

Volume 5 ▪ Issue 2 ▪ October 2014

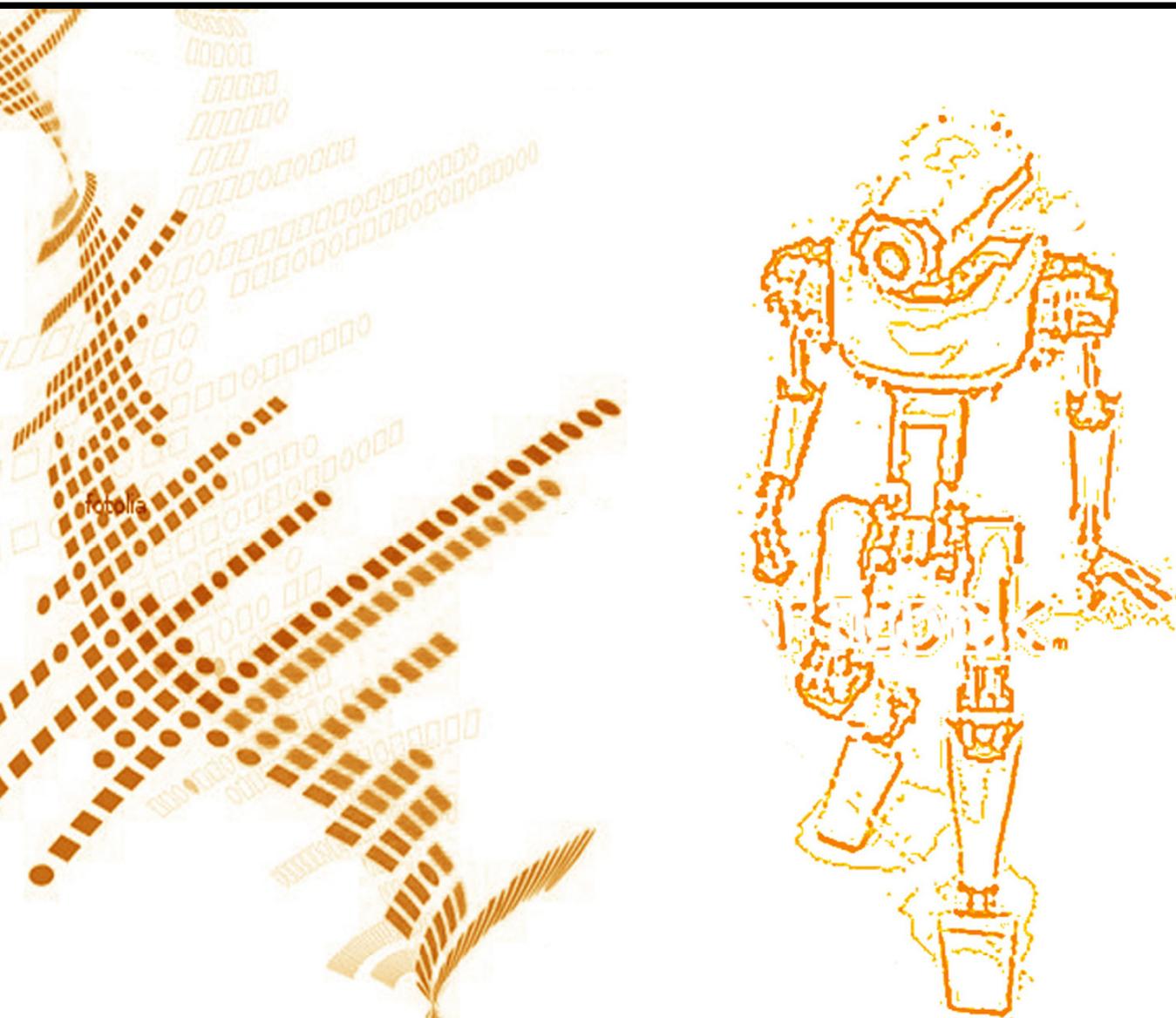
INTERNATIONAL JOURNAL OF  
ROBOTICS AND AUTOMATION (IJRA)

ISSN : 2180-1312

Publication Frequency: 6 Issues / Year

CSC PUBLISHERS

<http://www.cscjournals.org>



# **INTERNATIONAL JOURNAL OF ROBOTICS AND AUTOMATION (IJRA)**

**VOLUME 5, ISSUE 2, 2014**

**EDITED BY  
DR. NABEEL TAHIR**

ISSN (Online): 2180-1312

I International Journal of Robotics and Automation (IJRA) is published both in traditional paper form and in Internet. This journal is published at the website <http://www.cscjournals.org>, maintained by Computer Science Journals (CSC Journals), Malaysia.

IJRA Journal is a part of CSC Publishers  
Computer Science Journals  
<http://www.cscjournals.org>

# **INTERNATIONAL JOURNAL OF ROBOTICS AND AUTOMATION**

## **(IJRA)**

Book: Volume 5, Issue 2, October 2014

Publishing Date: 10-10- 2014

ISSN (Online): 2180-1312

This work is subjected to copyright. All rights are reserved whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illusions, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication of parts thereof is permitted only under the provision of the copyright law 1965, in its current version, and permission of use must always be obtained from CSC Publishers.

IJRA Journal is a part of CSC Publishers

<http://www.cscjournals.org>

© IJRA Journal

Published in Malaysia

Typesetting: Camera-ready by author, data conversation by CSC Publishing Services – CSC Journals, Malaysia

**CSC Publishers, 2014**

## **EDITORIAL PREFACE**

Robots are becoming part of people's everyday social lives - and will increasingly become so. In future years, robots may become caretaking assistants for the elderly or academic tutors for our children, or medical assistants, day care assistants, or psychological counselors. Robots may become our co-workers in factories and offices, or maids in our homes. It is the *Second Issue* of Volume *Five* of International Journal of Robotics and Automation (IJRA). IJRA published six times in a year and it is being peer reviewed to very high International standards.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Started with Volume 5, 2014, IJRA appears in more focused issues. Besides normal publications, IJRA intends to organize special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

IJRA looks to the different aspects like sensors in robot, control systems, manipulators, power supplies and software. IJRA is aiming to push the frontier of robotics into a new dimension, in which motion and intelligence play equally important roles. IJRA scope includes systems, dynamics, control, simulation, automation engineering, robotics programming, software and hardware designing for robots, artificial intelligence in robotics and automation, industrial robots, automation, manufacturing, and social implications etc. IJRA cover the all aspect relating to the robots and automation.

The IJRA is a refereed journal aims in providing a platform to researchers, scientists, engineers and practitioners throughout the world to publish the latest achievement, future challenges and exciting applications of intelligent and autonomous robots. IJRA open access publications have greatly speeded the pace of development in the robotics and automation field. IJRA objective is to publish articles that are not only technically proficient but also contains state of the art ideas and problems for international readership.

In order to position IJRA as one of the top International journal in robotics, a group of highly valuable and senior International scholars are serving its Editorial Board who ensures that each issue must publish qualitative research articles from International research communities relevant to signal processing fields.

IJRA editors understand that how much it is important for authors and researchers to have their work published with a minimum delay after submission of their papers. They also strongly believe that the direct communication between the editors and authors are important for the welfare, quality and wellbeing of the Journal and its readers. Therefore, all activities from paper submission to paper publication are controlled through electronic systems that include electronic submission, editorial panel and review system that ensures rapid decision with least delays in the publication processes.

To build its international reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJRA. We would like to remind you that the success of our journal depends directly on the number of quality articles submitted for review. Accordingly, we would like to request your participation by submitting quality manuscripts for review and encouraging your colleagues to submit quality manuscripts for review. One of the great benefits we can provide to our prospective authors is the mentoring nature of our review process. IJRA provides authors with high quality, helpful reviews that are shaped to assist authors in improving their manuscripts.

