Topological map learning for a mobile robot in indoor environments*

Juan Andrade-Cetto and Alberto Sanfeliu
Institut de Robòtica i Informàtica Industrial, UPC-CSIC
cetto,sanfeliu@iri.upc.es

Abstract

A system that builds burrow-like topological maps and solves the localization of a mobile robot for indoor environments is presented. The approach uses visual features extracted from a pair of stereo images as landmarks. New landmarks are merged into the map and transient landmarks are removed from the map over time. A learning rule associated to each landmark is used to compute the landmark's *existence state*. The position of the robot in the map is estimated by combining sensor readings, motion commands, and the current map state by means of an Extended Kalman Filter. The combination of neural network principles for map updating and Kalman filtering for position estimation allows for robust robot localization in indoor dynamic environments.

Keywords: map learning, mobile robot navigation, topological maps.

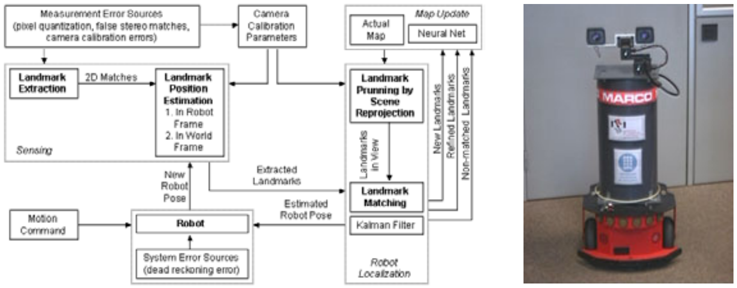
1 Introduction

Efficient indoor mobile robot navigation is limited mainly by the ability of a robot to perceive and interact with its surroundings in a deliberative way. And, for such interaction to take place, a model or description of the environment usually needs to be specified beforehand. If a global description or measurement of the elements present in the environment is not available, at least the descriptors and methods that will be used for the autonomous building of one are required. This is, either the robot has a global map, or it is given the means to learn one.

Many systems that incorporate human-made models of the environment have been successfully developed, even when only an approximate map is given, or in crowded environments [1, 2]. However, the autonomous building of a global, and possibly dynamic, map of the environment for a mobile robot is still a difficult problem. Three main difficulties arise during autonomous learning of an indoors map by a mobile robot, namely *dead reckoning*, *sensor uncertainty*, and *environment dynamics*. Map construction in mobile robotics has been made typically by updating grid maps of obstacles [3]. Probabilistic approaches that combine map learning and localization include [4, 5]. A method that combines sensor data from various robots to build a map using fuzzy set theory is presented in [6]. Some authors have recently proposed the use of goal oriented *cognitive maps* to learn the relationship between successively explored places [7, 8]. However, these methods are usually limited in that changing environments are only dealt with reactive behaviors.

* Work partially funded by CICYT project TAP98-0473, CONACYT and CETYS.

J.S. Sánchez and F. Pla editors, Proc. 9th Spanish Sym. Pattern Recog. Image Anal.
Benicassim, May 2001, Vol. 1, pp. 221-226.



(a) System Architecture (b) Physical View

Figure 1: MARCO Mobile Robot

A methodology for the construction and update of a dynamic topological map of a mobile robot is presented. Unlike grid-based techniques, it is scale independent. It was designed so that map updating can occur even in changing environments, and exploits the relationship between neighboring landmarks. It does not make any assumption on the distribution of the landmark positions, but it does expect white distribution of dead reckoning and sensor errors. The system architecture is shown in Fig. 1a, and includes three distinctive modules: sensing, robot localization, and map updating.

2 Sensing

The extraction of landmarks from the environment is based solely on visual information. A *salient feature locator* that uses Beaudet's cornerness measure [9], with further refinement using the *variance descent approach* [10] was implemented. These salient features are then pairwise matched in the stereo set by correlation, and by the enforcement of epipolar constraints. Each feature's 3-D position with respect to the robot is reconstructed from stereo geometry. The 3-D position of a feature $\mathbf{z}_i^R$ with respect to the robot, and an associated vector of appearance properties, constitute a *landmark*. The appearance properties, which are used to validate future map landmark matches, include the pixel gray-level mean and distribution over a small window around the salient feature, and the energy of the feature computed from Beaudet's cornerness measure.

