Robotics and Autonomous Systems 60 (2012) 487–497

Contents lists available at SciVerse ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

A Bayesian approach for place recognition

Fabio Ramos [a,*], Ben Upcroft [b], Suresh Kumar [a], Hugh Durrant-Whyte [a]

[a] Australian Centre for Field Robotics, School of Information Technologies, The University of Sydney, Sydney, NSW 2006, Australia

[b] School of Engineering Systems, Faculty of Built Environment and Engineering, Queensland University of Technology, Brisbane, QLD 4000, Australia

ARTICLE INFO

Article history:
Received 30 December 2005
Received in revised form
5 August 2011
Accepted 8 November 2011
Available online 20 November 2011

Keywords:
Place recognition
Bayesian inference
Dimensionality reduction
Mobile robots

ABSTRACT

This paper presents a robust place recognition algorithm for mobile robots that can be used for planning and navigation tasks. The proposed framework combines nonlinear dimensionality reduction, nonlinear regression under noise, and Bayesian learning to create consistent probabilistic representations of places from images. These generative models are incrementally learnt from very small training sets and used for multi-class place recognition. Recognition can be performed in near real-time and accounts for complexity such as changes in illumination, occlusions, blurring and moving objects. The algorithm was tested with a mobile robot in indoor and outdoor environments with sequences of 1579 and 3820 images, respectively. This framework has several potential applications such as map building, autonomous navigation, search-rescue tasks and context recognition.

1. Introduction

Localisation in complex environments is one of the main challenges for autonomous navigation and planning. This task can be performed by solving the simultaneous localisation and map building problem (SLAM) for which there exists a large literature (see [1] for an overview and further references). In SLAM the robot's position is estimated based on its relative position with respect to landmarks. At the same time, a map of the environment is built with the estimated position of those landmarks. Identification of landmarks in an unstructured environment is an area of intensive research. Landmarks need to be selected to facilitate further detection and association while improving the quality of the map [2]. Humans, however, rather than navigating using relative coordinates, have an abstract notion of distance while still being able to estimate their position in space. This ability is provided mainly from visual information associated with an internal (map) representation that localises the person based on the appearance of a scene rather than on precise distance measurements to landmarks [3,4].

The idea of building an appearance-based model is explored in the current paper for the purpose of robot localisation in indoor and outdoor environments. The proposed framework can be applied, for example, to planning tasks where the goal is to navigate to a particular place without necessarily specifying map coordinates. In this sense the interaction between the operator and the mobile robot is facilitated since it is easier to send commands such as leave the car park and go to the library rather than go to position (x, y).

In the proposed approach, places are learnt and recognised from images obtained with a camera mounted on a mobile robot. These images are subject to changes in illumination, blurrings, occlusions, moving objects, etc. The goal is thus to incrementally built a multi-class classifier from very few images so as to label new images as the robot navigates. No other extra information such as a topological map is provided. The difficulty of the problem can be seen in Fig. 1 where some images of the testing dataset are depicted.

Place recognition and localisation from images forms an approach to SLAM [5]. Some previous approaches use image histograms and topological maps for classification [6,7]. Others use invariant features such as [8,9]. In another approach, image features were used to estimate the position of the robot for map building problems [10–12]. Cummins et al. demonstrated loop closure over a 1.6 km path length based solely on the appearance of the image [13]. However, the offline learning procedure takes three hours and cannot be subsequently updated. Milford and Wyeth demonstrate multiple loop closures over a large scale topological map using a biologically inspired visual system [14] but require extensive parameter tuning. The novelty of this work lies in robust recognition from very few training images (usually 3–10 per place), without the need of a map, in a theoretical sound Bayesian framework. Furthermore, the recognition system was tested in both indoor and outdoor environments proving to be robust for practical applications.

* Corresponding author. Tel.: +61 293517156; fax: +61 293517474.
E-mail addresses: f.ramos@acfr.usyd.edu.au (F. Ramos), ben.upcroft@qut.edu.au (B. Upcroft), suresh@acfr.usyd.edu.au (S. Kumar), hugh@acfr.usyd.edu.au (H. Durrant-Whyte).

0921-8890/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2011.11.002

Fig. 1. Images used for place recognition. As can be observed, blurring, occlusions, changes in illumination and moving objects such as people and cars are some of the issues the place recognition algorithm has to cope with.

In the solution proposed, the world is interpreted as a set of places. Each place has a probabilistic representation learnt from images. Localisation is performed in near real-time by evaluating the responses of each model given a new image. The place recognition task is treated as a Bayesian learning problem in a space of *essential features*. Initially, training images are divided into small patches that constitute a high dimensional set. The dimensionality of this set is then reduced with nonlinear and neighbourhood preserving techniques to create a low dimensional set. These two sets are used to learn a mixture of linear models for nonlinear regression, from points in the high to the low dimensional space. Points in this low dimensional space constitute the set of *essential features* and are used in the next step where the variational approximation for Bayesian learning is computed to create a probabilistic density for each place. Recognition is performed by computing the log-likelihood of an entire image over each place model. This approach was tested with sequences of images obtained by the mobile robot in Fig. 2. The platform operated under differing conditions including moving objects, changes in illumination, different viewpoints, occlusions, outdoor and indoor environments, demonstrating robust localisation.

This paper is organised as follows. In Section 2, the approach for place recognition and how dimensionality reduction, nonlinear regression and variational Bayesian learning can be combined for a multi-class classification problem are explained. Section 3 shows some details of the implementation while Section 4 presents experimental results. Finally, conclusions are given in Section 5.

Fig. 2. Pioneer AT used in our experiments.

2. Algorithm description

This paper focuses on a classification procedure to map images to labels. Each label corresponds to a place learnt from a set of images. The learning algorithm is supervised as every image in the training set has an assigned label. Thus, given a training set of n pairs $(\mathbf{I}_i, p_i)$, where $\mathbf{I}_i$ is the ith image and p_i is the label of that particular image, the algorithm needs to generate a model to classify new images. These images can be obtained at different view points or have partial occlusions.

Additionally, a very desirable feature in recognition algorithms for robotics applications is the ability to learn accurate models from very small training sets. This minimises the tedious task of manual labelling while reducing training time and the storage demands for the data. Therefore, we derive an algorithm that uses all the information available from the image, as opposed to conventional approaches e.g. [8,9] where only image features are used, disregarding the rest of the image. The algorithm has two main parts, the first is an unsupervised dimensionality reduction method where the extracted features are compressed while preserving most of the information content in a principled manner. The second step builds a generative model for each

place using Bayesian learning techniques. This provides a very attractive and fundamentally sound method to learn from few examples by combining prior information with training data. Bayesian quantities, such as the marginal likelihood employed in this work, contains the Occam's Razor principle [15] which essentially states that the simplest model is always preferable. This avoids overfitting and allows the comparison of different statistical models. In this paper, the models to be compared are mixtures of Gaussians with different number of components.

The algorithm starts by dividing a given image into sets of non-overlapping patches of the same size. The disposition of these patches follow a grid or a lattice. Thus an image $\mathbf{I}_i$ is represented as a set of m patches $\{I_{i,1}, \ldots, I_{i,m}\}$. As a colour camera is used, each pixel in the patch has three values representing red, green and blue intensities. To further quantify texture, each patch is convolved with a sequence of Gabor wavelets at different scales and orientations. This is similar to the result obtained with steerable pyramids; a well known and accepted procedure to extract texture information in computer vision [16,17]. This convolution has also a biological interpretation as it provides a good approximation of natural processes for spectral decomposition that occurs in the primary visual cortex. Each patch now has a feature-vector representation $x_{i,j} = [I_{i,j}, \phi(I_{i,j})]^T \in \mathbb{R}^D$, where ϕ is an operator to indicate the Gabor wavelet convolutions.

