2003 International Symposium on Advanced Intelligent Systems
September 25-28, 2003, Jeju, Korea

Mobile Robot Navigation using a Dynamic Multi-sensor Fusion

San-Ju Kim*, Tae-Seok Jin*, Oh-Keol Lee** and Jang-Myung Lee *

*Department of Electronics Engineering, Pusan National University
**Department of Electrical Engineering, Dongeui Institute of Technology
email: ksj_elec@hotmail.com

***Abstract*— In this study, as the preliminary step for developing a multi-purpose Autonomous robust carrier mobile robot to transport trolleys or heavy goods and serve as robotic nursing assistant in hospital wards. The aim of this paper is to present the use of multi-sensor data fusion such as sonar, IR sensor for map-building mobile robot to navigate, and presents an experimental mobile robot designed to operate autonomously within both indoor and outdoor environments. Smart sensory systems are crucial for successful autonomous systems. We will give an explanation for the robot system architecture designed and implemented in this study and a short review of existing techniques, since there exist several recent thorough books and review paper on this paper. It is first dealt with the general principle of the navigation and guidance architecture, then the detailed functions recognizing environments updated, obstacle detection and motion assessment, with the first results from the simulations run.**

I. INTRODUCTION

Sensing of the environment and subsequent control is important feature of the navigation of an autonomous mobile robot. When a mobile robot navigates in an unknown or partially known environment, several types of sensors are commonly used for this purpose such as ultrasonic sensors, infrared sensors, laser range finders and vision systems for obstacle avoidance or path planning. Recently, it is increasing the use of vision system because it has inexpensive and is able to be fast real-time environmental recognition (Allen *et al.*, 1991; Camillo *et al.*, 1998). In this paper we present a statistical method for dealing with the general problem of concurrent localization and map building and show a global map using the local maps of each point. We furthermore address the problem of using occupancy grid maps for path planning in highly dynamic environments. The approaches have been tested extensively and several experimental results are given in the paper.

II. A GENERAL PATTERN OF SENSOR FUSION

Fig. 1 is mean to represent a general pattern of multi-sensor integration and fusion in a system. In this Fig., n sensors are integrated to provide information to the system. The output x_1 and x_2 from the first two sensors are fused at the lower left-hand node into a new representation $x_{1,2}$. The output x_3 from the third sensor could then be fused with $x_{1,2}$ at the next node, resulting in the representation $x_{1,2,3}$, which might then be fused at nodes higher in the structure.

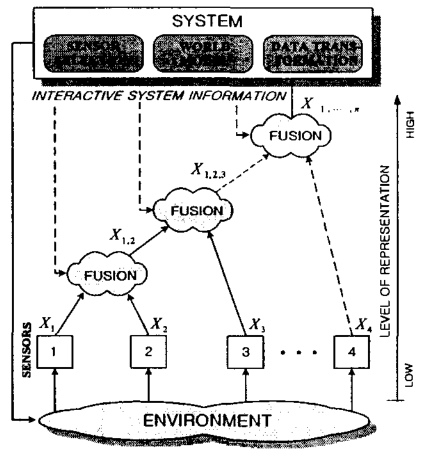

Fig. 1. General pattern of multi-sensor integration and fusion in system.

III. RELATED THEORIES

We will start by introducing the basic principles of data fusion in robots. We will then describe some existing mobile platform from a sensory processing perspective. The concept of using directed sonar sensing for navigation will be discussed in more detail, and we will concentrate on grid-based methods.

3.1. *Occupancy Grids*

Occupancy Grids is certainty a state of the art method in the field of grid based methods. The idea is to divide the environment into grid cell C_{ij}. Typically a 2-dimensional grid is enough to give interesting information about the environment. Each cell can be in two states $s(C_{ij})$ = OCCupied or $s(C_{ij})$ = EMPty, and to each cell there is a probability $P[s(C_{ij}) = \mathrm{OCC}]$ attached, which reflects the belief of the cell C_{ij} being occupied by an object. Since

$$P[s(C_{ij}) = \mathrm{EMP}] = 1 - P[s(C_{ij}) = \mathrm{OCC}] \quad (1)$$

The grid is initialized with $P[s(C_{ij}) = \mathrm{OCC}] = 1/2$. To update the cells when the robot traverses the environment, a stochastic sensor model $p(\mathrm{r} \mid \mathrm{z})$ is used. This model is obtained from experiments with the sensor in question and relates reading vector, r, to the true space range vector, z. Given a new range reading, r, from a sensor, the idea now is to use the sensor model $p(\mathrm{r} \mid \mathrm{z})$ to update the probabilities $P[s(C_{ij}) = \mathrm{OCC}]$ in the Occupancy Grid. This can be done by using Bayes theorem, see Equation (4),

$$P[s(C_{ij}) = \mathrm{OCC} \mid \mathrm{r}] = \frac{p[\mathrm{r} \mid s(C_{ij}) = \mathrm{OCC}]P[s(C_{ij}) = \mathrm{OCC}]}{\sum_{s(C_{ij})} p[\mathrm{r} \mid s(C_{ij})]P[s(C_{ij})]} \quad (2)$$

