A Stochastic Model for Natural Feature Representation

Kumar S., Ramos F., Upcroft B., Ridley M., Ong L., Sakkarieh S. and Durrant-Whyte H.
ARC Centre of Excellence for Research in Autonomous Systems
University of Sydney,
Sydney, NSW-2006
(s.kumar, f.ramos, b.upcroft, m.ridley, s.ong, salah, hugh)@acfr.usyd.edu.au

Abstract— **This paper presents a robust stochastic model for the incorporation of natural features within data fusion algorithms. The representation combines Isomap, a non-linear manifold learning algorithm, with Expectation Maximization, a statistical learning scheme. The representation is computed offline and results in a non-linear, non-Gaussian likelihood model relating visual observations such as color and texture to the underlying visual states.**

The likelihood model can be used online to instantiate likelihoods corresponding to observed visual features in real-time. The likelihoods are expressed as a Gaussian Mixture Model so as to permit convenient integration within existing nonlinear filtering algorithms. The resulting compactness of the representation is especially suitable to decentralized sensor networks. Real visual data consisting of natural imagery acquired from an Unmanned Aerial Vehicle is used to demonstrate the versatility of the feature representation.

I. Introduction

Autonomous navigation and data fusion tasks require robust feature extraction and representation. Traditional schemes in autonomous navigation have focussed on the selection of stable point features through the use of ranging devices (laser [1], sonar [2]). While such techniques have been deployed in unmanned air, ground and underwater vehicles, they do not provide rich characterizations of an unstructured environment in terms of color, texture or other sensory properties.

The use of visual sensing for automatic feature extraction and representation in decentralized sensor networks has been limited. Nettleton [3] used visual sensing to automatically extract and propagate point features that corresponded to high contrast beacons in a decentralized network. Brand [4] exploits point feature correspondence in an urban environment to perform localization in sensor networks.

The computer vision community has developed several stochastic feature representation schemes, although their application in robotics has been limited. Lee et al [5] present a generative visual model based on Independent Components Analysis (ICA) which provides a linear and non-Gaussian framework for feature representation. In the work of Karklin et al [6], a hierarchical probabilistic framework is presented for the detection of higher order statistical structure in natural imagery. A key limitation of these linear models is that they do not necessarily preserve the inherent similarities and distinctions in the original visual data. This minimizes their utility in classical estimation and data association tasks.

This research focusses on the automated extraction and representation of natural visual features for use in decentralized sensor networks to enable rich, probabilistic characterizations of the environment. This work has potential applications in feature selection for Simultaneous Localization and Mapping (SLAM), terrain classification and tactical picture compilation. Prior placement of beacons within the environment is not assumed and well defined features such as corners that are prevalent in urban scenarios are not required. The extracted features are not restricted to be points and are meaningful regions of natural imagery in general. The feature representation scheme results in a compact Gaussian Mixture Model (GMM) that is especially suited to decentralized sensor networks.

This work explicitly assumes that all visual data (e.g. color, texture) is sampled from the vicinity of a low dimensional manifold embedded in the observation space comprised of raw pixels. The concepts of Nonlinear Dimensionality Reduction (NLDR) [7] are then combined with Expectation Maximization (EM) [8] to compute stochastic representations of natural features. Such a representation leads to a compact, nonlinear and non-Gaussian description of high dimensional visual observations such as color and texture. Critically, this representation can be learnt offline and used to infer the underlying states of visual observations in real-time.

The feature extraction scheme adopted in this work is summarized in Section II. Section III presents an overview of NLDR schemes and outlines the basic theory of the Isomap method used in this work. Section IV describes the methodology to compute a statistical representation of NLDR algorithms as a GMM using EM. The statistical representation of natural features results in a *probabilistic likelihood model* that may be integrated within existing non-linear filtering algorithms for state estimation of visual features [9]. The methodology is finally applied to real natural imagery acquired from an Unmanned Aerial Vehicle (UAV). It is demonstrated that the model results in a compact, neighborhood preserving, statistical representation of the underlying state of visual features.

II. Note on Feature Extraction

A. Overview

The feature extraction algorithm used in this work is based on concepts from information theory to extract novel features from the sensory space. Novelty is informally defined to correspond to features with a low probability of occurrence and thus a high information content [10].

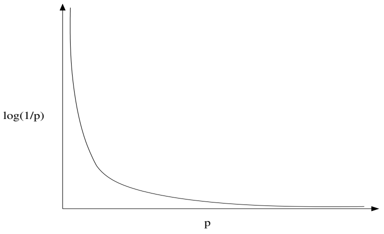

Fig. 1. Novelty in a random variable vs the probability of occurrence

The notion of novelty (Figure 1) as defined here can be used to develop an information theoretic feature selection scheme. The objective of effective feature selection is to identify regions in natural imagery with unique properties (e.g. color, texture or other relevant visual cues). The frequency of occurrence of the properties can be quantified through property histograms and the feature selection problem is addressed by working with the least likely features. This feature extraction scheme is expected to be especially useful for SLAM wherein unique landmarks are vital for robust loop closure. Terrain classification applications involve the opposing philosophy of extraction of the most likely visual features.

B. Illustration

The feature extraction scheme is illustrated through a sample image acquired from an Unmanned Aerial Vehicle (UAV).