The dimension D is usually intractable for building efficient representation models. For example, if the size of the patch is 5×5 pixels and using four Gabor wavelets, D equals 175 ($5 \times 5 \times 3$ corresponding to colour values, plus $5 \times 5 \times 4$ corresponding to four Gabor wavelet convolutions). To cope with this high dimensional problem, dimensionality reduction techniques are applied to extract the *essential* information of each patch and represent them in a lower dimensional space. This procedure, however, needs to preserve important characteristics of the data such as keeping the neighbourhood of points unchanged. This ensures that patches with similar appearance, for example representing trees and grass are located nearby in the low-dimensional representation.

Points in the low-dimensional space can then be classified using generative models for each place. These models are learnt using the variational Bayesian expectation maximisation algorithm which is described later in this section. Fig. 3 depicts a diagram of the learning and the classification procedures.

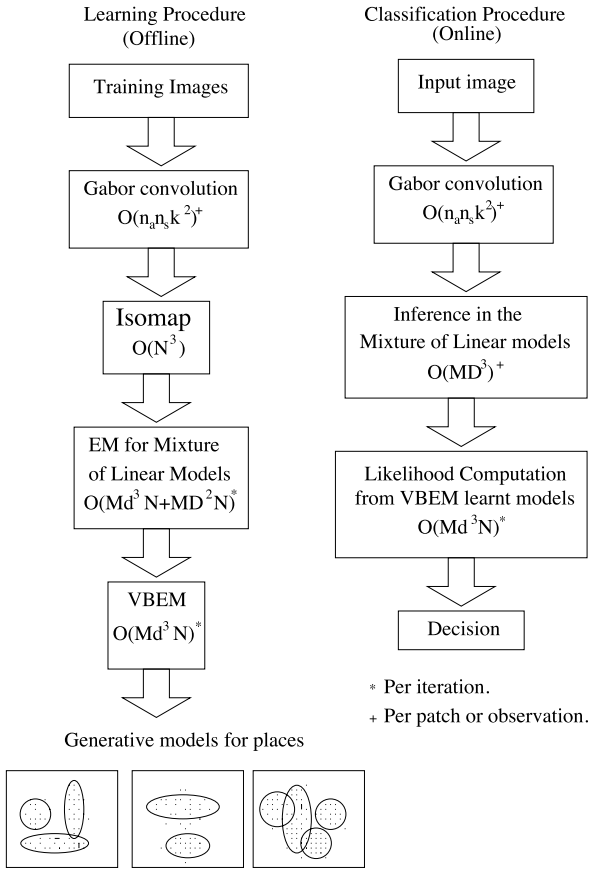

Fig. 3. Learning and classification procedures and complexities for place recognition. In this diagram, N is the number of samples; M is the number of components; D and d are the sizes of the high and the low dimensional spaces; k is the size of the width of patches; n_s and n_a are the number of scales and angles for the Gabor convolution.

2.1. Neighbourhood-preserving dimensionality reduction

Dimensionality reduction is one of the techniques that can manage the amount of information robotics' applications face. In this work, a nonlinear technique, Isomap [18], is applied to reduce the dimensionality of image patches into a feasible number where further statistical learning methods can be used. As opposed to principal components analysis (PCA) [19] and multidimensional scaling (MDS) [20], Isomap has the desired property of preserving the neighbourhood of points in the low dimensional manifold.

Isomap works in three steps. In the first, distances between points in the high dimensional space are computed in order to determine neighbours. In the second step, Isomap estimates the geodesic distances d_G between all pairs of points by computing their shortest path distances. In the final step, classical MDS is used to compute a graph embedding in k (low) dimensional space that closely respects the geodesic distances. The coordinate vectors w_i are chosen to minimise the norm $\sqrt{\sum_{i,j}(\tau(d_G) - \tau(d_W))^2_{ij}}$, where d_W is the matrix of output space distances and τ is an operator that converts distances into inner products. The global minimum of the cost function is computed by setting the output space coordinates w_i to the top l eigenvectors of $\tau(d_G)$.

Isomap is applied to the training set of patches returning a set of points in a low dimensional space d, $\{y_{i,1}, \ldots, y_{i,m}\}$, where $y_{i,j} \in \mathbb{R}^d$. Note that the dimensionality of the embedded manifold can be directly estimated from the Isomap algorithm by observing the residual variance of the norm above.

2.2. Fast dimensionality reduction with non-linear regression

Isomap and indeed most nonlinear dimensionality reduction algorithms are inherently deterministic algorithms and do not provide a measure of uncertainty of underlying states of high dimensional observations. In addition, Isomap does not output a model or function to directly compute the low dimensional coordinates of new observations, thus requiring k-neighbours based algorithms that can be cumbersome in real-time applications.

An alternative solution is to learn a generative model $p(x|y)$, where x is a vector in the high dimensional space and y is its low dimensional representation. This model encapsulates the uncertainties inherent in the inference of low dimensional points from noisy high dimensional observations. It can be learnt in a supervised manner to derive compact mappings that generalise over large portions of the input and embedding space. The input–output pairs of Isomap can serve as training data for an invertible function approximator in order to learn a parametric mapping between the two spaces. Once the model is learnt, the low-dimensional representation for patches from new images can be computed efficiently. This is the key point to enable real-time recognition since the essential features of a new image can be quickly computed from the model by making probabilistic inferences.

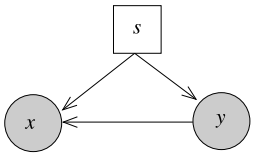

Fig. 4. Graphical model for computation of parametric models from non-linear dimensionality reduction algorithms. An arrow directed into a node depicts a dependency on the originating node. The discrete hidden variable s represents a specific neighbourhood on the manifold.

Given the results of Isomap, a mixture of linear models can be learnt to perform dimensionality reduction quickly. This mixture model is very attractive computationally as it can be trained with Expectation Maximisation (EM) [21] and inference can be efficiently computed. The joint probability distribution for the mixture model $p(x, y, s)$ contains a hidden discrete variable s representing the weights of the components. Mixture of linear models are similar to mixtures of factor analysers, that are commonly used to perform simultaneous clustering and local dimensionality reduction [22]. The only differences are that the low dimensional variable y is observed (through Isomap), not hidden, and the Gaussian distributions $p(y \mid s)$ have nonzero mean vectors ν_s and full covariance matrices Σ_s. Learning when the variable y is observed seems to discover a solution of better quality than in the opposite situation as in the conventional mixture of factor analysers [23].

The graphical model in Fig. 4 depicts the assumed dependences. The discrete hidden variable s introduced in the model physically represents a specific neighbourhood on the manifold over which a mixture component is representative. This representation conveniently handles highly nonlinear manifolds through the ability to model the local covariance structure of the data in different areas of the manifold. It can be trained with very large datasets and the computational cost of inferences does not depend on the number of training samples. This is the main advantage over non-parametric techniques such as Gaussian processes [24].

The complete generative model can now be summarised based on the assumed dependences (Eqs. (1)–(3)). The joint probability distribution in the graphical model is expressed as

$$p(x, y, s) = p(x \mid y, s)p(y \mid s)p(s) \tag{1}$$

where the conditional distributions are given by

$$p(x \mid y, s) \propto \exp\left\{-\frac{1}{2}\left[x - \Lambda_s y - \mu_s\right]^T \Psi_s^{-1}\left[x - \Lambda_s y - \mu_s\right]\right\}, \tag{2}$$

$$p(y \mid s) \propto \exp\left\{-\frac{1}{2}\left[y - \nu_s\right]^T \Sigma_s^{-1}\left[y - \nu_s\right]\right\}. \tag{3}$$

In the distributions above, Λ_s is a linear transformation matrix, μ_s is the high dimensional mean, Ψ_s is a diagonal matrix with variances in the high dimensional space, ν_s is the low dimensional mean and Σ_s the full covariance matrix for the low dimensional space. All of them defined for component s.

