The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Towards autonomous habitat classification using Gaussian Mixture Models

Daniel M. Steinberg, Stefan B. Williams, Oscar Pizarro & Michael V. Jakuba
Australian Centre for Field Robotics (ACFR)
University of Sydney,
NSW 2006, Australia
Email: d.steinberg@acfr.usyd.edu.au

Abstract— **Robotic agents that can explore and sample in a completely unsupervised fashion could greatly increase the amount of scientific data gathered in dangerous and inaccessible environments. Our application is imaging the benthos using an autonomous underwater vehicle with limited communication to surface craft. Robotic exploration of this nature demands in situ data analysis. To this end, this paper presents results of using a Gaussian Mixture Model (GMM), a Hidden Markov Model (HMM) filter, an Infinite Gaussian Mixture Model (IGMM) and a Variation Dirichlet Process model (VDP) for the classification of benthic habitats. All of the models are trained using unsupervised methods. Furthermore, the IGMM and VDP are trained without knowing the the number of classes in the dataset. It was found that the sequential information the HMM filter provides to the classification process adds lag to the habitat boundary estimates, reducing the classification accuracy. The VDP proved to be the most accurate classifier of the four tested, and also one of the fastest to train. We conclude that the VDP is a powerful model for entirely autonomous labelling of benthic datasets.**

I. Introduction

Autonomous underwater vehicles (AUVs) are often limited to performing fixed, pre-planned surveys [1], [2]. When used in this manner, the AUV may not completely capture processes of interest. This is exacerbated if there is little a priori information on the areas to be surveyed during the mission planning phase. To overcome this limitation, AUVs, and other autonomous agents, should be able to *adapt* their mission plans to suit scientific goals.

Adaptive sampling requires real-time processing and interpretation of data that is being gathered; so subsequent actions can be taken that are likely to increase the value of the resulting data products. One way of interpreting data is to aggregate, or cluster, it into classes that have a semantic categorical meaning, such as different habitats.

Often prior knowledge of the process of interest is required before the start of an adaptive mission in order to train the probabilistic models used for inference [3], [4]. We aim to have the AUV enter an environment with little or no prior training, and in a completely unsupervised manner, form its own representation of the environment. This requires the AUV to autonomously identify how many habitats are in the data it has gathered, and then classify the data accordingly.

Rather than using discriminative regression models for classification, such as logistic or probit regression, multinomial logit models, Support Vector Machines and Gaussian Process Classifiers, we have chosen to use generative Gaussian Mixture Model (GMM) based density estimators. This class of probabilistic model allows us to exploit aspects of the structure of the data that may not be possible with regression based methods. For instance, the Hidden Markov Model (HMM) allows us to exploit sequential correlations in the data, and the Infinite Gaussian Mixture Model (IGMM) and Variational Dirichlet Process model (VDP) can automatically infer the number of classes present.

The sequential nature of data captured by an AUV suggests that correlations will exist between the benthic habitats in subsequent images. We wish to determine whether accounting for these correlations improves the performance of habitat classification. HMMs have been used to provide contextual prior information for visually-based place and object recognition algorithms, with significant improvements in classification error rates [5]. HMMs have also previously been used to estimate the state of the environment surrounding an AUV in order to trigger some adaptive behaviour. Applications include the monitoring of various chemical features in a volume of water. The HMM infers whether the AUV is 'in' or 'out' of the feature. This can then trigger a sample of the chemical feature to be taken [6], or for the AUV to track the boundary of a feature such as a chemical spill [7].

This paper presents an investigation into the GMM and three variants – the HMM, the IGMM and the VDP – for the classification of benthic habitats. Data gathered by a stereo camera on the Seabed class AUV, *Sirius* [8], at a recent deployment in Scott Reef in Western Australia is used to illustrate the performance of these models. We have used unsupervised methods to train these models to demonstrate learning with minimal interaction from a human user. Furthermore we show that the IGMM and VDP can be learned entirely autonomously – without knowledge of the number of habitats in an environment.

The remainder of this paper is organised as follows. Sections II – V introduce the GMM, HMM, IGMM and VDP and their training and classification algorithms. Section VI discusses the visual rugosity habitat descriptor used and the Scott Reef dataset. Results are presented comparing the GMM, HMM, IGMM and VDP in Section VII. Finally, Sections VIII and IX present a discussion, the conclusion of this study, and outline our on-going work in this area.

II. Gaussian Mixture Model

An important assumption made in this paper is that the observable data is distinctly multimodal, and can be represented as a Gaussian Mixture Model – which are simply a linear sum of Gaussian distributions [9]. GMMs represent a continuous distribution of an observation $\mathbf{y}_i$ that is dependent on a discrete, unobservable (or latent), variable $\mathbf{z}_i$. The observations are assumed to be independently and identically distributed (i.i.d.). In this case the latent variable, $\mathbf{z}_i$, is the type of habitat in the image (also referred to as the label). The observation, $\mathbf{y}_i$, is a descriptor, or a vector of descriptors, extracted from the image such as the rugosity descriptor presented in Section VI.