1) The image is first converted to the Hue-Saturation-Value (HSV) space, and the hue histogram of the image is computed. The hue based information content (Figure 2 top right) at each pixel is computed as proportional to $log\left(\frac{1}{p}\right)$.
2) The raw image is convolved with Gabor wavelets [11] at 2 scales and 2 orientations, and the histogram of the resultant amplitude of the response is used to compute a texture based information content (Figure 2 bottom left) at each pixel.
3) Features that maximize mutual information between hue and texture are obtained by fusing the hue and texture based information content at each pixel (Figure 2 bottom right).

C. Feature Stability

For robust feature association, it is critical that the extracted features persist in the environment. Numerical experiments over real natural imagery acquired from terrestrial, underwater and aerial vehicles have demonstrated that features that maximize mutual information between multiple visual cues such as color, texture and intensity gradients exhibit greater stability than those based on single visual cues. Figure 3 demonstrates the persistence of the extracted features over a small image sequence acquired from a UAV.

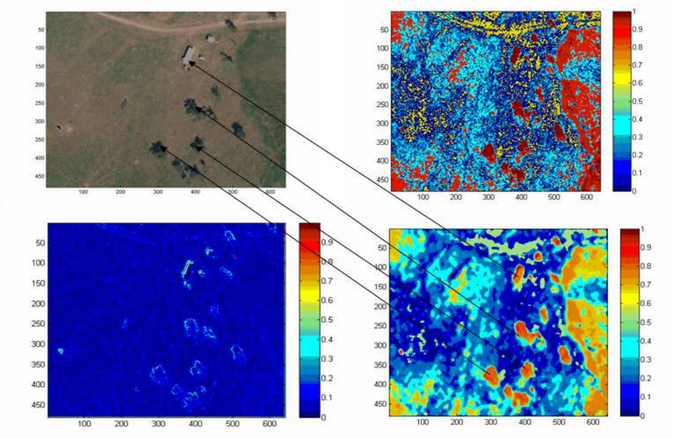

Fig. 2. **Top row** - Original image acquired by a UAV (Left) and Information content based on a hue histogram (Right). **Bottom row** - Information content in texture space computed by convolving image with Gabor wavelets (Left) and Features that maximize mutual information between hue and texture highlighted in red (Right)

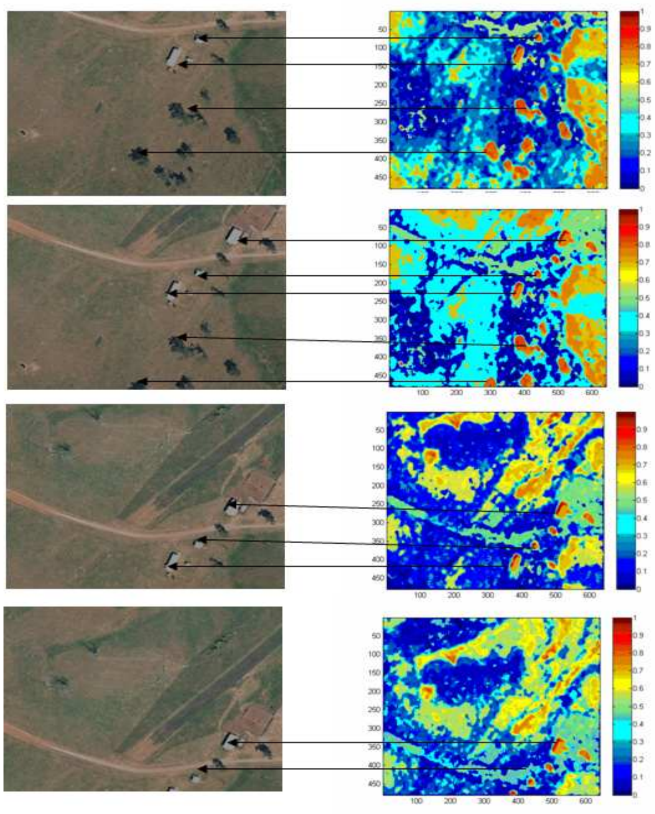

Fig. 3. Features that maximize mutual information between multiple visual cues such as color and texture appear persistent, and hence are worth tracking. These features are color coded red in the right column that displays the mutual information content between hue and texture at each pixel location.

III. Nonlinear Dimensionality Reduction

The feature representation scheme presented here is independent of any specific feature extraction algorithm. In this implementation, information theoretic concepts are used to extract features with unique properties within the sensory space as summarized in section II. While this feature selection scheme is ideal for SLAM, task driven feature extractors may be more appropriate in other scenarios.

Each extracted feature is potentially set in a very high dimensional space that is not readily amenable to simple interpretation and reasoning tasks. The development of compact and useful representations of natural features in unstructured dynamic worlds is critical to the development of next generation autonomous systems. These algorithms have numerous potential applications ranging from data compression, robust data association to assist autonomous navigation and unsupervised feature selection to create terrain models.

While traditional dimensionality reduction methods such as Principal Component Analysis (PCA) and its numerous variants provide theoretically optimal representations from a data-compression standpoint, they are unable to provide neighborhood preserving representations that are crucial to data association. This limitation has motivated the development of various nonlinear embedding methodologies such as Kernel PCA [12], Isomap [7], Laplacian Eigenmaps [13] and Locally Linear Embedding (LLE) [14]. Most NLDR techniques presume that the data lies on or in the vicinity of a low dimensional manifold and attempt to map the high dimensional data into a single low dimensional, global coordinate system. The Isomap algorithm is adopted in this work to provide a low dimensional description of high dimensional features primarily because it estimates the intrinsic dimensionality of the manifold in addition to the underlying states.

