Mobile robot : Simultaneous localization and mapping of unknown indoor environment

Mohamed Emharraf, Mohammed Rahmoun, Mohammed Saber, Mostafa Azizi
Mohammed First University
Oujda, Morocco
m.emharraf@gmail.com, moharahmoun@hotmail.fr, mosaber@gmail.com, Azizi.mos@gmail.com

***Abstract*— this paper presents an approach for unknown indoor environment exploration using a simultaneous localization and mapping system. The approach addresses the problem of unknown indoor environments exploration, based on robot mobile moving and sonar scanning. The measurements given by the localization system (odometry for the test system), update for the robot self-localization. The map building process maintaining two grid maps: (1) map grid models the environment occupancy (OM), (2) map grid memorize the robot trajectory(TM). The use of two grid maps provides an efficacy description and use of the environment information over time. Results in simulation and real robots experiments using random exploration show the fusibility of our approach.**

Keywords*— autonomous robot mobile; unknown indoor environment; self-localization; map building.

I. Introduction

Localization and map building are fundamental requirements for an autonomous mobile robot, enabling it to travel from an initial configuration to a final one, in an unknown environment a priori. A recent and general study on this subject can be found in [1, 2]. The robot is provided with a set of measurements obtained from odometry and exteroceptive sensors (e.g. sonar, laser, camera images) to solve the problem. The odometry is used for localization while the exteroceptive sensor measurements are utilized for mapping.

In the literature, we find two localization approach families, Relative and absolute. Each of the two families has advantages and disadvantages that are compared in Table 1. Relative location; the determination of mobile robot current position, based on previous positions and the measurements of its travels. For the absolute location, the determination of the current position is performed by measurements on a known position beacons.

Relative localization or position tracking is essential for mobile robotics explore unknown indoor environment. The relative localization with odometry is a simple, inexpensive system and easily workable in real time. The disadvantage of odometry is its unbounded accumulation of errors. two types of odometry errors: systematic errors and non-systematic errors.

Systematic errors have several sources: unequal wheels diameters, misaligned wheels, uncertainty about the length of the center distance, limitation of the encoder resolution and sampling frequency. All these errors lead to an unbounded accumulation of position error. A study's dealing with such errors is presented in [3,19,4].Non-systematic errors come from navigation on uneven environment, passing over unexpected objects on the floor, slippery ground, too much acceleration, a too sharply turning, external forces. Martinelli proposed in [5], [6] a method of evaluation of these kind of errors.

An occupancy grid map [7, 8] is a discrete map of the environment. To achieve this, we divide the environment into cells. Each cell has an occupation probability which is calculated based on the exteroceptive sensor measurements. Thrun's method [9] trains an artificial neural network using Back-Propagation to map sonar readings to occupancy values. Multiple sonar interpretations are then integrated over time using Bayes rule to form a global metric grid. Arleo et al [10] use a similar neural network technique to obtain the local grid-based map, but this local map is subsequently used only to identify obstacle boundaries in order to build a variable-resolution partitioning map. Song and Chang's method [11] extends from heuristic asymmetric mapping (HAM) [17], in which a sonar reading indicates the probabilities of multiple cells that correspond to physical occupied region and empty region. The probability of each cell is then integrated into a global grid map through a first-order digital filter to generate a certainty value from -1 to 1. Oriolo et al [12, 13] provide a fuzzy reasoning method to update the map. Borenstein and Koren [14] uses a simple metric sonar model that increases the cell value measured by the sonar and decreases the cells corresponding to free areas.

This paper is organized as follows: Section 2 discusses our localization system; Section 3 presents the overall framework for grid map building, and we finish with Summary.

TABLE I. SUMMARY OF ADVANTAGES AND DISADVANTAGES OF THE TWO FAMILIES LOCATION

	Relative localization	**Absolute localization**
Depends on the origin	yes	no
Operates continuously	yes	no
Depends on the environment	no	yes
The precision depends on the position	no	yes

	Relative localization	Absolute localization
Depends on the origin	yes	no
Accuracy drift over time	yes	no
Easy modeling errors	yes	no

II. MAP BUILDING

The proposed solution of map grid represents the environment by using evenly-spaced grid cells. A map can be defined as a vector MG (A, B, L), where (A, B) represent respectively the rows and columns; L is the length of cell size. To determine the coordinate (a,b) of the robot in the grid map, we transform the coordinate (x,y) given by localization system (odometry). The physical position of the robot is mapped into a position of the grid-based map so that corresponding information can be saved in a map. The coordinate mapping equations are given as follows.

$$a = \mathrm{int}(x/l) \tag{1}$$

$$b = \mathrm{int}(y/l) \tag{2}$$

Where int(a) represent the inter part of a.

