Vision based navigation using Kalman approach for SLAM

S. Panzieri* **F. Pascucci**** **G. Ulivi***

* Dip. Informatica e Automazione
Università degli Studi "Roma Tre"
Via della Vasca Navale 79, 00146 Roma, Italy
{panzieri,ulivi}@uniroma3.it

** Dip. Informatica e Sistemistica
Università degli Studi "La Sapienza"
Via Eudossiana, 18, 00189 Roma, Italy
pascucci@dia.uniroma3.it

Abstract

The aim of this paper is to present a vision based algorithm for simultaneous localisation and map building (SLAM) for indoor environment. The approach is based on the well known predictor-corrector structure of the Extended Kalman Filter and makes use of vision data in the correction phase. To this aim a low cost vision system is presented, with a fast image processing procedure for features extraction, based on the recognition of artificial sources of light in an office-like environment. Experimental tests, carried out on a robotic wheelchair equipped with the vision system, show the satisfactory performance of the mobile platform under realistic operating conditions.

1 Introduction

Mobile robotic systems require both reliable localisation and sufficiently precise map of the navigation area to autonomously accomplish navigation tasks. For this reason a localisation module is usually included in the control architecture of mobile robot together with an *a priori* knowledge of the environment (i.e., a map). With such a structure, the localisation module is able to provide reliable estimation of position of the vehicle in the environment using the map.

A more challenging problem arises when both the map and the robot location are unknown. In this case the robot starts in an unknown position in an unknown environment and tries to incrementally build a map while using the same map to compute its pose in the environment [1]. This problem is referred in literature as simultaneous localisation and map building (SLAM), and several approaches have been investigated to solve it in the last few years. The proposed solutions can be grouped in two main classes on the basis of the way the world is mapped: the first class deals with abstract and qualitative representation of the environment (i.e., *topological* mapping)[2]-[4], while the second one uses geometric relations between objects and a fixed reference frame to describe the navigation area (i.e., *metric* mapping) [1], [5], [6].

Although techniques based on topological maps have many potential advantages in terms of limiting the need for both accurate modeling and computational requirements, more promising results are obtained exploiting metric maps together with probability theory to describe the uncertainty sources.

In this framework a common approach uses the well known predictor-corrector structure of the Kalman filter: a solution for SLAM is given by modeling the environment and sensors and assuming that errors have a Gaussian distribution [7]. The measures of the proprioceptive sensory system (e.g., encoders, gyros) of the vehicle are used to compute a raw location estimation, while the exteroceptive sensors, usually range finders, refine this estimation, maintaining the map.

This article proposes a solution for the SLAM problem in office like environments, following the classical estimation theoretic approach, but using a vision system as exteroceptive sensor. As it is known, visual based localisation schemas require, generally, very complex algorithm and dedicated hardware to accomplish the features extraction [8]. In this paper we investigate the possibility to implement a simultaneous localisation and mapping system for a mobile robot using a low cost standard hardware, i.e. a PC with a web cam. To this end we have mounted the web cam on the mobile robot focused to the ceiling to use the lamps as reference points.

The presence of a low cost hardware generates some drawbacks. The first is due to the limited transfer rate that, in addition to the time consumed by the feature extraction algorithm, introduces a considerable delay in the navigation loop. Moreover, the poor quality of the image provided by the web cam and the optical distortion produced by its lens should be taken into account [9]. To partially alleviate this problem we propose a fast feature extraction algorithm that includes a model of the lens aberration in the SLAM filter.

The paper is organised as follow: in section 2 we detail the vision system, in section 3 the SLAM filter equation are explained, while experimental results are

presented in section 4. Conclusion are reported in 5

2 Vision system

The vision system used in this work has a camera on board configuration. A low cost web cam is mounted on a mobile robot, focusing at the ceiling, and it is connected to a notebook.

The web cam captures 352×288 pixels images using only the YUV format at the rate of 5 frames per second. The notebook processes the frames to provide the position of the lamps in the image according to a three-steps algorithm: features extraction, features validation and computation of beacons position.