### **Editorial Board Members**

International Journal of Robotics and Automation (IJRA)

## **Editorial Board Members**

### **ASSOCIATE EDITORS (AEiCs)**

---

**Professor. Hongbo Wang**

Yanshan University  
China

### **EDITORIAL BOARD MEMBERS (EBMs)**

---

**Dr. Andrew Agapiou**

Architecture Strathclyde University  
United Kingdom

**Dr. Xianwen Kong**

Heriot-Watt University  
United Kingdom

**Dr SUKUMAR SENTHILKUMAR**

Universiti Sains Malaysia  
Malaysia

**Associate Professor. Tejbanta Chingtham**

Sikkim Manipal Institute of Technology  
India

**Dr Hassab Elgawi Osman**

The University of Tokyo, Japan  
Japan

## TABLE OF CONTENTS

Volume 5, Issue 2, October 2014

### Pages

- |         |                                                                                                                                                                                |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15 - 23 | A Novel Wall Following Algorithm For Mobile Robots<br><i>Min Raj Nepali, Nishesh Yadav, Dubey Aditya Housila Prasad, Susheel Balasubramaniah, Venkatesh EN, Ashutosh</i>       |
| 24 - 35 | A Simple Integrative Solution For Simultaneous Localization And Mapping<br><i>Min Raj Nepali, Ashutosh, Dubey Aditya Housila Prasad, Susheel Balasubramaniah, Venkatesh EN</i> |

## A Novel Wall Following Algorithm For Mobile Robots

**Min Raj Nepali**

Research Associate

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

nminraj92@gmail.com

**Nishesh Yadav**

Final Year Bachelors, EEE

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

hiddenshadowy@gmail.com

**Dubey Aditya Housila Prasad**

Research Associate

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

adityad031@gmail.com

**Susheel Balasubramaniam**

Research Associate

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

susheels88@gmail.com

**Venkatesh EN**

Research Associate

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

venkateshen1@gmail.com

**Ashutosh**

Research Associate

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

ashu.studsat@gmail.com

---

### Abstract

This paper presents how a mobile robot can be directed to follow a wall using 2D data obtained with the help of a laser range finder. The robot is controlled using a simple kinematic controller and the controller is embedded in NXP LPC1768 microcontroller. The controller guides the robot to follow the wall along the required direction while maintaining a constant distance from the wall. The data obtained from the laser range finder is analyzed and processed to generate signals to control the speed as well as orientation of the robot. Implementation results show that the proposed algorithm controls the robot to follow walls, detecting the edges and corners successfully.

**Keywords:** Kinematic Controller, 2D Laser Range Finder, Autonomous Mobile Robot, Edge Detection, Corner Detection, wall Following.

---

### 1. INTRODUCTION

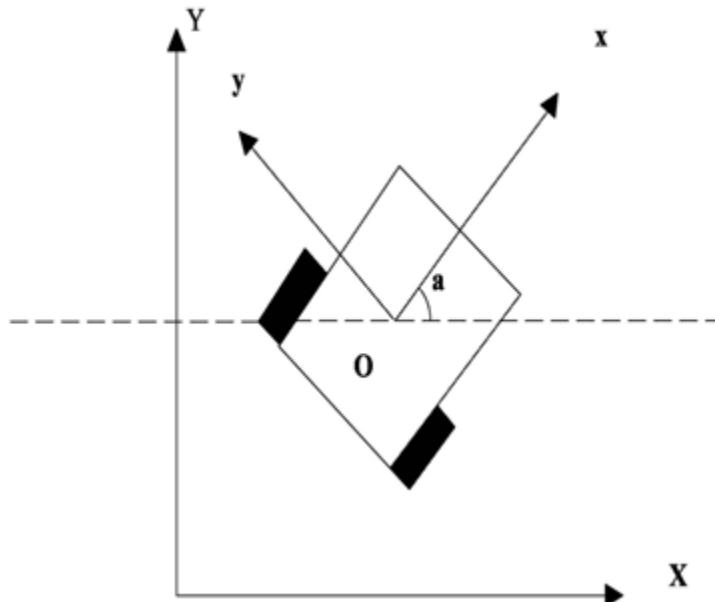
Autonomy has become a key concept in mobile robot navigation to complete desired task efficiently and with least human intuition. Mobile robots have numerous applications in various fields like space exploration, defence, education, etc. Considering different applications the robots should be incorporated with various algorithms like path planning, obstacle avoidance, etc. Wall

following is one of those algorithms which is to be implemented to make the robot autonomous. In this experiment laser range finder was used to map the environment and later on that information was used to navigate the robot and follow the wall accordingly. The perceived environment was classified as edges and corners based on different features and the robot was duly driven. Extensive research has been performed in the area of mobile robots and their applications.

A wall following method presented in [1] [2] uses IR sensors and divides the environment into a number of cells representing the wall as set of lines, points, etc. Another wall following algorithm presented in [3] [4] uses image processing to extract features like edges and corners after capturing the images or videos through a camera by using background subtraction method which later on is fed to control systems to drive the robot. Visual servoing was implemented to track and follow the wall using camera. In [5] [6] SONAR is used to perceive the environment and a Fuzzy logic controller is implemented to control the robot. Fuzzy logic controllers prove to be one of the best controllers since they represent real world knowledge and handle real world uncertainties so adeptly[7] [8].The paper [9] proposes a robust model to avoid obstacles which is further optimized to achieve wall following task by using a suitable neural controller. The sensors used to obtain information from the environment were sonar sensors.

The three important questions that have to be countered by the robot are (Where I am, Where to go, and how to go). To answer these questions correctly the robot has to have precise knowledge regarding the environment. The robot then should adjust itself relatively to the environment.

## 2. ROBOT MODEL



**FIGURE 1:** Showing the relationship between Global frame and Robot body Frame.

Throughout the experiment the robot was considered a rigid body on wheels moving on a plane surface. In order to postulate the position of the robot on the plane we create a relation between the global reference frame of the plane and the local reference frame of the robot, as in figure above. The axes X and Y define a random inertial basis on the plane as the global reference frame or the origin. To specify the position of the robot, we choose a point **O** on the robot chassis as its position reference point. The basis {x, y} defines two axes relative to **O** on the robot chassis and is thus the robot's local reference frame. The position of the robot **O** in the global reference

frame is specified by coordinates x and y, and the angular variation between the global and local reference frames is given by ‘ $\alpha$ ’. We can describe the pose of the robot as a vector with these three elements.

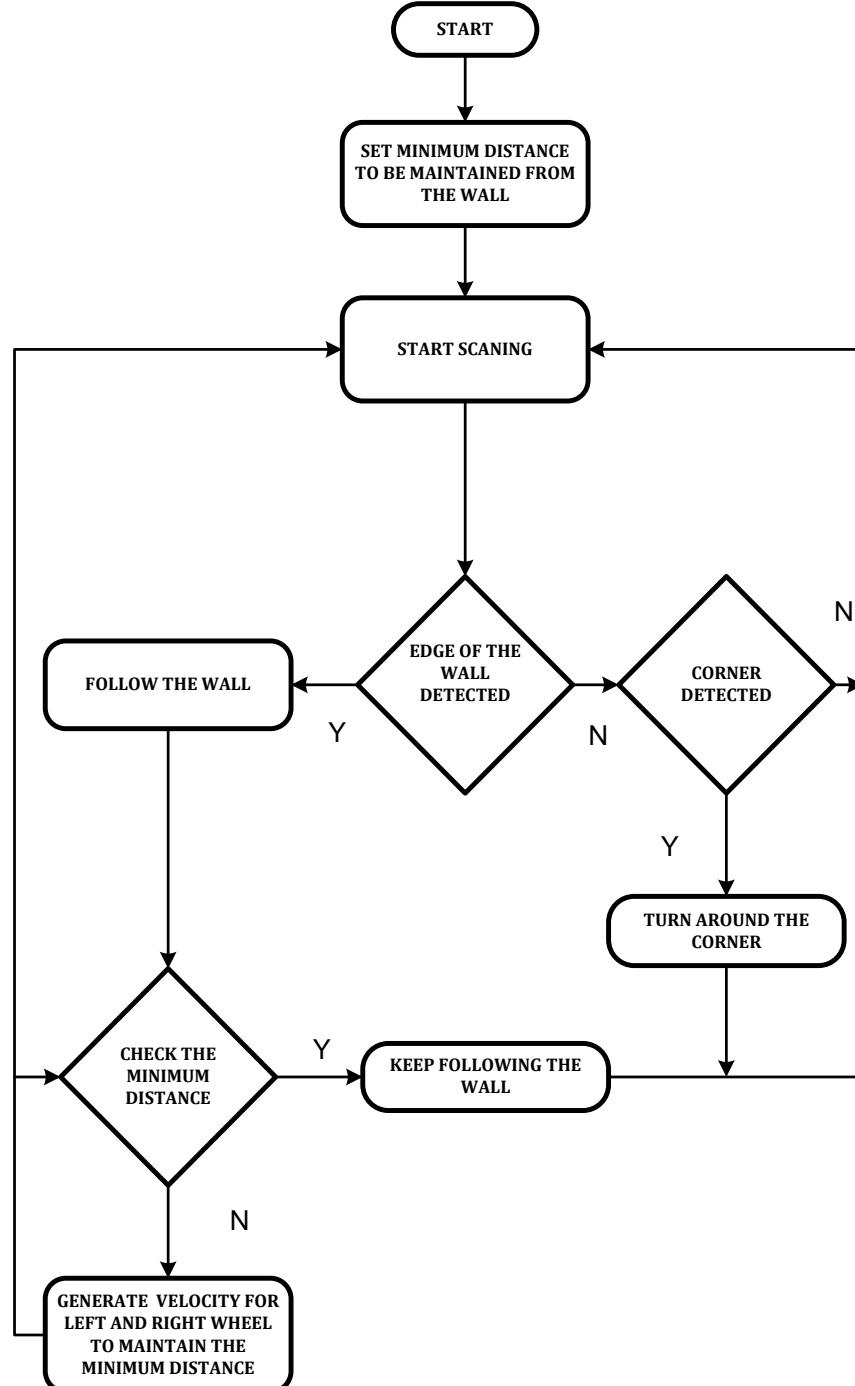
$$O_{XY} = \begin{bmatrix} x \\ y \\ \alpha \end{bmatrix}$$

