Multi-level State Estimation in an Outdoor Decentralised Sensor Network

B. Upcroft, M. Ridley, L.L. Ong, B. Douillard, T. Kaupp, S. Kumar, T. Bailey, F. Ramos, A. Makarenko, A. Brooks, S. Sukkarieh, H.F. Durrant-Whyte

Australian Centre for Field Robotics, The University of Sydney, Australia
`b.upcroft@cas.edu.au`

Summary. Decentralised estimation of heterogeneous sensors is performed on an outdoor network. Attributes such as position, appearance, and identity represented by non-Gaussian distributions are used in in the fusion process. It is shown here that real-time decentralised data fusion of non-Gaussian estimates can be used to build rich environmental maps. Human operators are also used as additional sensors in the network to complement robotic information.

1 Introduction

This paper presents the development and demonstration of multi-level Bayesian state estimation using an outdoor sensor network consisting of an autonomous air vehicle, a manual ground vehicle, and two human operators (Fig. 1). Decentralised fusion of position and identity information provided a common description (or map) of natural features in an unstructured environment. Unique aspects of this demonstration were (1) integration of human operators as information sources, (2) use of colour information from vision sensors to build rich world models, and (3) decentralised operation enabling a practical system which was robust, modular, and scalable.

Applications that benefit from multi-sensor data fusion include environmental sensing, surveillance, and search-and-rescue [1, 2, 3]. In each of these problems, individual nodes of the network make local measurements or observations of the common environment and attempt to combine the measurements to produce a global estimate of the observed state. The fusion approach adopted here is motivated by the need to survey and map large outdoor natural environments in which distributed sensor networks are prone to node and/or communication failure. In contrast to hierarchical and centralised distributed methods [4, 5], decentralised architectures ensure robustness to these failures while allowing scalability and modularity [6]. These properties arise as there are no central nodes to which global knowledge is communicated.

Previous approaches to robust decentralised data fusion have included tracking position features provided by range devices such as radar or laser [2], tracking artificial visual features with known range [7], monitoring temperature [1] or people

Fig. 1. The platforms used in the network: autonomous air vehicle, ground vehicle, and human operator. Close-ups of the sensor payloads including cameras are shown in the insets.

movement [8] in an office environment. Unlike these approaches, we concentrate on modelling *natural* features at three levels: *position*, *identity*, and *appearance* [9]. The resultant rich environment models are often required in real applications for both autonomous operation *and* to support human decision-making.

When observing complex environments, the observations made by robotic sensors and humans are likely to be complementary in terms of sensor modality, uncertainty, and robustness. Robotic sensors perform well in low-level descriptions such as geometric properties. In contrast, human operators can be valuable for higher-level tasks such as object recognition. Presence of such complementary information sources offers an opportunity for effective information fusion.

Probabilistic fusion of human and robotic information sources has been previously addressed only in theory or with non-probabilistic human observations [10, 11, 12]. The model presented here offers multiple abstractions of the available information to support analysis and decision-making, and permits the incorporation of higher-level human observations. The model is represented probabilistically as a Dynamic Bayesian Network (DBN) encoding statistical correlations between beliefs on all levels and the dynamics of the model. The methodology is a first step towards a more general framework for multi-level information fusion.

For this paper, position is considered to be independent of appearance and identity as far as the representation is concerned. At the geometric level, non-Gaussian Bayesian estimation was used for fusing bearing-only position observations from monocular vision sensors. Non-Gaussian distributed algorithms for sensor calibration have previously been developed by Ihler *et al.* [13], while Rosencrantz *et al.* demonstrated non-Gaussian distributed state estimation in the context of robotic laser tag [2]. In both cases, correlated estimation errors due to common information communicated between nodes in the past (also known as data incest) [14, 15] were not considered. Accounting for data incest in decentralised architectures is the key problem in ensuring mathematically consistent and convergent solutions. We also use an approximate node-node fusion algorithm [16] using Gaussian mixture models (GMMs) and a variant of the Covariance Intersect algorithm [17]. This algorithm provides consistent estimates in practice but like Ihler and Rosencrantz's work, there is no guarantee of a convergent solution.

The remainder of the paper describes the representation of the world that was used and realisation of the decentralised system with experimental results and practical experiences.

2 The Hierarchical World Representation

This section presents the representation used to describe features in the outdoor environment. Three levels are used: *position*, *appearance* and *identity*.