The solution of mapping use two map grid cell. Obstacle map (OM) and Trajectory map (TM). The OM map values OM (i,j) indicates the measure of confidence that an obstacle exists within the cell (i, j) area, where i = 1, 2, …, A, and j=1, 2, …, B. The TM map values TM (i,j) indicates. How often robot traverses the cell (i, j). The TM is designed to record the previously traversed trajectory as well as the time consumed by the robot to traverses the cell area. The information about TM can be used for robot online path-planning. The information is saved and updated in the maps every control period (10ms for our robotic system). The update algorithm is described in Section C.

A. *Sonar reading*

For the obstacles detection, we use an ultrasonic sensor. The sensor mounted on an engine in the robot face. The engine allows a rotation from 65 ° up to 155 °, which allows the sensor to cover the front of the robot. Six measurements are fixed for engine turne (65 ° to 155 °), which provide a good detection of obstacles in front of the robot.

To calculate the positions of the obstacle in the surround (xo, yo),we based on the current position of the robot (Xr, Yr, Θr) and the obstacle position defined by (d, Θo), with d is the distance between the robot and the obstacle in the angle Θo (Figure 1).

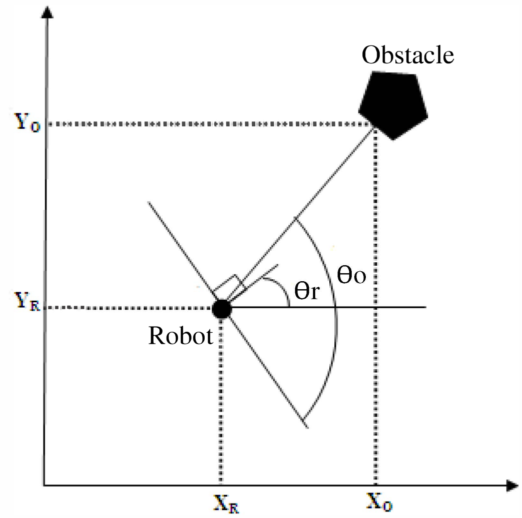


Fig. 1. *Calculation of theobstacle position*

$$\Theta = \Theta r + (\pi/2 - \Theta o) \tag{3}$$

$$Xo = xr + d \cos(\Theta) \tag{4}$$

$$Yo = yr + d \sin(\Theta) \tag{5}$$

B. *Map building system*

The map building proposed approach shown in Figure 2. The grid map is built based on robot's localization system and sonar scanning. The approach includes two modules: map update and map postprocessing. This section provides a short description of their design ideas.

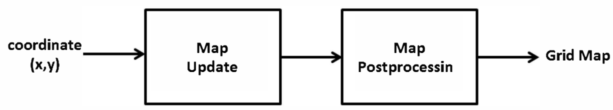


Fig. 2. *Map building system*

1) *Map update*

The map update module is used to analyze the sensor readings and integrate them into OM map that models the environment. The sonar readings are mapped into frequency values (i.e. OM's values) which represent the confidence of the cells where they are occupied by obstacles or not. The OM values are integrated over time to have a single, combined estimate of occupancy in a map (i.e. OM map) by simple addition or subtraction of frequency values. For the update of TM map, only one cell where the robot is currently located is incremented in each control period. The detail is presented in Section C.

2) *Map postprocessing*

The postprocessing model is a filter that learned map offline in order to remove some misclassified cells and to obtain more consistent and complete environment map. At first we use a threshold operation in order to remove some misclassified cells from the perspective of cell's intensity (i.e. magnitude of OM value). In the second step we use a template operation in order to remove most misclassified cells from the perspective of neighboring correlation. Finally, we use an insert operation in order to add some undetected cells. The detail is presented in Section D.

C. The map update

The map updated need two steps. Sensor readings interpretation for detecting the obstacle in the local map. Then the local map is integrated into a global map, that keeps global information throughout the entire control period and the corresponding cells are updated.