To describe robot motion in terms of component motions, it will be compulsory to map motion along the axes of the global reference frame to motion along the axes of the robot's local reference frame. Of course, the mapping is a function of the current pose of the robot. This mapping is achieved by orthogonal rotation matrix:

$$R(\alpha) = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

This matrix can be used to map motion in the global reference frame {x, y} to motion in terms of the local reference frame {x, y}.

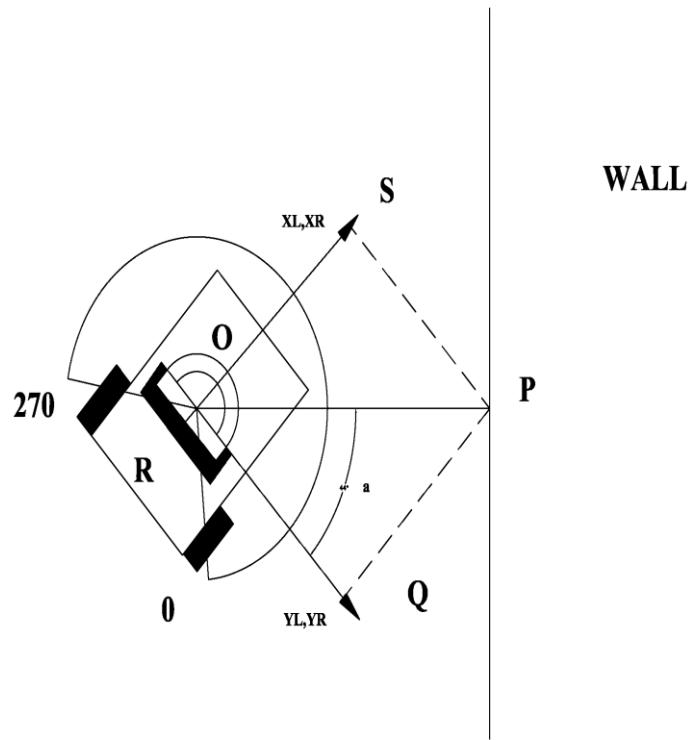
### 3. DEVELOPMENT OF THE ALGORITHM



**FIGURE 2:** Flow diagram of the controller.

The model assumes there is always a wall present somewhere in the range of the laser range scanner. The wall is perceived as a 2D plane. The wall detected is classified as edges and

corners based on different features. Segmentation, split and merge techniques may also be employed to classify the wall or to extract different features related to the wall. As far as the edge is considered the controller generates linear velocity but as soon as a corner is detected the controller generates angular velocity such that the robot is just able to tackle the corner. The laser is mounted in the robot such that the axes of the robot and the axes of the laser coincide. The laser scanner provides us with the distance between the wall and the reference point. Here the reference point is considered as the point where the laser range finder is mounted. The distance obtained from the laser range finder is resolved in its respective x component, y component and an angle from which exact location of the robot is obtained.

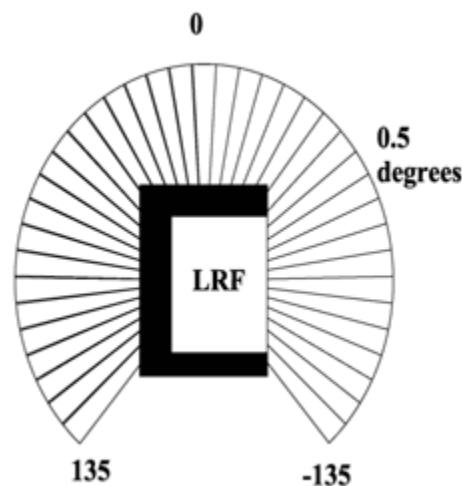


**FIGURE 3:** Robot position with respect to wall.

The object **O** mounted on the top of the robot **R** is the laser range finder. It is clearly visible that both the axes of robot and the laser are same. As shown in the figure the point **P** which is part of the wall is being resolved into **PS** and **PQ** as y and x component respectively. The robot will always try to maintain minimum distance with the wall and always try to orient **PS** perpendicular to the wall.

#### 4. CONTROLLER IMPLEMENTATION

The laser range scanner scans the environment. The scanner has capability to scan a range of 270 degrees that is from -135 to +135 as shown in the figure. The angular resolution is set to 0.5, which implies that in one scan time the laser outputs a bulk of 541 data. In this experiment two zones of 45 degrees were considered that is from -135 to -90 degrees and 135 to 90 degrees.



**FIGURE 4:** Scan range of LMS 100.

The controller is designed in such a way that it particularly takes two cases into consideration. If the wall is present on the left side of the robot, the scanned data corresponding from 135 to 90 is taken into consideration for controlling the motion of the robot. If the wall is present on the right side of the robot the scanned data corresponding from -135 to -90 is taken into consideration for controlling the motion of the robot. The controller implemented here is simple kinematic controller, which is capable of generating sufficient angular and linear velocities to drive the robot in a proper way. A threshold distance ( $d_{th}$ ) is defined for the robot to maintain while following the wall. If distance between the robot and wall is greater than the threshold distance the robot is driven towards the wall and vice-versa. The controller may be described as follows:

Where,

$V$ =linear velocity generated  
 $W$ =angular velocity generated  
 $K_V$ =velocity error gain  
 $K_d$ =distance error gain  
 $K_t$ =theta error gain