The latent can be represented by a 1-of-K vector, where K is the number of possible habitats that can be observed. For example, if $K = 4$, and $z_{i,3} = 1$ then $\mathbf{z}_i = \{0, 0, 1, 0\}$. Generally each habitat type, k, will correspond to one Gaussian in the mixture. The latent variable has the following marginal distribution:

$$p(\mathbf{z}_i) = p(\{z_{i,1} \dots z_{i,K}\}) = \{\pi_1, \dots, \pi_K\}, \tag{1}$$

where π_k are the mixing coefficients, or weights given to a Gaussian in the mixture. π_k must be in the range $[0, 1]$ and $\sum_K \pi_k = 1$. The marginal distribution of the observation, $p(\mathbf{y}_i)$, is a linear sum of Gaussians,

$$p(\mathbf{y}_i) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{y}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\,. \tag{2}$$

Here the parameter $\boldsymbol{\mu}_k$ is the mean vector and $\boldsymbol{\Sigma}_k$ is the covariance matrix, of the k^{th} multivariate Gaussian. From (1) and (2) we can see the conditional distribution of the mixture is,

$$p(\mathbf{y}_i | z_{i,k} = 1) = \mathcal{N}(\mathbf{y}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\,. \tag{3}$$

This is the likelihood of observing $\mathbf{y}_i$, given each Gaussian distribution corresponds to a class. Using this fact it is possible to find the probability of a mixture component generating an observation, or the probability of a habitat generating a type of image. The graphical model of the interaction between the observed and latent variables is shown in Figure 1.

A. Training

Classifying habitats using a mixture of Gaussians requires that the parameters of the distributions are learned. Specifically the means and covariances of the Gaussians ($\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$), as well as the Gaussian mixture weights (π_k). This is achieved using the Expectation-Maximisation (EM) algorithm, which learns the expectations of these parameters with respect to the data. Naturally the training set used has to be representative of the environment to be classified.

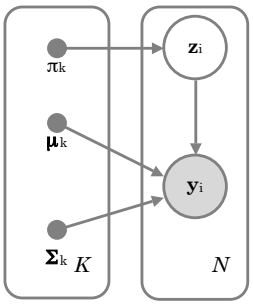


Fig. 1: GMM graphical model. $\mathbf{z}_i$ is the latent variable or class, and $\mathbf{y}_i$ is the observation. The filled points refer to point estimates (Dirac delta functions) of the parameters, rather than distributions with associated uncertainty.

B. Classification Algorithm

Given an observation $\mathbf{y}_i$, the probability of an image being of a specific habitat, $p(z_{i,k} = 1|\mathbf{y}_i)$, can be calculated using Bayes' rule and (1) – (3),

$$\begin{aligned} p(z_{i,k} = 1|\mathbf{y}_i) &= \frac{p(z_{i,k} = 1)\, p(\mathbf{y}_i | z_{i,k} = 1)}{p(\mathbf{y}_i)} \\ &= \frac{\pi_k \mathcal{N}(\mathbf{y}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\big(\mathbf{y}_i | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\big)}. \end{aligned} \tag{4}$$

This is also known as the responsibility of $z_{i,k}$ for explaining $\mathbf{y}_i$ [9].

III. Hidden Markov Model

The scale of a single image acquired by the AUV is generally far less than the scale at which habitats change. Coupled with the fact that an AUV follows a trajectory while acquiring images of the benthos suggests that there will be a sequential correlation between habitats in the images. The i.i.d. assumption made for the Gaussians Mixture Model does not allow this sequential information to be exploited for classification purposes.

A Hidden Markov Model (HMM) may be viewed as a generalisation of the GMM that relaxes this i.i.d. assumption. That is, subsequent samples are no longer assumed independent [9], [10]. Now the latent variables are correlated with each other as shown in Figure 2. Because of d-separation in this first-order model, the posterior estimate of habitat label $\mathbf{z}_n$ is only dependent on $\mathbf{z}_{n-1}$ (also $\mathbf{z}_{n+1}$ if we wish to use a smoother).

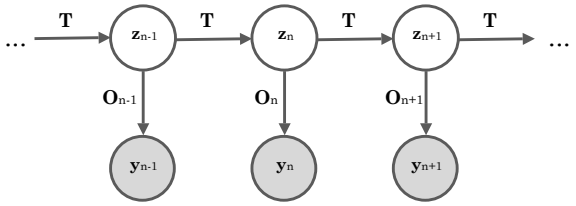


Fig. 2: Graphical representation of a Hidden Markov Model

The dependence of habitat label $\mathbf{z}_n$ on label $\mathbf{z}_{n-1}$ is represented by a *transition* matrix, $\mathbf{T}$, which is a table

of conditional probabilities [11]. For instance, the probability of transition from label k to label k^* is $\mathbf{T}_{kk^*} = p(z_n = k|z_{n-1} = k^*)$, or generally

$$\mathbf{T} = p(\mathbf{z}_n|\mathbf{z}_{n-1})\,. \tag{5}$$

In this instance observations are mixtures of Gaussians, so the observation matrix, $\mathbf{O}_n = p(\mathbf{y}_n|\mathbf{z}_n)$, is the likelihood distribution given by (3). The observations in a HMM do not always need to be Gaussian mixtures – other common distributions are binomial and multinomial.

A. Training

As with the GMM classifier, the HMM parameters, $\boldsymbol{\mu}_k$, $\boldsymbol{\Sigma}_k$, π_k and $\mathbf{T}$, can be learned by using the EM algorithm.

Unlike GMM training, the HMM EM algorithm relies upon having a sequence to learn valid transition probabilities. Because of this, it was found that the HMM variant of the EM algorithm needed substantially more training data than the GMM variant.

B. Filtering (forward) Algorithm

To classify each image, a discrete filtering or forward recursive algorithm was used. The forward/filter algorithm is a recursive Bayes' filter with predict and observe/update stages. The posterior estimate of the habitat is then

$$p(\mathbf{z}_n|\mathbf{y}_n) \propto p(\mathbf{y}_n|\mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} p(\mathbf{z}_n|\mathbf{z}_{n-1})\, p(\mathbf{z}_{n-1}|\mathbf{y}_{n-1})\,. \tag{6}$$

Equation (6) can be represented in matrix form [11],

$$\mathbf{f}_n \propto \mathbf{O}_n * \mathbf{T}^\top \mathbf{f}_{n-1}, \tag{7}$$

where $\mathbf{f}$ is a forward message (or posterior distribution), and '$*$' is an element-wise multiplication[1].

