the features autonomous exploration along with simultaneously environmental mapping. Hector algorithm is adopted for building the map around the robot which is equipped with single 1D laser range finder .Feedback from the three ultrasonic sensors provided on the mobile robot Navigate the robot .The corresponding to The position of the Robot surrounding map is created in the ROS platform using Rviz. The Arduino Mega is used as the medium between ROS and LIDAR. The aim is to build a cost efficient mapping robot with limited computational techniques, without losing accuracy and robustness. These instructions give you guidelines for preparing papers for IEEE Transactions and Journals. Use this document as a template if you are using Microsoft *Word* 6.0 or later. Otherwise, use this document as an instruction set. The electronic file of your paper will be formatted further at IEEE. Paper titles should be written in uppercase and lowercase letters, not all uppercase. Avoid writing long formulas with subscripts in the title; short formulas that identify the elements are fine (e.g., "Nd–Fe–B"). Do not write “(Invited)” in the title. Full names of authors are preferred in the author field, but are not required. Put a space between authors’ initials. Define all symbols used in the abstract. Do not cite references in the abstract. Do not delete the blank line immediately above the abstract; it sets the footnote at the bottom of this column.**

Index Terms— 2D mapping, LIDAR Lite V3, ROS, Differential Mobile Robot

I. INTRODUCTION

The sensory acquisition of the surrounding environment is the most important task of mobile robotics, as it is the foundation of any ability that the robot should have. Autonomous navigation of mobile robots has been attracting attention from researchers. A robot must be able to position itself with regard to its environment. The best way for a robot to obtain information about its current position is to have a genuine map of its current surroundings stored inside its memory. It is a challenging task for a mobile robot to explore an unknown environment and build a real-time navigation map[4][5]. Numerous mapping techniques have been developed by previous researchers utilizing different devices, including sonar sensors, cameras and laser scanners where, the laser scanner has been said to outperform its counter parts.

Here we consider the case of creating maps using a 1D LIDAR moving in 1-DOF. A key advantage of using a LIDAR is its insensitivity to ambient lighting and optical texture in the scene[1]. Recent developments in LIDAR have reduced their weight and size. The LIDAR can be held on a small mobile robot, or even attached to a micro aerial vehicle[8]. This robot uses Arduino Mega 2560 micro-processor to interface the LIDAR. Data is sent through serial communication between the micro-processor and the ROS software. The ROS is used to generate the Surrounding map of the environment. ROS is considered as one of the best framework for all robotics software. ROS is free and open software. ROS was developed to address a set of big challenges of building robots (Quigley et al., 2009). There are many packages are available in ROS for building surrounding map like hector slam, g mapping, karto slam, core slam[6] . For this work Hector mapping is preferred over g mapping because of the advantages like Hector SLAM relies only in scan matching and it does not make use of odometry, While it give accurate and robust result of surrounding environment[4] [5]. The accuracy and robustness of proposed mobile robot is demonstrated by the real time mapping in the unknown room

Due to the importance of the LIDAR sensor, a lot of researches work has focused on LIDAR rangefinder data interpretation and improvement LIDAR Lite V3 is a latest sensor in the market with cheap prize. We have not found many references addressing map building using single 1D LIDAR sensor. The aim of this paper is provide a 2D map using 1D LIDAR which reduce the cost while we have comparable accuracy with the 2D LIDAR. The advantages and applications of ROS(Robotic Operating System) is explained detail in [2]. Where various algorithms for SLAM like hector slam, G Mapping etc along with their advantages, disadvantages and application area are explained in [3][4]. [5] Proposed a map building robot using digital magnetic compass and ultrasonic sensor which will not give accuracy of LIDAR sensor and they are using LABVIEW as map building platform. In [8] they explained the challenges of mapping using ultrasonic sensors and they are using autonomous robot for map building .RP LIDAR which is A 2D LIDAR is used in the [7], But they are using MatLab as there map building platform and result is not as good as that we got from the ROS platform. A ROS-based human-robot interaction for indoor exploration and mapping is done on the[1]. They used an 2D LIDAR (RP LIDAR) to make 2D map of surroundings. The mapping robot needs human assistance to move around a unknown environment.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

the paper is organized as follows. in sect. ii, the robot and system overview is presented, and the software and hardware components of the system are introduced in sect. iii. more details on experiment and result of experiment we carried out. sect iv, deals with conclusion of autonomous 2d mapping.