The update method of the proposed grid map involves two parts: the update of OM map and the update of TM map. Both OM and TM are set to zero in the beginning (unknown environment).

$$M_{OM}(i,j) = \begin{cases} M_{OM}(i,j)+1, & \text{if } M_{OM}<OM_{max} \\ OM_{max}, & \text{Otherwise} \end{cases} \tag{6}$$

The OM map update method increments only one cell for each angle reading (i.e the cell represent the obstacle). At the same time it keeps the values of those cells that represent free areas in this range reading. The algorithm makes the update fast. the incremental cell is the one that corresponds to the measured distance d and lies on the angle. The incremental cell is updated by Eq. (6)

Where, OMmax is a constant represent the maximum OM cell's value. The increment is 1 and OMmax is 9, experimentally determined in our robotic system. Finally, we only update the cells that are located inside a circular sector of radius centered at the angle position. This circular sector is called the "confidence sector". The radius of the sector is 1 meter, which is an acceptable value that we have confidence to obtain the correct sonar readings for the robotic system. This reduces artifacts produced by sonar noises. Because of this update strategy, a likelihood distribution of occupancy is obtained by continuously sampling the sensor in each 6 turn angles as the robot moving.

$$M_{TM}(i,j) = \begin{cases} M_{TM}(i,j)+1, & \text{if } M_{OM}<TM_{max} \\ TM_{max}, & \text{Otherwise} \end{cases} \tag{7}$$

The update method of the TM matrix is simple (Eq.7). Only one cell where the robot is currently located is incremented in each control cycle.

Where TMmax is a constant for a grid cell's maximum value. This maximum value is 30, experimentally determined in our robotic system. There is no decrement for TM matrix, which means that the trajectory experienced by the robot might not be forgotten.

During every control cycle (10ms in our robotic system), Algorithm C.1 is called once to update a grid map.

Algorithm C.1

Input: (x0, y0) = current robot location;
O0 = current robot heading angle;
di (i = 0, 1, …, 5) = sonar readings from 6 turn angles.

Output: OM_N = The OM matrix; TM_N = The TM matrix.

BEGIN:

Step 1. Update the TM matrix TM_N.

Step 1.1. Do the coordinate mappings to transform current robot coordinate (x0, y0) into coordinate (a0, b0) of grid map by Eqs. (1) and (2);

Step 1.2. Update the TM value TM(a0, b0) of corresponding cell (a0, b0) in TM_N By Eq. (7);

Step 2. Update the OM matrix OM_N.

Read every sonar angle Si (i=0 to 5)

IF the sonar reading di is less than the radius rc of confidence sector, THEN

Step 2.1. Calculate the coordinate (xSd, ySd) of obstacle cell Sd based on the sonar angle's coordinate (xS0, yS0) and sonar reading di;

Step 2.2. Do the coordinate mapping to transform (xSd, ySd) into grid coordinate (mSd, nSd) by Eqs. (1) and (2);

Step 2.3. Increment the OM value OM(mSd, nSd) of cell (mSd, nSd) in OM_N By Eq. (6);

End

D. Map postprocessing

Map postprocessing module is to filter the constructed map offline in order to remove noises. The final map is obtained after the raw learned map (i.e. the OM map) that orderly processed by the modules of threshold operation, a template operation, and an insert operation.

1) The threshold operation

Eliminates some misclassified cells from the perspective of cell's intensity. The misclassified cells are the free cells mistakenly classified as occupied because of the sonar readings errors. Using threshold operation, the OM's value of each cell set to free if it is not larger than threshold TH, otherwise it is set to the maximum value OMmax. The parameter TH for our robotic system is 3, which implies that each occupied cell is eligible to reserve in the final map only if the cell's area is detected at least 3 times.