The position of a landmark in world coordinates is a nonlinear function of the robot position, the landmark position in robot coordinates, and the uncertainty in sensor measurement: $\mathbf{z}_i^W(k) = h(\mathbf{z}_i^R(k), \mathbf{x}(k), \mathbf{v}(k))$. A noise-free approximate measure of this quantity is given by

$$\tilde{\mathbf{z}}_i^W(k) = \mathbf{R}\mathbf{z}_i^R + \mathbf{t}, \quad \mathbf{R} = \begin{bmatrix} \sin(\theta) & -\cos(\theta) & 0 \\ \cos(\theta) & \sin(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{t} = (x, y, 0)^T \tag{1}$$

and a linearized version for this quantity can be expressed as a Taylor Series with the higher order terms dropped: $\mathbf{z}_i^W(k) \approx \tilde{\mathbf{z}}_i^W(k) + \mathbf{H}(k)(\mathbf{x}(k) - \tilde{\mathbf{x}}(k)) + \mathbf{v}(k)$. The prediction error $\mathbf{x}(k) - \tilde{\mathbf{x}}(k)$ is computed during robot localization, and the Jacobian or measurement innovation matrix takes the simple form

$$\mathbf{H}(k) = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}(\mathbf{z}_i^R(k), \tilde{\mathbf{x}}(k), 0) = \left[\begin{array}{cc|c} 1 & 0 & \\ 0 & 1 & \dot{\mathbf{R}}\mathbf{z} \\ 0 & 0 & \end{array} \right] \tag{2}$$

3 Map Update

To update the map of the environment, each landmark coming from sensor measurements is searched for a match in the current map. This search is limited to a reduced number of landmarks; all current map entries are first reprojected into the visual space, and only those map landmarks that fall within the field of view at the robot's current position are considered during the map update process.

If the sensed landmark position falls within the *uncertainty region* of a map landmark, and their appearance properties vectors are highly correlated, then we have a scene-to-map landmark match. Once a match is obtained, the distribution parameters for the uncertainty of the landmark position are updated, as well as the vector of appearance properties. The distribution of the uncertainty of a landmark position in the map is parameterized by a normal distribution with mean $\bar{\mathbf{z}}_i^W$ and covariance $\mathbf{S}_i$.

One of the main contributions of this work resides on the formulation of the landmark *existence state*, a measuring device of how persistent a landmark is in order to be considered a strong reference for environment representation and robot localization. Temporary landmarks and those coming from noisy sensor readings are pruned from the map as their existence state diminishes over time. On the other hand, those landmarks that are repeatedly seen are considered stronger indicators of the structure of the environment.

We have resorted to neural network principles for the formulation of the landmark existence state, because of the exponential decay properties of the perceptron update rule, and the possibility to link neighboring landmarks in a networked representation. Within this framework, two approaches have been analyzed for the construction of a topological map: first, we consider the landmarks as uncorrelated features that characterize the robot environment; and secondly, their reciprocal relationship is examined.

3.1 Independent Landmarks

For each landmark in the map, there exists an associated perceptron that will register how persistent, and how old the landmark is. The state of the perceptron x will be considered as the *existence state* or *strength* of a given landmark, and the training information for the neuron is the landmark identification stamp at time k. The proposed update rule equations for the existence of landmark i in the map are

$$x_i(k+1) = \frac{1}{1 + e^{-w_i(k)x_i(k)}}, \quad w_i(k+1) = w_i(k) + (1 - x_i(k))(\bar{e}_i(k) - c_f) \tag{3}$$

where $0 \leq c_f \leq 1$ is a *forgetting factor*, $w_i(k)$ is the neuron weight for landmark i at time k, and $\bar{e}_i(k) = \{0, 1\}$ its identification stamp. If $c_f \approx 0$ all landmarks will prevail in the map regardless of how old they are or how many identification stamps they got. On the other hand, when $c_f \approx 1$, the landmarks will be forgotten faster, a desirable situation in highly dynamic environments. For the later case only persistent landmarks will remain in the map, rewarding the associated neuron with its presence and penalizing it if they are not identified from the current view of the scene. It should be stressed that the neuron states are only updated for those landmarks in the map that fall within the viewable area at time k. Finally, if the *existence state* $x_i(k)$ falls below a *forgetting threshold* t_f, it means the corresponding landmark has been forgotten, and it is immediately removed from the map.