II. ROBOT AND SYSTEM OVERVIEW

In This Section We Describe The Hardware For The Robot Which Utilized For Building The Map Building Purpose

A) Lidar Lite V3

The LIDAR used was a Garmin LIDAR Lite V3. This sensor has 40 meters of range and records data interns of 1D. It is basically a 1D data, which can be converted into a 2D volume of data with the proper rotations of the sensor. The LIDAR-Lite 3 Laser Rangefinder by Garmin is an essential, powerful, scalable and economical laser based measurement solution supporting a wide variety of applications (ex. drones, general robotics, industrial sensing and more). Measures distance, velocity and signal strength of cooperative and non cooperative targets at distances from zero to more than 40 meters. LIDAR-Lite v3 features an edge-emitting, 905nm (1.3 watts), single-stripe laser transmitter, 4 m Radian x 2 m Radian beam divergence, and an optical aperture of 12.5mm. The LIDAR-Lite operates at 5V DC with a current consumption rate of <100mA at continuous operation. On top of everything else, the LIDAR-Lite is user-configurable, allowing adjustment between accuracy, operating range and measurement time. It can be interfaced via I^{2}C or PWM. LIDAR is rotated with an Servo motor which attached to the LIDAR. Servo Rotates around a semi circumference of 180 degrees with LIDAR sensor which helps to convert 1D sensors to make 2D map

B) Arduino Mega 2560

The Arduino Mega is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega 2560 R3 also adds SDA and SCL pins next to the AREF. In addition, there are two new pins placed near the RESET pin. One is the IOREF that allow the shields to adapt to the voltage provided from the board. The other is a not connected and is reserved for future purposes. The Mega 2560 R3 works with all existing shields but can adapt to new shields which use these additional pins

C) Differential Mobile wheeled Robot

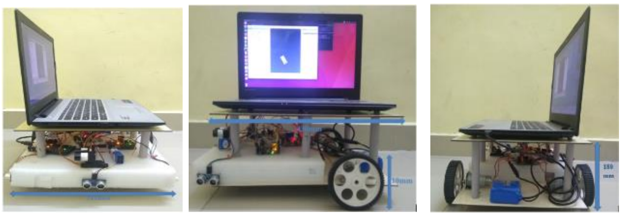

Fig 2:Fully assembled Mobile Robot

As a backbone for our robot, we decided Build our own Robot . The type of robot used in this experiment is a differential mobile robot (DMR). Right wheel and left wheel has an independent driver. The differential Mobile robot consists of 3 Wheels, One omnidirectional wheel on front and Two 10cm wheels on rear of the robot. The chassis of the Mobile robot is made by 2mm Thick mica board .The Dimension of the robot is 39x32x18 cm. The driving force for the mobile Robot given by the Two 12V side geared shaft DC Motor to the Two rear wheels. The geared motor can give the maximum stall torque of 45kg-cm.The Motor gives 30rpm at 12V and it is powered by 4.4Ah,11.1V lithium-ion battery. The Motors are controlled by the L298D Motor Driver

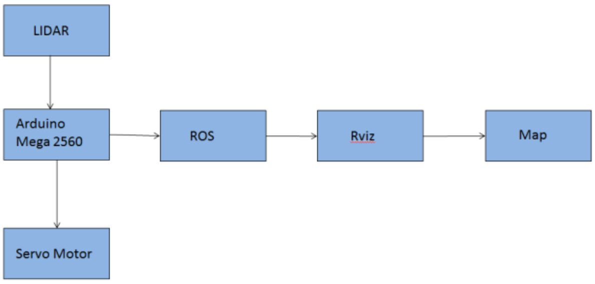

Fig 2:hardware connection for lidar

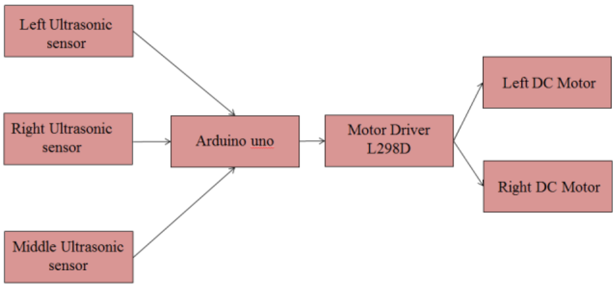

Fig 3: Hardware connection for mobile robot

The LIDAR is mounded on a Servo motor. Servo motor rotates 180 degree Servo motor is rotated continuously back and forth by the command send from the Arduino mega. The LIDAR is also connected to the Arduino Mega, For each 180 degree rotation of Servo Motor 100 points are collected. Arduino is connected to computer by serial port. After establishing serial port communication with arduino and ROS Arduino Mega send the points by Serial Port. Using the Hector Slam algorithm Map is created in Rviz. The entire setup is mounded on the Differential mobile robot. The mobile robot is controlled by the data from the Three ultrasonic sensors those situated in the middle, left side and Right side of the Mobile Robot. Distance gathered by the all the ultrasonic sensors are calculated and compared with each other in the arduino uno, After comparison corresponding command is send to the Motor driver L298D and L298D rotates the DC Motor according to the command send by the arduino. L298D is powered by an 11.1V 4.4ah battery and ultrasonic sensors, Arduino powered by two 9v batteries.