Two other major algorithms exist for HMMs; the forward-backward algorithm which is a smoothing algorithm, and the Viterbi algorithm which is a maximum likelihood algorithm. These algorithms may yield better classification results, however they are not realtime algorithms, and would be of most use in a post-processing scenario. As such, we have focused on the forward/filter algorithm.

IV. Infinite Gaussian Mixture Model

The GMM and HMM are examples of parametric models that require the number of habitats to be specified prior to training. For autonomous exploration this may be unknown. Conversely, non-parametric models are largely inferred from structures that are present in the data itself. The general form of the non-parametric model used in this work is the Dirichlet Process Mixture Model (DPMM) [12], [13]. It has the appealing property that the number of mixtures, or clusters that are present in a dataset does not need to be known a priori. DPMMs assume there are an infinite number of clusters, but only a few are actually present in a given dataset, the number of which is dependant on an aggregation parameter, α. This parameter is inferred from the data, and generally only allows the number of mixtures to increase as more data is observed.

The form of DPMM presented in this section the *univariate*[2] Infinite Gaussian Mixture Model (IGMM) [14]. This model uses the Polya urn scheme for representing the DPMM [15]. It is simply a Bayesian formulation of a finite Gaussian mixture model that has been generalised to have an infinite number of mixture components,

$$p(y_i) = \sum_{k}^{\infty} \pi_k \mathcal{N}\big(y_i|\mu_k, s_k^{-1}\big)\,. \tag{8}$$

where s_k is the precision, or inverse variance. These are sampled from a conditionally conjugate Normal-Gamma prior distribution with hyperparameters $\Theta = \{\beta, w, \lambda, r\}$,

$$\mu_k \sim \mathcal{N}\big(\lambda, r^{-1}\big) \quad \text{and} \quad s_k \sim \mathcal{G}\big(\beta, w^{-1}\big)\,.$$

These hyperparameters also have distributions from which they are drawn. Most are conjugate except for β, which requires Adaptive Rejection Sampling [16]. We can also sample the mixture weights according to a Dirichlet distribution [17] once we know the number of mixture components, K, with one or more observations assigned to them ($n_k > 0$),

$$\boldsymbol{\pi} \;\sim\; Dir\left(n_1 + \alpha/K, \ldots, n_K + \alpha/K\right). \tag{9}$$

It is important to note that the weights are integrated out in the IGMM, so (9) is not explicitly in the formulation [14]. The hierarchy of this model is illustrated in Figure 3.

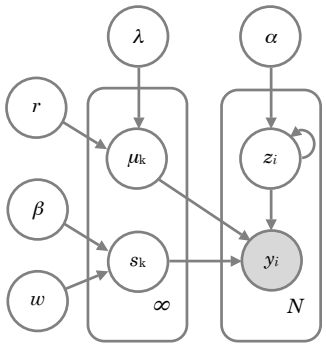


Fig. 3: Graphical model of the IGMM (adapted from [14], [17]). The hyperparameters are α and $\Theta = \{\beta, w, \lambda, r\}$. Each mixture component has its own parameters, $\{\mu_k, s_k^{-1}\}$, drawn from the hyperparameter space.

A. Training

Training an IGMM is done by Gibbs sampling [14] the class or mixture labels, z_i, in the following manner,

$$p(z_i = k|\mathbf{z}_{-i}, \alpha, \mu_k, s_k) \propto \frac{n_{-i,k}}{n-1-\alpha} \mathcal{N}\big(y_i|\mu_k, s_k^{-1}\big)\,, \tag{10}$$

$$p(z_i \neq k\ \forall k|\mathbf{z}_{-i}, \alpha, \Theta) \propto \frac{\alpha}{n-1-\alpha} \times \int \mathcal{N}\big(y_i|\mu_k, s_k^{-1}\big)\, p\big(\mu_k, s_k^{-1}|\Theta\big)\, \mathrm{d}\mu_k\, \mathrm{d}s_k^{-1}. \tag{11}$$

[1] This element-wise multiplication can be avoided by making the observation a matrix specified by $diag(\mathbf{O}_n)$.

[2] A multivariate formulation of the IGMM also exists [14].

Here z_i indicates the assignments[3] of observations, y_i to an *existing* mixture component k. The subscript $-i$ means all observations except i, so $n_{-i,k}$ means the number of observations, not including i, that belong to mixture component k. The probability of an observation belonging to an existing Gaussian mixture component is given by (10). The probability of the observation belonging to a new component is given by (11). Unfortunately the integral in (11) is intractable because the prior, $p\big(\mu_k, s_k^{-1}|\Theta\big)$, is only conditionally conjugate (see [14], [18] for a discussion). To overcome this, the prior is sampled to generate an estimate of the probability of generating a new class. Once a new observation is classified, the model hyperparameters and parameters are updated. Specifically, Algorithm 8 from [19] is used – for more detail see [14] and for sampling α see [20].

The Gibbs sampler is run until apparent convergence, and since we are using the IGMM for classification, only *one* sample of the IGMM is used as the classification model. A few low weight mixtures may still remain in the chosen sample, so one hard-assignment EM iteration is run over the observation labels. This makes the resulting density more reasonable for classification [21].

B. Classification Algorithm

For this paper, the IGMM uses the same classification algorithm as the GMM, given in (4). However we may also use (10) and (11) to classify incoming observations during a mission, while also updating the hierarchy in Figure 3. This assigns a probability to an observation belonging to a previously unseen habitat classes, and continues the learning process through the entire mission [21].