2.1 Lamps recognition algorithm

The algorithm performs feature extraction using the luminance information coded in the Y parameter of the frame. Assuming that the beacons have a frame representation with large Y values, a threshold should be first applied but, to grab position information also for lamps that have small luminance or large reflections phenomena, a morphological operator performing a closure operation is previously applied. At the end of the two phases a graph search of all connected components is performed using the 8-adjacency relationship between pixels.

After this scanning, more than one connected component is usually retrieved. Sometimes, reflections can produce small connected components that can be discriminated evaluating their area. The lamps used in the present work produce circular connected components with about 90 pixels, therefore the algorithm processes them only if greater than 80 pixels: in this way erroneous information provided by reflections and lamps only partially in the viewing window are discarded.

At this point it is possible to compute the position of the centre of gravity (CoG), in the image coordinates, of each connected component using the well known formula:

$$(u_c, v_c) = \left(\frac{\sum_{i=0}^{N} Y_i (u_i - 0.5)}{\sum_{i=0}^{N} Y_i}, \frac{\sum_{i=0}^{N} Y_i (v_i - 0.5)}{\sum_{i=0}^{N} Y_i} \right) \quad (1)$$

where (u_j, v_j) is the position of the $j-$th pixel in the image coordinate, Y_i its luminance, and N the pixels number of the lamps.

In Fig. 1 two different images, A and C, are elaborated with the algorithm above described: even the second one, of much lower quality, can be usefully interpreted and its CoG correctly retrieved.

Knowing the scale factor between pixels and metres, and the *configuration* (i.e., position and orientation) of the web cam in the robot reference frame, it is easy to compute the position of the beacons in the latter reference frame using a coordinate transformation.

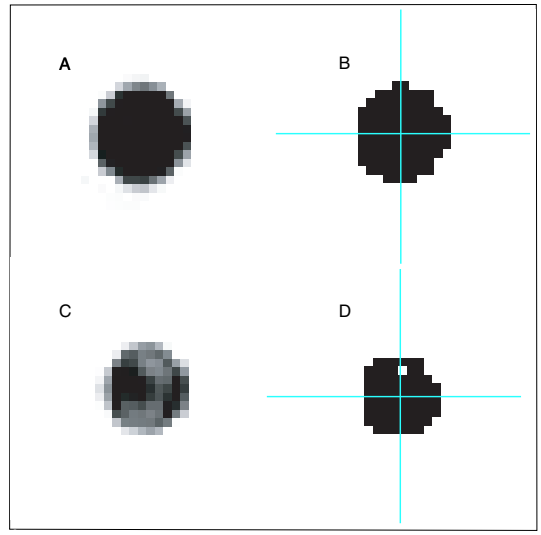

Figure 1: *two different spot images (A, C) and the result of their elaboration (B, D) respectively*

Note that the scale factor between pixels and metres is constant, as the distance between the web cam and the ceiling is always the same in an office like environment.

3 SLAM filter

The solution for the SLAM problem adopted in this work is based on the estimation theory. In this framework we assume that all uncertainty sources have unimodal Gaussian distributions and provide a model for the vehicle, the lamp positions, and the sensors. The mobile platform considered is a robot with the kinematics of an unicycle. The robot is equipped with encoders and gyro, as proprioceptive sensors, while uses the vision system presented in the above section as exteroceptive sensor. The measurements provided by the exteroceptive sensory system are expressed in the robot coordinates and represent the position of the beacons in the viewing windows of the web cam.

Under the SLAM framework, the filter is able to localise the robot and concurrently build a map (a list of beacons position). During the navigation task, the system detects new features when exploring new areas.Once those features become reliable, they are included into the map.

The structure of the SLAM filter, shown in Fig. 2, proceeds directly from the EKF equations.

At the k-th step, the kinematic model of the robot and the measurements encoders and gyro are used to compute from an odometric and map prediction $\hat{x}_{k|k-1}$ with an associated covariance matrix $P_{k|k-1}$; an observation prediction $\hat{z}_k$ is formed and compared with the measure z_k provided by the vision system.

