Combining Object Recognition and SLAM for Extended Map Representations

Fabio Ramos, Juan Nieto, and Hugh Durrant-Whyte

ARC Centre of Excellence for Autonomous Systems
Australian Centre for Field Robotics, The University of Sydney
Sydney, NSW 2006, Australia
{f.ramos,j.nieto,hugh}@acfr.usyd.edu.au

1 Introduction

Building a map while navigating in an unknown environment is a major problem in robotics. The robot has to incrementally build a map of the environment, while concurrently using this map to localise itself. As the number of landmarks increases the problem becomes more complex and expensive to compute - the complexity is quadratic in the number of landmarks. Various approaches have tackled the complexity problem [11, 4, 15, 21, 3], however two challenging issues remain in SLAM: reliable data association and operation in dynamic environments.

Successful data association involves association of the correct measurement with the respective underlying state, initialising new tracks and detecting and rejecting spurious measurements. As the robot moves, the uncertainty of its pose and landmark estimation increases until a known landmark is re-observed. Depending on the sensor noise, the landmark representation (usually point features) and the distance travelled, the uncertainty in position can be large enough to cause failure in the data associations. Due to this reason, data association algorithms based entirely on position estimates tend to fail in long-term trips. This imposes a serious problem for loop-closing and consequently robust SLAM applications. In addition, the real world is dynamic. Objects and people may move which can cause static map representations to fail if moving objects are errouneously used as landmarks.

When representing the world, not only feature maps are important for robotics tasks. Individual representations of objects in the map could also be interesting in problems that go beyond navigation tasks such as finding victims of earthquakes inside buildings or bushfire fighting. With extended representations that model appearance, it is possible to send commands such as "find a house that looks like this" or "make a map of objects that are similar to this".

In this paper, object recognition and segmentation techniques are used in conjunction with EKF-SLAM [6, 20] to create extended representations of the environment. The proposed algorithm provides a solution to data association problems that combines visual with position information. As individual representations of objects are created and updated with new observations, the algorithm can be applied to dynamic environments while providing not only a map but also appearance models of the objects in the map.

The combination of visual and position estimates of features to improve data association have been previously addressed in [16]. In their work visually salient features are extracted from images and used in combination with laser scans for loop-closing. The use of feature representations significantly improves data association but does not provide meaningful representations of the objects in the scene. Also, because salient features are based on particular configurations (positions) of objects, this approach has problems when moving objects are present in the environment.

A new trend is to use stereo vision and feature extraction algorithms such as SIFT [13] to build maps of the environment [19, 9]. Those approaches, however, rely on the assumption that SIFT features can always be extracted and matched, and that there is an accurate observation model for the stereo camera range. Also, in dynamic environments, SIFT is not reliable since it may extract invariant features not directly associated to objects known to be static.

The algorithm presented in this paper combines 2D laser scan with camera to help object segmentation and position estimation. This fusion process is possible by calculating the extrinsic parameters of the laser with respect to the camera. Landmarks are thus recognised by their appearance and position. Once an landmark has been identified, laser measurements are used to estimate the robot and the object (landmark) positions. In addition, a representation of the object is created and updated with new observations.

2 Algorithm Overview

The algorithm uses two common sensors, a 2-D laser and a camera, which are present in a variety of robot platforms currently available. It is divided into two parts: in the offline phase a general representation of the object to be mapped is created. In the online phase, instances of this object are segmented using both laser and camera. Once an object is found, a specific generative model of its appearance is created and stored for further data association. Also, its position is included in the state vector of SLAM. Figure 1 shows a schematic representation of the proposed algorithm.

Before building the map, images of the objects of interest should be obtained for the offline phase. In the case of general outdoor objects such as trees or cars, these images can be obtained from a search in the Internet or previous missions. They should contain different instances of the object in multiple

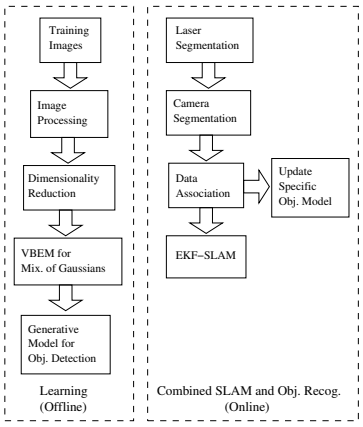

Figure 1. Diagram with offline and online phases of the proposed algorithm.

viewpoints in order to provide a reasonable generalisation of the object-class appearance.