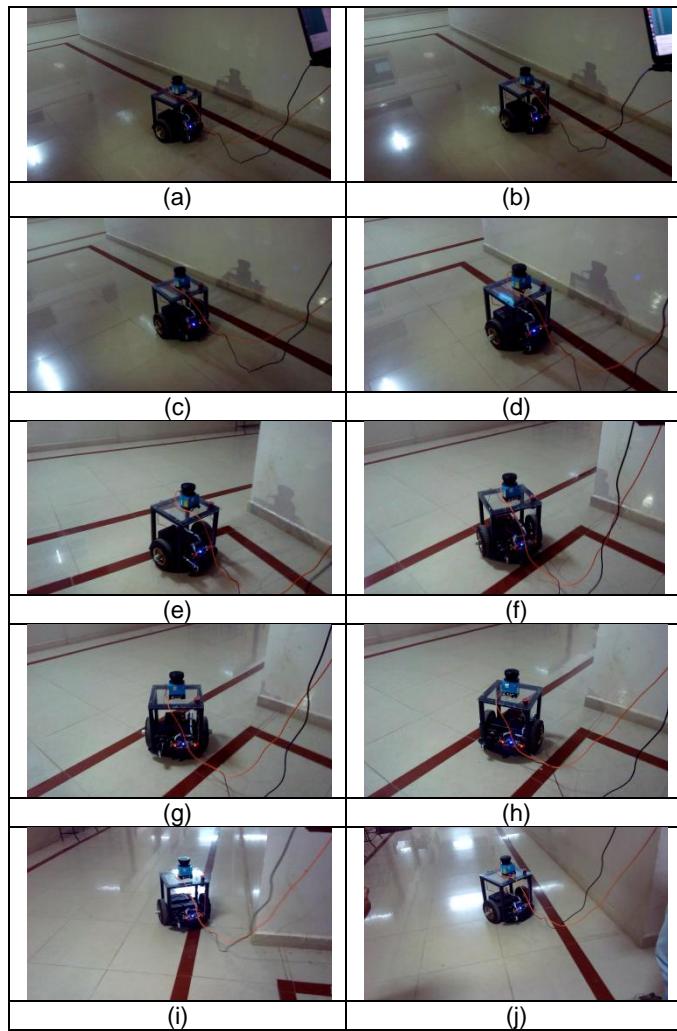
$d_{perp}$  =perpendicular or minimum distance to be maintained while following the wall  
 $d_{error}$  =error in distance  
 $a_{error}$  =error in angle

## 5. EXPERIMENTAL SETUP AND RESULTS

A series of experiments were conducted to ensure the controller works precisely. The gain values were determined by trial and error and the whole experiment was carried out in an indoor environment. The mobile robot was made to follow a series of walls and the performance of the controller was observed. The result shows that the laser is being able to detect the edges and corners successfully. After detecting the edges and corners respective control signals are being generated with the help of which the controller is driving the robot by passing down the control signals to the motors. The controller was experimentally shown to work and the series of walls were successfully followed.



**FIGURE 5:** Model of LMS 100.



**FIGURE 6:** Experimental results (a) to (j) depicting the detection of edge and corner and maneuvering the robot accordingly.

## 6. CONCLUSION AND FUTURE SCOPE

Mobile robots are capable to move around in their surrounding location and are not fixed to one location when tasks like navigation, survey, etc are considered. An autonomous robot knows some facts about where it is and how to reach various goals and or waypoints lying on its way. "Localization" or knowledge of its current position is obtained with the help of one or more means, using sensors such motor encoders, vision, lasers and global positioning systems. A novel wall following algorithm for mobile robots equipped with a 2D laser range finder has been presented. In this implementation, two cases i.e., if the left wall is present and if the right wall is present were taken into consideration. In scenarios such as if a sharp corner is present right in front of the robot; further expansion to the algorithm is required. However, the mobile robot was experimentally shown to reliably follow a series of walls given the constraints mentioned above. In an unknown environment it is easy for a robot to navigate, if it can find wall like structures and follow it. Wall following method would better help robot rather than path following if structures like wall are present in the unknown environment. It happens in many cases like underground, underwater, etc wall like structures are present and by making use of them wall following robot can easily navigate through the environment due to which the robot gains more autonomy. This kind of autonomous algorithms can be incorporated with SLAM. It would be easier to carry out the SLAM if the robot be totally autonomous. Wall follow algorithm may perform extremely well if it is carried along with mapping. From mapping we can easily obtain edges and corners of the wall and make robot to follow accordingly.

## 7. REFERENCES

- [1] P. Van Turennout, G. Honder d, L. J. vanSchelven, Wall-following control of a Mobile Robot, IEEE, International Conference on Robotics and Automation, pp. 280-285, May,1992.
- [2] J. Borenstein, Y. Kor en, The Vector Field Histogram – Fast Obstacle Avoidance for Mobile Robots, IEEE Transaction on Robotics and Automation, Vol. 7, No. 3, pp. 278-288, June,1991.
- [3] R. Frizera Vassallo, H. J. Schneebeli, J. SantosVictor, Visual navigation: combining visual servoing and appearance based methods,Proceedings of the International Symposium onIntelligent Robotic Systems, SIRS'98,Edinburgh, Scotland, July, 1998.
- [4] Francisco Bonin-Font, Alberto Ortiz, Gabriel Oliver, Visual Navigation for MobileRobots: A Survey. 2008.
- [5] R. Carelli and E. Freire, "Stable Corridor Navigation Controller for Sonar-Based Mobile Robots," Technical Report, INAW, Universidad Nacional de San Juan, San Juan, 2001.
- [6] D. Ratner and P. McKerrow,"Navigation an Outdoor Robot along Continuous Landmarks with Ultrasonic Sen- sing," Robotics and Autonomous Systems, Vol. 45, No. 1, 2003, pp. 73-82. doi:10.1016/S0921-8890(03)00096-4.
- [7] R. Malhotra and A. Sarkar, "Development of a Fuzzy Logic Based Mobile Robot for Dynamic Obstacle Avoidance and Goal Acquisition in an Unstructured Enviroment," IEEE/ASME Proceeding of International Conference on Advanced Intelligent Mechatronics, Kobe, 20- 24 July 2003, pp. 235-247.
- [8] P. M. Peri, "Fuzzy Logic Controller for an Autonomous Mobile Robot," Master Thesis, Cleveland State University, Cleveland, 2005.
- [9] R. Braunsting, J. Mujika and J. P. Uribe, "A Wall Fol- lowing Robot with a Fuzzy Logic Controller Optimized by a Genetic Algorithm," Fuzzy Systems, Vol. 5, 1995, pp. 77-82.
- [10] M. Ergezer, "Multivariable Control Method for Wall Tracking Robot," Project Paper,Department of Electrical and Computer Engineering, Cleveland State University, Cleveland, 2006.
- [11] R. Benporad, M. Di Marco and A. Tesi, "Wall-Following Controller for Sonar-Based Mobile Robots," Proceeding of IEEE Conference on Decision and Control, San Diego, 10-12 December 1997, pp. 3063-3068.
- [12] Specifications and Operating Commands for SICK LMS100 Laser Range Finder. Can be found on [www.sick-automation.ru/images/File/pdf/DIV05/LMS100\\_manual.pdf](http://www.sick-automation.ru/images/File/pdf/DIV05/LMS100_manual.pdf).

- [13] R. Clark, A. El-Oser y, K. Wedeward and S.Bruder, A Navigation and Obstacle Avoidance Algorithm for Mobile Robots Operating inUnknown, Maze-Type Environments, Proc.International Test and Evaluation AssociationWorkshop on Modeling and Simulation, LasCruces, NM, December 2004.

# A Simple Integrative Solution For Simultaneous Localization And Mapping

## Min Raj Nepali

Research Associate

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

nminraj92@gmail.com

## Dubey Aditya Housila Prasad

Research Associate

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

adityad031@gmail.com

## Susheel Balasubramaniam

Research Associate

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

susheels88@gmail.com

## Venkatesh EN

Research Associate

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

venkateshen1@gmail.com

## Ashutosh

Research Associate

Nitte Meenakshi Institute of Technology  
Bangalore, 560064, India

ashu.studsat@gmail.com

## Abstract

Simultaneous Localization and Mapping is a method used to find the location of a mobile robot while at the same time build a constructive map of its surrounding environment. This paper gives a brief description about a simple integrative SLAM technique using a Laser Range Finder (LRF) and Odometry data, primarily for indoor environments. In this project, a solution for the SLAM problem was implemented on a differential drive mobile robot equipped with a SICK laser scanner.

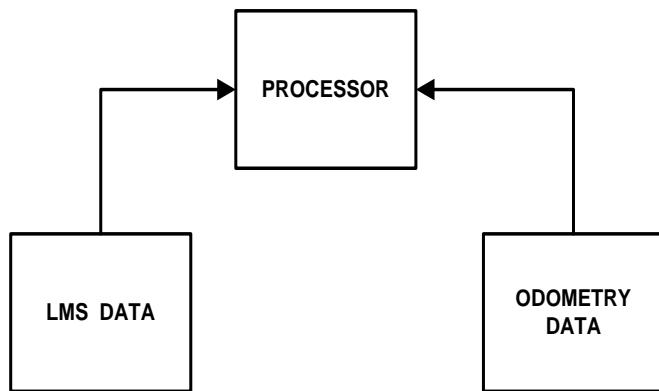
**Keywords:** Localization, LRF, Feature Based Mapping, Odometry, SLAM, Split and Merge.