D) ROS(Robotic Operating System)

The Robot Operating System (ROS) is not an actual operating system, but a framework and set of tools that provide functionality of an operating system on a heterogeneous

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

computer cluster. Its usefulness is not limited to robots, but the majority of tools provided are focused on working with peripheral hardware. ROS is an open-source, meta-operating system for your robot. It provides the services you would expect from an operating system, including hardware abstraction, low-level device control, implementation of commonly-used functionality, message-passing between processes, and package management. Figure 4 shows the basic concept of ROS with distributed system capabilities.

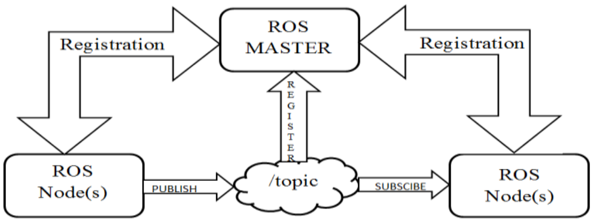

Fig 4: ROS basic concept

In ROS, a node is an instance of an executable. A node may equate to a sensor, motor, processing or monitoring algorithm, and so on. The Master is a node declaration and registration service, which makes it possible for nodes to find each other and exchange data. ROS is not a real time framework, though it is possible to integrate ROS with real time code. Data is exchanged asynchronously by means of a topic and synchronously via a service**. A** topic is a data transport system based on a subscribe/publish system. One or more nodes are able to publish data to a **topic**, and one or more nodes can read data on that topic**. A** message is a compound data structure. A message comprises a combination of primitive types (character strings, Booleans, integers, floating point, etc.) and messages (a message is a recursive structure). A topic is an asynchronous communication method used for many-to-many communication. **A** service meets a different kind of need; that for synchronous communication between two nodes. The idea is similar to that of a remote procedure call.

In support of this primary advantage of sharing and collaboration, there are several other advantages of the ROS framework:

- Thin: ROS is designed to be as thin as possible, so that code written for ROS can be used with other robot software frameworks. A corollary to this is that ROS is easy to integrate with other robot software frameworks: ROS has already been integrated with OpenRAVE, Orocos, and Player.
- ROS-agnostic libraries: the preferred development model is to write ROS-agnostic libraries with clean functional interfaces.
- Language independence: the ROS framework is easy to implement in any modern programming language.
- Easy testing: ROS has a built in unit/integration test framework called ROS TEST that makes it easy to bring up and tear down test fixtures.
- Scaling: ROS is appropriate for large runtime systems and for large development processes.

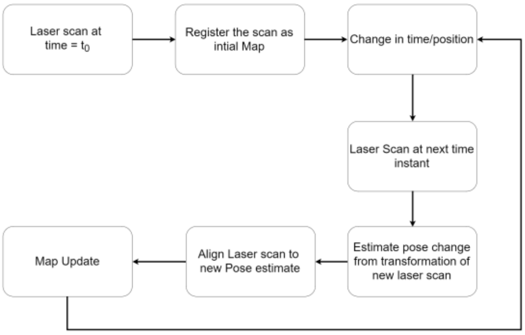
Laser scan at time = t_0
Register the scan as intial Map
Change in time/position
Laser Scan at next time instant
Estimate pose change from transformation of new laser scan
Align Laser scan to new Pose estimate
Map Update

Fig 5:Map generation

2D SLAM map generated by, performing by initially took a laser scan of the surrounding area and register that map as initial map. When robot moves next scan by the LIDAR will be different. Using a scan matching algorithm it will estimate pose change from transformation of new scan using Guass-Newton equation. After the scan matching it will align laser scan to new pose estimate and update the map. It will repeat until mapping is finished