A common inference in this model is the evaluation of the posterior $p(y, s \mid x_i)$. This posterior represents the probability of the low dimension point given an ith observation in the high dimensional space. From the joint distribution, it is calculated as

$$p(y, s \mid x_i) = \frac{p(x_i \mid y, s)p(y \mid s)p(s)}{\sum_{s'} \int p(x_i \mid y, s')p(y \mid s')p(s')\, dy}. \tag{4}$$

The solution of this expression results in a mixture of Gaussians with means given by

$$\begin{aligned}\mu_{y|s,x_i} &= E\left[y \mid s, x_i\right] \\ &= \nu_s + \left(\Sigma_s^{-1} + \Lambda_s^T \Psi_s^{-1} \Lambda_s\right)^{-1}\left(\Lambda_s^T \Psi_s^{-1}\right)(x_i - \mu_s - \Lambda_s \nu_s)\end{aligned} \tag{5}$$

and covariances

$$\begin{aligned}\Sigma_{y|s,x_i} &= E\left[yy^T \mid s, x_i\right] \\ &= \Sigma_s - \Sigma_s^T \Lambda_s^T \left(\Psi_s + \Lambda_s \Sigma_s^T \Lambda_s^T\right)^{-1} \Lambda_s \Sigma_s.\end{aligned} \tag{6}$$

Since Ψ_s is a diagonal matrix and Σ_s is assumed to be non-singular (since it is a covariance matrix), the inverse of $\Psi_s + \Lambda_s \Sigma_s^T \Lambda_s^T$ can be efficiently computed by using the matrix inversion lemma (Sherman–Morrison–Woodbury).

Weights can be computed by marginalising the joint probability $p(x, y \mid s)$ over y to obtain:

$$p(s \mid x_i) = \frac{p(x_i \mid s)p(s)}{\sum_{s'} p(x_i \mid s')p(s')}, \tag{7}$$

where

$$\begin{aligned}p(x_i \mid s) = &\frac{1}{(2\pi)^{D/2} \mid \Psi_s + \Lambda_s \Sigma_s^T \Lambda_s^T \mid^{1/2}} \\ &\times \exp\left\{-\frac{1}{2}(x_i - \mu_s - \Lambda_s \nu_s)^T \right. \\ &\left. \times \left(\Psi_s + \Lambda_s \Sigma_s^T \Lambda_s^T\right)^{-1}(x_i - \mu_s - \Lambda_s \nu_s)\right\}.\end{aligned}$$

The result of the inference process is thus a mixture of Gaussians with means $\mu_{y|s,x_i}$, covariances $\Sigma_{y|s,x_i}$ and weights $p(s \mid x_i)$. To make it feasible for the Bayesian learning this mixture is collapsed so as to have a single mean, which will be used as a training point, and a covariance matrix which will be used in the initialisation of the hyper-parameters.

2.3. VBEM for mixtures of Gaussians

The data now represented with its essential features in the low dimensional space is used to learn a generative model for each place. This problem is formulated in a Bayesian framework where the model selection task consists of calculating the posterior distribution over a set of models (which in this case will be mixtures of Gaussians with different numbers of components) given the prior knowledge and the dataset. Denoting $\mathbf{s}$ for the hidden variable representing the weights, $\mathbf{y}_i$ for the observations of a place i, θ for the parameters of a model M and $p(M)$, $p(\theta \mid M)$ for the prior over models and their parameters, respectively, the posterior over models M given these observations is given by:

$$p(M \mid \mathbf{y}_i) = \frac{p(\mathbf{y}_i \mid M)\, p(M)}{p(\mathbf{y}_i)}. \tag{8}$$

The best model M is the model that maximises this posterior. The first term in the numerator of Eq. (8) is known as *marginal likelihood* and is the key expression in the Bayesian formulation for model selection, since it represents an average of how good a particular model fits the observations over all possible parametrisation, convoluted by the prior. This quantity permits the comparison of different models given the data by encoding Occam's Razor principle. Unfortunately, the computation of the marginal likelihood involves the solution of integrals which in most cases do not have an analytical form. To circumvent this issue, the variational Bayesian approach is employed. The main idea is to approximate the marginal likelihood with a lower bound by using variational calculus techniques [25–27].

Introducing a free distribution q over $\mathbf{s}$ and θ, with $\int \sum_{\mathbf{s}} q(\mathbf{s}, \theta)\, d\theta = 1$, and applying Jensen's inequality [28], it is possible to

compute a lower bound on the log of the marginal likelihood:

$$\ln p\left(\mathbf{y}_i \mid M\right) \geq \int \sum_{\mathbf{s}} q\left(\mathbf{s}, \theta\right) \ln \frac{p\left(\mathbf{s}, \mathbf{y}_i, \theta \mid M\right)}{q\left(\mathbf{s}, \theta\right)} d\theta. \tag{9}$$

Maximising this lower bound with respect to the free distribution $q\left(\mathbf{s}, \theta\right)$ is difficult. A better strategy is to factorise this free distribution to yield a variational approximation in which $q\left(\mathbf{s}, \theta\right) \approx q_{\mathbf{s}}\left(\mathbf{s}\right) q_{\theta}\left(\theta\right)$:

$$\ln p\left(\mathbf{y}_i \mid M\right) \geq \int \sum_{\mathbf{s}} q_{\mathbf{s}}\left(\mathbf{s}\right) q_{\theta}\left(\theta\right) \ln \frac{p\left(\mathbf{s}, \mathbf{y}_i, \theta \mid M\right)}{q_{\mathbf{s}}\left(\mathbf{s}\right) q_{\theta}\left(\theta\right)} d\theta \tag{10}$$

$$= \mathcal{F}_M\left(q_{\mathbf{s}}\left(\mathbf{s}\right), q_{\theta}\left(\theta\right), \mathbf{y}_i\right). \tag{11}$$

The quantity $\mathcal{F}_M$ is a functional of the free distributions $q_{\mathbf{s}}\left(\mathbf{s}\right)$ and $q_{\theta}\left(\theta\right)$ and is known as the negative free energy. The variational Bayesian algorithm iteratively maximises $\mathcal{F}_M$ with respect to the free distributions until the function reaches a stationary value. By taking the functional derivatives of $\mathcal{F}_M$ with respect to $q_{\mathbf{s}}\left(\mathbf{s}\right)$ and $q_{\theta}\left(\theta\right)$, and equating them to zero, $\frac{\partial}{\partial q_{\mathbf{s}}(\mathbf{s})} \mathcal{F}_M\left(q_{\mathbf{s}}\left(\mathbf{s}\right), q_{\theta}\left(\theta\right)\right) = 0$, $\frac{\partial}{\partial q_{\theta}(\theta)} \mathcal{F}_M\left(q_{\mathbf{s}}\left(\mathbf{s}\right), q_{\theta}\left(\theta\right)\right) = 0$, produces:

VB-E Step:

$$q_{\mathbf{s}}^{(t+1)}\left(\mathbf{s}\right) \propto \exp\left[\int \ln p\left(\mathbf{s}, \mathbf{y}_i \mid \theta, M\right) q_{\theta}^{(t)}\left(\theta\right) d\theta\right] \tag{12}$$

VB-M Step:

$$q_{\theta}^{(t+1)}\left(\theta\right) \propto p\left(\theta \mid M\right) \exp\left[\sum_{\mathbf{s}} \ln p\left(\mathbf{s}, \mathbf{y}_i \mid \theta, M\right) q_{\mathbf{s}}^{(t+1)}\left(\mathbf{s}\right)\right]. \tag{13}$$

An interesting implementation of VBEM uses conjugate priors that are analytically tractable and easy to interpret. Thus, Dirichlet, Normal and Wishart multivariate distributions [29] are used as priors over weights, means and covariances. They are denoted as $\mathcal{D}\left(\pi; \lambda\right)$, $\mathcal{N}\left(\mathbf{x}; \mu, \mathbf{\Sigma}^{-1}\right)$ and $\mathcal{W}\left(\Gamma; \alpha, \mathbf{B}\right)$ and are functions of their hyper-parameters. Also, a multivariate Student-t distribution $\mathcal{S}\left(\mathbf{x}; \rho, \mathbf{\Lambda}, \omega\right)$ is used to represent the predicted density.

