Indoor Environment Mobile Robot Localization

IMTHIYAS M.P.
Student, VLB Janakiammal College of Engineering and Technology, Coimbatore

Abstract— Robot Localization is an emerging area in recent research and applications. The determination of location or localization is the basic requirement for robots to move in their office environment. This proposed work aims to build a map from a sparse set of noisy observations, taken from known locations by multiple sensors and is validated experimentally in indoor office environment. A set of training data is collected from each environment and processed offline to produce a GP Model (Gaussian Process Model). The robot uses this model to localize while traversing each environment. The sensors are used to extract information about the robot's environment. Because a mobile robot moves around, it will frequently encounter unforeseen environmental characteristics. The sensors have only a limited range, and so it must physically explore its environment to build a map. So, the robot must not only create a map but also it must do so while moving and localizing to explore the environment. In the robotics terminology, this is called the simultaneous localization and mapping (SLAM), and then changing the robot's trajectory as informed by its sensors during robot motion is called the Obstacle avoidance. The proposed system is used for avoiding real time obstacle in smooth surface by using feature extraction.

Index Terms— Gaussian Process Model, Mapping, Sensors, obstacle avoidance. SLAM

I. INTRODUCTION

An autonomous mobile robot must be able to determine its relationship to the environment by making measurements with its sensors and then using those measured signals. Sensor measurements may have error and, therefore, uncertainty associated with them. Therefore, sensor inputs must be used in a way that enables the robot to interact with its environment successfully in spite of measurement uncertainty.

There are two strategies for using uncertain sensor input to guide the robot's behavior. One strategy is to use each sensor measurement as a raw and individual value. Such raw sensor values could, for example, be tied directly to robot behavior, whereby the robot's actions are a

function of its sensor inputs. Alternatively, the raw sensor values could be used to update an intermediate model, with the robot's actions being triggered as a function of this model rather than the individual sensor measurements. The second strategy is to extract information from one or more sensor readings first, generating a higher-level percept that can then be used to inform the robot's model and the robot's actions directly. Robots will interpret sensors to varying degrees, depending on each specific functionality. For example, in order to guarantee emergency stops in the face of immediate obstacles, the robot may make direct use of raw forward facing range readings to stop its drive motors. For local obstacle avoidance, raw ranging sensor strikes may be combined in an occupancy grid model, enabling smooth avoidance of obstacles meters away. For map-building and precise navigation, the range sensor values and even vision sensor measurements may pass through the complete perceptual pipeline, being subjected to feature extraction followed by scene interpretation to minimize the impact of individual sensor uncertainty on the robustness of the robot's mapmaking and navigation skills. The pattern that thus emerges is that, as one move into more sophisticated, long-term perceptual tasks, the feature extraction and scene interpretation aspects of the perceptual pipeline become essential.

1.1 Feature definition

Features are recognizable structures of elements in the environment. They usually can be extracted from measurements and mathematically described. Good features are always perceivable and easily detectable from the environment. Low-level features (geometric primitives) such as lines, circles, or polygons are distinguished between high-level features (objects) like edges, doors, tables, or a trash can. At one extreme, raw sensor data provide a large volume of data, but with low distinctiveness of each individual quantum of data. Making use of raw data has the potential advantage that every bit of information is fully used, and thus there is a high conservation of information. Low-level features are abstractions of raw data, and as such provide a lower volume of data while increasing the distinctiveness of each feature. Low-level features are filtering out poor or useless data, but of course it is also likely that some valid information will be lost as a result of the feature extraction process. High-level features provide maximum abstraction from the raw data, thereby reducing the volume of data as much as possible while providing highly distinctive resulting features. Once again, the abstraction process has the risk of filtering away important information, potentially lowering data utilization. Although features must have some spatial locality, their geometric extent can range widely. For example, a corner feature inhabits a specific coordinate location in the geometric world. In contrast, a visual fingerprint identifying a specific room in an office building applies to the entire room, but has a location that is spatially limited to the one particular room. In mobile robotics, features play an especially important role in the creation of environmental models. They enable more compact and robust descriptions of the environment, helping a mobile robot during both map-building and localization. When designing a mobile robot, a critical decision revolves around choosing the appropriate features for the robot to use. A number of factors are essential to this decision:

1.2 Target environment

For geometric features to be useful, the target geometries must be readily detected in the actual environment. For example, line features are extremely useful in office building environments due to the abundance of straight wall segments, while the same features are virtually useless when navigating Mars.