2.1 Geometrical Features

Position observations z_k were a sequence of azimuth (ψ) and elevation (θ) measurements [18]

$$z_k = [\psi\ \theta]^T = \begin{bmatrix} \tan^{-1}(y_k/x_k) \\ \tan^{-1}(z_k/\sqrt{x_k^2 + y_k^2} \end{bmatrix} + v_k \tag{1}$$

where v_k is the measurement noise.

A Gaussian mixture model (GMM) was used to fit the observations and thus an approximate transformation of the likelihood from the bearing-only measurement space to a sensor-centric Cartesian coordinate space was required [16]. In our experiments, the approximation for the GMM was learnt off-line using the Expectation-Maximisation (EM) algorithm [19]. The initial parameters for EM were equally weighted Gaussians spread evenly over a range of 200m on the x-axis. Approximately 50 iterations resulted in a good fit to the distribution. The learnt model was then rotated and translated appropriately for specific observations.

Local filtering was achieved using Bayes Theorem for the update and the Chapman-Kolmogorov equation for prediction [16, 20].

2.2 Appearance and Identity

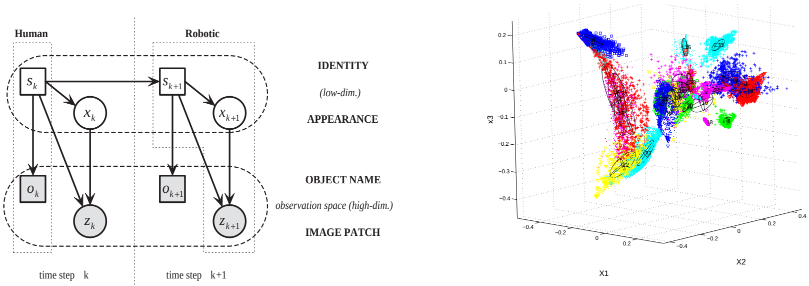

Fig. 2. LHS: The hierarchical appearance/identity model. Two time slices are shown to represent the filtering process as a DBN. Shaded nodes are observed by human operators or robotic cameras. RHS: The projection of the input points obtained by Isomap as well as the low dimensional components of the learnt model. The axes are the three eigenvectors of the low-dimensional space.

Fig. 2(LHS) shows a DBN representation of the identity/appearance states for a single feature. Round nodes are continuous random variables, square nodes are discrete, and shaded nodes are observed. The joint space z, x, s, represents a mixture of factor analysers [21, 22] and is learnt offline from training data. The s node is multinomial and determines the number and the weight of the GMM components.

2.3 Offline Model Learning

All parameters of the DBN model shown in Fig. 2(LHS) need to be specified. The transition model $P(s_{k+1}|s_k)$ is not required since s is assumed to be stationary.

The joint model $P(x, s, z)$ is learned on a typical set of labelled image patches using a combination of a nonlinear dimensionality reduction called Isomap [23] and Maximum Likelihood (ML) [21]. Isomap computes a low-dimensional output (x space) by preserving similarities in the high-dimensional input (z space). The purpose is to generate a fully supervised data set $\langle z, x\rangle$ which is then extended by manually labeling s. Maximum Likelihood (ML) is then used to estimate all the parameters.

Fig. 2(RHS) shows the learned model which was used for the experimental demonstration. Each 729-dimensional image patch in z space is reduced to a 3-dimensional representation in x space as suggested by the residual in Isomap. The ellipsoids represent a set of Gaussians with s representing the associated weights. Each Gaussian component represents a cluster containing data points whose corresponding image patches have similar colour appearance. The dimension of s is chosen to be 27, i.e. 27 components (or identities) based on the visible number of clusters. The number of samples used is 12388.

2.4 Human Observations

The correspondence of GMM components to feature identity leads to the interpretation of the s node as a higher-level abstraction of its appearance – an identity state. This state can be observed by human operators specifying object names o. Object names can represent a subset of all GMM components; e.g. a "tree" observation is represented by 7 GMM components.

The remaining conditional probability distribution $P(o|s)$ is the identity sensor model relevant for human observations. The o node's dimension N_o is chosen to be 4: "tree", "shed", "white_object" and "red_car". The sensor model is represented by a table of size $N_s \times N_o$ (27×4). For each object name, the weights of its corresponding components are distributed equally. The online computation of likelihoods is thus a simple table lookup. It is assumed here that human operators almost perfectly classify feature classes, which is reasonable for this application. For more complex classification tasks, experiments would be required to find a model.