3 Learning Phase

Learning is performed to provide visual models of objects (landmarks). These models are used to segment objects out of images for further data association. Whenever an object is re-observed, its data association model is updated to incorporate the new information, which can be the object appearance at a different vantage point. The output of the learning phase is a generative model for object segmentation from image patches. The steps for the creation of the generative model are detailed as follows.

3.1 Image Processing and Dimensionality Reduction

Image processing is the first operation in the offline part of the algorithm and is also performed in the camera segmentation of the online part. It comprises the division of the image into patches of the same size, Gabor convolution at multiple scales and orientations (in order to obtain texture information) and dimensionality reduction.

More precisely, each image $\mathbf{I}$ is divided into a grid of square patches of equal size $\mathbf{I} = \{I_1, \ldots, I_n\}$ where I_i is the i-th patch. For example, an image with resolution 640×480 would result in a set of $640/9 \times 480/9 = 3763$ square patches of 9×9 pixels. Each patch is then convolved with a set of Gabor wavelets (in the experiments two orientations and two scales were used). After convolution, each patch can be seen as a point in a $L \times L \times (3 + G)$-dimensional space, where L is the size of the patch, 3-colour pixels, and G

Gabor convolutions. This new set, in the high-dimensional space, is then projected down to a low-dimensional space using principal component analysis (PCA)[1] [7].

During the offline part, all images are used to compute the eigenvectors of PCA. These eigenvectors are then stored and used in the online part to project image patches into a lower d-dimensional space. Points in the d-dimensional space constitute the feature-vector $\mathbf{z}_A^{\mathbf{I}} = \left\{z_{A,1}^{\mathbf{I}}, z_{A,2}^{\mathbf{I}}, \ldots, z_{A,n}^{\mathbf{I}}\right\}$ of image $\mathbf{I}$. They represent appearance observations of patches in the same image (the subscript A is used to indicate appearance).

The last operation is the inclusion of neighbourhood information in the feature-vector which significantly improves segmentation performance. This is performed by adding the norm of the neighbours appearance (patches on the left, right, above and below) in the feature-vector:

$$z_i^{\mathbf{I}} = \left[z_{A,i}^{\mathbf{I}}, \left|z_{A,i-up}^{\mathbf{I}}\right|, \left|z_{A,i-down}^{\mathbf{I}}\right|, \left|z_{A,i-left}^{\mathbf{I}}\right|, \left|z_{A,i-right}^{\mathbf{I}}\right|\right]^T,$$

with $|.|$ denoting the norm of a vector. The final image representation, used for training and further segmentation, is then the set $\mathbf{z}^{\mathbf{I}} = \left\{z_1^{\mathbf{I}}, z_2^{\mathbf{I}}, \ldots, z_n^{\mathbf{I}}\right\}$.

3.2 VBEM for GMMs

The Variational Bayesian Expectation Maximisation (VBEM) is applied to learn a generative model in the d-dimensional space. VBEM was initially proposed in [1]. It provides an automatic manner to discover the best model (structure) and parameters of probabilistic models. Also, it has been applied in robotics [18] and computer vision [8] with great success. As a direct Bayesian approximation it has two main advantages over non-Bayesian approaches for structure and parameters learning [17]:

1. Structures are compared over the whole set of possible parameters values which produce a better scoring function than Minimum Description Length (MDL), Bayesian Information Criterion (BIC) or Akaike Information Criterion (AIC) [12];
2. Prior information can be used to reduce the learning process and the number of training data points.

VBEM is used here to automatically discover the number of components in a Gaussian mixture model (GMM). VBEM directly penalises complexity (Occam's Razor property [14]) which results in more accurate generative models. Assuming a particular GMM, 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 structure, the likelihood of an observation $z_n^{\mathbf{I}}$ in a d-dimensional space is

[1] The current implementation uses PCA to meet speed requirements. Further implementations with Isomap will be investigated in future work.

$$p\left(z_n^{\mathbf{I}} \mid \theta, m\right) = \sum_{s=1}^{S} p\left(s_n = s \mid \pi\right) p\left(z_n^{\mathbf{I}} \mid \mu_s, \mathbf{\Gamma}_s\right), \tag{1}$$

where each component is a Gaussian with $p\left(z_n^{\mathbf{I}} \mid \mu_s, \mathbf{\Gamma}_s\right) = \mathcal{N}\left(z_n^{\mathbf{I}}; \mu_s, \mathbf{\Gamma}_s\right)$ and where $p\left(s_n = s \mid \pi\right)$ is a multinomial distribution representing the probability of the observation $z_n^{\mathbf{I}}$ being associated with component s.