A. Theoretical Aspects of the Isomap Method

The Isomap method [7] formulates NLDR as the problem of finding a Euclidean feature space embedding of a set of observations that attempts to explicitly preserve their intrinsic metric structure; the metric structure is quantified as the geodesic distances between the points along the manifold.

The Isomap method assumes that the sensor data $\vec{Z}$ lies on a smooth nonlinear manifold embedded in the high dimensional observation space and attempts to reconstruct an implicit mapping $f : \vec{Z} \rightarrow \vec{X}$ that transforms the data to a low dimensional Euclidean feature (state) space $\vec{X}$, that optimally preserves the distances between the observations as measured along geodesic paths on the manifold. Significant steps in the Isomap algorithm are summarized next.

B. Nearest Neighbor Computation

Neighboring points on the manifold are determined based on the input space distances $d_z(i,j)$ between pairs of points $i,j \in \vec{Z}$. Each input point is connected to adjacent points based either on the K nearest neighbors or all points within a fixed distance ϵ from the point under consideration. The neighborhood relations are expressed as a weighted graph G over the data points with edges of weight $d_z(i,j)$ between neighboring points.

C. Computation of Geodesic Distances

The length of a path in G is defined as the sum of the link weights along the path. The shortest path lengths d_G^{ij} between two nodes i and j in the graph G are computed through the Floyd's algorithm [15] that generally scales as $O(N^3)$ or the Dijkstra algorithm [16] that scales as $O(N^2 log(N))$, where N is the number of data points.

D. Graph Embedding Through Multi-Dimensional Scaling

Classical Multi-dimensional Scaling (MDS) [17] is now used to compute a graph embedding in k dimensional space that respects closely the geodesic distances d_G^{ij} computed through the dynamic programming algorithms. The coordinate vectors $x_i \in \vec{X}$ are chosen to minimize the cost function $E = \|\tau(d_G) - \tau(d_X)\|_{L^2}$, where d_X is the matrix of output space distances and the norm is the matrix L^2 norm $\sqrt{\sum_{i,j} (\tau(d_G) - \tau(d_X))^2_{ij}}$. τ is an operator that converts distances into inner products and is defined as $\frac{1}{2}HSH$, where the centering matrix $H_{ij} = \delta_{ij} - (1/N)$ and the matrix of squared distances $S_{ij} = D^2_{ij}$. The global minimum of the cost function is computed by setting the output space coordinates x_i to the top k eigenvectors of $\tau(d_G)$.

E. Practical Implementation

Any feature extraction scheme identifies specific regions in an image that exceed a general information threshold. Each such feature is comprised of several pixels in general, and each pixel can be described by the raw color intensities, multi-scale texture and other visual cues (e.g. intensity gradient, brightness gradient, texture gradient).

In a practical implementation, each extracted feature is subdivided into image patches of a fixed size (e.g. 11×11) and the image patch is described by a single vector of properties of all individual pixels within the patch. These vectors $\vec{Z}$ constitute high dimensional visual observations and form the input to Isomap. Isomap computes a low dimensional representation of high dimensional image patches such that the inherent similarities (or distinctions) in the original patches are preserved. The low dimensional representation physically corresponds to the intrinsic coordinates (or equivalently the visual state) of each patch on the nonlinear manifold.

F. The S Manifold - A 2 Dimensional Manifold Embedded in 3 Dimensions

The nonlinear manifold representation computed by Isomap is illustrated through a synthetic example. An analytically generated two dimensional manifold [14] embedded in three dimensions is depicted in Figure 4. The objective of Isomap is to automatically discover the global coordinates intrinsic to the two dimensional embedding without any explicit directives on how the data is to be mapped onto the low dimensional space.

Figure 4 shows the original manifold and 2000 randomly generated samples. The samples are color coded according

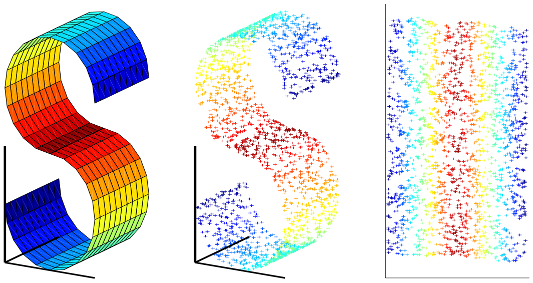

Fig. 4. S Manifold - Original Manifold (**Left**) Sampled Manifold (**Middle**) Isomap Embedding (**Right**)

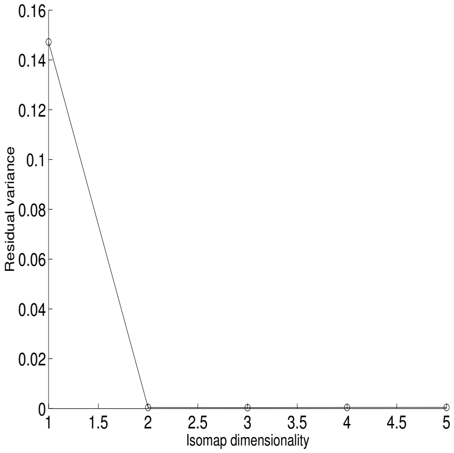

Fig. 5. Residual variance vs. Manifold dimensionality computed by Isomap

to their spatial positions in three dimensions. The objective of Isomap is to find a two dimensional representation of the samples that essentially preserves the neighborhood structure depicted by the color coding.