3 State Estimation

3.1 Local Estimation

The first step in the online implementation for a node connected to a robotic sensor was to extract relevant features from raw observations using template matching [24]

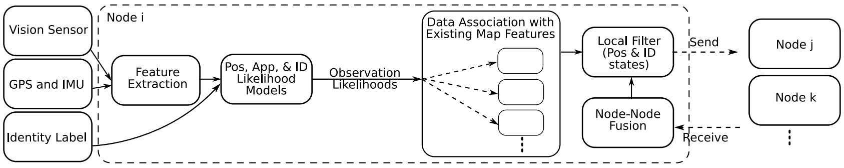

Fig. 3. Algorithmic framework for each node in the network.

(Fig 3). A node attached to a human operator also accepted identity likelihoods as observations.

The feature extractor generated observations of two types: (1) azimuth and elevation of the object in the extracted patch, (2) colour histograms formed from binning the RGB information contained in the extracted patch. The observation for position was then converted to a GMM in Cartesian space (see Sec. 2.1) and estimation proceeded using a Bayesian filter [25, 20, 26].

For the appearance observations, each histogram was processed as an observation of a track label. The fusion was performed through a Bayesian update

$$p(s_k|\mathbf{Z}_k) = \frac{p(\mathbf{z}_k|s_k)p(s_k|\mathbf{Z}_{k-1})}{p(\mathbf{z}_k|\mathbf{Z}_{k-1})} \tag{2}$$

where the observation model is given by

$$p(\mathbf{z}|s) = \int_{\mathbf{x}} p(\mathbf{z},\mathbf{x}|s)d\mathbf{x} = \int_{\mathbf{x}} p(\mathbf{z}|\mathbf{x},s)p(\mathbf{x}|s)d\mathbf{x} \tag{3}$$

and the terms $p(\mathbf{z}|\mathbf{x},s)$ and $p(\mathbf{x}|s)$ arise from the expansion of the joint distribution $p(\mathbf{z},\mathbf{x},s) = p(\mathbf{z}|\mathbf{x},s)p(\mathbf{x}|s)p(s)$.

3.2 Decentralised Estimation

All local node information was then transmitted to neighbouring platforms. The net result was that each platform locally maintained a complete map (or belief) of all features observed by all nodes in the network. Multiple observations of the same feature, possibly by different platforms, resulted in an increasingly accurate estimate of the feature location, its appearance properties, and its identity for *all* nodes.

It can be shown that fusion of the raw correlated information between nodes i and j is [14, 15]

$$p(\mathbf{x}|\mathbf{Z}_i \cup \mathbf{Z}_j) = \frac{1}{c}\frac{p(\mathbf{x}|\mathbf{Z}_i)p(\mathbf{x}|\mathbf{Z}_j)}{p(\mathbf{x}|\mathbf{Z}_i \cap \mathbf{Z}_j)} \tag{4}$$

where $\mathbf{Z}_{i(j)}$ are all the observations available to node $i(j)$, $p(\mathbf{x}|\mathbf{Z}_i \cup \mathbf{Z}_j)$ is the posterior probability over the unknown state given information from both nodes, $p(\mathbf{x}|\mathbf{Z}_{i(j)})$ are the posteriors based only on locally available information, $p(\mathbf{x}|\mathbf{Z}_i \cap \mathbf{Z}_j)$ is the information the two nodes have in common, and c is a normalising constant.

Thus the problem of constructing the union $\mathbf{Z}_i \cup \mathbf{Z}_j$, reduces to finding the common information $\mathbf{Z}_i \cap \mathbf{Z}_j$ and is the *key* to the decentralised communication problem. The incorporation of redundant information in DDF systems may lead to bias, over-confidence, and divergence in estimates.

The common information for the multinomial class labels could be calculated anaytically according to Eq. 4. However, analytical solutions to the division for GMMs are unknown. A non-optimal solution for node-to-node fusion of Gaussian representations is the Covariance Intersect (CI) filter which conservatively combines the information in two incoming channels assuming that the correlation is unknown [17]. Here, we use Gaussian mixture models (GMMs) and a variant of the Covariance Intersect algorithm [16]. However, as in the work of Ihler *et al.* [13] and Rosencrantz *et al.* [2] divergent solutions are possible.