For the particular case of a Gaussian mixture model M with S components, where each component has weight given by π_s, mean μ_s and covariance $\mathbf{\Gamma}_s$, the set of parameters can be written as $\theta = \{\pi, \mu, \mathbf{\Gamma}\}$ where $\pi = \{\pi_1, \pi_2, \ldots, \pi_S\}$, $\mu = \{\mu_1, \mu_2, \ldots, \mu_S\}$ and $\mathbf{\Gamma} = \{\mathbf{\Gamma}_1, \mathbf{\Gamma}_2, \ldots, \mathbf{\Gamma}_S\}$.

Given these parameters and the model, the likelihood of an observation $y_{i,j}$ in a d-dimensional space can be written as

$$p\left(y_{i,j} \mid \theta, M\right) = \sum_{s'=1}^{S} p\left(s = s' \mid \pi\right) p\left(y_{i,j} \mid \mu_s, \mathbf{\Gamma}_s\right), \tag{14}$$

where each component is a Gaussian with $p\left(y_{i,j} \mid \mu_s, \mathbf{\Gamma}_s\right) = \mathcal{N}\left(y_{i,j}; \mu_s, \mathbf{\Gamma}_s\right)$ and $p\left(s = s' \mid \pi\right)$ is a multinomial distribution representing the probability of the observation $y_{i,j}$ be associated with component s'.

The prior over the parameters is given by

$$p\left(\theta \mid M\right) = p\left(\pi\right) \prod_{s} p\left(\mathbf{\Gamma}_s\right) p\left(\mu_s \mid \mathbf{\Gamma}_s\right), \tag{15}$$

where the weight prior is a symmetric Dirichlet $p\left(\pi\right) = \mathcal{D}\left(\pi; \lambda_0 \mathbf{I}\right)$, the prior over each covariance matrix is a Wishart $p\left(\mathbf{\Gamma}_s\right) = \mathcal{W}\left(\Gamma; \alpha_0, \mathbf{B}_0\right)$ and the prior over the means given the covariance matrices is a multivariate normal $p\left(\mu_s \mid \mathbf{\Gamma}_s\right) = \mathcal{N}\left(\mu_s; \mathbf{m}_0, \beta_0 \mathbf{\Gamma}_s\right)$. The joint likelihood of the data, assuming the samples are independent and identically distributed (i.i.d.), can be computed as

$$p\left(\mathbf{y_i}, \mathbf{s} \mid \theta, M\right) = \prod_{n=1}^{N} p\left(s_n = s \mid \pi\right) p\left(y_{i,n} \mid \mu_s, \mathbf{\Gamma}_s\right), \tag{16}$$

where $\mathbf{y}_i = \left\{y_{i,1}, y_{i,2}, \ldots, y_{i,n}\right\}$ and $\mathbf{s} = \{s_1, s_2, \ldots, s_S\}$.

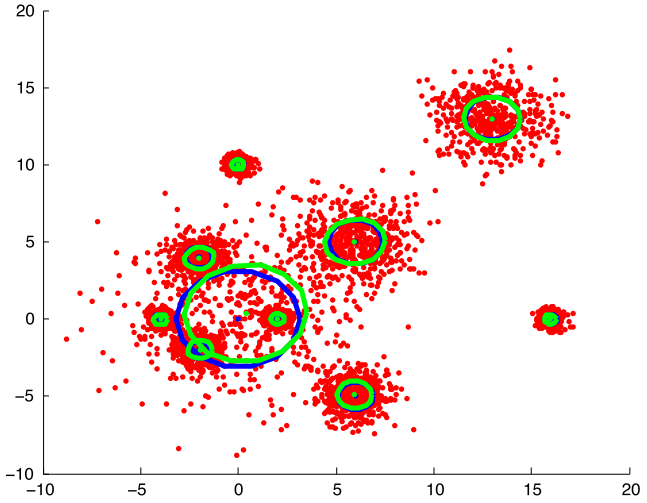

Fig. 5. VBEM results for a simulated dataset. The red dots are the sampled points from the mixture of Gaussians represented with blue ellipsoids (1 standard deviation). The estimated covariances from VBEM are represented by the green ellipsoids. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

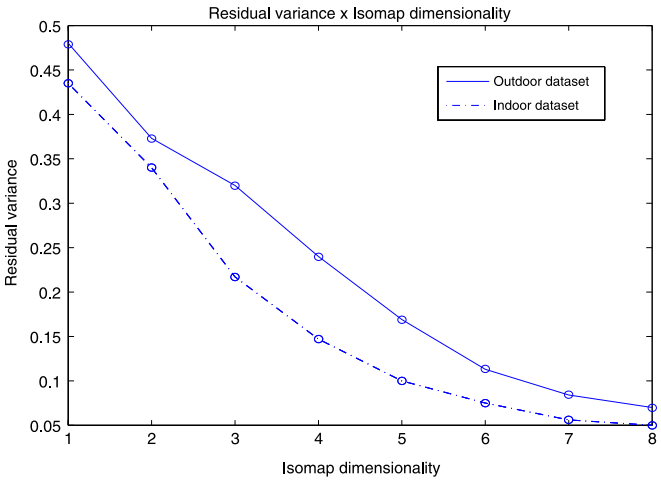

Fig. 6. Residual variance of Isomap as a function of the number of dimensions.

The variational approximation for the log marginal likelihood leads to the following free densities q.

- For the covariance matrices, $q\left(\mathbf{\Gamma}\right) = \prod_s q\left(\mathbf{\Gamma}_s\right)$ with $q\left(\mathbf{\Gamma}_s\right) = \mathcal{W}\left(\Gamma; \alpha_s, \mathbf{B}_s\right)$;
- For the means, $q\left(\mu \mid \mathbf{\Gamma}\right) = \prod_s q\left(\mu_s \mid \mathbf{\Gamma}_s\right)$ with $q\left(\mu_s \mid \mathbf{\Gamma}_s\right) = \mathcal{N}\left(\mathbf{x}; \mathbf{m}_0, \beta_s \mathbf{\Gamma}_s\right)$;
- For the mixing coefficients, $q\left(\pi\right) = \mathcal{D}\left(\pi; \lambda\right)$, where $\lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_s\}$;
- For the hidden variable $\mathbf{s}$, $q\left(\mathbf{s}\right) = \prod_s q\left(s_s\right)$.

Taking the functional derivatives of the free energy with respect to the free densities q produces the update rules of VBEM for the mixture of Gaussian cases. In the VB-E Step the weights of the hidden variables are calculated and in the VB-M Step parameters and hyper-parameters are updated. These rules are omitted here for brevity but can be found in [30].