Figure 5 demonstrates that Isomap has accurately assessed the intrinsic two dimensional nature of this manifold through the vanishing residual variance in higher dimensions. It is clear from a comparison of the color coding of points in the sampled manifold and the two dimensional embedding (middle and extreme right in Figure 4) that high dimensional neighborhoods are preserved in the computed two dimensional embedding. This property is crucial to applications in robotics as similar visual observations of color and texture in the original sensor space must have similar underlying visual states.

IV. Statistical Models

A. The Generative Model

The Isomap algorithm and indeed most NLDR algorithms are inherently *deterministic* algorithms that do not provide a measure of the *uncertainty* of the underlying visual states of high dimensional observations. The integration of the low dimensional states computed by Isomap into a probabilistic, Bayesian filtering framework requires the definition of a generative likelihood model $P(\vec{Z}|\vec{X})$, where $\vec{Z}$ and $\vec{X}$ are the observation and state spaces respectively. This likelihood model encapsulates the uncertainties inherent in the inference of a low dimensional state from noisy high dimensional observations. The incorporation of natural feature states within a non-Gaussian and non-linear filter is expected to significantly enhance data association as the low dimensional appearance states and kinematic variables are complementary.

Methods from supervised learning can be used to derive compact mappings that generalize over large portions of the observation and state 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.

Given the results of Isomap, a probabilistic model of the joint distribution $P(\vec{Z},\vec{X})$ can be learnt through the EM algorithm [8]. The joint distribution can be used to map inputs to outputs and vice versa by computing the expected values $E[\vec{Z}|\vec{X}]$ and $E[\vec{X}|\vec{Z}]$. The joint distribution is represented by a generalization of a GMM that is termed as a mixture of factor analyzers [18]. The joint distribution is graphically displayed in Figure 6 with the assumed dependencies. The discrete hidden variable s in the model physically represents a specific neighborhood on the manifold over which a mixture component is representative. This representation conveniently handles highly nonlinear manifolds through the capability to model the local covariance structure of the data in different areas of the manifold.

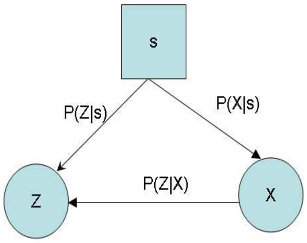

Fig. 6. Graphical model for computation of parametric models from NLDR algorithms. An arrow directed into a node depicts a dependency on the originating node. The discrete hidden variable s represents a specific neighborhood on the manifold.

The complete three step generative model can now be summarized based on the assumed dependencies (Equations 1-3). The joint probability distribution of all the random variables in the graphical model is expressed as

$$P\left(\vec{z},\vec{x},s\right) = P(\vec{z} \mid \vec{x},s)P(\vec{x} \mid s)P\left(s\right) \tag{1}$$

where $\vec{z} \in \vec{Z}$, $\vec{x} \in \vec{X}$ and the dependencies are given by

$$P(\vec{z} \mid \vec{x},s) = \frac{1}{\left(2\pi\right)^{D/2}\left|\Psi_s\right|^{1/2}} \times \tag{2}$$

$$\exp\left\{-\tfrac{1}{2}\left[\vec{z}-\Lambda_s\vec{x}-\mu_s\right]^T\Psi_s^{-1}\left[\vec{z}-\Lambda_s\vec{x}-\mu_s\right]\right\}$$

$$P(\vec{x}\mid s)=\frac{1}{(2\pi)^{d/2}\left|\Sigma_s\right|^{1/2}}\times \tag{3}$$

$$\exp\left\{-\tfrac{1}{2}\left[\vec{x}-\nu_s\right]^T\Sigma_s^{-1}\left[\vec{x}-\nu_s\right]\right\}$$

B. Parameter Estimation

In this model, the set of parameters θ that need to be estimated from the observed high and low dimensional spaces are the prior probabilities $P(s)$, which follow a multinomial distribution, the mean vectors $\vec{\nu_s}$ and $\vec{\mu_s}$, the full covariance matrix Σ_s, the diagonal covariance matrix Ψ_s and the loading matrices Λ_s. The EM algorithm performs iterative parameter estimation by maximizing the log-likelihood of the data given the model and the set of parameters. The observable parameters in the graphical model are denoted as $\{\vec{z}_n,\vec{x}_n\}_{n=1}^N$ where N is the number of samples. EM iteratively maximizes the log-likelihood of the observations

$$\mathcal{L}=\sum_{n=1}^{N}\log\sum_{i=1}^{M}P\left(\vec{z}_n,\vec{x}_n,s_i\mid\theta\right), \tag{4}$$

where M is the number of mixtures considered in the model. Since direct maximization over the above expression is hard to be calculated analytically, an auxiliary distribution $q\left(s_i\right)$ over the hidden variable is introduced:

$$\mathcal{L}=\sum_{n=1}^{N}\log\sum_{i=1}^{M}q\left(s_i\right)\frac{P\left(\vec{z}_n,\vec{x}_n,s_i\mid\theta\right)}{q\left(s_i\right)} \tag{5}$$