Covariance Intersect Filter

Consider two estimates μ_a and μ_b with covariances $\mathbf{\Sigma}_a$ and $\mathbf{\Sigma}_b$ respectively. The CI algorithm computes an updated covariance matrix as a convex combination of the two initial covariance matrices in the form

$$\mathbf{\Sigma}_c^{-1} = \omega \mathbf{\Sigma}_a^{-1} + (1-\omega)\mathbf{\Sigma}_b^{-1} \tag{5}$$

$$\mathbf{\Sigma}_c^{-1}\mu_c = \omega \mathbf{\Sigma}_a^{-1}\mu_a + (1-\omega)\mathbf{\Sigma}_b^{-1}\mu_b \tag{6}$$

where $0 \leq \omega \leq 1$ with ω computed so as to minimise a chosen measure for the size of the covariance matrix.

The resultant estimate is based on all possible correlations and thus removes the need for the division in Eq. 4.

Pairwise Component Covariance Intersect

An extension to the CI algorithm involves a pairwise CI update between each of the Gaussian components in the two mixtures that are to be fused. The weight update for each component is given by

$$\pi_c = \alpha \pi_a \pi_b \tag{7}$$

where

$$\alpha = \frac{1}{(2\pi)^{D/2}\left|\varSigma_\omega\right|^{1/2}} e^{-1/2(\mu_a-\mu_b)^T \varSigma_\omega^{-1}(\mu_a-\mu_b)} \tag{8}$$

is the scaling constant resulting from the multiplication of two Gaussians, D is the dimension of the space, and $\varSigma_\omega = \varSigma_a/\omega + \varSigma_b/(1-\omega)$.

We have found that this update remains non-divergent for all the practical scenarios we have encountered and is always better than a straight multiplication in which the common information is not accounted for at all [16]. Although this conservative behaviour is not guaranteed we hope that in future work convergence bounds can be obtained.

4 Experimental Results

The algorithms were demonstrated using a four node network with three of the platforms illustrated in Fig. 1. Both vehicles were equipped with Global Positioning

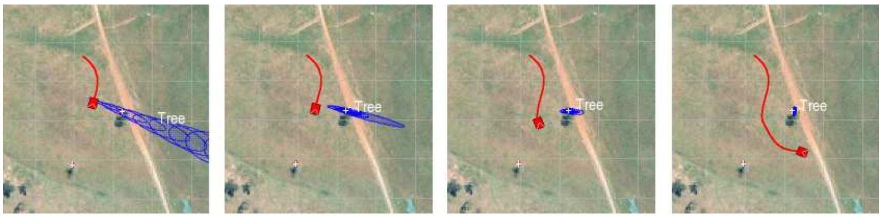

Fig. 4. Surveyed differential GPS locations are marked as white crosses on the aerial photograph. Ellipses of the same colour represent 2-D projections of a Gaussian mixture representing the position estimate of a single feature. Labels represent the component with the highest probability inferred from the visual model. The ellipses and labels combine to make up the entire map. The red line represents the trajectory taken by the vehicle.

System (GPS) and Inertial Measurement Unit (IMU) sensors in addition to a single vision sensor. The two human operators were able to input and receive information using tablet PCs with attached hand-held GPS units. Physical communication between nodes was achieved with standard IEEE 802.11b wireless network adaptors using the UDP protocol. The system architecture as a whole was developed using the Active Sensor Network framework [27] while the software implementation of the system adopted the component paradigm of the Orca robotics project[1]. This ensured a modular approach to development and ease of software component interaction and communication which is essential in such a system. Decentralised communication and testing of various probabilistic representations also required the development of a low-level communication software library[2] and a general library for probabilistic algorithms[3].

The experiments were performed at an outdoor test facility over an area of a few square kilometres. The position of each of the vehicles and the fused estimates at each of the nodes could be monitored with an online graphical user interface (GUI) overlaying a geo-referenced aerial image taken prior to the demonstration. A number of objects such as trees, sheds, and cars were surveyed using differential GPS measurements allowing comparisons to ground truth.

Fig. 4 illustrates GUI screenshots of a sequence of map updates for observations of a tree from a ground vehicle. The camera was mounted sideways so forward movement automatically increased the baseline between observations. As the vehicle moves past the tree, the updated estimate increases in accuracy and converges to the true location of the object (indicated by the white cross). Note that the label, representing the component with the highest probability inferred from the visual model, also correctly identifies the object.