Once parameters and the model were obtained, the predictive density for a particular patch $p\left(y' \mid \mathbf{y_i}, M\right)$ has a closed-form solution of a mixture of Student-t distributions,

$$p\left(y' \mid \mathbf{y_i}, M\right) = \sum_{s=1}^{S} \bar{\pi}_s \mathcal{S}\left(\mathbf{y}'; \rho_s, \mathbf{\Lambda}_s, \omega_s\right), \tag{17}$$

Fig. 7. Sample images of the indoor dataset.

with $\omega_s = \alpha_s + 1 - d$ degrees of freedom, where the means are $\rho_s = \mathbf{m}_s$ and the covariances are $\mathbf{\Lambda}_s = \left(\left(\beta_s + 1\right)/\beta_s\omega_s\right)\mathbf{B}_s$. The weights are computed based on the hyper-parameters of the Dirichlet distribution with $\bar{\pi}_s = \lambda_s / \sum_{s'} \lambda_{s'}$.

2.3.1. Heuristic for searching

VBEM allows direct model comparison by evaluating the free energy function of different models. In the case of mixtures of Gaussians this model can be a single component or a mixture with hundreds of components. Theoretically, there is no limit for the number of components and the search for the best model can be cumbersome. To cope with this problem, a heuristic based on birth and death of components is used. This heuristic appears to be appropriate for robotics problems since simpler models are evaluated before more complex ones, thus decreasing the computational complexity.

The birth–death heuristic used here has the same stopping and splitting criteria as in [27] for mixtures of factor analysers. The selection for splitting is based on the component with the smallest individual free energy. The search ends when all existing components were divided but none of those divisions result in free energy improvement.

As VBEM proceeds in estimating the parameters of a particular model and optimising the objective function, it is necessary to know when to stop and compare the free energy of this model with another one. This stopping criterion must be invariant to the dimensionality and amount of data. For this reason, the obvious manner of checking the free energy difference between two iterations is not adequate since it is hard to define a threshold for this quantity that scales appropriately with dimensionality, complexity and amount of data. Also, this method would require the computation of the free energy in every iteration, which can slow down the process as a whole. The alternative implementation examines the rate of change in the weight estimation given by $q\left(s_i\right)$. A measure of this quantity averaged over all data is given by the *agitation*:

$$\text{agitation}(s)^{(t)} \equiv \frac{\sum_{i=1}^{n} \left| q\left(s_i\right)^{(t)} - q\left(s_i\right)^{(t-1)} \right|}{\sum_{i=1}^{n} q\left(s_i\right)^{(t)}}, \tag{18}$$

where (t) denotes the iteration number. When the *agitation* is smaller than a threshold, the estimation is said to be mature enough and VBEM can stop, compute the free energy and compare it with other models. The same idea was used in [27] for a mixture of factor analysers. Note that by using *agitation*, the free energy needs to be calculated only once for each model – at the end, when the estimation is mature – saving computations since *agitation* is much simpler to calculate than the free energy.

In the implemented birth and death heuristic, the search starts evaluating a model with a single component. Once VBEM reaches maturity for this model and its free energy is computed, the component can be split into two and the parameters of this new model are evaluated. At this point, however, it is necessary to define a direction or a method to divide the points associated with the previous component into the other two. The procedure implemented in this work samples a direction $\mathbf{d}$ and defines an allocation indicator variable for each data point, $\mathbf{d} \sim \mathcal{N}\left(\mathbf{d}; \mu_s, \mathbf{\Sigma}_s\right)$ and

$$r_i = \begin{cases} 1 & \text{if } \left(\mathbf{y}_i - \mu_s\right)^{\mathrm{T}} \mathbf{d} \geq 0 \\ 0 & \text{if } \left(\mathbf{y}_i - \mu_s\right)^{\mathrm{T}} \mathbf{d} < 0 \end{cases} \quad \text{for } i = 1, \ldots, n. \tag{19}$$

From these two expressions, it is then possible to reassign points associated with the *parent* component $q(s_i)$ to the two children, introducing a hardness parameter α_h, ranging from 0.5 to 1:

$$q\left(s_i^a\right) = q(s_i)\left[\alpha_h r_i + (1-\alpha_h)(1-r_i)\right], \tag{20}$$

$$q\left(s_i^b\right) = q(s_i)\left[(1-\alpha_h) r_i + \alpha_h (1-r_i)\right]. \tag{21}$$

The hardness parameter defines how much weight is transferred to the assigned child. When $\alpha_h = 0.5$, the weight is shared equally. With these responsibilities, the parameters of the new model are updated in the VB-M Step which continues until maturity.

A selection criterion needs to be defined to choose which component to split. This is achieved by analysing the individual free energies of the components. The component with the smallest free energy is the preferable one for division since it is the worst in modelling the data correctly (in a free energy sense).

The heuristic continues, trying to split existing components and checking if there is improvement on the free energy. A tentative model is disregarded if there is death of a component or if there is no free energy improvement. This is easily verified by checking if any of the responsibilities goes below a certain threshold, meaning that a component has little or no importance for the data description. The process ends when all existing components were divided but none of those divisions result in free energy improvement.

2.3.2. Example

To measure the quality of VBEM and the searching heuristic, an artificial example is presented in this section. 5000 samples from a mixture of Gaussians with 10 components were sampled in two dimensions. This mixture contains components with covariances of different shapes including components "inside" one standard deviation of other components. The task of VBEM was to estimate the number of components, their means and covariances from the data supplied. The results can be seen in Fig. 5. VBEM correctly finds the correct number of components and estimates means and covariances accurately. Other tests for the same problem were also performed reducing the number of points sampled. VBEM starts missing some components when the number of samples is less than half of the initial set. In this situation, there is not enough points to represent the complexity of the distribution. Even in these cases, however, the estimated mixture model is reasonable and closed to the original distribution. Given the difficulty of the problem, VBEM seems to be robust in estimating complex distributions even when the number of samples is small.

2.4. Multi-class classification

As opposed to most classification problems where the input is a single feature-vector, in this approach the whole set of patches of an image is used. Each patch has equal contribution to the final classification decision and it is evaluated under the different models representing the places. The idea is to compute the log-likelihood of a set of image patches for every model learnt. The log-likelihood with the largest value is the final decision of the classification. Thus, the label of an image i is the label corresponding to the place model that maximises the expression:

$$M^* = \arg\max_M \sum_{j=1}^{m} \log p\left(y_{i,j} \mid M\right). \tag{22}$$

The computation for the log-likelihood in selecting the model that best explains the set of patches can be quickly computed. Also, it is possible to include more models, allowing sequential learning implementations. This is one of the demands for autonomous navigation as the robot visits new places, representations of them should be incorporated and correlated with the current knowledge.