Then, it is possible to obtain a lower bound for $\mathcal{L}$ by applying the Jensen's inequality [10]:

$$\mathcal{L}\geq\sum_{n=1}^{N}\sum_{i=1}^{M}q\left(s_i\right)\log\frac{P\left(\vec{z}_n,\vec{x}_n,s_i\mid\theta\right)}{q\left(s_i\right)} \tag{6}$$

$$=\sum_{n=1}^{N}\sum_{i=1}^{M}q\left(s_i\right)\log P\left(\vec{z}_n,\vec{x}_n\mid\theta\right)+ \tag{7}$$

$$\textstyle\sum_{n=1}^{N}\sum_{i=1}^{M}q\left(s_i\right)\log\frac{P(s_i|\vec{z}_n,\vec{x}_n,\theta)}{q(s_i)}$$

$$=\sum_{n=1}^{N}\log P\left(\vec{z}_n,\vec{x}_n\mid\theta\right)- \tag{8}$$

$$\textstyle\sum_{n=1}^{N}\sum_{i=1}^{M}q\left(s_i\right)\log\frac{q(s_i)}{P(s_i|\vec{z}_n,\vec{x}_n,\theta)}.$$

Thus, maximizing $\mathcal{L}$ with respect to $q\left(s_i\right)$ is equivalent to minimizing the second term of (Equation 9) which is the Kullback-Leibler divergence between the free distribution $q\left(s_i\right)$ and the posterior probability $P\left(s_i\mid\vec{z}_n,\vec{x}_n,\theta\right)$.

For each iteration EM alternates between the Expectation step where the posterior probability of s given the observations is computed through

$$P(s\mid\vec{z}_n,\vec{x}_n)=\frac{P\left(\vec{z}_n\mid\vec{x}_n,s\right)P\left(\vec{x}_n\mid s\right)P\left(s\right)}{\sum_{s'}P\left(\vec{z}_n\mid\vec{x}_n,s'\right)P\left(\vec{x}_n\mid s'\right)P\left(s'\right)} \tag{9}$$

and the Maximization step, where this posterior is used to re-estimate the parameters. The update rules for the Maximization step are presented below:

Defining $\gamma_{sn}=P(s\mid\vec{z}_n,\vec{x}_n)$ and $\omega_{sn}=\frac{\gamma_{sn}}{\sum_{n'}\gamma_{sn'}}$ the updates are:

$$\vec{\nu_s}\leftarrow\sum_n\omega_{sn}\vec{x}_n, \tag{10}$$

$$\Sigma_s\leftarrow\sum_n\omega_{sn}\left[\vec{x}_n-\vec{\nu_s}\right]\left[\vec{x}_n-\vec{\nu_s}\right]^T, \tag{11}$$

$$\Lambda_s\leftarrow\sum_n\omega_{sn}\vec{z}_n\left(\vec{x}_n-\vec{\nu_s}\right)^T\Sigma_s^{-1}, \tag{12}$$

$$\vec{\mu_s}\leftarrow\sum_n\omega_{sn}\left[\vec{z}_n-\Lambda_s\vec{x}_n\right], \tag{13}$$

$$\Psi_s\leftarrow\sum_n\omega_{sn}\left[\vec{z}_n-\Lambda_s\vec{x}_n-\vec{\mu_s}\right]\left[\vec{z}_n-\Lambda_s\vec{x}_n-\vec{\mu_s}\right]^T, \tag{14}$$

$$P\left(s\right)\leftarrow\frac{\sum_n\gamma_{sn}}{\sum_{s'n'}\gamma_{s'n'}}. \tag{15}$$

The algorithm continues execution until the difference between the log-likelihood of two iterations is smaller than a given threshold.

C. Comments

Once the parameter estimation is completed, the joint distribution $P(\vec{z},\vec{x},s)$ is fully characterized, and a likelihood model $P(\vec{z}=\vec{z_i}|\vec{x})$ (Equation 16) can be computed by making an observation $\vec{z_i}$ in the high dimensional space. Along the lines of the derivation in [18], it can be shown that this likelihood can be expressed as a GMM.

$$P\left(\vec{z}=\vec{z_i}|\vec{x}\right)=\sum_s P\left(s|\vec{z}=\vec{z_i}\right)P\left(\vec{x}|\vec{z_i},s\right) \tag{16}$$

Such a model can be easily integrated within a non-linear, non-Gaussian filtering scheme [9]. Further, as the NLDR algorithms compute essentially invariant properties in the underlying low dimensional state, a process model is not required to describe their evolution. Thus an update based on Bayes theorem is sufficient for integration within a nonlinear filtering algorithm.

A crucial difference between this model and an unsupervised mixture of factor analyzers is that both the high and low dimensional spaces are observed as opposed to the low dimensional states remaining hidden in conventional factor analysis. The fact that dimensionality reduction is decoupled from the learning is expected to significantly stabilize iterative EM based parameter estimation approaches due to the observability of the low dimensional states [14].