The result is the innovation term v_k and its covariance matrix S_k, that is used by the EKF to produce

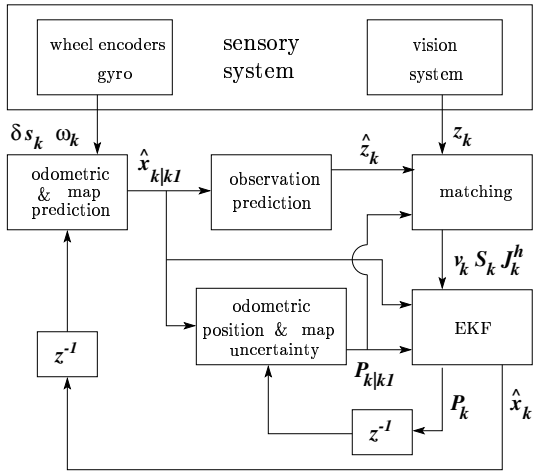

Figure 2: *Structure of SLAM filter (z^{-1} denotes the unit delay operator)*

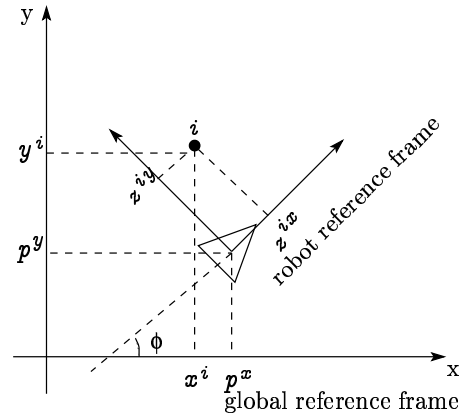

Figure 3: *Reference frames used in the SLAM filter*

the new state estimate $\hat{x}_k$ and the associated covariance P_k. In the following, each phase is explained in some detail.

Note that vision system data are available at a lower frequency than the encoder and gyro measures, so more than one prediction step is performed between two update steps. Moreover, a delay arises between the frame acquisition instant and the time in which geometrical data becomes available to the filter. The solution adopted, like in [10], was to store in a buffer some past state values and odometric data and, each time a new frame (relative to time t_f) is processed, go back with the filtering algorithm to $t = t_f$, include the geometrical data in its evaluation, and then propagate to the actual instant the state estimate using the odometric data previously stored.

3.1 Odometric prediction

As described above, the state of the whole system at the k-th sampling interval consists in the configuration of the robot together with the positions of all beacons w.r.t. a global reference frame (see Fig. 3):

$$x_k = ({x_k^r}^T, {x_k^b}^T)^T. \tag{2}$$

Define the robot state vector as

$$x_k^r = (p_k^x, p_k^y, \phi_k, b_k)^T \tag{3}$$

where b is the gyro bias, and define the inputs for the robot model as

$$u_k = (\delta s_k, \omega_k)^T \tag{4}$$

where δs_k is the vehicle displacement and ω_k its angular velocity during the k-th sampling interval.

The robot configuration is estimated using the equation of the unicycle model:

$$\hat{x}_{k|k-1}^r = f(\hat{x}_{k-1}^r, u_k) = \tag{5}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -\delta t_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \hat{x}_{k-1}^r + \begin{bmatrix} cos\tilde{\phi}_k & 0 \\ sin\tilde{\phi}_k & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} u_k$$

where $\tilde{\phi}_k = \hat{\phi}_{k-1} + (\omega_k - \hat{b}_{k-1}\delta t_k)/2$ is the average robot orientation during the sampling time interval δt_k.

The beacon state vector is defined as

$$x_k^b = [p_k^1, ..., p_k^N]^T \tag{6}$$

where $p_k^i = (x_k^i, y_k^i)$ is the location of the i-th beacon in the global reference frame.

The state transition equation for the beacons can be written as

$$\hat{x}_{k|k-1}^b = \hat{x}_k^b \tag{7}$$

since beacons are assumed to be static.