2) The template operation

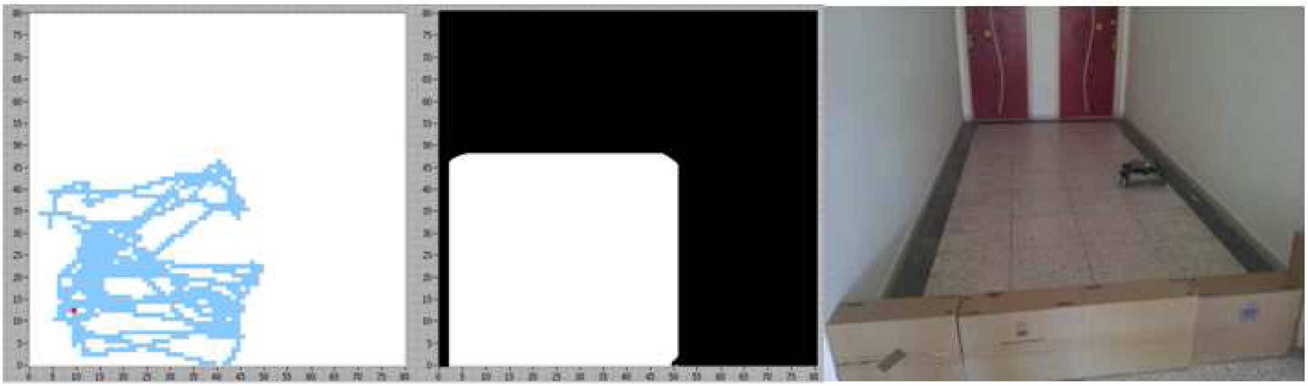
The goal of the operation is to realize the following heuristic rule: Isolated cells (i.e. cells whose neighbors are not occupied) come mostly from erroneous sonar readings. Every cell of the processed map does not have at least surrounded by four occupied neighbors is matched. For example, if all neighboring cells are occupied, this case is matched successfully, otherwise it fails. The OM's value of the cell is maintained in the final map if it's matched successfully. If it fails to match, the OM's value of the cell is set to free (i.e. a free cell).

3) The insert operation

Adds some undetected cells. The undetected cells are those occupied cells that are mistakenly classified as free cells because the sonars miss. Those cell areas due to the robot moving. The purpose of this operation is to realize the following heuristic rule: the cells, whose neighbors on both sides are occupied, should also be occupied. If ever it matched successfully, the OM value of the matched cell is set to the maximum value OMMAX, otherwise its value is maintained.

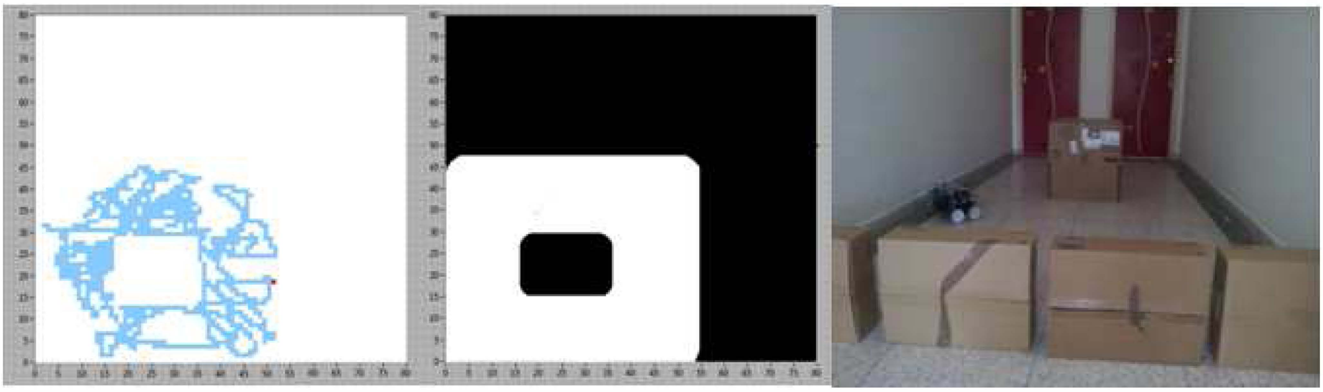
E. EXPERIMENTATION

In order to evaluate the map accuracy, we test our robotic system that integrates an interactive control model for communication [18], in small unknown indoor environment (~20m2), where the sizes of cells are different for each experiment. For the navigation system we use for test a random exploration system; avoid the obstacles while the robots navigate.



Trajectory Map **Obstacle Map** **Obstacle Map**

Fig. 3. *Simple environment exploration*



Trajectory Map **Obstacle Map** **Obstacle Map**

Fig. 4. *Environment with obstacle*

III. LOCALIZATION

Odometry is the most used system in land mobile robotics especially for the environment with small dimensions (up to a hundred meters). Its work as follows: a wheeled mobile robot with two independents wheels not aligned with the direction of movement and equipped with increments encoders (optical encoders). These encoders used to determine the displacement of each wheel. These movements called dUg and dUd (figure 5).