D. S-Manifold - Statistical Representation

Figures 7 and 8 show the statistical model of the S-manifold computed with 32 factor components. The Gaussian distributions $P(\vec{z}|\vec{x}, s)$ accurately model the locally linear neighborhoods of the manifold from which the data was generated. The quality and numerical robustness of the representation results from the observation of both $\vec{x}$ and $\vec{z}$ [14].

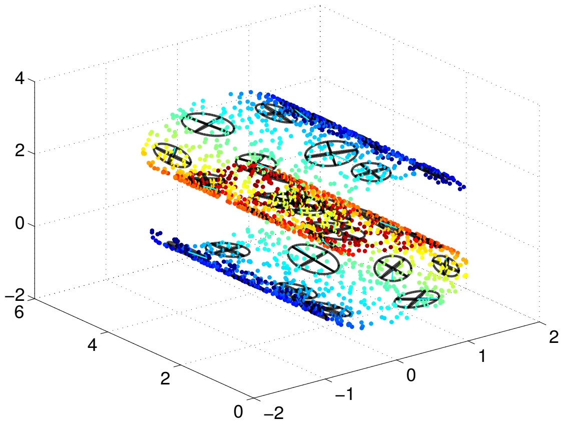

Fig. 7. Mixture of 32 factor analyzers learnt from data sampled randomly on the S-manifold. The covariances corresponding to $P(\vec{z}|\vec{x}, s)$ are overlaid on the plot. The mixture model captures the local covariance structure of the data over different regions of the manifold.

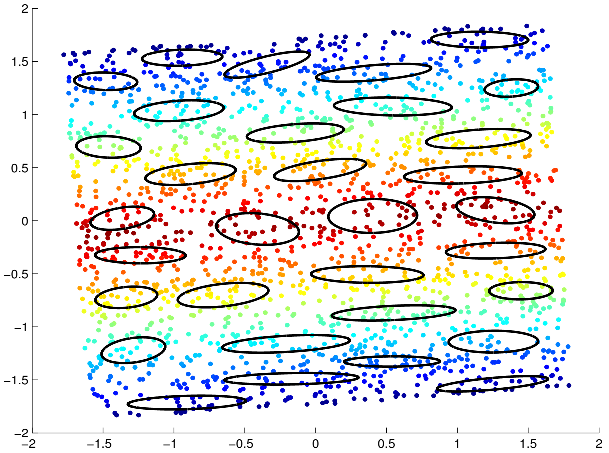

Fig. 8. The covariances of $P(\vec{x}|s)$. The uncertainty in the low dimensional state of a noisy high dimensional observation $\vec{z}$ is represented through the statistical model.

E. Integration within a Bayesian Filtering Framework

Bayes theorem provides an incremental and recursive, probabilistic method for combining high dimensional visual observations $\mathbf{Z}^k$ of a state $\mathbf{x}_k$,at time t_k, with a prior belief of the state $P(\mathbf{x}_{k-1})$. Features are extracted in real time from incoming natural imagery and are represented as a conditional probability distribution or likelihood $P(\mathbf{z} = z_k|\mathbf{x}_k)$, and the resultant combination is a revised posterior distribution on the state:

$$P(\mathbf{x}_k|\mathbf{z}^k) = \frac{P(\mathbf{z} = z_k|\mathbf{x}_k)P(\mathbf{x}_{k-1}|\mathbf{Z}^{k-1})}{P(z_k|\mathbf{Z}^{k-1})} \qquad (17)$$

where $\mathbf{Z}^k = \{z_k, \mathbf{Z}^{k-1}\}$ is the set of high dimensional visual observations from all nodes in the decentralized sensor network. The representation of the visual likelihoods as a Gaussian Mixture Model simplifies this update step into an algebraic computation of the product of two Gaussian Mixture Models [9].

V. Example Applications

The generative graphical model outlined earlier can be used off-line to compute the model parameters comprised of the means and covariance matrices of the constituent conditional Gaussian distributions. A rigorous approach would necessitate an extensive training set comprised of numerous high dimensional features from representative natural environments that are sampled under realistic ambient conditions.

A. UAV Acquired Imagery - Inference of Underlying Visual States

A random sample of about 7500 high dimensional points physically representing colors and textures of typical objects in the environment such as sky, trees, bush and grass was selected from a sequence of images acquired from a camera mounted on a UAV at the Marulan test facility operated by the Australian Center for Field Robotics. Texture information was included in the high dimensional input space by convolving 11×11 pixel patches with a bank of Gabor wavelets [11] at 2 scales and 2 orientations, resulting in an input space dimensionality of 847. Isomap was used to compute a low dimensional embedding of the training data and the intrinsic dimensionality of the manifold was estimated to be about 5.

The top two eigenvectors of the computed low dimensional embedding are shown in Figure 9. It is readily observed that image patches corresponding to bush and sheds are on the extreme left and right respectively, while grass and transitional patches are between these two extremes. The EM algorithm was used to learn the parameters of the generative model (Equations 1-3).

B. Inference

The learnt model was subsequently used to infer the low dimensional states (through appropriate marginalization of the joint probability densities) within a typical test image that was acquired in the same environment. Note that feature extraction is not performed within this image. Instead, the entire image is sub-divided into 11×11 patches, and the visual cues (color, texture) within each patch form the high dimensional observations.

The results of inference on the test image (Figure 9 top) in terms of the means of the inferred eigenvectors scaled to gray-scale limits (0-255) are shown in Figures 10-12.