The covariance matrix associated with the prediction error is computed as

$$P_{k|k-1} = J_x^f(\hat{x}_{k-1})\, P_{k-1}\, (J_x^f(\hat{x}_{k-1}))^T + J_u^f(u_k)\, C\, (J_u^f(u_k))^T + Q. \tag{8}$$

Here, $J_x^f(\cdot)$ and $J_u^f(\cdot)$ are the Jacobian matrices of $f(\cdot)$ with respect to $\hat{x}_{k-1}$ and u_k, P_{k-1} is the covariance matrix at time instant t_{k-1}, $C = \text{diag}\{\sigma_{\delta s}^2, \sigma_\omega^2\}$ is the covariance matrix of the Gaussian white noise which corrupts input measures and $Q = \text{diag}\{\sigma_{px}^2, \sigma_{py}^2, \sigma_{p\phi}^2, \sigma_b^2, \mathbf{0}\}$(with $\mathbf{0} \in R^{2N}$) is the covariance matrix of the Gaussian white-noise which directly affects the state in the kinematic model. Accordingly with estimation theory, we assume that landmarks are not in stochastic motion: this ensure some convergence properties of the EKF for the SLAM as it proved in [1].

3.2 Observation prediction and matching

The observation equation describes the relation between robot configuration and position of beacons in the viewing windows of the web cam (referred as *active beacon* in from now on). The observation prediction vector

$$\hat{z}_k = h(\hat{x}_{k|k-1}) \tag{9}$$

consists of sub vectors $\hat{z}_k^i$, $i = 1, ..., M$, where M is the number of active beacons and

$$\hat{z}_k^i = h^i(\hat{x}_{k|k-1}) = \begin{bmatrix} \hat{z}_k^{ix} \\ \hat{z}_k^{iy} \end{bmatrix} = \mathbf{R}_k^{\hat{\phi}} \begin{bmatrix} \hat{x}_{k|k-1}^i - \hat{p}_{k-1}^x \\ \hat{y}_{k|k-1}^i - \hat{p}_{k-1}^y \end{bmatrix} \tag{10}$$

being $\mathbf{R}_k^{\hat{\phi}}$ the rotation matrix between the robot reference frame and the global frame reference (see Fig. 3):

$$\mathbf{R}_k^{\hat{\phi}} = \begin{bmatrix} cos\hat{\phi}_{k|k-1} & sin\hat{\phi}_{k|k-1} \\ -sin\hat{\phi}_{k|k-1} & cos\hat{\phi}_{k|k-1} \end{bmatrix} \tag{11}$$

The innovation term and the associated covariance are, then, computed as

$$v_k = z_k - \hat{z}_k \tag{12}$$

$$S_k = J_x^h(\hat{x}_{k|k-1})\, P_{k|k-1}\, (J_x^h(\hat{x}_{k|k-1}))^T + R_k \tag{13}$$

where $J_k^h(\cdot)$ is the Jacobian matrix of $h(x_{k|k-1})$ respect to $x_{k|k-1}$. The first term used to compute S_k represents the uncertainty on the observation due to the uncertainty on the odometric and map prediction. The second term is the observation noise covariance matrix.

The measurement noise depends on the interaction between the sensory system and the environment, which is not invariant in space and time. For example, due to the web cam lens aberration the accuracy of vision system measurement is related to the position of the image points in the image plane. In order to avoid a time consuming calibration step and model this fact, we have chosen to label each pixel of the image with a value representing the reliability of the vision system measure, and to modify the correspondent rows of the covariance matrix R accordingly [11].