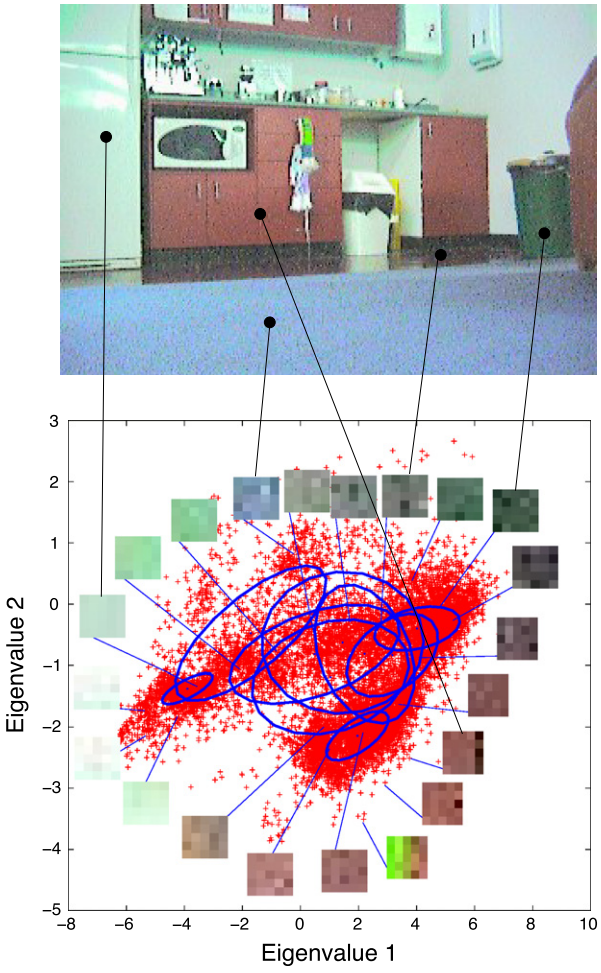

Fig. 8. Generative model learnt for a kitchen. Points are plotted on the direction of the two largest eigenvalues of the essential features. Ellipses correspond to the covariance matrices of the components learnt with VBEM. The association between the patches and their location in the real scene is also indicated.

3. Implementation

The framework was tested with a Pioneer 2-AT. 320 × 240 images were obtained with a 24-bit colour camera. Each image is then divided into 3072 non-overlapping patches with 5 × 5 pixels. In addition to colour, the patches are convolved with 4 Gabor wavelets to account for texture information. The resulting input space has a dimensionality of 175 ($5 \times 5 \times 3$(RGB) $+ 5 \times 5 \times$ 4(Gaborwavelets) $= 175$).

Learning is performed offline with labelled images from the above set. The training images were selected to give a multi-view perspective of the place. In the indoor experiment for example, if an office has a rectangular shape, 4 training images are taken close to the walls but the algorithm should still be able to recognise the place when observing it from the centre.

The essential features obtained with Isomap were estimated to have 8 dimensions. This number was obtained by computing the residual value of Isomap for different dimensions as depicted in Fig. 6.

The mixture of linear models was learnt with EM using 16 components for both environments. EM was initialised with k-means [15] for the low dimensional means and randomly for the other parameters. Convergence is verified when the increment in the log-likelihood was less than 0.01%. Inference is performed using Eqs. (5)–(7) resulting in mixtures of Gaussians

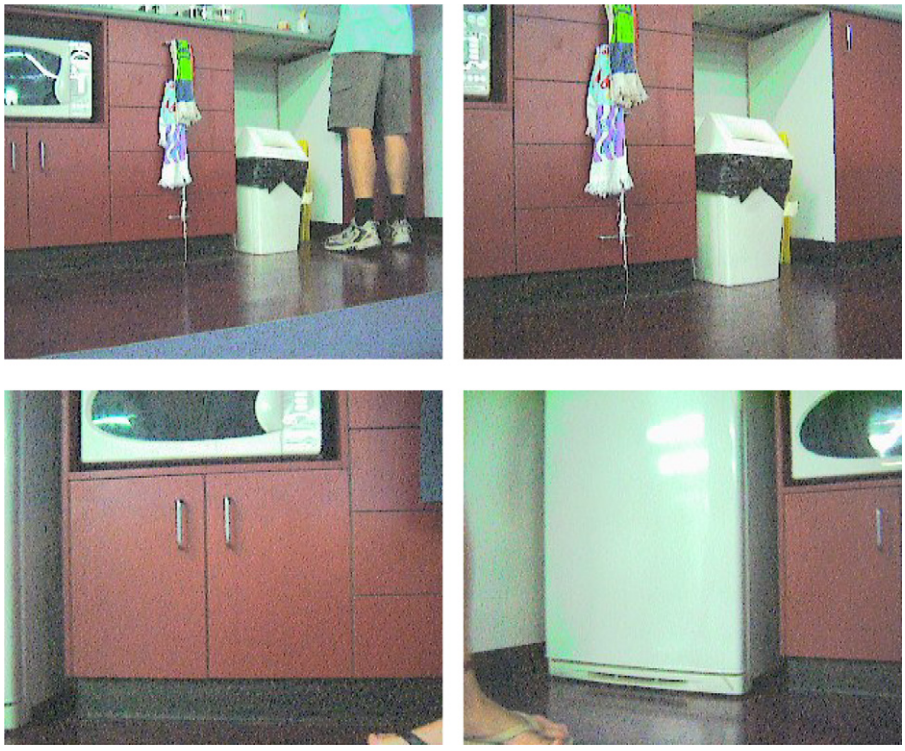

Fig. 9. The training images used for the kitchen model. The model learnt from them was able to recognise wider views of the place such as the image in Fig. 8.

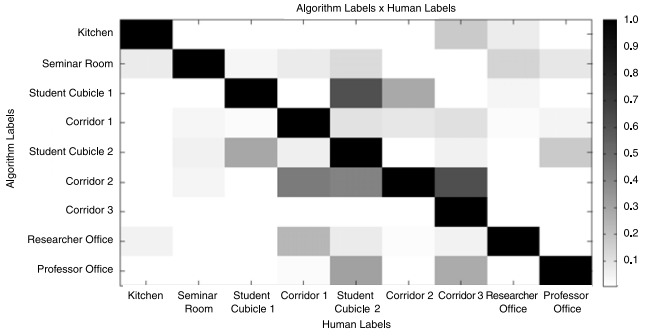

Fig. 10. Graphical representation of the confusion matrix for the indoor experiment. Darker cells represent larger values.

which are further collapsed to single components. This allows the computation of the log-likelihood for the different place models.

The variational Bayesian learning starts searching for the best model from a single-component and follows the heuristic of birth and death as described before. The covariance obtained from the mixture of linear models is used to initialise the parameters α_0 and $\mathbf{B}_0$ of the Wishart distribution. As shown in Eq. (6), the covariance is independent of particular observations, having the meaning of how uncertain the model is.

A diagram showing the steps and their computational complexities for learning and inference is depicted in Fig. 3. Note that learning is performed *offline* while the online classification procedure runs in (near) real-time.

When testing the algorithm, the whole set of 3072 patches representing an image is used. The process takes about 1 s per image in a Pentium M 1.7 GHz which comprises Gabor convolutions, inference in the mixture of linear models, and log-likelihood computation for each model learnt. Future implementations may use a subset of patches sampled from the original set to further reduce the classification time. This, however, may decrease the accuracy thus characterising an accuracy-time trade-off.

4. Experiments[1]

Two different experiments were performed to evaluate the algorithm in different conditions—indoor and outdoor environments. In both experiments, there were people walking by the places and sometimes occluding the robot's view. In the outdoor experiment, there were also cars and bicycles which add more complexity to the problem as the environment becomes dynamic.

4.1. Indoor dataset

The indoor dataset consists of 55 training images of 9 different places—each place has 5–9 training images only. The test set has 1579 sequence images obtained by the robot when navigating inside the lab. The cl are {kitchen, seminar room, student cubicle 1, corridor 1, student cubicle 2, corridor 2, corridor 3, research office and professor office}. Fig. 7 shows some images of the indoor places. The generative model for the kitchen is depicted in Fig. 8. It shows the covariance matrices learnt through VBEM from the essential features. The correlation between the patches and their real position is also indicated.

To assess the performance of the algorithm empirically, information retrieval metrics were used. This allowed us to evaluate correctness of the place recognition in a model-independent manner. The two measures used are precision and recall.