3.2 Correlated Landmarks

The learning rule in Eq. 3 omits the relationships that exist among different landmarks, thus neglecting correlational information. To model these correlations, consider now the following learning rule

$$x_i(k+1) = \frac{1}{1 + e^{-\sum_{j \in I(k)} w_{ji}(k) x_j(k)}}, \quad w_{ij}(k+1) = w_{ij}(k) + x_i(k)(1 - x_j(k))(\bar{e}_i(k) - c_f) \quad (4)$$

with $I(k)$ the set of landmarks in the map that, when reprojected at time k, persist in the field of view. Note that different from the perceptron rule used for the uncorrelated case, we need now to update as much as $|I(k)|$ weights for each landmark instead of just one. The time and space complexities of the map updating algorithm are $|I(k)|$ times greater in average for the correlated case.

The proposed map update schemes have the following advantages over other map learning algorithms: (1) The map preserves its topological structure. The prevailing relationship among existing features are their own Euclidean metrics, as well as the learned weights for the correlated case; (2) The map is not limited in resolution, as opposed to grid-based maps. This permits the modeling of different size environments without the need to modify its general structure; and (3) The dynamic property of the map allows for the robust modeling of changing or noisy environments. It also restricts the map from growing indefinitely, a situation that could affray with system resources (search speed and memory).

4 Robot Localization

Localization techniques can be divided mainly in three groups: (1) *correlation methods* that match sensor signals against previously stored maps; (2) *Kalman filters* that estimate the current robot position from current and previous sensor readings, past position estimates, and motion commands, as well as uncertainty measurements of sensory and motion information; and (3) *Markov* localization techniques, which use a probabilistic framework to maintain a position probability density over the whole set of robot poses [2]. Kalman filters are typically robust for local localization, whereas Markov localization is better suited for global localization. The former technique requires that the initial location of the robot be

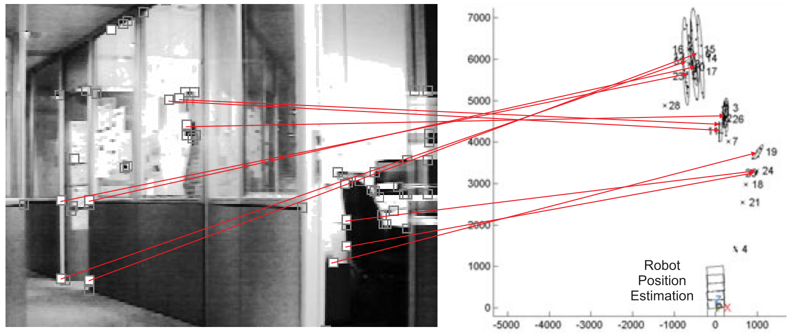


Figure 2: Detected Landmarks and their Uncertainties

known, whereas the latter method usually requires stronger assumptions about the nature of the environment.

In this work, we are limited to the construction of an environment map starting from a known position or *origin*. Also, since the computation of the actual robot position is very sensitive to the accumulation of dead-reckoning error, we have opted for the use of an Extended Kalman Filter for robot localization.

The time update equations for the *a priori* estimate of the robot position at time $k+1$, and its error covariance are $\hat{\mathbf{x}}^-(k+1) = \hat{\mathbf{x}}(k) + \mathbf{u}(k)$, and $\mathbf{P}^-(k+1) = \mathbf{P}(k) + \mathbf{Q}(k)$, respectively. The motion command vector is $\mathbf{u}(k)$; and the robot motion noise $\mathbf{Q}(k)$ is typically set to a constant value, and can be computed by running a set of motion commands and parameterizing the deviation from the desired pose. If the number of samples is sufficiently large, white noise parameters can be estimated for $\mathbf{Q}$. The position error covariance estimate, can be initialized to $\mathbf{I}$ and updated with $\mathbf{P}(k) = (\mathbf{I} - \mathbf{K}(k)\mathbf{H}(k))\mathbf{P}^-(k)$.