## 1. INTRODUCTION

Simultaneous Localization and Mapping (SLAM) is a method used by mobile robot placed in an unknown location in an unknown environment to incrementally build a consistent map of this environment while simultaneously keeping the track of current location within this map. An intuitive understanding of the SLAM process can be conveyed through following example: Consider a simple mobile robot: a set of wheels connected to a motor and a laser range finder (LRF). The LRF sends out series of beams at small regular intervals and obtains the distance of objects from it at a given angular resolution. This information is stored in a file that can be used to build the map.

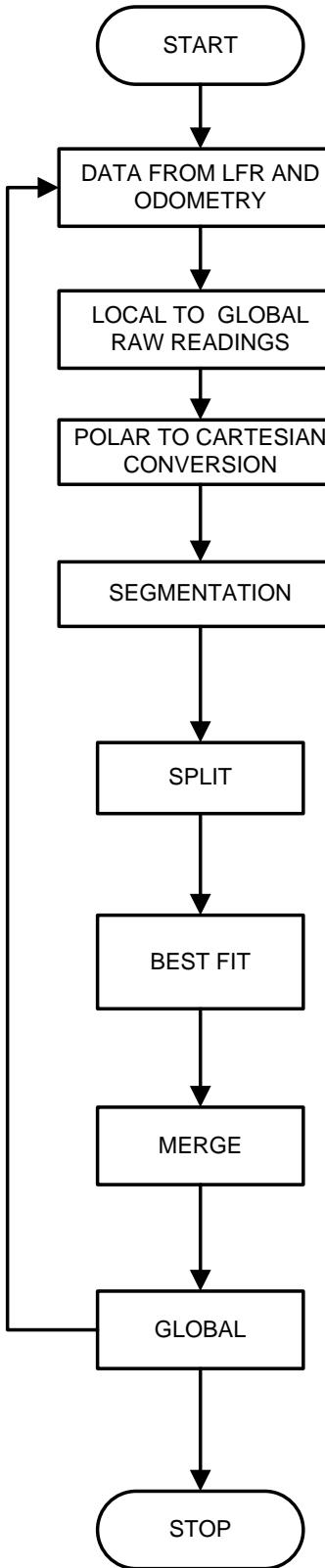
Rather than keep track of a large number of points to build a map (termed as point-based mapping) we instead extract geometric features from the information obtained and keep track of those instead. This results in a significant reduction in computational complexity. The geometric features chosen in this implementation are lines, since they are present predominantly in indoor environments.

The SICK LMS-100 Laser range finder is used in this implementation. It has a 270° field view around the robot with angular step readings at 0.5°/0.25° resolution. The long range scan of about 20m makes it distinctive for its selection. Wheel encoders allow us to ascertain the orientation and speed of the robot. Motor drivers are used to control the speed of the wheels using a PID algorithm. The encoders from which odometry data is ascertained and the drivers are interfaced with a microcontroller.



**FIGURE 1:** System setup.

The microcontroller transmits Odometry data to the computer for localization. The flowchart describing the whole process is shown below.

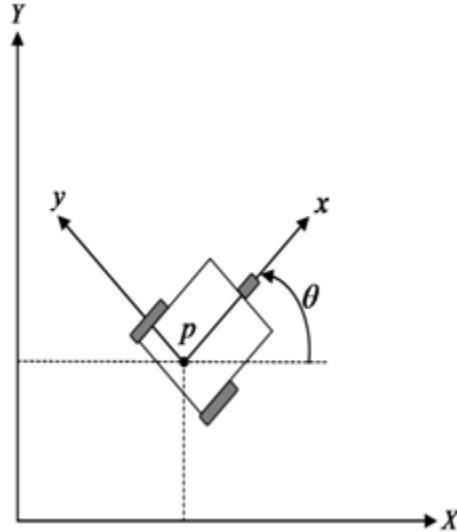


**FIGURE 2:** Flowchart of SLAM process.

## 2. ODOMETRIC LOCALIZATION

### 2.1. Representing Robot Position

The axes X and Y state an random inertial basis on the plane as the global reference frame or the origin.



**FIGURE 3:** Global reference frame and the robot local reference frame.

To state the location of the robot, we select a point P on the robot frame as its position reference point. {x, y} outlines two axes relative to P on the robot frame and is thus the robot's local reference frame. The position of the robot P in the global reference frame is detailed by coordinates x and y, and the angular alteration between the global and local reference frames is given by  $\theta$ . We can describe the pose of the robot as a vector with these three elements.

$$P_{xy} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (1)$$

To describe robot motion in terms of component motions, it will be needed to map motion along the axes of the global reference frame to motion along the axes of the robot's local reference frame. This mapping is obtained using the orthogonal rotation matrix:

$$R(\theta) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

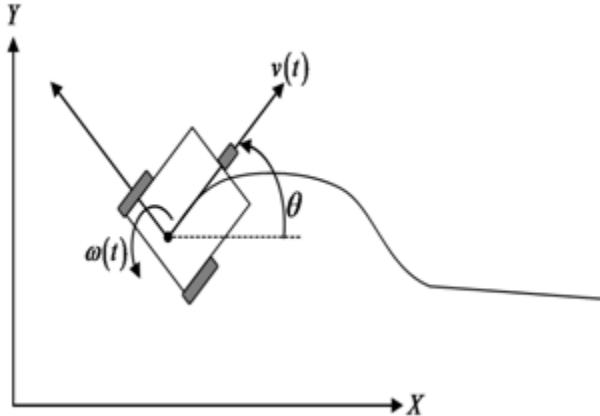
This matrix can be used to map motion in the global reference frame {x, y} to motion in terms of the local reference frame {x, y}. This operation is denoted by  $R(\theta) P_{xy}$  because the computation of this operation depends on the value of  $\theta$ :

$$P_{xy} = R(\theta) P_{xy} \quad (3)$$

### 2.2. Forward Kinematic Model

The forward kinematics of the mobile robot captures how the robot moves, given its geometry and the speeds of its wheels. For example, the differential drive robot has two wheels, with radius  $r_r$  and  $r_l$ . Robot position reference point P centered between the two drive wheels, and distance between the wheels I. Given  $r_r$ ,  $r_l$ , I,  $\theta$  and the spinning speed of each wheel,  $\varphi_r$  and  $\varphi_l$ , a forward

kinematic model would predict the robot's total speed in the global frame of reference. i.e  $P_{xy} \{x, y, \theta\}$ .



**FIGURE 4:** A differential-drive robot in its global reference frame.

From  $P_{xy}$  we know that we can compute the robot's movement in the global reference frame from motion in its local reference frame:  $\dot{P}_{xy} = R(\theta)^{-1} \dot{P}_{xy}$ .

Suppose that the robot's local reference frame is aligned such that the robot moves forward along  $+x$ , as shown in figure. First consider the contribution of each wheel's spinning speed to the translation speed at  $P$  in the direction of  $+x$ . If one wheel spins while the other wheel contributes nothing and is stationary, since  $P$  is halfway between the two wheels, it will move instantaneously with half the speed:  $\dot{x}_r = (1/2)r_r \varphi_r$  and  $\dot{x}_l = (1/2)r_l \varphi_l$ . In a differential drive robot, these two contributions can simply be added to calculate the  $\dot{x}$  component of  $\dot{P}_{xy}$ . Consider, for example, a differential robot in which each wheel spins with equal speed but in opposite directions. The result is a stationary, spinning robot. As expected,  $\dot{x}$  will be zero in this case.. Neither wheel can contribute to sideways motion in the robot's reference frame, nor so  $\dot{y}$  is always zero. Finally, we must compute the rotational component  $\dot{\theta}$  of  $\dot{P}_{xy}$ . Once again, the contributions of each wheel can be computed independently and just added. Consider the right wheel (wheel 1). Forward spin of this wheel results in counter-clockwise rotation at point  $P$ . Recall that if the right wheel spins alone, the robot pivots around the left wheel. The rotation velocity  $\omega_r$  at  $P$  can be computed because the wheel is instantaneously moving along the arc of a circle of radius  $2l$ :

$$\omega_r = \frac{r_r \varphi_r}{2l} \quad (4)$$

The same calculation applies to the left wheel, with the exception that forward spin results in clockwise rotation at point  $P$ :

$$\omega_l = \frac{r_l \varphi_l}{2l} \quad (5)$$

Combining these individual formulas yields a kinematic model for the differential-drive robot:

$$\dot{P}_{xy} = R(\theta)^{-1} \begin{bmatrix} \frac{r_r \varphi_r + r_l \varphi_l}{2} \\ 0 \\ \frac{r_r \varphi_r - r_l \varphi_l}{2l} \end{bmatrix} \quad (6)$$