- *Precision*: this corresponds to the ratio of correct labels (true positives) found to the total number of images classified as a particular class (true positives plus false positives). Intuitively, it is a measure of how many of the identified places are actually correct (exactness).

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}. \tag{23}$$

[1] Videos with the experiments and the datasets are available at: http://www.acfr.usyd.edu.au/people/postgrads/ftozeto.

Fig. 11. Sample images of the outdoor dataset.

- *Recall*: this corresponds to the ratio of correct labels (true positives) found to the total number of images in the particular class. Intuitively, it is a measure of how many of the correct places were found (completeness);

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}. \tag{24}$$

Table 1 shows precision and recall results for this 9-class problem and Fig. 10 shows a graphical representation of the confusion matrix. In general, classification is accurate as can be seen by the strong diagonal in the confusion matrix. Considering that in many occasions the robot moves less than one metre away from walls and furniture, significantly reducing the angle of view, these results are quite promising. In those situations even manual classification is difficult. Also, some places are very similar, for example, the corridors and the student cubicles. These places can be distinguished from a few objects such as paintings in the case of corridors, and books or computers on the desks of student cubicles. However, both objects might not be observed by the robot since they are in a higher position. Fig. 9shows the training images for the kitchen model. Even without a training image of the kitchen with a wider view, the classifier was able to recognise the kitchen from the image of Fig. 8. This generalisation is one of the main properties of the algorithm which is achieved through a compact representation obtained with dimensionality reduction combined with a generative Bayesian model for places. The dimensionality reduction step extracts the important information from the image while the Bayesian model integrates several images of the same place to produce a generalised model.

Table 1
Precision and recall results for the indoor dataset.

Place name	Precision	Recall
Kitchen	81.82	76.60
Seminar room	68.07	81.82
Student cubicle 1	52.86	48.05
Corridor 1	74.96	84.09
Student cubicle 2	62.96	46.12
Corridor 2	40.00	62.50
Corridor 3	100	15.13
Researcher office	73.01	90.16
Professor office	62.96	71.83
Average	68.52	64.03

Table 2
Precision and recall results for the outdoor dataset.

Place name	Precision	Recall
ACFR front	71.04	87.44
ACFR park	83.19	71.05
Eng. Building	65.40	77.31
Eng. Road	91.43	45.43
Eng. Carpark	26.17	87.10
Mech. Building	64.47	17.63
Mech. Corridor	90.68	65.54
ACFR carpark	55.24	76.40
ACFR road	87.00	74.34
Garage	23.94	87.18
Office	99.47	61.84
Average	68.92	68.30

4.2. Outdoor dataset

The outdoor dataset consists of 57 training images of 11 different places at University of Sydney, with each place having 3–8 training images. The test set has 3820 images obtained from a half-kilometre journey around the university. The classes are {ACFR front, ACFR park, Eng. Building, Eng. Road, Eng. Carpark, Mech. Building, ACFR carpark, ACFR road, garage (indoor) and office (indoor)}. Fig. 11 shows pictures of those places. The generative model for the class ACFR park is shown in Fig. 12. Also annotated is the correlation between patches and their location in the manifold.

Table 2 presents precision and recall results and Fig. 13 the confusion matrix. In general, the results are better than the indoor dataset. The most difficult problem of the outdoor dataset was to distinguish between the two carparks. When the robot was very close to a car, it was not able to have a more general view of the place which resulted in classifying the image as the other carpark. Also, "Mech. Building" and "Eng. Building" are physically in the same building and the limits of where one starts and the other finish are not very clear.

5. Conclusions

This paper introduced a new algorithm for place recognition that can be learnt from very small datasets with significant generalisation properties. Specifically, the proposed framework has three main contributions: it shows that mixture of linear models can be used as a tool for nonlinear regression problems with noise such as in Isomap mappings; it demonstrates how variational Bayesian learning with a free-energy heuristic can choose the right number of components of a mixture of Gaussians; it shows how the log-likelihood can be applied to multi-class problems where classification is given from a set of samples rather than from a single point.

If compared to other algorithms such as [31], the proposed approach can be trained incrementally and with much less data while achieving similar performance as the hidden Markov model (HMM) employed. Furthermore, the dataset used here was obtained with a mobile robot with a camera fixed in a lower position, that is much more sensitive to irregularities in the terrain, occlusions and blurring than the camera mounted in a human helmet used by them.

Patches of images and images themselves are treated here as independent and identically distributed. In the case of patches, further implementations can include the positions of each patch as additional dimensions in the feature-vector. Also, spatial relations among them can be included in more sophisticated relational statistical models. This, however, should preserve the main benefits of the model such as learning from few images and efficient classification.

Most of the false classifications took place when the robot was close to a wall or an object occluding a wider view of the scene.

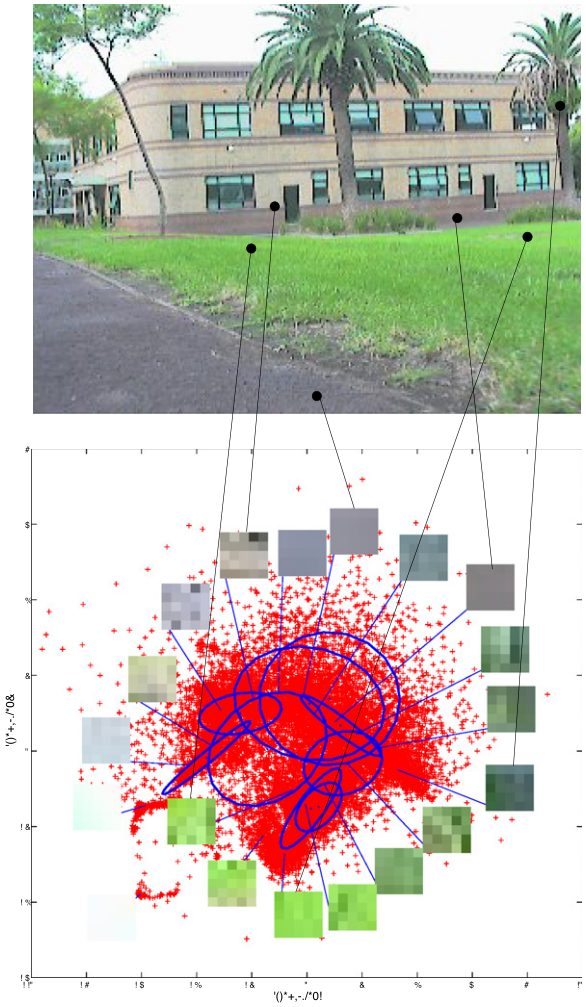

Fig. 12. The same as Fig. 8 but for the "ACFR-park" model.

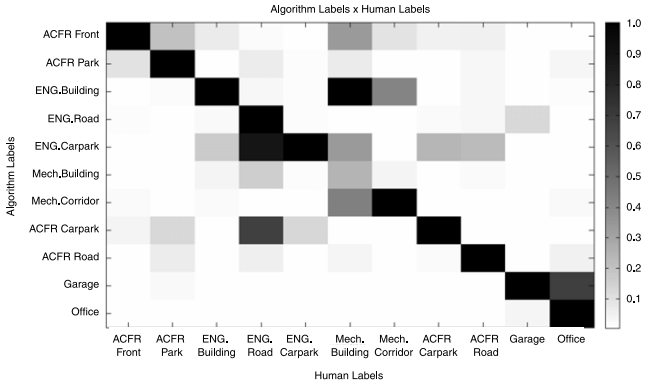

Fig. 13. Graphical representation of the confusion matrix for the outdoor experiment. Darker cells represent larger values.