Moreover, since sources of uncertainties in a mobile system do not obey normal distribution, we need a mechanism for rejecting invalid measures (*outliers*) that are common in robot sensing and would seriously affect filter performance. In order to avoid the inclusion of outliers in the correction phase, a *validation gate* is defined using Mahalanobis distance [12]. Suppose that v_k^i is the innovation term for the i-th beacon; it will be processed by the EKF only if it satisfied the χ^2 test.

$$(v_k^i)^T (S_k^i)^{-1} v_k^i \leq \chi^2 \tag{14}$$

where

$$S_k^i = J_x^{hi}(\hat{x}_{k|k-1}) P_{k|k-1} (J_x^{hi}(\hat{x}_{k|k-1}))^T + R_k^i. \tag{15}$$

Figure 4: *The robotised wheelchair*

3.3 Extended Kalman Filter

The Extended Kalman Filter is used to correct the odometric configuration and beacons position estimate one the basis of the validated observations. Particularly, the final state estimate is obtained as

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k[z_k - \hat{z}_k] \tag{16}$$

where K_k is the Kalman gain matrix

$$K_k = P_{k|k-1}(J_x^h(\hat{x}_{k|k-1}))^T S_k^{-1}. \tag{17}$$

The covariance associated with the final state estimate $\hat{x}_k$ is given by

$$P_k = P_{k|k-1} - K_k\, S_k\, K_k^T. \tag{18}$$

4 Experimental results

Experimental trials have been carried out using a robotised wheelchair prototype, see Fig. 4, built at the robotics lab of the University of "Roma Tre" and a Philips Vesta Pro Scan. The vehicle has two driving wheels equipped with low resolution incremental encoders (6.4 pulses per millimetre of the wheel movement). The proprioceptive sensory system is completed by a piezoelectric gyro (MuRata), that measures rotation velocity. The gyro has a good accuracy (3%) but is affected by temperature depending bias.

Five ultrasonic sensors with a radiation cone of 90° are also available, but they are not used in this work.

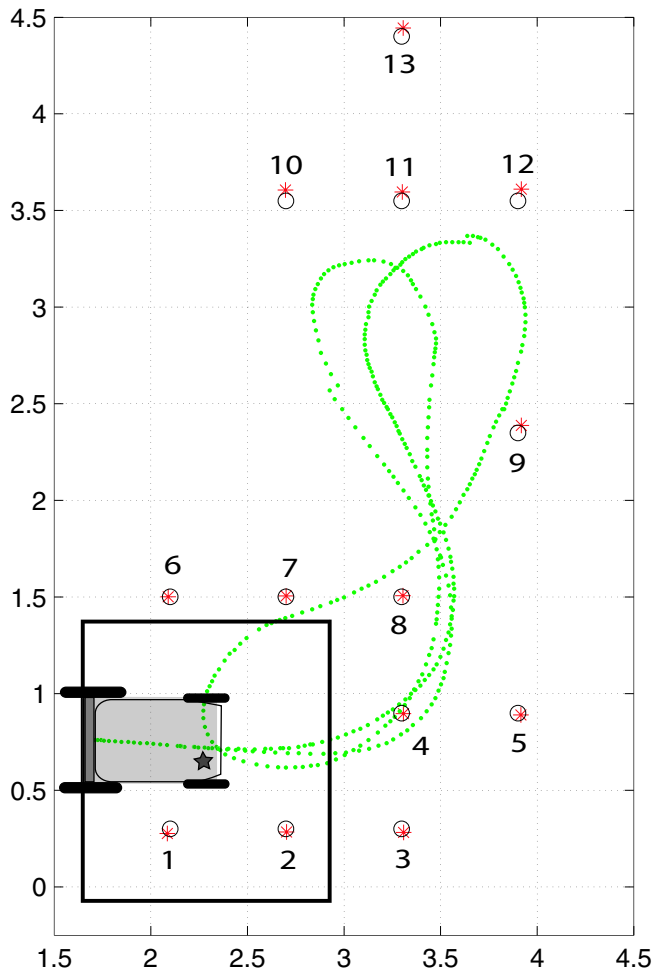

Figure 5: *Final map and estimated path*

The software of the overall system has been implemented on two notebooks connected over an Ethernet link: the first is a laptop installed on the wheelchair, while the second is devoted to process images from the vision system. The on-board laptop is a 486-PC and runs the control software. Since it is equipped with a data acquisition card (DAQPad 1200 by National Instruments), the control software has been developed using the graphical language of LabVIEW, while some of the most time-critical routines, as the SLAM filter, have been written in C. Rapid prototyping has been the philosophy underlying the system development. It motivates the choice of high level programming tools. In return for this, the system suffers from some fragility [13].