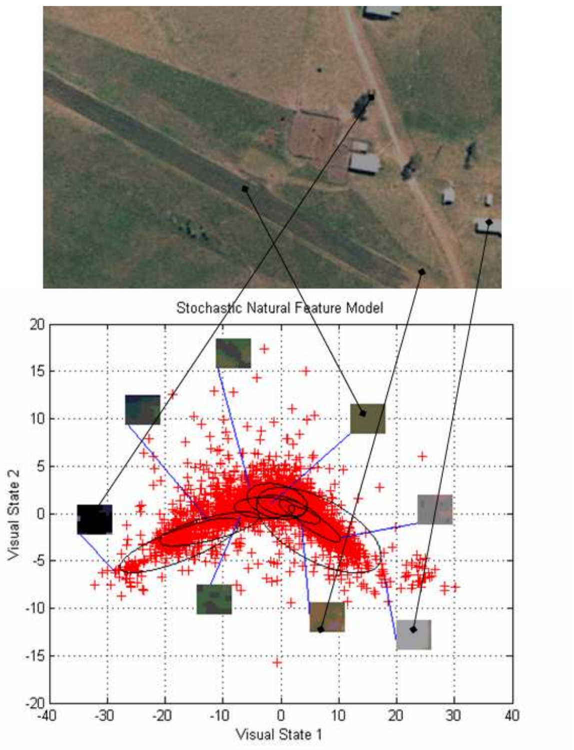

Fig. 9. Sample image acquired by the UAV (**top**) and low dimensional embedding of randomly sampled high dimensional image patches (**bottom**). The covariances Σ_s of each of the factor components are overlaid on the plot.

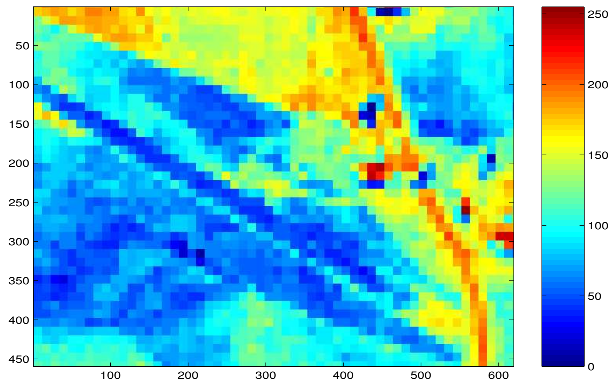

Fig. 10. Contour of the inferred means of the top eigenvector on each 14×14 image patch. This state enables a clear discrimination of the tracks and sheds (color coded red, range $\approx 200-250$) from all other visual groups in the image. This state is strongly correlated with the brightness of the image patches.

C. Comments

Each of the plots depicting the low dimensional states must be interpreted as a contour plot of the respective states in the image plane. It is important to realize that every patch consists of 847 correlated observations in the sensory space, while only a few uncorrelated states are sufficient to capture the similarities (or differences) between the patches after state inference.

The inferred low dimensional states are reasonable in that similar high dimensional image patches (such as those corresponding to bush, tracks or man made landmarks) are assigned similar low dimensional states, as is to be expected from a parametric model of neighborhood preserving manifold learning algorithms. Each inferred low dimensional state enables some degree of discrimination between important objects in the scene such as the tree, bush, tracks and the sky. The inferred low dimensional states provide an invariant descriptor of high dimensional visual observations, a property that is significant in the context of robust visual feature association.

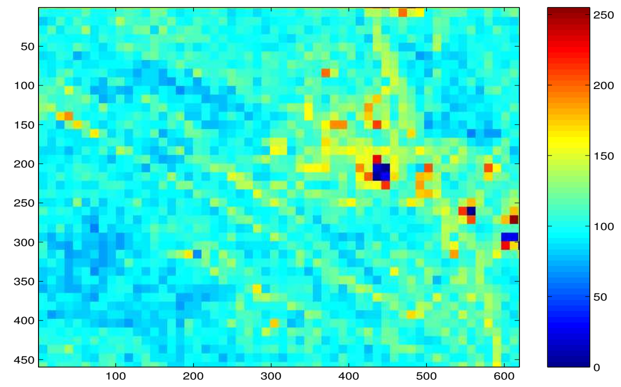

Fig. 11. Contour of inferred means of the second eigenvector. This state allows discrimination of the sheds (range $\approx 0-50$) from the grass and the tracks in the scene. This state exhibits strong correlations with the hue of the image patches.

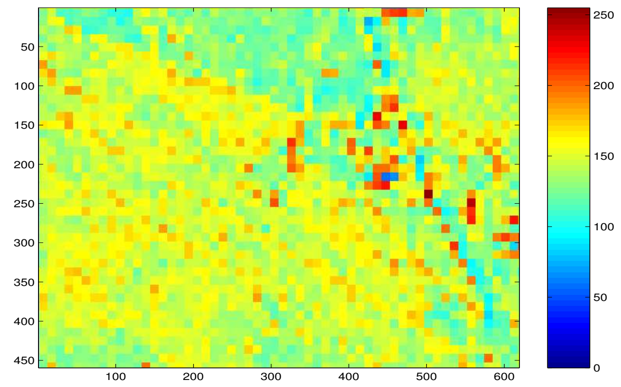

Fig. 12. Contour of inferred means of the third eigenvector. This state highlights intensity gradients in the image. This results from the fact that the Gabor wavelets essentially act as edge detecting operators.