1.3 Available sensors

The specific sensors and sensor uncertainty of the robot impacts the appropriateness of various features. Armed with a laser rangefinder, a robot is well qualified to use geometrically detailed features such as corner features owing to the high-quality angular and depth resolution of the laser scanner. In contrast, a sonar-equipped robot may not have the appropriate tools for corner feature extraction.

1.4 Computational power

Vision-based feature extraction can affect a significant computational cost, particularly in robots where the vision sensor processing is performed by one of the robot's main processors.

1.5 Environment representation

Feature extraction is an important step toward scene interpretation, and by this token the features extracted must provide information that is consonant with the representation used for the environmental model. For example, non geometric vision-based features are of little value in purely geometric environmental models but can be of great value in topological models of the environment.

1.6 Feature extraction based on range data (laser, ultrasonic, vision-based ranging)

Most of today's features extracted from ranging sensors are geometric primitives such as line segments or circles. The main reason for this is that for most other geometric primitives the parametric description of the features becomes too complex and no closed-form solution exists. Here, line extraction is described in detail, demonstrating how the uncertainty models presented above can be applied to the problem of combining multiple sensor measurements. Another successful feature of indoor mobile robots is the corner feature and these features can be combined in a single representation is also demonstrated.

1.6.1 Line extraction

Geometric feature extraction is usually the process of comparing and matching measured sensor data against a predefined description, or template, of the expect feature. Usually, the system is over determined in that the number of sensor measurements exceeds the number of feature parameters to be estimated.

1.6.2 Probabilistic line extraction from uncertain range sensor data

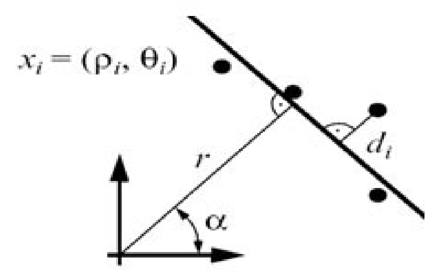

Figure 1.3

Estimating a line in the least-squares sense. The model parameters y (length of the perpendicular) and α (its angle to the abscissa) uniquely describe a line. The main aim is to extract a line feature based on a set of sensor measurements as shown in figure 1.3. There is uncertainty associated with each of the noisy range sensor measurements, and so there is no single line that passes through the set. Instead, the best possible match is related, given some optimization criterion. Given some measurement point, it can calculate the corresponding Euclidean coordinates as and $x = \rho \cos\theta$ $y = \rho \sin\theta$. If there were no error, we would want to find a line for which all measurements lie on that line is found.

$$\rho \cos\theta \cos\alpha + \rho \sin\theta \sin\alpha - r = \rho \cos(\theta - \alpha) - r = 0$$

Consider that the ranging measurement points in polar coordinates are produced by the robot's sensors. There is uncertainty associated with each measurement, so it can model each measurement using two random variables. In this analysis, it is assumed that uncertainties with respect to the actual value are independent.

1.7 Obstacle avoidance

The obstacle avoidance focuses on changing the robot's trajectory as informed by its sensors during robot motion. The resulting robot motion is both a function of the robot's current or recent sensor readings and its goal position and relative location to the goal position. The obstacle avoidance algorithms depend to varying degrees on the existence of a global map and on the robot's precise knowledge of its location relative to the map.

2. SYSTEM MODEL

2.1 Gaussian Process Model

A GP is the generalization of Gaussian distribution from random variables to functions. GP model is used to estimate spatially varying distribution over sensor readings. The steps required are:

a) Obtain a training set of geo-referred sensor measurement.
b) Extract low dimensional features from sensor data.
c) Fit a model to training set using Bayesian regression.
d) Use the model to produce a sensor likelihood function for Monte Carlo localization.