V. Variational Dirichlet Process Mixture Model

A second non-parametric model is used in this paper, and is a mean-field variational approximation to the 'stick breaking' representation of a DPMM [22]. It is called a Variational Dirichlet Process (VDP) and can be used for all exponential family mixture distributions [23]. The Gaussian mixture formulation of the model is similar to that of the IGMM,

$$p(\mathbf{y}_i) = \sum_k^{\infty} \pi_k \mathcal{N}\big(\mathbf{y}_i|\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k^{-1}\big)\,. \tag{12}$$

For this model we again use precision rather than variance ($\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$), and the latent class label, $\mathbf{z}_i$, is once again a 1-of-K vector. The mean and precision have a fully conjugate Gaussian-Wishart prior distribution with hyperparameters $\{\mathbf{m}_0, \beta_0, \mathbf{W}_0, \nu_0\}$,

$$\begin{aligned} p(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k) &= p(\boldsymbol{\mu}_k|\boldsymbol{\Lambda}_k)\, p(\boldsymbol{\Lambda}_k) \\ &= \mathcal{N}\big(\boldsymbol{\mu}_k|\mathbf{m}_0, (\beta_0\boldsymbol{\Lambda}_k)^{-1}\big)\, \mathcal{W}(\boldsymbol{\Lambda}_k|\mathbf{W}_0, \nu_0)\,. \end{aligned} \tag{13}$$

[3]This does not necessarily need to be a 1-of-K representation. In this case z_i takes on an integer value.

The mixture weights, $p(\mathbf{z}_i) = \{\pi_k \dots \pi_\infty\}$, have a 'stick breaking' prior with an infinite collection of 'stick lengths', $V = \{v_k \dots v_\infty\}$,

$$\pi_k(V) = v_k \prod_{j=1}^{k-1}(1 - v_j), \tag{14}$$

where $v_k \sim \boldsymbol{\beta}(1, \alpha)$.

The variational model makes the approximation that (13) and (14) can be represented by factored distributions, $q(\cdot)$. There is a factor distribution for each mixture component with parameters $\{\mathbf{m}_k, \beta_k, \mathbf{W}_k, \nu_k\}_{k=1}^T$,

$$\begin{aligned} p(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k) &\approx q(\boldsymbol{\mu}_k|\boldsymbol{\Lambda}_k)\, q(\boldsymbol{\Lambda}_k) \\ &= \mathcal{N}\big(\boldsymbol{\mu}_k|\mathbf{m}_k, (\beta_k\boldsymbol{\Lambda}_k)^{-1}\big)\, \mathcal{W}(\boldsymbol{\Lambda}_k|\mathbf{W}_k, \nu_k)\,, \end{aligned} \tag{15}$$

and also one for each mixture weight stick length with parameters $\{\phi_{k,1}, \phi_{k,2}\}_{k=1}^T$,

$$p(v_k|\alpha) \approx q(v_k|\boldsymbol{\phi}_k) = \boldsymbol{\beta}(\phi_{k,1}, \phi_{k,2})\,, \tag{16}$$

up to some mixture truncation level $k = \{1 \dots T\}$. Typically $T > K$ – the extraneous mixtures have negligible weights and naturally revert to their prior values. The graphical model of (12), (13) and (14) is shown in Figure 4.

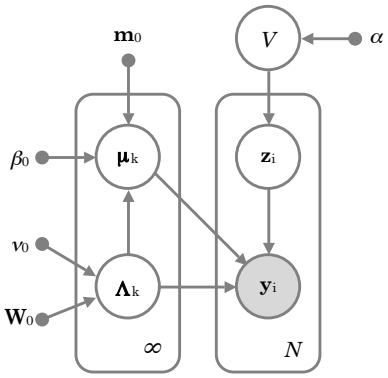


Fig. 4: VDP graphical model (not showing the approximation). Point estimates are made of the hyperparameters, unlike the IGMM, which specfies two hierarchical levels of priors.

Point estimates (or Dirac delta functions) of the hyperparameters are made in this model, rather than specifying a distribution over the hyperparmeters. The hierarchy can be extended to account for a distribution over α [24], however it has been shown that the aggregation parameter only has a weak influence on the learned mixture density for mean-field variational approximations [25].

A. Training

The VDP is trained by a variational Bayes Expectation-Maximisation (VBEM) algorithm, detailed in [23]. An accelerated VBEM algorithm that compresses the training data using kd-trees was also presented by the authors, however we have not yet implemented the accelerated version.

Unlike other variational Bayesian methods for mixture models ([9], [24], [26]), the VDP is nested with respect to the truncation level, T, [23]. This means that the optimal number of mixtures found is not a function of the chosen starting truncation level. This allows the truncation level to be increased *during* learning and still have VBEM converge to an optimal number of mixture components. If the model was not nested, it would be necessary to restart the learning process for each truncation level to make sure that the optimum number of components is found.

Training of the nested VDP is performed by firstly starting with a specified truncation level (we start with $T = 1$). VBEM is then run until it converges, after which the mixture components are split in a direction perpendicular to their principal components. The split that leads to the maximal reduction of free energy (analogous to log-likelihood) is chosen and also tested for convergence; i.e. if it improves the free energy by more than a convergence threshold it is accepted, and VBEM is resumed. If the split is not accepted the algorithm has converged to a learned distribution. We then also remove mixture components that have negligible weights (and consequently are indistinguishable from their prior values).