D. Qualitative Comparison

The validity of the inferred visual states is qualitatively evaluated through a comparison with a k nearest neighbor approach. The 12 nearest neighbor of each test sample in the training set are computed, and the top two visual states of the test sample are evaluated as a weighted average of the states of the nearest neighbors. The weights are chosen to be inversely proportional to the high dimensional distances between the test and training samples.

A close examination of Figure 13 reveals that the statistical inference and k nearest neighbor approach place the test samples in a similar region of the manifold. The k nearest neighbor approach distributes the test samples compactly as compared to the stochastic estimate.

It is to be emphasized that this comparison is qualitative, and the nearest neighbor approach should not be viewed as ground truth as it has some inherent limitations [19]. The stochastic estimate is versatile as the inferred covariances quantify the uncertainty in the visual state estimation. Thus the stochastic visual state model renders itself to a convenient integration into a nonlinear filtering algorithm akin to conventional sensor models.

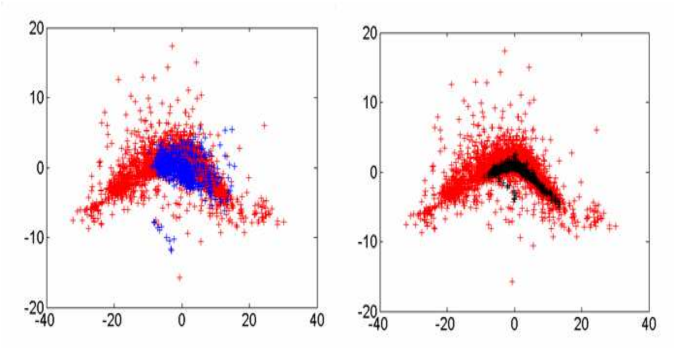

Fig. 13. Inferred low dimensional states (**left**) and nearest neighbor state estimate (**right**). The trained manifold is represented in red and corresponding test samples are overlaid on the manifold. The axes represent the top two visual state estimates from either approach.

VI. Conclusion

The combination of non-parametric manifold learning algorithms with statistical learning strategies leads to a consistent description of natural features in unstructured environments. While the entire learning procedure can be incorporated in the training phase of these models that is performed off-line, inference can be performed in real-time on any extracted features to compute likelihoods for the natural features as a Gaussian mixture model. Natural features can thus be fully integrated within existing non-Gaussian, non-linear filtering algorithms through the likelihood model so that tasks of estimation and data association are significantly enhanced through a combination of kinematic and visual states.

Acknowledgment

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

References

[1] T. Bailey, *Mobile Robot Localisation and Mapping in Extensive Outdoor Environments*. Ph.D. Dissertation, University of Sydney, 2002.

[2] S. Williams, *Efficient Solutions to Autonomous Navigation and Mapping Problems*. Ph.D. Dissertation, University of Sydney, 2001.

[3] E. Nettleton, *Decentralised Architectures for Tracking and Navigation with Multiple Flight Vehicles*. Ph.D. Dissertation, University of Sydney, 2003.

[4] M. Brand, M. Antone, and S. Teller, "Spectral Solution of Large Scale Extrinsic Camera Calibration as a Graph Embeddin Problem," in *European Conference on Computer Vision (ECCV)*, 2004.

[5] T. Lee, T. Wachtler, and T. J. Sejnowski, "Relations between the statistics of natural images and the response properties of cortical cells," *Vision Research*, vol. 42, pp. 2095–2103, 2002.

[6] Y. Karklin and M. Lewicki, "Learning higher order structure in natural images," *Computation in Neural Systems*, vol. 14, pp. 483–499, 2003.

[7] J. Tenenbaum, V. DeSilva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality redution," *Science*, vol. 290, pp. 2319–2323, 2000.

[8] 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.

[9] B. Upcroft, S. Kumar, M. Ridley, S. Ong, and H. Durrant-Whyte, "Fast Parameter Estimation for General Bayesian Filters in Robotics," in *Submitted to the Australian Conf. on Robotics and Automation*, 2004.

[10] D. J. MacKay, *Information Theory, Learning and Inference*. Cambridge University Press, 2003.

[11] D. J. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *Journal of the Optical Society of America*, vol. 4, pp. 2379–2394, 1987.

[12] B. Scholkopf, A. J. Smola, and K. R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, pp. 1299–1319, 1998.

[13] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," University of Chicago, Department of Computer Science, Tech. Rep., 2002.

[14] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by Locally Linear Embedding," *Science*, vol. 290, pp. 2323–2326, 2000.

[15] I. Foster, *Designing and Building Parallel Programs*. Addison Wesley, 1995.

[16] E. W. Dijkstra, "A note on two problems in connexion to graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.

[17] T. Cox and M. Cox, *Multidimensional Scaling*. Chapman and Hall, 1994.

[18] Z. Ghahramani and G. E. Hinton, "The EM algorithm for mixtures of factor analyzers," Department of Computer Science, University of Toronto CRG-TR-96-1, Tech. Rep., 1996.

[19] T. Hastie, R. Tibshirani, and J. Friedman, *Elements of Statistical Learning*. Springer Verlag, 2001.