### 3. MOTION MODEL

For a differential-drive robot the position can be estimated starting from a known position by integrating the movement (summing the incremental travel distances). For a discrete system with a fixed sampling interval  $\Delta t$  the incremental travel distances ( $\Delta x, \Delta y, \Delta \theta$ ) are:

$$\Delta x = \Delta s \cos\left(\theta + \frac{\Delta \theta}{2}\right) \quad (7)$$

$$\Delta y = \Delta s \sin\left(\theta + \frac{\Delta \theta}{2}\right) \quad (8)$$

$$\Delta \theta = \frac{\Delta s_r - \Delta s_l}{l} \quad (9)$$

$$\Delta s = \frac{\Delta s_r + \Delta s_l}{2} \quad (10)$$

Where,

$(\Delta x, \Delta y, \Delta \theta)$  = Path traveled in the last sampling interval.

$(\Delta s_r, \Delta s_l)$  = Travelled distances for the right and left wheel respectively.

Thus we get the updated position  $P(k+1)$

$$p(k+1) = p(k) + \begin{bmatrix} \Delta s \cos\left(\theta + \frac{\Delta \theta}{2}\right) \\ \Delta s \cos\left(\theta + \frac{\Delta \theta}{2}\right) \\ \Delta \theta \end{bmatrix} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \Delta s \cos\left(\theta + \frac{\Delta \theta}{2}\right) \\ \Delta s \cos\left(\theta + \frac{\Delta \theta}{2}\right) \\ \Delta \theta \end{bmatrix} \quad (11)$$

By using the relation for  $\Delta s$  and  $\Delta \theta$ , we further obtain the basic equation for odometric position update (for differential drive robots):

$$p(k+1) = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \frac{\Delta s_r + \Delta s_l}{2} \cos\left(\theta + \frac{\Delta s_r - \Delta s_l}{l}\right) \\ \frac{\Delta s_r + \Delta s_l}{2} \sin\left(\theta + \frac{\Delta s_r - \Delta s_l}{l}\right) \\ \frac{\Delta s_r - \Delta s_l}{l} \end{bmatrix} \quad (12)$$

As we discussed earlier, odometric position updates can give only a very rough estimate of the actual position. Owing to integration errors of the uncertainties of  $p(k)$  and the motion errors during the incremental motion( $\Delta s_r, \Delta s_l$ ), the position error based on odometry integration grows with time. In the next step we will establish an error model for the integrated position  $p(k+1)$  to obtain the covariance matrix of the odometric position estimate  $p(k+1)$ . To do so, we assume that at the starting point the initial covariance matrix  $P(k)$  is known. For the motion increment ( $\Delta s_r, \Delta s_l$ ) we assume the following covariance matrix Q

$$Q = cov(\Delta s_r, \Delta s_r) = \begin{bmatrix} k_r |\Delta s_r| & 0 \\ 0 & k_l |\Delta s_l| \end{bmatrix} \quad (13)$$

Where  $k_r$  and  $k_l$  are error constants representing the nondeterministic parameters of the motor drive and the wheel-floor interaction. As you can see, in equation above we made the following assumptions: (i) The two errors of the individually driven wheels are independent; (ii) The variance of the errors (left and right wheels) is proportional to the absolute value of the travelled distances. These assumptions, while not perfect, are suitable and will thus be used for the further development of the error model. The *motion errors* are due to imprecise movement because of deformation of wheel, slippage, unequal floor, errors in encoders, and so on. The values for the

error constants  $k_r$  and  $k_l$  depend on the robot and the environment and should be experimentally established by performing and analysing representative movements.

If we assume that  $P$  and  $q = (\Delta s_r; \Delta s_l)$  are uncorrelated and the derivation of  $f$  is reasonably approximated by the first-order Taylor expansion (linearization), we conclude, using the error propagation law:

$$P(k+1) = F_p(k) \cdot P(k) \cdot F_p^T(k) + F_q(k) \cdot Q \cdot F_q^T(k) \quad (14)$$

$$F_p(k) = \begin{bmatrix} \frac{\partial p}{\partial x} & \frac{\partial p}{\partial y} & \frac{\partial p}{\partial \theta} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\Delta s \sin\left(\theta + \frac{\Delta\theta}{2}\right) \\ 0 & 1 & \Delta s \sin\left(\theta + \frac{\Delta\theta}{2}\right) \\ 0 & 0 & 1 \end{bmatrix} \quad (15)$$

$$F_q(k) = \begin{bmatrix} \frac{\partial p}{\partial \Delta s_r} & \frac{\partial p}{\partial \Delta s_l} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \cos\left(\theta + \frac{\Delta\theta}{2}\right) - \frac{\Delta s}{2b} \sin\left(\theta + \frac{\Delta\theta}{2}\right) & \frac{1}{2} \cos\left(\theta + \frac{\Delta\theta}{2}\right) + \frac{\Delta s}{2b} \sin\left(\theta + \frac{\Delta\theta}{2}\right) \\ \frac{1}{2} \sin\left(\theta + \frac{\Delta\theta}{2}\right) + \frac{\Delta s}{2b} \cos\left(\theta + \frac{\Delta\theta}{2}\right) & \frac{1}{2} \sin\left(\theta + \frac{\Delta\theta}{2}\right) - \frac{\Delta s}{2b} \cos\left(\theta + \frac{\Delta\theta}{2}\right) \\ \frac{1}{b} & -\frac{1}{b} \end{bmatrix} \quad (16)$$

Once the error model has been established, the error parameters must be specified. One can compensate for deterministic errors properly calibrating the robot. However the error parameters specifying the nondeterministic errors can only be quantified by statistical (repetitive) measurements.

## 4. MAPPING

Mapping is the subject of integrating the readings gathered with the robot's sensors into a given representation. Principal aspects in mapping are the representation of the surroundings and the interpretation of sensor data. While on the other hand, localization is the problem of guesstimating the pose of the robot relative to a map. Normally, one identifies between pose tracking, where the initial pose of the vehicle is well-known, and global localization, in which no a priori knowledge about the starting position is set.

### 4.1. Feature Based Mapping

The primary issue is how to accurately match sensed data against information in a priori map or information that has been collected so far. There are two common matching techniques that have been used in mobile robotics: point based matching and feature-based matching. Instead of working directly with raw scan points, feature based matching first transforms the raw scans into geometric features. These extracted features are used in the matching in the next step. This approach has been studied and employed intensively in recent research on robot localization, mapping, feature extraction, etc. Being more compact than they require much less storage and still provide rich and accurate information, algorithms based on parameterized geometric features are expected to be more efficient compared to point-based algorithms.

Among many geometric primitives, line segment is the simplest one. It is easy to describe most office environments using line segments. A paper by Nguyen et al [2005] comes to the conclusion that the split and merge and *Incremental* are the preferred candidates for SLAM, because of their speed and good correctness. They mention that for real-time applications, Split-and-Merge is clearly the best choice by its superior speed. It is also the first choice for localization problems with a priori map, where false pose is not very important.