The prior over the parameters is given by

$$p\left(\theta \mid m\right) = p\left(\pi\right) \prod_{s} p\left(\mathbf{\Gamma}_s\right) p\left(\mu_s \mid \mathbf{\Gamma}_s\right) \tag{2}$$

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 (IID), can be computed as

$$p\left(\mathbf{z}^{\mathbf{I}}, \mathbf{S} \mid \theta, m\right) = \prod_{n=1}^{N} p\left(s_n = s \mid \pi\right) p\left(z_n^{\mathbf{I}} \mid \mu_s, \mathbf{\Gamma}_s\right) \tag{3}$$

where $\mathbf{S} = \{s_1, s_2, \ldots, s_S\}$ are the indexes of the components in the mixture.

VBEM works in two steps both of them maximising the objective function, the log-marginal likelihood of the data,

$$\ln p\left(\mathbf{z}^{\mathbf{I}} \mid m\right) = \ln \int \sum p\left(\mathbf{z}^{\mathbf{I}}, \mathbf{S} \mid \theta, m\right) p\left(\theta \mid m\right) d\theta, \tag{4}$$

which is computed for each mixture, each one having a different number of components. The algorithm then selects the mixture with the highest log-marginal likelihood. The search for the number of components is implemented in a birth-death heuristic similar to the one used in [2] for mixtures of factor analysers.

The predictive density for a new observation given the learnt structure $p(z' \mid \mathbf{z}^{\mathbf{I}})$ is a Student-t distribution which approximates a Gaussian as the number of training points N increases.

3.3 Patch Classification

Using the procedure just described, two generative models are learnt. One for patches belonging to the object and another for the background or general description of the environment. The learning phase is supervised in the sense that patches corresponding to an object must be manually selected and labelled in the training images. Assuming a Gaussian approximation of the Student-t predictive densities computed by VBEM, the two models are $p\left(z' \mid \mathbf{z}^{obj}\right)$ and $p\left(z' \mid \mathbf{z}^{noObj}\right)$, where $p\left(z' \mid \mathbf{z}^{obj}\right) = \pi_s^{obj} \mathcal{N}\left(z'; \mu_s^{obj}, \mathbf{\Gamma}_s^{obj}\right)$ and $p\left(z' \mid \mathbf{z}^{noObj}\right) = \pi_s^{noObj} \mathcal{N}\left(z'; \mu_s^{noObj}, \mathbf{\Gamma}_s^{noObj}\right)$. A patch z' is then classified as part of an *object* if

$$R = \frac{\omega_{obj} \times p\left(z' \mid \mathbf{z}^{obj}\right)}{\omega_{obj} \times p\left(z' \mid \mathbf{z}^{obj}\right) + \omega_{noObj} \times p\left(z' \mid \mathbf{z}^{noObj}\right)}$$

is larger than a threshold, usually 0.5. ω_{obj} and ω_{noObj} are the proportions of the training data to each model.

This classification method is able to learn non-linear boundaries for classification which, associated with the Occam's Razor property of VBEM, provides a powerful methodology for classification with very good generalisation performance [12].

4 Combined SLAM and Object Recognition

4.1 Laser-Camera Calibration

The first step in combining laser-camera is calibration; computation of extrinsic parameters (rotation and translation) of a point in the camera coordinate system w.r.t a point in the laser coordinate system.

Given a calibrated camera with intrinsic parameters K, the corresponding pixel coordinates $p = [u, v]^T$ of points in the world coordinate system $P = [X, Y, Z]^T$ can be computed as $p \sim K(RP + t)$ where R is a 3×3 rotation matrix and t is a 3-vector representing translation. Assuming that P_l is a point in the laser coordinate system and P_c in the camera, the equation $P_l = \Phi P_c + \Delta$ represents their transformation, where Φ and Δ are the rotation and translation parameters for the laser-camera calibration. To compute Φ and Δ, the method described in [22] was used. The outputs are the optimised parameters Φ and Δ that in conjunction with the intrinsic parameters K of the camera, allow the projection of laser points in image pixels.

4.2 Laser-Camera Object Segmentation

The segmentation algorithm works in two steps. In the first, the laser scan is processed to find discontinuous clusters of points that can represent objects. This step significantly reduces the area of the image where the camera segmentation algorithm must search for the object. Clusters are identified by looking for group of points whose distances to any other points are larger than a threshold. A more detailed description of the laser-clustering algorithm is omitted for brevity.

Once clusters are obtained, the image patches associated with the laser points are extracted along with all image patches in the same columns. The final step is the classification of each of those patches which is performed using the method described in Section 3.