B. Classification Algorithm

The VDP approximation uses a very similar classification algorithm as the GMM –

$$q(z_{i,k} = 1|\mathbf{y}_i) = \frac{\exp(S_{i,k})}{\sum_{j=1}^{K} \exp(S_{i,j})}. \tag{17}$$

Again $q(\cdot)$ is an approximating distribution (factored over i), and

$$S_{i,k} = \mathbb{E}_V \left[\log p(\pi_k|V)\right] + \mathbb{E}_{\boldsymbol{\mu}_k,\boldsymbol{\Lambda}_k} \left[\log \mathcal{N}(\mathbf{y}_i|\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k)\right]. \tag{18}$$

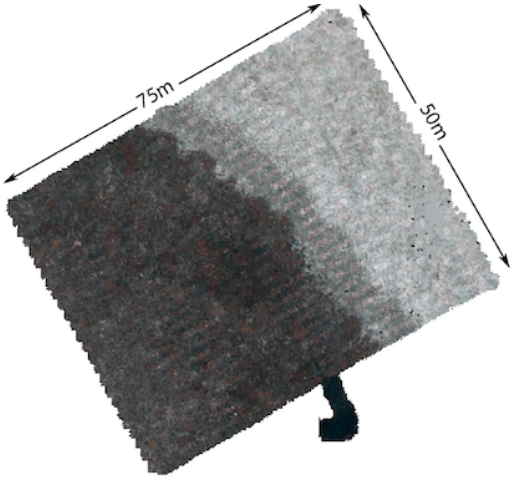
However these expectations are actually functions of the hyperparameters, $\{\phi_{k,1}, \phi_{k,2}, \mathbf{m}_k, \beta_k, \mathbf{W}_k, \nu_k\}$, rather than the parameters themselves, $\{\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k\}$, as is the case with EM.

VI. Visual Habitat Descriptors and Dataset

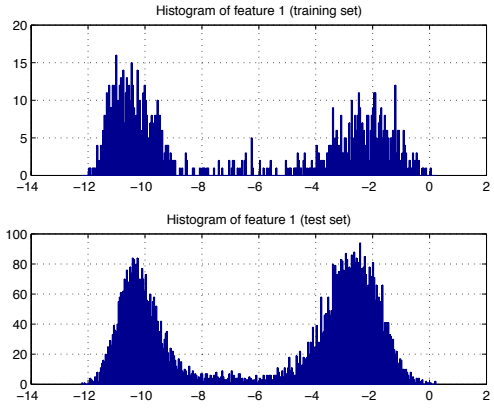
Stereo camera data gathered during a recent AUV deployment at Scott Reef in Western Australia is used to illustrate the performance of the models. The dive chosen was used to create a full-coverage 50 m by 75 m photomosaic of the benthos. It features clear transitions between dense coral cover, barren sand and an intermediate, partially populated substrate class, shown in Figure 5a. All of the data is georeferenced by a visually based extended information-form SLAM filter [27].

Although there are many choices of visual features [28], the focus of the paper is on the machinery capable of classifying observations into habitats without human intervention.

The habitat descriptor tested is a bathymetric rugosity index derived from the 3D stereo imagery [29]. The georeferenced stereo imagery provided by the AUV can be used to generate fine-scale bathymetric reconstructions in the form of 3D triangular meshes [30]. From these meshes, it is possible to derive multi-scale terrain complexity measures of rugosity. These measures proved to be very effective at discriminating between habitats. Rugosity, y_{rug}, is essentially the ratio between the surface (draped) area, and a plane that fits the surface. A flat surface has a rugosity of 1, while more complex structures have higher values.



(a)



(b)

Fig. 5: The Scott Reef Dataset. (a) Image reconstruction of the dense survey, consisting of 50 parallel tracklines, each 75 m long and spaced one meter apart. (b) Histograms of the rugosity feature for the training and test data.

Terrain complexity measures, such as rugosity, are commonly used to describe habitats by marine ecologists since they captures habitat complexity, which is known to correlate with biodiversity [31].

Histograms of rugosity for a single habitat tend to be distributed in a log-normal fashion in the range $[1, \infty)$, so we apply the transformation:

$$y = C \cdot \log(y_{rug} - 1), \tag{19}$$

which makes the habitat data have a Gaussian shape in the range of $(-\infty, \infty)$. C is an arbitrary scaling factor applied to increase the over-all variance of the resulting density. This

descriptor picks out the transition between the reef and sand habitats distinctly, which results in good separation between the modes that describe these habitats, Figure 5b.

We used the first 1000 stereo pairs as training data (the first few transects), and then the rest of the dataset serves as the test data (approx. 8000 stereo pairs), the histograms of this data is shown in Figure 5b.

The reason for using the first few transects as training data is we envisage a possible situation where an AUV will in fact complete a preliminary, pre-planned survey. The data from this will be used to autonomously train classifiers. Then the AUV will enter an adaptive phase while still deployed, where it will visit and classify areas which are deemed to increase the scientific value of the mission.

VII. Results

The class labels appear as coloured markers in Figure 6, each marker corresponding to a stereo image pair. If there is a high probability associated with an observation belonging to a particular class, its marker will be coloured accordingly. We have also plotted the posterior entropy of an observation belonging to a class,

$$H(\mathbf{z}_i|\mathbf{y}_i) = -\sum_K p(\mathbf{z}_i|\mathbf{y}_i)\log p(\mathbf{z}_i|\mathbf{y}_i)\,. \tag{20}$$

This is represented as the size of a marker in our plots, a higher entropy (or less-certain classification) being signified by a larger marker. Naturally there is no entropy associated with our hand-labelled ground truth (Figure 6e), so all of its markers are a uniform size.