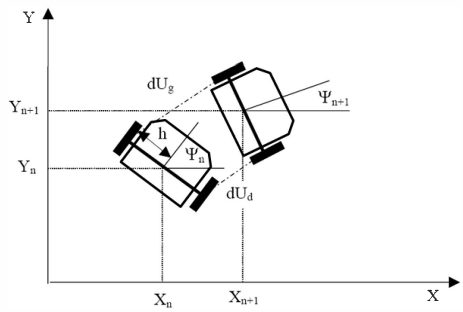


Fig. 5. *Determination of current position by odometry*

The odometry method uses the knowledge of previous positions to determine the new one. We start from the known position Pn(Xn,Yn,Ψn) and use the values given by the encoders to determine the new position Pn+1(Xn+1,Yn+1,Ψn+1). Using geometrical relations [15] we have the following equations:

$$d\Psi = (dU_d - dU_g) / 2h \tag{8}$$

$$dU = (dU_d + dU_g) / 2 \tag{9}$$

$$X_{n+1} = X_n + dU.\cos[(\Psi_n + \Psi_{n+1})/2] \tag{10}$$

$$Y_{n+1} = Y_n + dU.\sin[(Y_n + Y_{n+1})/2] \tag{11}$$

$$\Psi_{n+1} = \Psi_n + d\Psi \tag{12}$$

It appears in the form of the equations that the errors accumulated for each iteration. If we consider more slippage of the wheels on the ground, the weaknesses of this method is key. Two sources of errors that can degrade the accuracy of the localisation:

- Systematic errors wich can be identified and corrected using calibration methods.
- Non-systematic errors, it related to the validity of assumptions underlying model of rolling without sliding of the robot on level ground. Non-systematic errors are almost impossible to compensate without recourse to absolute localization methods. Martinelli et al. proposed in [16] a method of estimating this type of error.

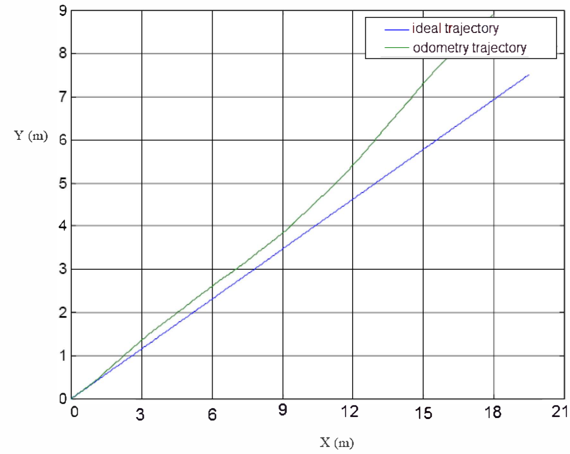
A. *Reduction of systematic errors*

In practice, the systematic errors are not very reliable for localization by odometry when the robot has traveled a certain distance (about ten meters). To reduce the influence of these errors, a first approach is to try to identify them in order to introduce a compensation term in the localization process. This identification or calibration is usually done by requiring the robot to follow a reference trajectory set (e.g. a square). We measure by means of external trajectory, the difference between the reconstructed by odometry and the actual trajectory path, it is possible to observe the systematic errors and correct them. An example of correcting these errors is given in [6].

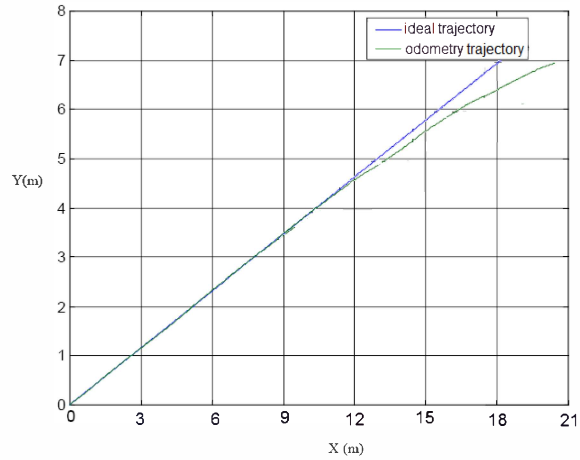
A second approach to reduce odometry errors is to get as close as possible to the conditions of validity of the model used in rolling without sliding odometry equations [15].

B. *Experimentation*

To test the reliability of our localization system, we performed the following experiment, in a plane area without slippage. The position at the beginning is known and the measurements of the wheels rotation (odometry) allow evaluating approximately the position of the robot. The results of our experiment are shown in Figure 6.



(a) Example 1



(b) Example 2

Fig. 6. *Odometry measurement*

In the first example we use a great acceleration, which implies a significant accumulation of errors with time. To reduce these errors we use an acceptable average acceleration (Example 2) which provides acceptable results.