2.2 Gaussian distribution

The Gaussian distribution, also called the normal distribution, is used across engineering disciplines when a well-behaved error model is required for a random variable for which no error model of greater felicity has been discovered. The Gaussian has many characteristics that make it mathematically advantageous to other ad hoc probability density functions. It is symmetric around the mean. There is no particular bias for being larger than or smaller than, and this makes sense when there is no information to the contrary. The Gaussian distribution is also unimodal, with a single peak that reaches a maximum at (necessary for any symmetric, unimodal distribution). This distribution also has tails (the value of f(x) as x approaches $-\infty$ and ∞) that only approach zero asymptotically. This means that all amounts of error are possible, although very large errors may be highly improbable. In this sense, the Gaussian is conservative. Finally, as seen in the formula for the Gaussian probability density function, the distribution depends only on two parameters:

$$f(x) = 1/(\sigma\sqrt{2}\prod)\exp - \left[(x-\mu)^2/2\,\sigma^2\right]$$

The Gaussian's basic shape is determined by the structure of this formula, and so the only two parameters required to fully specify a particular Gaussian are its mean μ , and its standard deviation σ .

Suppose that a random variable X is modeled as a Gaussian, it identifies the chance that the value of X is within one standard deviation of μ requires integration of f(x), the Gaussian function to compute the area under a portion of the curve:

$$\text{Area} = \int_{-\sigma}^{\sigma} f(x)\, dx$$

Unfortunately, there is no closed-form solution for the integral in above equation, and so the common technique is to use a Gaussian cumulative probability table. Using such a table, one can compute the probability for various value ranges of X. For example, 95% of the values for X fall within two standard deviations of its mean. This applies to any Gaussian distribution. It is clear from the above progression, under the Gaussian assumption, once bounds are relaxed 3σ to, the overwhelming proportion of values (and, therefore, probability) is subsumed. The data is to be normalized into zero mean and unit standard deviation and then PCA is applied to reduce dimensionality.

2.3 PCA compression

The mathematical concepts PCA (Principle Compound Analysis), covers standard deviation, covariance, eigenvectors and eigenvalues. Standard deviation operates on one dimension, so that it could only calculate the Gaussian distribution for each dimension of the data set independently of the other dimensions. However, it is useful to have a similar measure to find out how much the dimensions vary from the mean with respect to each other.

3. SYSTEM DESIGN

The total system is represented through two designs. The first one is SLAM building. Second one is working environment with obstacle avoidance.

3.1 SLAM building

A robot that localizes successfully has the right sensors for detecting the environment, so the robot ought to build its own map. A mobile robot's sensors have only a limited range, and so it must physically explore its environment to build such a map. So, the robot must not only create a map but also it must do so while moving and localizing to explore the environment. In the robotics terminology, this is often called the simultaneous localization and mapping (SLAM). If a mobile robot updates its position based on an observation of an imprecisely known feature, the resulting position estimate becomes correlated with the feature location estimate.

3.2 Working Environment with Obstacle Avoidance

The obstacle avoidance focuses on changing the robot's trajectory as informed by its sensors during robot motion. The resulting robot motion is both a function of the robot's current or recent sensor readings and its goal position and relative location to the goal position. The obstacle avoidance algorithms depend to varying degrees on the existence of a global map and on the robot's precise knowledge of its location relative to the map. Consider the Figure 3.1, the Robot has come across an obstacle and would turn to move in either of two possible directions. The obstacle avoidance behavior algorithm finds direction 2 desirable to achieve goal position, so it will turn to direction 2.

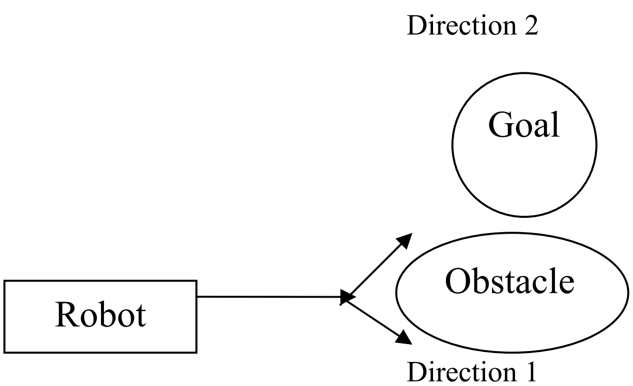

Figure 3.1
Working Environment

3.3 Bug 2 algorithm

Figure 3.2 shows the working of a Bug algorithm, i.e. Robot begins to follow the object's contour, and then departs from the point with the shortest distance toward the goal.