In Figure 6, K was set to three for the GMM and HMM EM algorithms. The VDP also could distinguish three distinct habitat classes, and for this run the IGMM returned three classes. Blue corresponds to the barren sand habitat, green to the intermediate/mixture habitat, and red to the reef habitat. It is somewhat hard to gauge which classifier is most representative of the hand-labelled ground truth from these plots, but it is clear that the HMM filter has the lowest entropies associated with its classifications. This is to be expected since additional state transition information is being incorporated into the model which is not present in the other mixture models. However, there is some lag associated with the boundary estimate between each transect, especially between the barren sand and mixture classes, where the transition boundary is less clearly defined.

TABLE I: Model classification performance.

Models	Correctly Classified (%)	Relative to GMM (%)
GMM	87.88	0
HMM	86.47	−1.41
IGMM	89.04	+1.16
VDP	**90.17**	**+2.29**

Quantitative results are presented in Table I and II. The VDP and IGMM have higher classification accuracy than the GMM and HMM when compared to the ground truth. The VDP yielding the most precise result, and the HMM the worst. This dataset is quite separable, so generally classification performance is high, and it is arguable that the HMM filtering algorithm is unnecessary. In fact the HMM also seems to have re-enforced the GMM classification errors in the reef habitat (lower left corner in Figure 6b).

TABLE II: Confusion Matrices – each column is normalised by the population of each class in the ground truth dataset.

		Truth		
	Classes	Barren	Mix	Reef
GMM	Barren	0.9977	0.4957	0
	Mix	0.0023	0.3894	0.0912
	Reef	0	0.1150	0.9088
HMM	Barren	0.9977	0.4089	0
	Mix	0.0023	0.4946	0.1373
	Reef	0	0.0965	0.8627
IGMM	Barren	0.9966	0.4957	0
	Mix	0.0034	0.3807	0.0685
	Reef	0	0.1236	0.9315
VDP	Barren	0.9962	0.4772	0
	Mix	0.0038	0.3861	0.0493
	Reef	0	0.1367	0.9507

The normalised confusion matrices presented in Table II show that the most confusion is in the classification of the mixture class. The HMM performs the best at classifying this class, but at the cost of inconsistent habitat boundary estimates, and reinforcing the mis-classification of the large reef class. The VDP outperforms the others in classifying the reef habitat, which contributes greatly to its overall accuracy for this dataset.

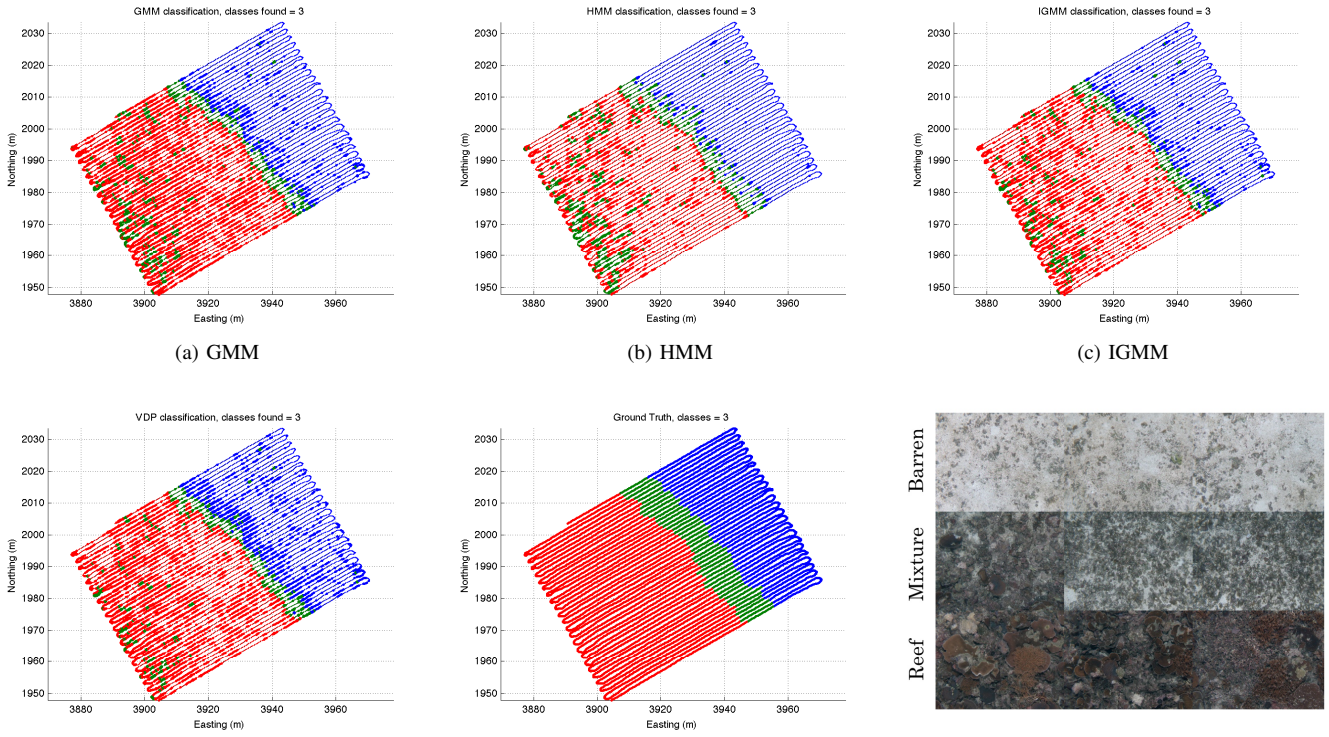
TABLE III: Example Gibbs Sampling runs of the IGMM.