### 4.2. Landmark Extraction

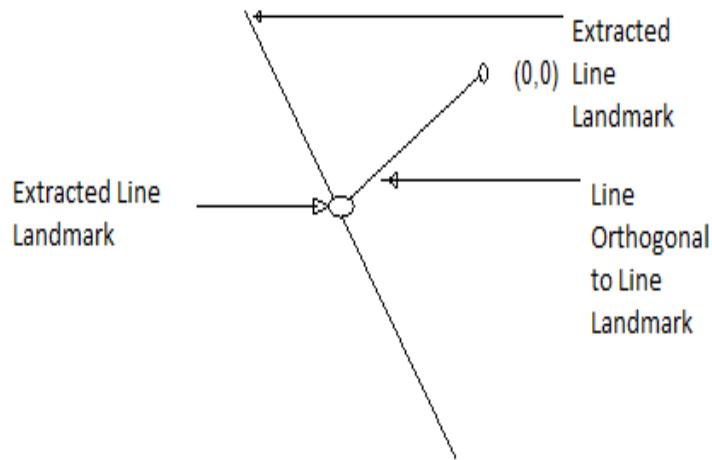
Landmarks are features which can easily be re-observed and distinguished from the environment. These are used by the robot to find out where it is (to localize itself). Landmarks should be re-

observable by allowing them for example to be viewed (detected) from different positions and thus from different angles.

Landmarks should be unique enough so that they can be easily identified from one time-step to another without mixing them up. In other words if you re-observe two landmarks at a later point in time it should be easy to determine which of the landmarks is which of the landmarks we have previously seen. If two landmarks are very close to each other this may be hard.

One can easily translate a line into a fixed point by taking another fixed point in the world coordinates and calculating the point on the line closest to this fixed point. Using the robots position and the position of this fixed point on the line it is trivial to calculate a range and bearing from this. Using simple trigonometry one can easily calculate this point.

Here illustrated using the origin as a fixed point:



**FIGURE 4:** Getting extracted line landmark as a point.

#### 4.3. Map Building

The sensor used to measure environment information and build environment map is a 2D laser scanner SICK. It has a scanning angle of 270 degrees with an angular resolution of 0.5 degree and a range error of less than 5mm. A local map is the representation of environment that laser range scanner perceives from its current position. A global map is the representation of environment that has been perceived previously. It is yielded by merging all local maps built previously.

#### 4.4. Local Map Building

The formation of a local map consists of a number of steps and is implemented as follows:

##### i) Clustering of points into segments

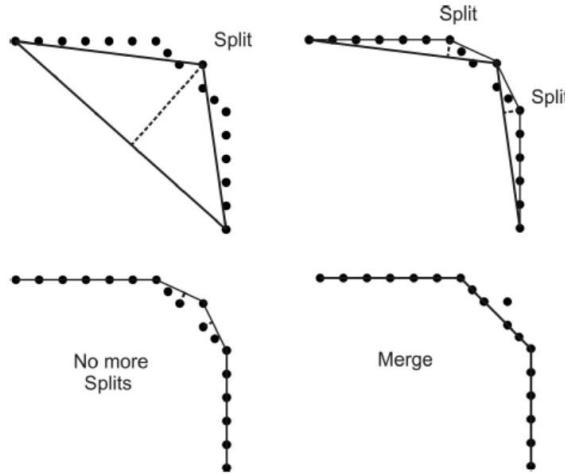
From the first to the last point, if the distance between two neighbouring points is great than a threshold, then whole scan is split between these two points. After testing all distance of neighbouring points, a scan is split into different regions, denoted with  $R_i$ . Each region  $R_i$  consists of a set of points from  $S_i$  to  $E_i$ , where  $S_i$  and  $E_i$  are respectively the first and the last point of the segment. Removal of Stray Segments: The number of points in each segment is counted. If the number of points in each segment is less than a given threshold, the segment is removed.

##### ii) The Split-and-Merge Algorithm

The algorithm splits the initial line by searching the particular point in between with the largest perpendicular line distance. This point is set as a new vertex, respectively a new segment endpoint, and builds two new segments with the former given endpoints. For the thereby new

generated segments the algorithm searches again the vertices. The algorithm stops the recursion in case that the distance to the farthest found point lies below a predefined threshold. The initial line is specified by the first and the last captured scan point. These therefore build the first and respectively the last vertex of our resulting Polyline.

The ‘Merging’ component of the Split-and-Merge Algorithm involves calculating the distance between the two adjacent points of each vertex derived from the Split procedure. If the distance is found to be greater than a given threshold, the vertex is considered a stray point, and is ignored.



**FIGURE 5:** Split and Merge (Iterative end point fit).

### iii) Finding the best-fit line using the method of least squares

Having obtained the points using the split-and –merge algorithm that correspond to the vertices in each segment, the best fit line in the least-squared sense is obtained, considering all the points in each segment derived.

The basic problem is to find the best fit straight line  $y = ax + b$  given that, for  $n \in \{1, \dots, Ng\}$ , the pairs  $(x_n, y_n)$  are observed. The method easily generalizes to finding the best fit of the form

$$y = a_1 f_1(x) + \dots + c_k f_k(x) \quad (17)$$

It is not necessary for the functions  $f_k$  to be linearly in  $x$  – all that is needed is that  $y$  is to be a linear combination of these functions. It can be shown that the expression for the landmark extracted ( $r$  and  $\alpha$ ) from the feature line is as follows:

$$\alpha = \frac{1}{2} \operatorname{atan} 2 \left\{ \frac{-2 \sum (\bar{y} - y_i)(\bar{x} - x_i)}{\sum (\bar{y} - y_i)^2 - (\bar{x} - x_i)^2} \right\} \quad (18)$$

$$r = \bar{x} \cos(\alpha) + \bar{y} \sin(\alpha) \quad (19)$$

$$\bar{x} = \frac{1}{n} \cdot \sum x_i \quad \bar{y} = \frac{1}{n} \cdot \sum y_i \quad (20)$$

Where

$r, \alpha$  - range and bearing of the extracted feature point  
 $x, y$  - x and y co-ordinates of all points in the line segment

iv) Finding projections of the line segments on the best-fit line

Having found out the parameters of the best fit line, which also are the feature points extracted, we find the projection of the vertices on the best fit line in the following manner:

$$m = -\frac{\cos\alpha}{\sin\alpha} = -\sec\alpha \quad (21)$$

$$c = \frac{R}{\sin\alpha} \quad (22)$$

$$y_s = my_p + mx_p + \frac{c}{m^2+1} \quad (23)$$

$$x_s = my_p + x_p - \frac{mc}{m^2+1} \quad (24)$$

But since if  $\alpha = 0, 180$  or  $-180$  we obtain an exception, we use the following formula in these cases:

$$X_{s/e} = X_{start/end} \quad (25)$$

$$Y_{s/e} = Y_{start/end} \quad (26)$$

#### 4.5. Global map building

The main problem of global map updating is to solve the correspondence problem between line segments respectively from local map and global map. If a line segment in local map has corresponding line segment in current global map, then they are merged into one to update this line segment in current global map. Those line segments in local map that have not corresponding line segment in current global map are inserted into current global map according to their ordinal relation. The global map is built by merging repeatedly local maps. The ordinal relations of line segments are retained in merging.

To determine whether given pair of lines is sufficiently similar to warrant merging, we apply a merge criterion based on the chi-squared test. The coordinates and covariance matrices of the two lines as found by our line fitting algorithm are first represented with respect to a common pose  $i$ . We then apply the chi-squared test to determine if the difference between two lines is within the 3 sigma deviance threshold defined by the combined uncertainties of the lines.

The merge criterion is:

$$\chi^2 = (\delta L)^T (P_{L1}^i + P_{L2}^i)^{-1} \delta L < 3 \quad (27)$$

Where

$P_{L1}^i, P_{L2}^i$  - corresponds to the covariance matrices of the landmark in the local and global frames respectively. The difference between the landmark co-ordinates in the local and global frames are given as follows:

$$\delta L = \begin{bmatrix} R_1^i - R_2^i \\ \alpha_1^i - \alpha_2^i \end{bmatrix} \quad (28)$$

If the chi-squared test is satisfied, the local and global co-ordinates are merged to obtain a single landmark.