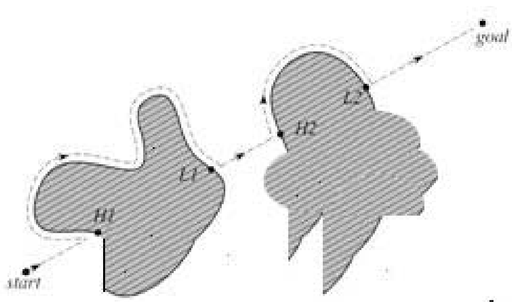

Figure 3.2
Planning and Navigation of Bug Algorithm
with H1, H2, outer boundary hit points and L1, L2, leave points

4. EXPERIMENTATION AND RESULTS

4.1 SLAM Building

The SLAM building is start with the reference point according to the algorithm that is, initialize the starting from an arbitrary initial point, a mobile robot should be able to autonomously explore the environment with its on-board sensors. Knowledge about it is gained. The scene is interpreted. Build an appropriate map is build. The output of setting reference point is shown in Figure 4.1.

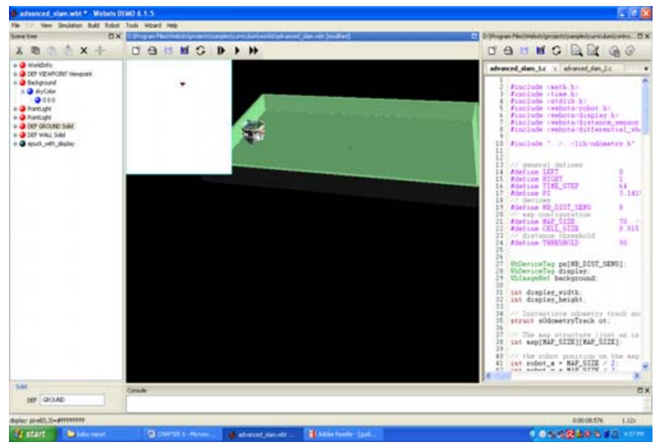

Figure 4.1
Setting Reference Point

A robot that localizes successfully has the right sensors for detecting the environment, and so the robot ought to build its own map. The Figure 4.2 shows the SLAM building in an office environment. A sample output for SLAM is formed.

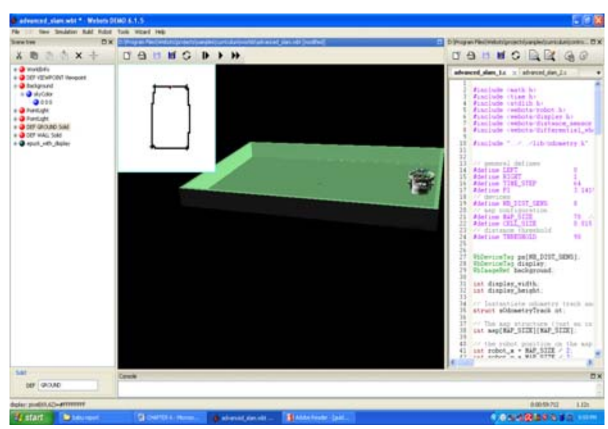

Figure 4.2
SLAM building in an office environment

4.2 OBSTACLE AVOIDANCE IN AN OFFICE ENVIRONMENT

The obstacle avoidance focuses on changing the robot's trajectory as informed by its sensors during robot motion. The resulting robot motion is both a function of the robot's current or recent sensor readings and its goal position and relative location to the goal position. Figure 4.3 represent an office environment with five obstacles.