Run No.	Classes	Classes ($\pi \geq 2\%$)	Log-Likelihood	Iter.
1	4	4	-2055	8
2	7	5	-2037	26
3	3	3	-2059	34
4	4	3	-2057	27
5	4	4	-2059	13
6	9	6	-2049	41

Unfortunately, the IGMM result presented here is not necessarily representative of the IGMM's general performance. This is because Gibbs sampling is a stochastic process, so it leads to different learned density parameters every run. Table III presents six training runs of the IGMM over the same training data. The number of mixtures, likelihoods, and iterations until convergence vary significantly. This makes it hard to compare Gibbs sampling to deterministic learning methods such as EM and Variational Bayes, which will converge to the same results given the same starting conditions and training data.

VIII. Discussion

The structure, and the performance, of each model presented in this paper is tightly coupled to its training algorithm. Training the GMM and HMM using EM is fast and only requires a few iterations before converging. Training the IGMM using Gibbs sampling requires considerably more time. Each sweep is slower than a corresponding EM iteration, and convergence is fairly arbitrary. VBEM is fast since it is similar to EM, and also has the flexibility of



(a) GMM (b) HMM (c) IGMM

(d) VDP (e) Hand-labelled ground truth (f) Sample images from the ground truth

Fig. 6: Classification results for the test data in the Scott Reef dataset. The colour of the markers represents the classified habitat type of an image. The size of the markers represents the classification entropy (larger is more entropy).

Gibbs sampling in that the number of classes does not need to be known a priori. Furthermore, VBEM is a Bayesian probabilistic method, so it is not susceptible to over-fitting like EM.

Typically training the IGMM using Gibbs sampling is for probability density estimation and not classification [14]. This is because multiple samples of the IGMM's parameters are combined linearly to then form the predictive distribution. There is no clear way to do this for classification, since we need to have only one mixture component correspond to each habitat class. We are then limited to only using one sample of the IGMM for classification, with no guarantees that this sample will be appropriate for classification. Removing low weight mixtures by performing a hard EM assignment step somewhat improves the IGMM for classification, but the mean-field variational approach seems more appropriate for classification using DPMMs. A non-linear, generative, classification model based upon DPMMs exists [32] which is also more appropriate for classification tasks, however it requires at least partially labelled data which is not appropriate for our application. See [21] for a more thorough discussion on the shortcomings of the IGMM for classification.

The results raise the question of whether a temporal filtering algorithm is appropriate for adaptive behaviour. Lag is introduced into the estimates of habitat boundaries that could be good triggers for adaptive behaviours. Furthermore, classification errors may be reinforced, as in the case of the reef habitat. This lag issue may be partially resolved by running a smoothing algorithm, e.g. the forward-backward algorithm, over a finite window as opposed to the filter algorithm used in this paper. Running the forward-backward algorithm over all of the data is not suited to online tasks, but is appropriate for post-processing tasks.

The need for smoothing can be mostly mitigated by using descriptors that effectively discriminate various habitats, such as the rugosity descriptor. Using a higher dimensional vector of good descriptors as the observation data would improve these results again. For a good discussion on smoothing, see [33].

A limiting factor with training the GMM and HMM using EM is that it assumes the number of habitats, K, is known prior to training. The IGMM (Gibbs sampling) and VDP (VBEM) do not require this information, and what is more they can be modified to recognise new, unseen habitats online as more information is gathered in the environment [21], [34].

IX. Conclusions and Future Work

In this paper we have shown that relatively simple classification models can be applied successfully to benthic classification if the habitat descriptor used is discriminating. Taking into account sequential correlations in this data, in the form of a Hidden Markov Model, does not improve classification results in this instance, and will introduce lag into class boundary estimates.

The non-parametric Dirichlet process mixture model derivatives; the Infinite Gaussian mixture model and Variational Dirichlet Process model are both capable of autonomously learning the number of classes in the benthic dataset presented. However, the VDP proved to be superior in that its learning algorithm is fast and deterministic like Expectation Maximisation, and although an approximation to a true DPMM, it still retains the inherent capability of being able to represent an infinite number of classes.

As future work we will look into incremental, online learning techniques for the VDP so not all of the habitats need to be present within the preliminary survey in order recognise new habitats, and cue adaptive behaviours. We are also in the process of using the VDP as a 'supervisor' for training Gaussian Process (GP) regression and classification models [35]. Combining the VDP with GPs in this way will allow spatial maps of benthic environments to be autonomously learned. These probabilistic maps will could then provide additional adaptive capabilities to an AUV.

ACKNOWLEDGMENTS

This work is supported by the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC), the New South Wales State Government and the Integrated Marine Observing System (IMOS) through the DIISR National Collaborative Research Infrastructure Scheme. We would like to thank Ariell Friedman for his rugosity code, and the Australian Institute for Marine Science for making ship time available. The crew of the R/V Solander was instrumental in deploying and recovering the AUV. We also acknowledge Duncan Mercer, George Powell, Matthew Johnson-Roberson, Ian Mahon, Stephen Barkby, Ritesh Lal, Paul Rigby, Jeremy Randle, Bruce Crundwell and the late Alan Trinder, who have contributed to the development and operation of the AUV.

REFERENCES

[1] D. R. Yoerger, M. Jakuba, A. M. Bradley, and B. Bingham, "Techniques for deep sea near bottom survey using an autonomous underwater vehicle," *The International Journal of Robotics Research*, vol. 26, no. 1, pp. 41–54, 2007.

[2] P. Rigby, "Autonomous spatial analysis using Gaussian process models," Ph.D. dissertation, The University of Sydney, 2008.

[3] D. R. Thompson and D. Wettergreen, "Intelligent maps for autonomous kilometer-scale site survey," in *i-SAIRAS 2008*, Hollywood, USA, 2008.

[4] P. Rigby, S. B. Williams, O. Pizarro, and J. Colquhoun, "Effective benthic surveying with autonomous underwater vehicles," in *OCEANS 2007*. IEEE, 2007.