We mention here that the right side of Equation (2) has to be developed further to be computable. Exactly how this is done can be found in (Abidi *et al.*, 1992; Elfes, 1989). The Occupancy grid method now provides a useful setting for fusing data from different sensors. There are, basically, two main approaches:
Considering handling position uncertainty of the robot one can have a global grid map of the environment stored on the robot platform. The fused robot map can then be matched against the global map to reduce uncertainty in position. One major drawback with using Occupancy Grids as described above is that when updating the grid, i.e., evaluating Equation (2) for each cell C_{ij}, the computational cost is high even for rather small grids. This implies that algorithms that use Occupancy grids for navigation are rather slow. One way of getting around this problem is to use Vector Field Histogram methods, which is described in the next section.

3.2. *Vector Field Histogram (VFH)*

The Vector Field Histogram is a way of handling fast map building and obstacle avoidance at the same time. The method was originally introduced by Borenstein and Koren [9,10,11]. A window moves with the robot, overlying a square region of active window in the histogram grid of Fig. 2. The contents of each active cell in the histogram grid are mapped into the corresponding sector of the polar histogram (see Fig. 2), resulting in each sector k(S_k) holding a value h_k. Thus, h_k is higher if there are many cells with high certainty value c_{ij} s in one sector. Intuitively, this value can be interpreted as the polar obstacle density in the direction of sector k(S_k).
However, to obtain a fast map update, the formula (2) is not used since it projects a probability profile onto all those cells affected by a range reading. Instead each cells have an associated *certainty value* c_{ij} of integer type which reflects the belief of the cell being occupied. The higher (lower) value of c_{ij}, the more confidence we have that the cell C_{ij} being occupied (empty). The question of how much the cells should be incremented or decremented depends on which type of sensors is used. So again it is important to have good sensor models in order to obtain good results. To prevent the integers c_{ij} to grow or decrease too much, they are saturated at some maximum and minimum integer. For each cell C_{ij} in a given sector, say S_k, one calculates an obstacle vector m_{ij}. The magnitude of m_{ij} is dependent of the certainty value c_{ij}, and also by the distance between the center of the grid (robot position) and the cell C_{ij}. After this procedure one sum all the obstacle vectors m_{ij} s in sector S_k to form an obstacle density entity h_k

$$h_k = \sum_{C_{ij} \in S_k} m_{ij}, \qquad k = 1, \ldots n. \quad (3)$$

At this point the entities $h_1, h_2, \ldots, h_n$ are uses to form a histogram, which can be used for building map. Areas in the histogram where the vector magnitudes are big indicate regions with high obstacle density, while areas with low vector magnitudes regions with low obstacle density. By adapting a threshold to the histogram it is possible to localize regions of sectors with low obstacle distance which can be used for obstacle avoidance.

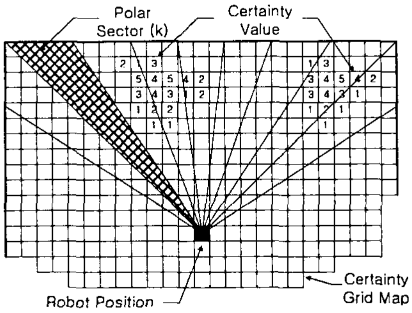

Fig. 2. The heart of the VFH method: Mapping active cells onto the polar histogram.

IV. Data Fusion

Data fusion is about deriving information about certain variables from observations of other variables. The application area is huge, see the special issue on data fusion in (Varshney, 1997) for a recent overview. An edited collection of survey papers on data fusion in robotics and machine intelligent is given in (Abidi *et al.*, 1992). Sensor fusion in general is discussed in (Rothman *et al.*, 1991; Thomopoulos, 1990).

4.1. *Statistical Foundations*

From a statistical perspective, we have the following problem. Given two vector random variables X and Y, what does the observation $Y = y$ tell us about X? The complete answer is given by the so-called conditional probability density function,

$$p_{X|Y}(x \mid y) = \frac{p_{X|Y}(x, y)}{p_Y(y)} \quad (4)$$

Here $p_{X,Y}(x, y)$ is the joint probability density for X and Y, and $p_Y(y)$ is the probability density for Y. By using the dual assumption, namely that $X = x$ is given, we obtained the very useful Bayes rule

$$p_{X,Y}(x, y) = p_{X|Y}(x \mid y)p_Y(y) = p_{Y|X}(y \mid x)p_X(x) \quad (5)$$

$$p_{X,Y}(x,y)=\frac{p_{Y|X}(y\mid x)p_X(x)}{p_Y(y)}, \qquad (6)$$

which is the key formula in Bayesian and maximum likelihood estimation theory.