IV. SUMMARY

This article proposes a self-localization and map learning approach namely grid mapping. The approach includes a localization model, map model, a map update method, and a map postprocessing method. The localization model uses an odometry system for determinate position uses odometry measurements and start position. The map adopts a grid-based representation and uses frequency value to measure the confidence that a cell is occupied by an obstacle. The fast map update makes the approach a candidate for real-time implementation on mobile robots. The proposed map postprocessing method, including a threshold operation, a template operation, and an insert operation, is useful to improve the accuracy of the learned map.

REFERENCES

[1] H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki, S. Thrun, Principles of Robot Motion: Theory, Algorithms, and Implementation, MIT Press, 2005.

[2] S.S. Ge, F. Lewis, Autonomous Mobile Robots: Sensing, Control, Decision Making and Applications, CRC Press, Taylor and Francis Group, Boca Raton, FL, 2006.

[3] J. Borenstein and L. Feng, 'Measurement and correction of systematic odometry errors in mobile robots'. IEEE Transactions on Robotics and Automation, volume 12, n°6, pp. 869-880, 1996.

[4] Agostino Martinelli, 'A possible strategy to evaluate the odometry error of a mobile robot'. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1946-1951, Maui, Hawaii, USA, October 29-November 03, 2001.

[5] Agostino Martinelli, 'The odometry error of a mobile robot with a synchronous drive system'. IEEE Transactions on Robotics and Automation, volume 18, n°3, pp. 399-405, June 2002.

[6] Agostino Martinelli, 'Evaluating the odometry error of a mobile robot'. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 853-858, EPFL, Lausanne, Switzerland, October 2002.

[7] A. Elfes. Using Occupancy Grids for Mobile Robot Perception and Navigation. Computer, vol. 22, pages 46–57, juin 1989.

[8] B. Schiele et J. L. Crowley. A Comparison of Position Estimation Techniques Using Occupancy Grids. In IEEE International Conference on Robotics and Automation, pages 1628–1634, San Diego, Etats-Unis, mai 1994.

[9] Thrun S. (1998b), "Learning metric-topological maps for indoor mobile robot navigation", Artificial Intelligence, Vol.99, pp.21-71

[10] Arleo A., Millan J.D.R., Floreano D. (1999), "Efficient learning of variable-resolution cognitive maps for autonomous indoor navigation", IEEE Trans. on Robotics and Automation, Vol.15, No.6, pp.990-1000

[11] Song K.T., Chen C.C. (1996), "Application of heuristic asymmetric mapping for mobile robot navigation using ultrasonic sensors", Journal of Intelligent and Robotic Systems, Vol.17, pp.243-264

[12] Oriolo G., Ulivi G., Vendittelli M. (1997), "Fuzzy maps: a new tool for mobile robot perception and planning", Journal of Robotic Systems, Vol.14, Iss.3, pp.179-197

[13] Oriolo G., Ulivi G., Vendittelli M. (1998), "Real-time map building and navigation for autonomous robots in unknown environments", IEEE Trans. on Systems, Man and Cybernetics, Vol.28, No.3, pp.316-333

[14] Borenstein J., Koren Y. (1991), "Histogramic in-motino mapping for mobile robot obstacle avoidance", IEEE Trans. on Robotics and Automation, Vol.7, No.4, pp.535-539

[15] Ming Wang C., 'Location estimation and uncertainty analysis for mobile robots'. IEEE International Conference on Robotics and Automation, volume 2, pp. 1230-1235, Philadelphia, 1988.

[16] Agostino Martinelli, Nicola Tomatis, Adriana Tapus and Roland Siegwart, 'Simultaneous localization and odometry calibration for mobile robot'. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1499-1504, Las Vegas, Nevada, October 2003.

[17] Song K.T., Chen C.C. (1996), "Application of heuristic asymmetric mapping for mobile robot navigation using ultrasonic sensors", Journal of Intelligent and Robotic Systems, Vol.17, pp.243-264

[18] Emharraf, M.; Rahmoun, M.; Saber, M.; Azizi, M., "Implementation of interactive control model for robots mobile teleoperation," Complex Systems (ICCS), 2012 IEEE International Conference.

[19] J. Borenstein and L. Feng. Correction of systematic odometry errors in mobile robots. In Proceedings of the 1995 IEEE International Conference on Robotics and Automation, pages 569{574, 1995.