Figure 4.3
An Office Environment with Obstacles

Consider the Figure 4.4, the Robot has come across an obstacle and would turn to move in either of two possible directions. The obstacle avoidance means modulating the trajectory of the robot in order to avoid collisions.

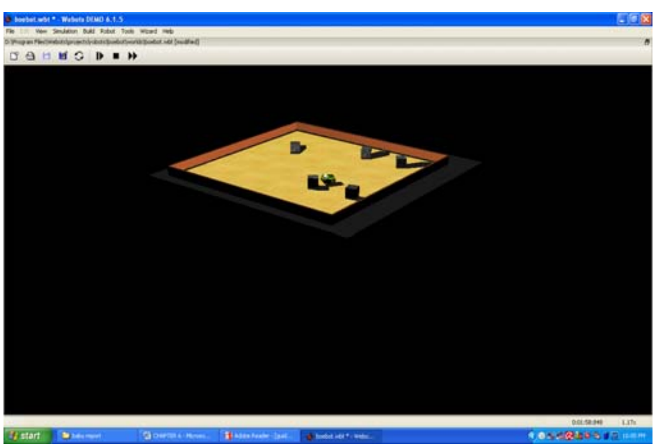

Figure 4.4
Find an Obstacle

Obstacle avoidance is one of the basic tasks required of most mobile robots. Range-based sensors provide effective means for identifying most types of obstacles facing a mobile robot. The Robot has come across an obstacle and would turn to move in either of two possible directions. Figure 4.5 represents an obstacle avoidance of a Robot in an office environment.

Figure 4.5
Obstacle Avoidance and achieve Goal direction

5. CONCLUSION

The indoor experiments were performed in an office environment. A training set is obtained by manually operating the Robot through the environment. A test dataset is generated through this operation. Localizing in this manner provides an accurate representation of the office environment by SLAM building.

REFERENCES

[1] Alex Brooks, Alexei Makarenko, and Ben Upcroft "Gaussian Process Models for indoor and outdoor Sensor-Centric Robot Localization", *IEEE Transaction on Robotics, Vol. 24, No.6, December 2008.*

[2] T Bailey and H Durrant-Whyte, "Simultaneous localization and mapping (SLAM)" *Robot. Autom. Mag., Vol.13, no. 3, September 2006.*

[3] M. Gibbs and D Mackay, "Efficient implementation of GP" Cavendish Lab., *Cambridge, U.K., Tech. Rep., 1997.*

[4] J. Borenstein, H. R. Everett, L. Feng, and D.Wehe, "Mobile robot positioning: Sensors and techniques," *J. Robot. Syst., vol. 14, pp. 231–249, Aug. 1997.*

[5] Clark F. Olson, Member, IEEE, "Probabilistic Self-Localization for Mobile Robots" *IEEE Transactions On Robotics And Automation, Vol. 16, No. 1, February 2000.*

[6] Aye Aye New, Wai Phyo Aung and Yin Mon Myint, "Software Implementation of obstacle detection and avoidance system for wheeled Robot", world academy of science Engineering and Technology 42, 2008.

[7] Borenstein, J., Koren, Y., "The Vector Field Histogram – Fast Obstacle Avoidance for Mobile Robots." IEEE Journal of Robotics and Automation, pages 278–288, 1991.

[8] F. Lu and E. Milios "Robot pose estimation in unknown environments by matching 2d range scans", In IEEE Computer Vision and Pattern Recognition Conference (CVPR), pages 935–938, 1994.

[9] I. Cox. Blanche. "An experiment in guidance and navigation of an autonomous robot vehicle", IEEE Transactions on Robotics and Automation, 7(2):193–204, 1991.

[10] M. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building

(SLAM) problem," IEEE Transactions on Robotics and Automation, vol. 17, no. 3, pp. 229–241, 2001.

BIOGRAPHY

1. Mr. Imthiyas M.P. received the degree in B.Tech (Computer Science and Engineering) from Cochin University of science and Technology, Cochin, Kerala, India. I am currently the final year ME student of V.L.B. Janakiammal College of Engineering and Technology, Coimbatore, Tamilnadu. My areas of interests are Robotics, Artificial Intelligence and Algorithm design.