The web cam is mounted on the robot and focuses the ceiling. The distance between the vision system and the beacon (circular lamps) is about 2.50 m, so each pixel is about 5mm large at the CIF resolution (352×288).

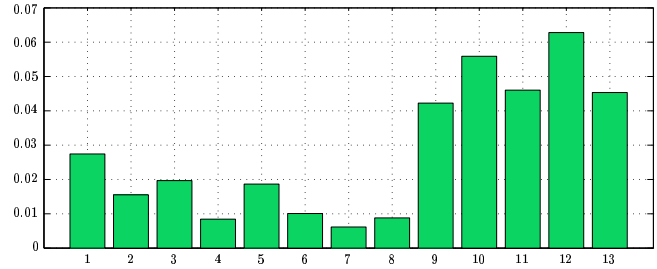

Figure 6: *Distance errors (m)*

The results of a first experiment is given in Fig. 5. The robot describes a double 8-path under the beacons and is able to localise itself during the navigation.

Table 1: *Map error (in cm)*

Beacons	Mean value	Standard deviation	Min error	Max error
Group A	1.4	0.7	0.6	2.5
Group B	5.0	0.9	4.2	6.3

The differences between the true wheelchair path and the path estimated using the SLAM filter are too small and cannot be appreciated in the figure.

In Fig. 5 are also shown the real dimensions of the wheelchair, the position of the web cam (identified by a star) and the viewing window in its home position.

Fig. 7, conversely, reports the odometric path as estimated on the basis of encoders and gyro data; note the large difference with Fig. 5. In the same figure, dots represent the history of the beacon positions as estimated on the basis of the odometry and the grabbed image.

The results obtained with map building show that, on the basis of the errors between actual and estimated positions reported in Fig. 6, the beacons can be divided into two groups, say group A (beacon 1-8) and group B (beacon 9-13). The distance from the groups is larger than the viewing window of the web cam, as it can be seen in Fig. 5.

This means that the robot uses only odometric prediction to estimate its configuration when travels from the zone A to the zone B, and the estimation error grows and cannot be reset using the beacons position, so the resulting map for group B is affected by an offset (see Tab.(1)).

5 Conclusion

This work presents a SLAM filter which successfully exploits low cost vision system measures to improve dead reckoning data and to maintain a correct