Figure 2 shows an example of the segmentation algorithm. Depicted are the laser scan, the laser points projected into the image, and the resulting segmentation of the object - in this case, the trunk of a tree. Although the segmentation is not perfect, there are few false positives (patches assigned to tree that are not, in reality, part of the tree). Even in this example, where the wall in the background has a similar colour to the trunk, and the black

Figure 2. Laser scan and field of view of the camera (top), laser points projected into the image (middle) and final segmentation of the trunk (bottom).

post on the left of the tree has similar shape, the segmentation of the trunk is good. This example demonstrates the advantages of combining laser and camera for segmentation.

4.3 Data Association

Whenever an object is segmented, the patches representing the object are stored (feature-vectors in a low dimensional space). These feature-vectors are then augmented with the global positions of the patches from Equation **??** and a probability density for them learnt through VBEM[2]. The objects are thus represented by probability densities that have both appearance and position information.

To associate objects, the Kullback-Leibler divergence[5] is used. Let $\mathbf{p}_L = \{p_1, p_2, \ldots, p_n\}$ be the set of probability distributions for n objects already mapped. A new observation $\mathbf{z}_{new}$ is associated with an object i if the $KL(p(x \mid \mathbf{z}_{new}) \parallel p_i(x))$ is smaller than a threshold. The KL divergence returns a measurement of the distance between two distributions that, in this case, are GMMs. Efficient algorithms to compute KL divergence for GMMs can be found in [10].

If the new observation is not associated with any of the object models, a new object model is created and included in $\mathbf{p}_L$.

4.4 Object Model Updating

When an observation is associated with a learnt object model, it can be used to improve this model with new information. This operation involves updating the object model sufficient statistics (SS) with the new observation. As the models are GMMs, their SS for each mixture component π_s, μ_s and $\mathbf{\Gamma}_s$ can be updated as follows

$$\pi_s \leftarrow \frac{M\pi_s^{new} + N\pi_s}{M+N},\ \mu_s \leftarrow \frac{M\mu_s^{new} + N\mu_s}{M+N},\ \mathbf{\Gamma}_s \leftarrow \frac{M\mathbf{\Gamma}_s^{new} + N\mathbf{\Gamma}_s}{M+N}, \quad (5)$$

[2] Priors from the general object detection model are used to improve convergence. Also, convergence can be accelerated by using a fixed number of components.

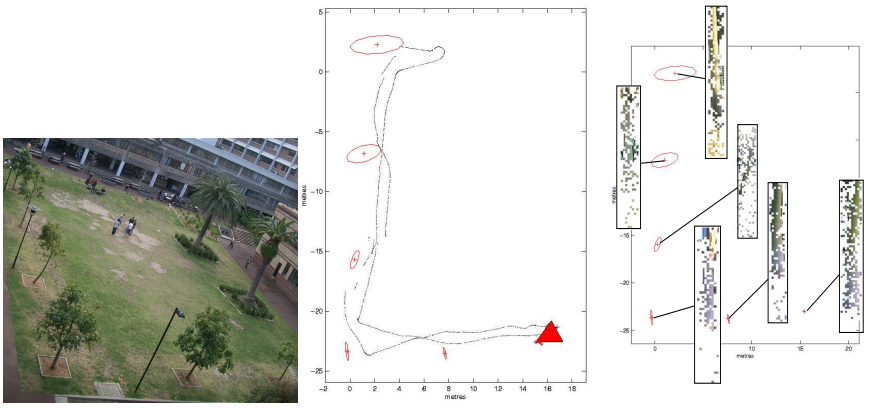

Figure 3. The explored area (left), map obtained with the algorithm (centre) and tree models learnt during SLAM(right). Note that the posts were not included in the map. The two palms on the right were not used to make data association more difficult as they are far from posts or other similar objects.

where π_s^{new}, μ_s^{new} and $\mathbf{\Gamma}_s^{new}$ are the SS of the new observation and N and M are the number of patches in the model and in the new observation respectively.

5 Experiments

The algorithm was tested outdoors to recognise and map trees in conjuction with EKF-SLAM. The platform was a Pioneer 2AT with Sick laser and camera calibrated according to the method described in Section 4. The area explored is relatively small (around 600 square metres) but with special attributes that make data association difficult for conventional algorithms. In this area (Figure 3 top) there are posts close to trees that have similar shapes to tree trunks. Since the task is to map trees, a laser-based data association algorithm would probably fail. Furthermore, during the data acquisition for SLAM, there were some people moving in the mapped area that would be erroneously included in the map by conventional techniques.

The algorithm presented overcomes these problems as can be seen from Figure 3. The map obtained has only trees (without the posts and the palm trees) and was not affected by the extraneous people moving in the area during the data acquisition. The trees were posed at the correct locations and no wrong associations have occurred.