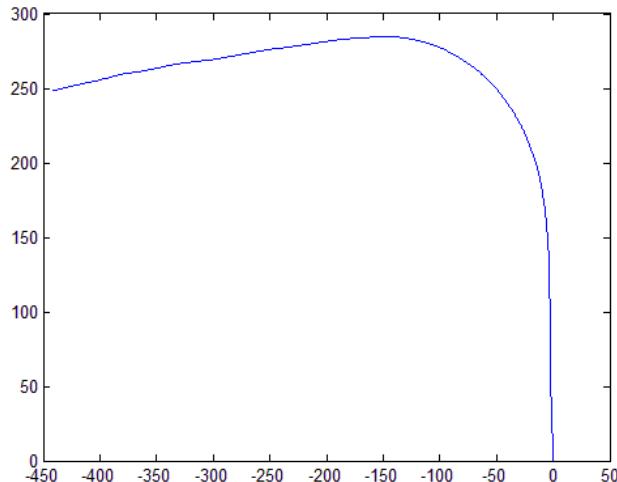
The final merged landmark co-ordinates and covariance matrix is given by:

$$L_m^i = P_{L_m}^i ((P_{L_1}^i)^{-1} L_1^i + (P_{L_2}^i)^{-1} L_2^i) \quad (29)$$

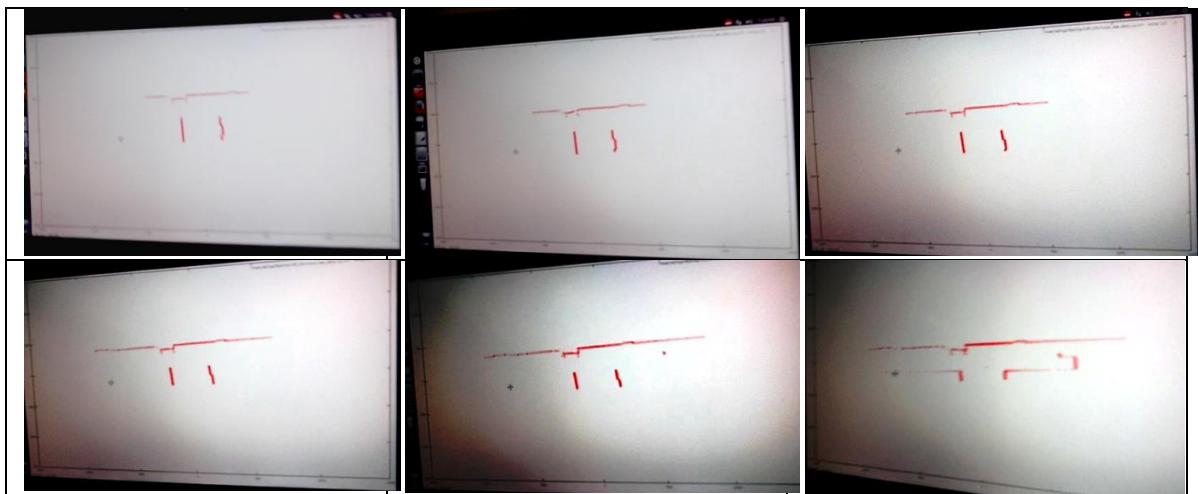
$$P_{L_m}^i = ((P_{L_1}^i)^{-1} + (P_{L_2}^i)^{-1})^{-1} \quad (30)$$

## 5.RESULTS AND ANALYSIS

The mobile robot was placed in a corridor. A wall following algorithm was implemented such that the robot followed the wall observed to the right. Fig (7) illustrates the odometry path followed by the robot following the wall and Fig (8) shows the map obtained, until the robot veered dangerously close to a staircase.



**FIGURE 6:** Localization data from odometry.



**FIGURE 7:** Plot of the map obtained after Global Mapping.

## 6. CONCLUSION

This paper illustrates a simplistic SLAM solution along with its implementation and results. The split-and-merge line feature and landmark extraction algorithms are explained in detail and are found to be accurate. The global map generated is found to be consistent in small-scale environments.

It is possible to refine the solution provided in this paper by implementing extended Kalman filters, in order to obtain the best estimate of the robot position and map points. Different algorithms that consider the SLAM process a non-linear process such as FAST-SLAM may be implemented as well. This current technology can be used in various applications such as aerospace, a mobile robot on the moon, a coal mine where human entry is difficult, underwater monitoring in small and difficult to reach places.

## 7. REFERENCES

- [1] H. F. Durrant-Whyte, "Where am I? A Tutorial on Mobile Vehicle Localization," *Industrial Robot*, vol. 21, no. 2, pp. 11±16, 1994.
- [2] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. "FastSLAM: A factored solution to the simultaneous localization and mapping problem". *AAAI National Conference on Artificial Intelligence*, pages 593–598, 2002.
- [3] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.
- [4] Lina M. Paz, Pedro Pini'és, Juan D. Tard'os, and Jos'eNeira, Large- Scale 6-DOF SLAM With Stereo-in-Hand, *IEEE Transactions On Robotics*, Vol. 24, No. 5, October 2008
- [5] J.A. Castellanos and J.D. Tard'os. "Mobile Robot Localization and Map Building: A Multisensor Fusion Approach". *Kluwer Academic Publishers*, Boston, MA, 2000.
- [6] W. Burgard, D. Fox, H. Jans, C. Matenar, and S. Thrun. Sonarbased mapping with mobile robots using EM. In Proc. 16th International Conf. on Machine Learning, pages 67–76. Morgan Kaufmann, San Francisco, CA, 1999.
- [7] S. Thrun, D. Fox, and W. Burgard, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Machine Learning* 31, 29–53 and *Autonomous Robots* 5, 253–271,(joint issue), 1998.
- [8] Jose Guivant, Eduardo Nebot and Stephan Baiker. "Autonomous Navigation and Map building Using Laser Range Sensors in Outdoor Applications". *Journal of Robotic Systems*, Volume 17, No 10, October 2000
- [9] Jaun D. Tardos , Jose Neira , Paul M. Newman , John J. Leonard "Robust Mapping and Localization in Indoor Environments using Sonar Data." *The International Journal of Robotics Research* Vol. 21, No. 4, April 2002.

## **INSTRUCTIONS TO CONTRIBUTORS**

Robots are becoming part of people's everyday social lives - and will increasingly become so. In future years, robots may become caretaking assistants for the elderly or academic tutors for our children, or medical assistants, day care assistants, or psychological counselors. Robots may become our co-workers in factories and offices, or maids in our homes.

The International Journal of Robotics and Automation (IJRA), a refereed journal aims in providing a platform to researchers, scientists, engineers and practitioners throughout the world to publish the latest achievement, future challenges and exciting applications of intelligent and autonomous robots. IJRA is aiming to push the frontier of robotics into a new dimension, in which motion and intelligence play equally important roles. IJRA scope includes systems, dynamics, control, simulation, automation engineering, robotics programming, software and hardware designing for robots, artificial intelligence in robotics and automation, industrial robots, automation, manufacturing, and social implications.

To build its International reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJRA.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Started with Volume 5, 2014, IJRA appears in more focused issues. Besides normal publications, IJRA intends to organize special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

We are open to contributions, proposals for any topic as well as for editors and reviewers. We understand that it is through the effort of volunteers that CSC Journals continues to grow and flourish.

### **IJRA LIST OF TOPICS**

The realm of International Journal of Robotics and Automation (IJRA) extends, but not limited, to the following:

- Automation Control
- Autonomous Robots
- Emergence of The Thinking Machine
- Household Robots and Automation
- Jacobian and Singularities
- Nanotechnology & Robotics (Nanobots)
- Robot Controller
- Robotic & Automation Software Development
- Robotic Surgery
- Robotic Welding
- Robotics Programming
- Robots Society and Ethics
- Spatial Transformations
- Unmanned (Robotic) Vehicles
- Automation Engineering
- Biotechnology & Robotics
- Forward Kinematics
- Inverse Kinematics
- Methods for Teaching Robots
- Orientation Matrices
- Robot Structure and Workspace
- Robotic Exploration
- Robotic Surgical Procedures
- Robotics Applications
- Robotics Technologies
- Software and Hardware Designing for Robots
- Trajectory Generation

## **CALL FOR PAPERS**

---

**Volume: 5 - Issue: 3**

**i. Paper Submission:** October 31, 2014

**ii. Author Notification:** November 30, 2014

**iii. Issue Publication:** December 2014

## **CONTACT INFORMATION**

### **Computer Science Journals Sdn BhD**

B-5-8 Plaza Mont Kiara, Mont Kiara  
50480, Kuala Lumpur, MALAYSIA

Phone: 00603 6204 5627

Fax: 00603 6204 5628

Email: [cscpress@cscjournals.org](mailto:cscpress@cscjournals.org)