The map shown in Fig. 5 is a live screenshot of the belief of one of the platforms after multiple nodes entered the network. Each set of coloured ellipses with a corresponding label represents a different feature. Qualitatively, it can be seen

[1] http://orca-robotics.sf.net/orca1
[2] http://crud.sf.net
[3] http://spasm.sf.net

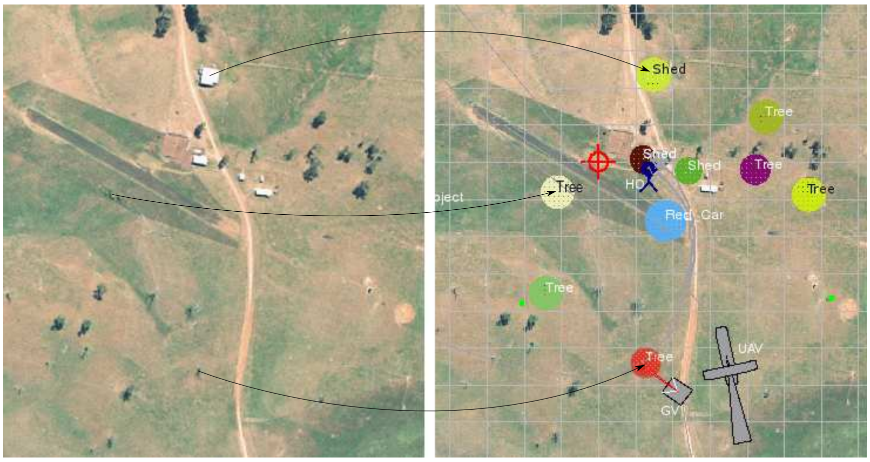

Fig. 5. RHS: The belief of one of the nodes in the network. Each set of coloured ellipses corresponds to a particular feature and the labels represent the identity state with highest probability. The icons represent each of the different nodes in the network; UAV = air vehicle, GV = ground vehicle, HO = human operator. LHS: the original aerial image with arrows highlighting a few of the correspondences with the belief of the node.

that an accurate map with correct feature classification was achieved. Simulations in previous papers show detailed analysis of the algorithms [16, 28].

5 Practical Experiences

Our practical experience of deploying such a large scale system has highlighted the benefits of a decentralised architecture. Although implementation initially proved to be time consuming and intensive, once the basic software infrastructure was in place, physical and functional extension to the system was relatively easy. This success was due to system division into independent, interacting software components. The component paradigm also improved software-hardware integration to a much larger extent than was expected.

Component interaction was specified using Orca's graphical utility Gorca, which provided a simple drag-&-drop interface with automatic creation of configuration files for each platform. Thus we were able to easily build and observe the interaction between many software components over different platforms. Each component could be started remotely through Orca's monitoring utility and thus each platform could easily be integrated into the network.

Demonstration of such a large system also pointed out the weaknesses in our approach, the main one being the use of consumer-grade wireless network hardware. Although low in price, we found that the buggy firmware was detrimental to system operation. Often the hardware would work between two nodes over a relatively large distance, however failure would occur as more nodes contributed information. Thus, reliable equipment and extensive testing at all levels is required for system robustness.

6 Conclusion

We have shown that decentralised data fusion with non-Gaussian represenations can be performed in real-time. Additional to position, appearance and identity labels were included in the estimation process increasing the richness of information. Human operators were able to submit observations at the identity level allowing human and robotic information to complement each other. Future research will involve analysing the advantages of including additional attributes particularly for data association. Further understanding of non-Gaussian data estimation is also needed to ensure consistent estimates are maintained at the node-node fusion level.