Figure 3 (right) shows the location of the trees with their appearance models learnt. The images of the trees were obtained from the projection of the segmented patches back to the high dimensional space, using the eigenvec-

tors given by PCA. Although the segmentation is not perfect, the number of patches segmented were enough for accurate density estimation.

6 Conclusions

This paper has investigated three major problems of autonomous exploration: SLAM in dynamic environments, data association combining both laser with camera information, and appearance modelling of landmarks (objects). The algorithm demonstrates its robustness to deal with difficult data association problems while providing appearance models of the objects mapped. The combined laser-camera segmentation algorithm has the advantages of each of the sensors resulting in better object segmentation.

There are a number of improvements that can be made in future work. In the object segmentation method, spatial statistics techniques can be used to take into account spatial correlation, if feasible for real-time implementations. Markov random fields, conditional random fields and Gaussian process (also called kriging) are among the techniques to be investigated. Non-linear dimensionality reduction techniques such as Isomap or LLE can be used to improve landmark representation. More extensive experiments have also to be performed to test the algorithm in larger areas. Another possibility is the mapping of different objects and their context relation.

References

1. H. Attias. Inferring parameters and structure of latent variable models by variational bayes. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 21–30, San Francisco, USA, 1999. Morgan Kaufmann.
2. M. J. Beal. *Variational Algorithms for Approximate Bayesian inference*. PhD thesis, The Gatsby Computational Neuroscience Unit, University College London, May 2003.
3. M. Bosse, P. Newman, J. Leonard, and S. Teller. Simultaneous localization and map building in large-scale cyclic environments using the Atlas framework. *International Journal of Robotics Research*, 23(12):1113–1139, 2004.
4. J. A. Castellanos, M. Devy, and J. D. Tardos. Simultaneous localisation and map building for mobile robots: A landmark-based approach. In *Proceedings of IEEE international Conference on Robotics and Automation: Workshop on Mobile robot Navigation and Mapping*, San Francisco, USA, 2000.
5. T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, Inc, New York, 1991.
6. G. Dissanayake, P. Newman, S. Clark, and H. F. Durrant-Whyte. A solution to the simultaneous localisation and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation*, 17(3):229–241, 2001.
7. R. Duda, P. Hart, and D. Stork. *Pattern Classification*. Wiley-Interscience, New York, second edition, 2001.

8. L. Fei Fei, R. Fergus, and P. Perona. A Bayesian approach to unsupervised one-shot learning of object categories. In *Proceedings of the International Conference on Computer Vision*, 2003.
9. M. A. Garcia and A. Solanas. 3D simultaneous localization and modeling from stereo vision. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, New Orleans, USA, 2004.
10. J. Goldberger, S. Gordon, and H. Greenspan. An efficient image similarity measure based on approximations of kl-divergence between two gaussian mixtures. In *Proceedings of the International Conference on Computer Vision*, 2003.
11. J. Guivant and E. Nebot. Optimisation of the simultaneous localisation and map building algorithm for real time implementation. *IEEE Transactions on Robotics and Automation*, 17(3):242–257, 2001.
12. T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer-Verlag, New York, USA, 2001.
13. D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
14. D. J. C. Mackay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, Cambridge, UK, 2003.
15. M. Montemerlo and S. Thrun. Simultaneous localization and mapping with unknown data association using FastSLAM. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Taipei, Taiwan, 2003.
16. P. Newman and K. Ho. SLAM - loop closing with visually salient features. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, 2005.
17. D. Pelleg and A. Moore. X-means: Extending K-means with efficient estimation of the number of clusters. In *Proc. 17th International Conf. on Machine Learning*, pages 727–734. Morgan Kaufmann, San Francisco, CA, 2000.
18. F. Ramos, B. Upcorft, S. Kumar, and H. F. Durrant-Whyte. A Bayesian approach for place recognition. In *Proceedings of IJCAI-05 Workshop on Reasoning with Uncertainty in Robotics*, Edinburgh, UK, 2005.
19. S. Se, D. Lowe, and J. Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *International Journal of Robotics Research*, 21(8):735–758, 2002.
20. R. Smith, M. Self, and P. Cheeseman. A stochastic map for uncertain spatial relationships. In *Fourth International Symposium of Robotics Research*, pages 467–474, 1987.
21. S. B. Williams, G. Dissanayake, and H. F. Durant-Whyte. An efficient approach to the simultaneous localization and mapping problem. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Washington, USA, 2002.
22. Q. Zhang and R. Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japan, 2004.