[5] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin, "Context-based vision system for place and object recognition," in *Ninth IEEE International Conference on Computer Vision, 2003*, 2003, pp. 273–280 vol.1.

[6] C. McGann, F. Py, K. Rajan, J. Ryan, and R. Henthorn, "Adaptive control for autonomous underwater vehicles," in *AAAI Conference on Artificial Intelligence (2008)*, Chicago, 2008.

[7] Z. Jin and A. L. Bertozzi, "Environmental boundary tracking and estimation using multiple autonomous vehicles," in *46th IEEE Conference on Decision and Control*, New Orleans, LA, USA, 2007.

[8] H. Singh, A. Can, R. Eustice, S. Lerner, N. McPhee, O. Pizarro, and C. Roman, "Seabed AUV offers new platform for high-resolution imaging," vol. 85, no. 31, pp. 289, 294–295, Aug. 2004.

[9] C. M. Bishop, *Pattern Recognition and Machine Learning*. Cambridge, UK: Springer Science+Business Media, 2006.

[10] M. I. Jordan, *An Introduction to Probabilistic Graphical Models*, Berkeley, 2002.

[11] S. Russell and P. Norvig, *Artificial Intelligence, A Modern Approach*, 2nd ed. New Jersey: Prentice Hall, 2003.

[12] T. Ferguson, "A Bayesian Analysis of some Nonparametric Problems," *The Annals of Statistics*, vol. 1, no. 2, pp. 209–230, 1973.

[13] C. E. Antoniak, "Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems," *The Annals of Statistics*, vol. 2, no. 6, pp. 1152–1174, 1974. [Online]. Available: http://www.jstor.org/stable/2958336

[14] C. Rasmussen, "The Infinite Gaussian Mixture Model," *Advances in Neural Information Processing Systems*, vol. 12, pp. 554–560, 2000.

[15] D. Blackwell and J. MacQueen, "Ferguson distributions via Polya urn schemes," *The Annals of statistics*, vol. 1, no. 2, pp. 353–355, 1973.

[16] W. Gilks and P. Wild, "Adaptive rejection sampling for Gibbs sampling," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 41, no. 2, pp. 337–348, 1992.

[17] E. Sudderth, "Graphical models for visual object recognition and tracking," Ph.D. dissertation, Massachusetts Institute of Technology, 2006.

[18] M. West, P. Muller, and M. Escobar, "Hierarchical priors and mixture models, with application in regression and density estimation," *Aspects of uncertainty: A Tribute to DV Lindley*, pp. 363–386, 1994.

[19] R. M. Neal, "Markov chain sampling methods for Dirichlet process mixture models," *Journal of Computational and Graphical Statistics*, vol. 9, no. 2, pp. 249–265, 2000. [Online]. Available: http://www.jstor.org/stable/1390653

[20] M. Escobar and M. West, "Bayesian density estimation and inference using mixtures." *Journal of the American Statistical Association*, vol. 90, no. 430, pp. 577–588, 1995.

[21] D. M. Steinberg, O. Pizarro, M. V. Jakuba, and S. B. Williams, "Dirichlet process mixture models for autonomous habitat classification," in *OCEANS 2010*. Sydney: IEEE, May 2010.

[22] J. Sethuraman, "A constructive definition of Dirichlet priors," *Statistica Sinica*, vol. 4, no. 2, pp. 639–650, 1994.

[23] K. Kurihara, M. Welling, and N. Vlassis, "Accelerated variational Dirichlet process mixtures," *Advances in Neural Information Processing Systems*, vol. 19, p. 761, 2007.

[24] D. Blei and M. Jordan, "Variational methods for the Dirichlet process," in *Proceedings of the twenty-first International Conference on Machine Learning*. ACM New York, NY, USA, 2004.

[25] O. Zobay, "Mean field inference for the dirichlet process mixture model," *Electronic Journal of Statistics*, vol. 3, pp. 507–545, 2009.

[26] K. Kurihara, M. Welling, and Y. W. Teh, "Collapsed variational Dirichlet process mixture models," in *Proceedings of the International Joint Conference on Artificial Intelligence*, vol. 20, 2007.

[27] I. Mahon, S. Williams, O. Pizarro, and M. Johnson-Roberson, "Efficient view-based SLAM using visual loop closures," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1002–1014, 2008.

[28] D. Forsyth and J. Ponce, *Computer vision: a modern approach*. Prentice Hall Professional Technical Reference, 2002.

[29] A. Friedman, O. Pizarro, and S. B. Williams, "Rugosity, slope and aspect from bathymetric stereo image reconstructions," in *OCEANS 2010*. Sydney: IEEE, May 2010.

[30] M. Johnson-Roberson, O. Pizarro, and S. Williams, "Towards three-dimensional heterogeneous imaging sensor correspondence and registration for visualization," in *OCEANS 2007 - Europe*. IEEE, 2007.

[31] M. I. McCormick, "Comparison of field methods for measuring surface topography and their associations with a tropical reef fish assemblage," *Marine Ecology Progress Series*, vol. 112, no. 1-2, pp. 87–96, 1994.

[32] B. Shahbaba, "Improving classification models when a class hierarchy is available," Ph.D. dissertation, University of Toronto, 2007.

[33] B. Douillard, "Laser and Vision Based Classification in Urban Environments," Ph.D. dissertation, Australian Centre for Field Robotics, 2009.

[34] R. Gomes, M. Welling, and P. Perona, "Incremental learning of nonparametric Bayesian mixture models," in *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008*, 2008, pp. 1–8.

[35] C. Rasmussen and C. Williams, *Gaussian processes for machine learning*. Springer, 2006.