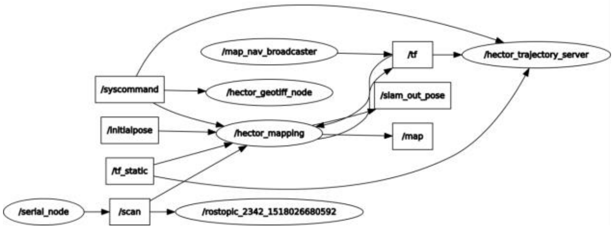

Fig 6:ROS nodes connections

Figure 6 shows the nodes and topic used for this work. The nodes used created by the developers, third party developers and some of them are created by our self specifically for this work .Rviz is a visualization node used to observe the robot's 2D reconstruction of the environment. Furthermore hector_mapping, hector_geotiff_node are part of a third party developed node called hector slam. This is a SLAM algorithm developed by Team Hector(Heterogeneous Cooperating Team of Robots) of TU Darmstadt. This node takes in the laser data from the LIDAR and, without any additional data, it is capable of making high quality maps with loop closure, with a fairly low usage of CPU and memory. There are many SLAM nodes in ROS [3]; however, the ability of Hector slam to use only the laser data to make its maps, without any requirement to utilize IMU or odometry data in the majority of use cases, is exceptional. This node allows the robot to create 2D map of environment. While, Serial node is developed our self for specifically for this work. Serial node is actually the Arduino Mega with LIDAR. The readings from the LIDAR are published to the ROS with USB 2.0 with the help of Arduino Mega 2560.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

III. RESULT

Our test area is a closed room with dimensions of 440x190 cm as shown in Fig 7. Hardware configurations is as shown per Fig 1 and configured as per Fig 2. The LIDAR with 180 degree rotating servo motor is mounted on the front of the mobile robot for mapping. The reading by the LIDAR sensor taken into the computer via serial cable USB 2.0 . Arduino Mega is used as medium between LIDAR and arduino Mega. The robot moved around the room with the room along with mapping the area. The drawing in the left side indicate the actual drawing of the room and right one indicted map obtained from the LIDAR scanner. The final map we got is comparable with the actual map

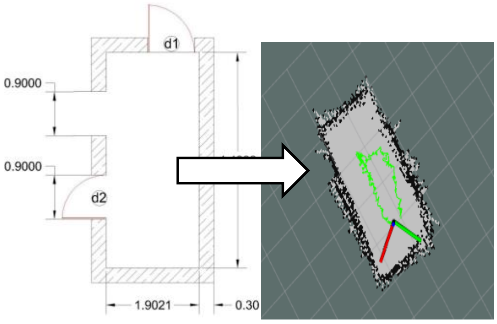

Fig 7: Actual room and map of the room

The green line shows the path of the mobile robot. The axis shape is the current position of the robot. The Mobile robot almost completed the one cycle around the room

IV . CONCLUSION

This work demonstrates that even with 1D LIDAR we can achieve 2D mapping of an unknown area. The robot is made autonomously so that robot can move around the obstacle while mapping. The final map we get by using 1D LIDAR is comparable with the map we usually get from high cost 2D mapping LIDAR. ROS is remarkable frame work in which we made the map around the robot. ROS is great tool for build robot software systems which can be useful to a variety of hardware platforms, research settings, and runtime requirements.

V. ACKNOWLEDGMENT

We want to thank, after Almighty God, Prof. KRISHNA S for dedicated supervision of this research and constant help throughout the project.

VI. REFERENCE

[1] Hendriyawan Achmad , Mohd Razali Daud, Saifudin Razali, Dwi Pebrianti, A ROS- based human-robot interaction for indoor exploration and mapping, International Journal of Advanced and Applied Sciences, 3(4) 2016, Pages: 88-92.

[2] Quigley, Morgan, et al., ROS, an open-source Robot Operating System, ICRA workshop on open source software, vol. 3, no. 3.2. 2009.

[3] Mustafa Eliwa, Ahmed Adham, Islam Sami and Mahmoud Eldeeb, A critical comparison between Fast and Hector SLAM algorithms, REST Journal on Emerging trends in Modelling and Manufacturing 3(2) 2017, 44-49

[4] Jo˜ao Machado Santos, David Portugal and Rui P. Rocha, An Evaluation of 2D SLAM Techniques Available in Robot Operating System. IEEE 2013,41-47

[5] Dhiraj Arun Patil, Sakshi Vinod Agiwal, Design and Implementation of Mapping Robot using Digital Magnetic Compass and Ultrasonic Sensor, International Journal of Engineering Research & Technology, Vol. 4 Issue 06, June-2015

[6] Muslim Tan, and Ali Yeon Md Shakaff, Indoor Scanning and Mapping using Mobile Robot and RP Lidar, Int'l Journal of Advances in Mechanical & Automobile Engg. (IJAMAE) Vol. 3, Issue 1(2016)

[7] R. Rodin and I. Štajduhar, The Challenge of Measuring Distance to Obstacles for The Purpose of Generating a 2-D Indoor Map Using an Autonomous Robot Equipped with an Ultrasonic Sensor, MIPRO 2017, May 22- 26, 2017, Opatija, Croatia

(This work is licensed under a Creative Commons Attribution 4.0 International License.)