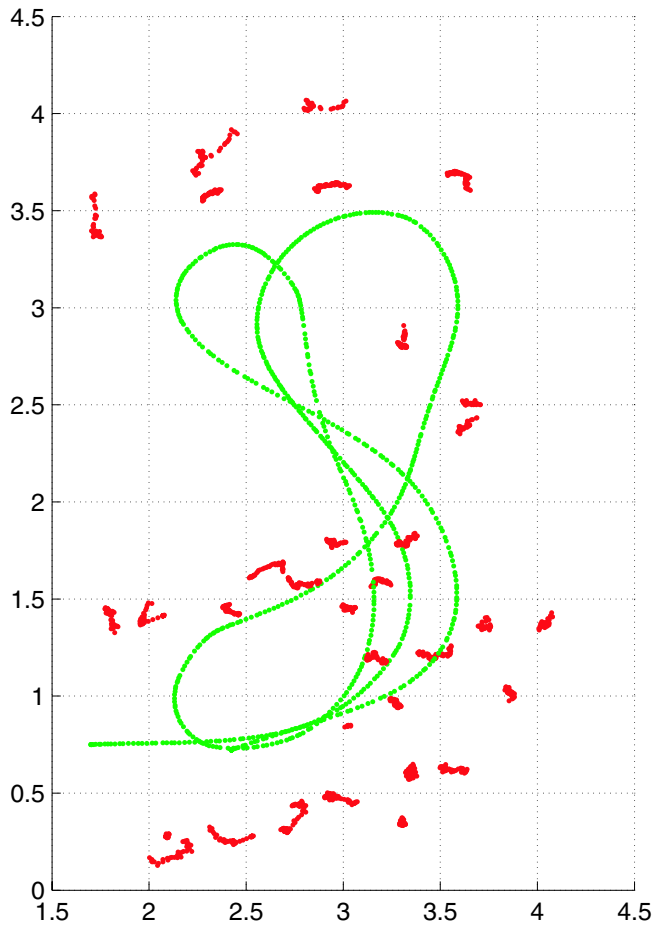

Figure 7: *Odometric path with beacons*

estimation of the configuration of a mobile robot while building a map of the environment.

Low cost vision hardware imposes the use of fast image processing and care of camera aberration.

To avoid computational complexity, we propose to model lens aberration changing the error covariance matrix R and use features easy to extract.

Experimental trials have shown the satisfactory performance of the method under realistic operating conditions. Future works will be devoted to reduce the overall computation complexity of the filter.

References

[1] Clark S., Dissanayake G., Newman P. and Durrant-Whyte H., "A Solution to Simultaneous Localization and Map Building (SLAM) Problem," *IEEE Transaction of Robotics and Automation*, Vol. 17, No 3. pp. 229-241, 2001.

[2] Levitt T.S., Lautwon D.T., "Qualitative navigation for mobile robots," *Artificial Intelligence Journal*, Vol. 44, No 3. pp. 305-360, 1990.

[3] Fabrizi E., Panzieri S., Ulivi G., "Extracting topological feature of indoor environment from sonar-based fuzzy maps" *Intelligent Autonomous Systems 6*, Pagello E., Groen F., Arai T., Dillmann R., Stenz A.(eds), IOS press, Amsterdam, 2000

[4] Kuipers B., Byun Y.T., "A robot exploration and mapping strategy based on semantic hierarchy of spatial representation", *Journal of Robotics and Autonomous Systems*, Vol. 8, pp. 47-63, 1991.

[5] Guivant J., Nebot E., "Optimization of Simultaneous Localization and Map Building Algorithm for Real Time Implementation" *IEEE Transaction of Robotics and Automation*, Vol. 17, No 3. pp. 242-256, 2001.

[6] Thrun S., Fox D., Burgard W., "A probabilistic approach to concurrent mapping and localization for mobile robots," *Mach. learning Autonom. Robots*, Vol. 31, pp. 29-53, 1998.

[7] Gelb A.C., "Applied optimal estimation," MIT Press, Cambridge, MA, 1994

[8] Panzieri S., Pascucci F., Setola R., Ulivi G., "A low cost vision based localization system for mobile robots", *Proc. of 9th Mediterranean Conf. on Control and Automation*, Doubrovnik, Croatia, 2001.

[9] Weng J., Cohen P., Herniou M., "Camera calibration with distortion model and accuracy evaluation," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 14, No 10, 1992.

[10] Panzieri S., Renzi S., Ulivi G., "A stereo vision based docking procedure for a mobile robot," *Proc. of 6th Int. IFAC Symp. On Robot Control SYROCO 2000*, Vienna, Austria, pp. 403-408, 2000

[11] Fabrizi E., Oriolo G., Panzieri S., Ulivi G. "Enhanced uncertainty modeling for robot localization," *Proc. of 7th Int. Symp. on Robotics with Application (ISORA'98)*, Anchorage, AL, 1998.

[12] Castellanos J. A., Montiel J.M.M., Neira J., Tardos J.D., "The SPmap: a probabilistic framework for simultaneous localization and map building", *IEEE Transaction on Robotics and Automation*, Vol. 15, No. 5, pp. 948-952, 1999.

[13] E. Fabrizi, S. Panzieri, G. Ulivi, "An integrated sensing-guidance system for a robotized wheelchair", *14th IFAC World Congress*, pp. 457-462, Beijing, China, July 5-9, 1999.