References

1. M. A. Paskin and C. E. Guestrin, "Robust probabilistic inference in distributed systems," *In Proceedings of the Twentieth Conference on Uncertainty in Artificial Intelligence (UAI-04)*, 2004.
2. M. Rosencrantz, G. Gordon, and S. Thrun, "Decentralized sensor fusion with distributed particle filters," *In Proceedings of the Conference on Uncertainty in AI (UAI), Acapulco, Mexico*, 2003.
3. F. Bourgalt and H. Durrant-Whyte, "Communication in general decentralized filters and the coordinated search strategy," *In Proceedings of The 7th International Conference on Information Fusion*, 2004.
4. H. Hashemipour, S. Roy, and A. Laub, "Decentralized structures for parallel Kalman filtering," *IEEE Trans. Automatic Control*, vol. 33, pp. 88–93, 1988.
5. H. Nii, "Blackboard systems," *AI Magazine*, 1986.
6. J. Manyika and H. Durrant-Whyte, *Data Fusion and Sensor Management: A Decentralized Information-Theoretic Approach.* Ellis Horwood, 1994.
7. E. Nettleton, P. Gibbens, and H. Durrant-Whyte, "Closed form solutions to the multiple platform simultaneous localisation and map building (slam) problem," *Sensor Fusion: Architectures, Algorithms, and Applica-tions IV*, pp. 428–437, 2000.
8. A. Brooks, A. Makarenko, T. Kaupp, S. Williams, and H. Durrant-Whyte, "Implementation of an indoor active sensor network," *Proceedings of the 9th International Symposium on Experimental Robotics*, 2004.
9. T. Kaupp, B. Douillard, B. Upcroft, and A. Makarenko, "Hierarchical environmental model for fusing information from human operators and robots," *In IEEE/RSJ International Conference Proceedings on Intelligent Robots and Systems*, 2006.
10. G. Pavlin, P. de Oude, and J. Nunnink, "A mas approach to fusion of heterogeneous information," *In IEEE/WIC/ACM Joint Conference on Intelligent Agent and Web Technology*, 2005.

11. K. Ross, R. Chaney, G. Cybenko, D. Burroughs, and A. Willsky, "Mobile agents in adaptive hierarchical bayesian networks for global awareness," *In IEEE International Conference on Systems, Man, and Cybernetics*, vol. 3, pp. 2207–2212, 1998.
12. D. J. Bruemmer and M. C. Walton, "Collaborative tools for mixed teams of humans and robots," in *NRL Workshop on Multi-Robot Systems*, Washington, DC, USA, 2003, pp. 219–229.
13. A. Ihler, J. Fisher III, R. Moses, and A. Willsky, "Nonparametric belief propagation for self-calibration in sensor networks," *Information Processing in Sensor Networks, Berkeley, California, USA*, 2004.
14. Y. Bar-Shalom, Ed., *Multitarget-multisensor tracking: advanced applications.* Artech House, 1990.
15. Y. Bar-Shalom and X. Li, *Multitarget-multisensor tracking: principles and techniques*, 1995.
16. B. Upcroft, L. L. Ong, S. Kumar, M. Ridley, T. Bailey, S. Sukkarieh, and H. Durrant-Whyte, "Rich probabilistic representations for bearing-only decentralised data fusion," *In IEEE Conference Proceedings on Information Fusion 2005*, 2005.
17. S. Julier and J. Uhlmann, "A nondivergent estimation algorithm in the presence of unknown correlations," *In Proceedings of the American Control Conference*, vol. 4, pp. 2369–2373, 1997.
18. L. Stone, C. Barlow, and T. Corwin, "Bayesian multiple target tracking," *Artech House, Norwood*, 1999.
19. A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society B*, vol. 39, pp. 1–37, 1977.
20. D. Alspach and H. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Transactions on Automatic Control*, vol. AC-17, p. 439, 1972.
21. F. Ramos, S. Kumar, B. Upcroft, and H. Durrant-Whyte, "Representing natural objects in unstructured environmentss," *In Advances in Neural Information Processing Systems, Workshop on Machine Learning Based Robotics in Unstructured Environments*, 2005.
22. Z. Ghahramani and G. Hinton, "The em algorithm for mixtures of factor analysers," *University of Toronto Technical Report CRG-TR-96-1*, 1996.
23. J. Tenenbaum, V. DeSilva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality redution," *Science*, vol. 290, pp. 2319–2323, 2000.
24. D. Scott, *Multivariate density estimation.* Wiley, 1992.
25. H. Sorenson and D. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, p. 465, 1971.
26. D. Alspach, "A Gaussian sum approach to the multi-target identification-tracking problem," *Automatica*, vol. 11, p. 285, 1975.
27. A. Makarenko, A. Brooks, S. Williams, H. Durrant-Whyte, and B. Grocholsky, "A decentralized architecture for active sensor networks," *In IEEE International Conference on Robotics and Automation*, 2004.
28. L. L. Ong, M. Ridley, B. Upcroft, S. Kumar, T. Bailey, S. Sukkarieh, and H. Durrant-Whyte, "A comparison of probabilistic representations for decentralised data fusion," *In Proceedings of Intelligent Sensors, Sensor Networks and Information Processing*, 2005.