Lastly, the Kalman Filter Gain $\mathbf{K}(k)$ is chosen such that the a posteriori error covariance $\mathbf{P}(k)$ is minimized: $\mathbf{K}(k) = \mathbf{P}^-(k)\mathbf{H}^T(k)\left(\mathbf{H}(k)\mathbf{P}^-(k)\mathbf{H}^T(k) + \mathbf{S}(k)\right)^{-1}$, with $\mathbf{S}$(k) being the measurement residual error covariance at time k for each landmark. The reader is referred to [11, 12] for a detailed explanation on Kalman Filtering techniques. Our formulation follows closely that of [12].

5 Experiments

The mobile robot platform MARCO used in our experiments is shown in Fig. 1b. A sample run of our map building method is shown in Fig. 2, where the hollow boxes show a group of salient features extracted from one of the images, and the filled boxes correspond only to those features that have been matched properly in the stereo pair. Fig. 2 shows also a top view of the estimated position of the matched landmarks with respect to the robot as well as their position error covariance estimate in the form of uncertainty ellipses. The sample run shown in Fig. 2 considers only the case of uncorrelated corner landmarks. The scene contains a total of 28 landmarks. Landmarks 7, 18, 21, and 28 were seen only once, and they represent noise or spurious data. Stronger landmarks, such as the ones numbered 3, 19, 24, and 26, have smaller error covariance estimates when compared to landmarks 14, 15, 16, and 17, which are seen fewer times and are also further apart from the stereo head. The updated robot position is also shown in the figure.

References

[1] A. Ohya, A. Kosaka, and A. C. Kak. Vision-based navigation by a mobile robot with obstacle avoidance using single-camera vision and ultrasonic sensing. *IEEE Trans. Robot. Automat.*, 14(6):969–978, Dec. 1998.

[2] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *J. Artif. Intell. Res.*, 30:391–427, Nov. 1999.

[3] G. Oriolo, G. Ulivi, and M. Vendittelli. Real-time map building and navigation for autonomous robots in unknown environments. *IEEE Trans. Syst., Man, Cybern. B*, 28(3):316–332, Jun. 1998.

[4] S. Thrun. Bayesian landmark learning for mobile robot localization. *Mach. Learn.*, 33(1):41–76, Oct. 1998.

[5] J. A. Castellanos, J. M. M. Montiel, J. Neira, and J. D. Tardós. The SPMap: A probabilistic framework for simultaneous localization and map building. *IEEE Trans. Robot. Automat.*, 15(5):948–952, Oct. 1999.

[6] M. Lpez-Snchez, F. Esteva, R. Lpez de Mntaras, C. Sierra, and J. Amat. Map generation by cooperative low-cost robots in structured unknown environments. *Auton. Robot.*, 5:53–61, 1998.

[7] R. Araújo and A. T. de Almeida. Learning sensor-based navigation of a mobile robot in unknown worlds. *IEEE Trans. Syst., Man, Cybern. B*, 29(2):164–178, Apr. 1999.

[8] A. Arleo, J. del R. Millán, and D. Floreano. Efficient learning of variable-resolution cognitive maps for autonomous indoor navigation. *IEEE Trans. Robot. Automat.*, 15(6):990–1000, Dec. 1999.

[9] R. Deriche and G. Giraudon. A computational approach for corner and vertex detection. *Int. J. Comput. Vision*, 10(2):101–124, 1993.

[10] R. Deriche and T. Blaszka. Recovering and characterizing image features using an efficient model based approach. In *Proc. 8th IEEE Conf. Comput. Vision Pattern Recog.*, pages 530–535, New York, Jun. 1993.

[11] P. S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 1. Academic Press, New York, 1979.

[12] G. Welch and G. Bishop. An introduction to the Kalman filter. Technical Report TR 95-041, Dept. of Comput. Sci., Univ. of North Carolina at Chapel Hill, 1995.