Different estimates of X can now be constructed from its distribution. The (conditional) minimal variance of X equals the conditional mean of X given $Y=y$,

$$\hat{x}=E[X\mid Y=y]=\int_{-\infty}^{\infty} x p_{X|Y}(x\mid y)dx \qquad (7)$$

Another useful estimate is the maximum a posteriori estimate, which maximizes the function $p_{X|Y}(x\mid y)$. The rest is design and analysis issues, i.e. formulating the underlying model, specifying probability density functions and calculating equality/variance properties. The most used probability density function is the Gaussian one (the Normal distribution). The main reason is that the conditional density function also will be Gaussian, and analytic expressions of the minimal variance estimate can thus be obtained.

Let X and Y be jointly Gaussian, i.e. $Z=[X'\ Y']'$ is Gaussian with mean and covariance

$$m_z=\begin{bmatrix}\bar{x}\\ \bar{y}\end{bmatrix}, \Sigma_{zz}=\begin{bmatrix}\Sigma_{xx} & \Sigma_{xy}\\ \Sigma_{yx} & \Sigma_{yy}\end{bmatrix} \qquad (8)$$

Then X conditional on $Y=y$ has a Gaussian distribution with mean and covariance

$$m_{x|y}=\bar{x}+\Sigma_{xy}\Sigma_{yy}^{-1}(y-\bar{y}), \Sigma_{x|y}=\Sigma_{xx}-\Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx} \qquad (9)$$

Hence the conditional mean of X given $Y=y$, equals

$$\hat{x}=E[X\mid Y=y]=\bar{x}+\Sigma_{xy}\Sigma_{yy}^{-1}(y-\bar{y}) \qquad (10)$$

Almost all practical estimators are special cases of the above result. The expression is called *the fundamental equations of linear estimation* in (Bar-Shalom, 1993). This reference also provides a very good introduction to estimation theory, in general, and tracking, in particular.

V. ROBOT TYPE IN EXPERIMENT SETUP

The mobile robot used in the experiments is an *IRL-2001* developed in the *IRL*, PNU which is designed for an int elligent service robot.

Fig. 3: *IRL*-2001 robot.

This robot is shown in Fig. 3 along with some of its sensory components. Its main controller is made on system clock 1.2 GHz, Pentium IV Processor. The sensors, 16-ultrasonic and a robust odometry system are installed on the mobile robot. Ultrasonic sensors and infrared sensors in eight sides(25°) sense obstacles of close range, and the main controller processes this information.

5.1. Building a local map

Building a robust and reliable avoid behavior has been found to require some kind of memory. Inspired by the work of Borenstein and Koren [4] we have implemented a grid based local map for the robot. So far this map has been updated using only the sonar data. At this early stage we have been using a ray-trace model for the sonar, which is justified by the motto, try simple first and supported by [10]. The results of these tests show that the avoid behavior is improved. Below (Fig. 4) is a sketch of the experimental environment. To show what the local maps look like, four samples of such maps are shown in Fig.s 5-7. The size of the cells in these maps were 20X20 mm and the number of cells were 200X200, giving a total size of 4X4m. Note that the coordinate system of the local maps are robot centered. The approximate location of the robot when the maps were saved is given in the sketch (Fig. 4) by the letters A-D.

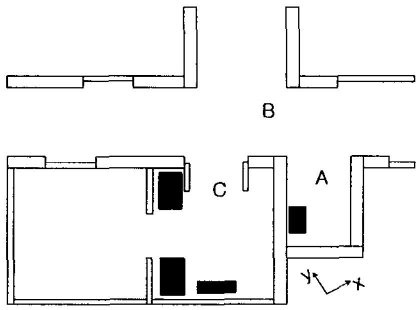

Fig. 4. A sketch of the environment around the robot lab.

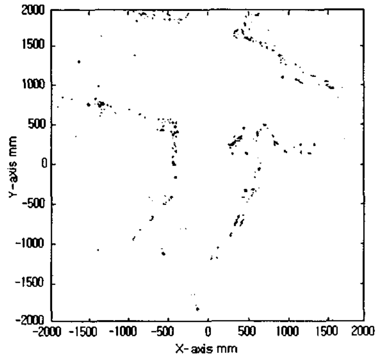

Fig. 5. Sonar based local map of the corridor outside room with three closed wall and one open. A in the sketch.