This problem could be avoided if a topological map of the environment were encoded in a HMM to constrain the search to fewer places. In future works, algorithms for learning HMMs incrementally will be investigated as well as how to integrate them in the existing framework.

Acknowledgements

This work is supported by the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales (NSW) State Government.

References

[1] S. Thrun, J. Leonard, Simultaneous localisation and mapping, in: Springer Handbook of Robotics, 2008, pp. 871–889.
[2] H. Strasdat, C. Stachniss, W. Burgard, Which landmark is useful? learning selection policies for navigation in unknown environments, in: IEEE International Conference on Robotics and Automation, ICRA, 2009.
[3] B.J. Stankiewicz, G.E. Legge, J.S. Mansfield, E.J. Schlicht, Lost in virtual space: Studies in human and ideal spatial navigation, Journal of Experimental Psychology: Human Perception and Performance 32 (3) (2006) 688–704.
[4] B. Garsoffky, S. Schwan, M. Huff, Canonical views of dynamic scenes, Journal of Experimental Psychology: Human Perception and Performance 35 (1) (2009) 17–27.
[5] K. Daniilidis, J. Eklundh, 3-D vision and recognition, in: Springer Handbook of Robotics, 2008. pp. 543–562.
[6] I. Ulrich, I. Nourbakhsh, IEEE International Conference on Robotics and Automation, ICRA, in: Appearance-based place recognition for topological localization, vol. 2, San Francisco, USA, 2000, pp. 1023–1029.
[7] T. Mitchell, F. Labrosse, Visual homing: a purely appearance-based approach, in: Proc. of Towards Autonomous Robotic Systems, Colchester, UK, 2004.
[8] J. Kosecka, F. Li, Vision based topological Markov localization, in: IEEE International Conference on Robotics and Automation, ICRA, 2004.
[9] J. Wolf, W. Burgard, H. Burkhardt, Robust vision-based localization for mobile robots using an image retrieval system based on invariant features, in: IEEE International Conference on Robotics and Automation, ICRA, Washington, USA, 2002.
[10] S. Se, D. Lowe, J. Little, Global localization using distintive visual features, in: Proc. of International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, 2002, pp. 226–231.
[11] R. Sim, G. Dudek, Learning environmental features for pose estimation, Image and Vision Computing 19 (11) (2001) 733–739.
[12] A. Davison, D. Murray, Simultaneous localisation and map-building using active vision, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2002) 865–880.
[13] M. Cummins, P. Newman, Probabilistic appearance based navigation and loop closing, in: International Conference on Robotics and Automation, 2007.
[14] M. Milford, G. Wyeth, Mapping a suburb with a single camera using a biologically inspired slam system, IEEE Transactions on Robotics 24 (5) (2008).
[15] D.J.C. Mackay, Information Theory, Inference, and Learning Algorithms, Cambridge University Press, Cambridge, UK, 2003.
[16] T. Serre, L. Wolf, T. Poggio, Object recognition with features inspired by visual cortex, in: 2005 IEEE International Conference on Computer Vision and Pattern Recognition, 2005.
[17] K. Jarrett, K. Kavukcuoglu, M.A. Ranzato, Y. LeCun, What is the best multi-stage architecture for object recognition? in: 2009 IEEE International Conference on Computer Vision, 2009.
[18] J. Tenenbaum, V. de Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality redution, Science 290 (2000) 2319–2323.
[19] R. Duda, P. Hart, D. Stork, Pattern Classification, second ed., Wiley-Interscience, New York, 2001.
[20] T. Cox, M. Cox, Multidimensional Scaling, Chapman and Hall, 1994.
[21] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society. Series B 39 (1977) 1–37.
[22] Z. Ghahramani, G.E. Hinton, The EM algorithm for mixtures of factor analyzers, Technical Report, Department of Computer Science, University of Toronto CRG-TR-96-1, 1996.
[23] L.K. Saul, S.T. Roweis, Think globally, fit locally: Unsupervised learning of low dimensional manifolds, Journal of Machine Learning Research 4 (2003) 119–155.
[24] N.D. Lawrence, Gaussian process latent variable models for visualisation of high dimensional data, in: Advances in Neural Information Processing Systems 17, NIPS, MIT Press, Vancouver, Canada, 2004.
[25] M.I. Jordan, Z. Ghahramani, T. Jaakkola, L.K. Saul, An introduction to variational methods for graphical models, Machine Learning 37 (2) (1999) 183–233.
[26] T.P. Minka, Using lower bounds to approximate integrals. Informal notes available at: http://www.stat.cmu.edu/minka/papers/learning.html, 2001.
[27] M.J. Beal, Variational algorithms for approximate Bayesian inference, Ph.D. Thesis, The Gatsby Computational Neuroscience Unit, University College London, May 2003.
[28] T.M. Cover, J.A. Thomas, Elements of Information Theory, John Wiley & Sons, Inc., New York, 1991.
[29] R.J. Muirhead, Aspects of Multivariate Statistical Theory, Wiley, New York, USA, 1982.
[30] H. Attias, A variational Bayesian framework for graphical models, in: Proceedings of Neural Information Processing Systems 12, MIT Press, Cambridge, MA, USA, 2000.
[31] A. Torralba, K. Murphy, W. Freeman, M. Rubin, Context-based vision system for place and object recognition, in: Proceedings of the International Conference of Computer Vision, ICCV, Nice, France, 2003.

Fabio Ramos is a Senior Lecturer at the School of Information Technologies, University of Sydney. He received the B.Sc. and the M.Sc. degrees in Mechatronics Engineering at University of Sao Paulo, Brazil, in 2001 and 2003, respectively, and the Ph.D. degree at University of Sydney, Australia, in 2007. Since 2007 he has been a senior research fellow at the Australian Centre for Field Robotics. He has received the Best Paper Award in the International Conference on Intelligent Robots and Systems (IROS) in 2005, and the Australian Research Council (ARC) Post Doctoral Fellowship in 2008. His research focuses on statistical learning techniques for dimensionality reduction, stochastic processes modelling and object recognition with applications in robotics and mining.

Ben Upcroft is a Senior Lecturer in Mechatronics at the University of Queensland. He completed his undergraduate degree in Science with Honours and continued a Ph.D. in ultracold atomic physics in 2003. Throughout his Ph.D., Ben had a keen interest in robotics, which led him to a Postdoctoral position (2003) at the ARC Centre of Excellence for Autonomous Systems, School of Aerospace, Mechatronics, and Mechanical Engineering, University of Sydney. He has run major industrial projects involving autonomous aircraft, offroad and urban vehicles, and network communications. Ben currently focuses on computer vision aided localisation and navigation for autonomous ground vehicles.

Suresh Kumar received a B.Tech in Engineering from the Indian Institute of Technology, Chennai (1992), M.S. (1994) and Ph.D. (1997) in Computational Engineering Mechanics from the State University of New York, Buffalo, respectively. His research interests include finite and boundary element methods for solution of partial differential equations, inverse problems (acoustics, EEG, electro-cardiography) and machine learning methods for data representation and interpretation.

Hugh Durrant-Whyte received the B.Sc. in Nuclear Engineering from the University of London, UK, in 1983, and the M.S.E. and Ph.D. degrees, both in Systems Engineering, from the University of Pennsylvania, USA, in 1985 and 1986, respectively. From 1987 to 1995, he was a University Lecturer in Engineering Science, the University of Oxford, UK and a Fellow of Oriel College Oxford. Since 1995 he has been a Professor of Mechatronic Engineering at University of Sydney. He has been awarded two Australian Research Council (ARC) Federation Fellowships, in 2002 and 2007. His research work focuses on robotics and sensor networks. He has published over 350 research papers and has won numerous awards and prizes for his work.