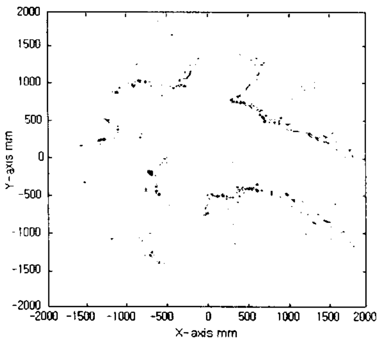

Fig. 6. Sonar based local map of the corridor beside room. B in the sketch.

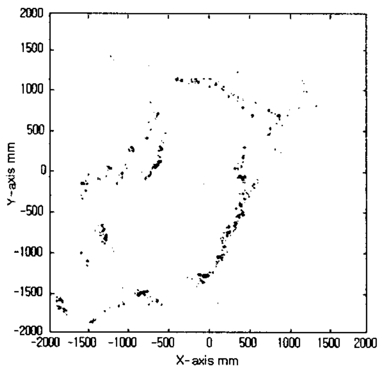

Fig. 7. Sonar based local map of the door-passage into room. C in the sketch.

If the local map is extended to a size that can hold much more information the figure below (Fig. 8) show a possible result. It can be clearly seen that most of the features of the environment (corners, wall, etc) are accurately mapped. The intention of the local map is not to be this large, but rather to have a size more like the once shown above. The global map is updated based on the local map, i.e., the sensor data is not directly used in the global map.

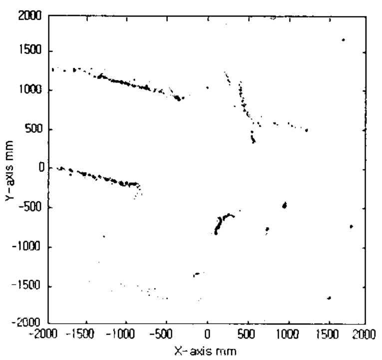

Fig. 8. Sonar based map of the experimental environment.

VI. CONCLUSION

In this paper, we have presented two more or less orthogonal approaches for using sonar sensor and map-building by multi-sensor mobile robot to navigate within an indoor setting. Important regions of the robot workspace (locales) are represented using grid-based map collected during the exploration phase. From a scientific/academic perspective it is important to study very general issues and approaches, were the ultimate aim is full autonomy. However, the engineering perspective is the opposite, i.e. one wants to solve a specific problem, e.g. a sonar sensor based feedback control algorithm for going through narrow doorways. However, the main issue for such research is scalability, i.e. is the solution of more general interest and can it be extended to more complex situations.

For future works, it is straightforward to control robot locate a certain target with multi-sensor upon navigation. Also it will be interesting to have the robot to learn and map an unknown or pseudo-unknown environment.

REFERENCES

[1] A. Elfes, *A Probabilistic Framework for Robot Perception and Navigation.* PhD these, Carnegie-Mellon University, 1989.

[2] A. Elfes, onar-based real-world mapping and navigation, *IEEE Journal of Robotics and Automation*, RA-3(3): pp. 249-265, 1987.

[3] Allen P., Yoshimi B., and Timecenko A., eal-time visual servoing, *in Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 851-856.1991.

[4] Borenstein, J., and Koren, Y., he vector field histogram - fast obstacle avoidance for mobile robots, *IEEE Transactions on Robotics and Automation* 7(3): pp. 278 88, 1991.

[5] B. Triggs and C. Laugier, utomatic Camera Placement for Robot Vision, *Proc. IEEE Int Conf. Robotics and Automation*, Vol. 2, pp. 1,732-1,738, Nagoya, Japan, 1995.

[6] Buhmann, J., Burgard, W., Cremers, A. B., Fox, D., Hofmann, T., Schneider, F., Strikos, J., and Thrun, S., he mobile robot Rhino, *AI Magazine* 16(2), pp. 31–38, 1995.

[7] Camillo J. T. and David J. K., ision-Based Motion Planing and Exploration Algorithms for Mobile Robots, *IEEE Trans. on Robotics and Automation*, Vol.14, No.3, pp. 417-426, 1998.

[8] D. Fox, *A Probabilistic Framework for Mobile Robot Localization and Naviagation*. PhD thesis, Dept. of Computer Science, University of Bonn, Germany, December, 1998.

[9] J. Borenstein and Y. Koren, rror eliminating rapid ultrasonic firing for mobile robot obstacle avoidance, *IEEE Transactions on Robotics and Automation*, II (1): pp. 132-138, February 1995.

[10] J. Borenstein and Y. Koren, eal-time obstacle avoidance for fast mobile robots, *IEEE Transactions on Systems, Man, and Cybernetics*, 19(5), September/ October 1989.