Multi-modality learning from visual and remotely sensed data

Dushyant Rao, Mark De Deuge, Navid Nourani-Vatani,
Stefan B. Williams and Oscar Pizarro

***Abstract*— Autonomous vehicles are often tasked to explore unseen environments, aiming to acquire and understand large amounts of visual image data and other sensory information. In such scenarios, remote sensing data may be available *a priori*, and can help to plan and execute an autonomous mission. In this paper, we propose a multi-modality learning algorithm to model the relationship between visual images taken by an Autonomous Underwater Vehicle (AUV) during a survey, and remotely sensed acoustic bathymetry (ocean depth) data that are available prior to the survey. The algorithm is based on a mixture of Restricted Boltzmann Machines, and models the joint distribution between the bathymetry and visual modalities. The model is able to cluster the input data, generate useful features for classification, and predict visual image features in unseen dive sites from just the ocean depth information, facilitating image-based queries. These capabilities are useful in planning future AUV dives in unseen environments.**

I. INTRODUCTION

Autonomous robots are often deployed to explore and gather information about the world around them, through cameras and other sensing means. Remotely sensed data can provide useful *a priori* information about a robot's surroundings, and help to make informed decisions about where to go and what tasks to perform.

In the case of Autonomous Underwater Vehicles (AUVs), it is particularly important to consider remotely sensed data from shipborne multibeam SONAR, because AUV dives can only cover a very small fraction of the ocean floor. Specifically, modelling the relationship between bathymetry (ocean depth) data and in-situ visual images allows the AUV to (1) better classify bathymetric data; and (2) predict what kinds of visual features might be observed in unseen areas. The former capability allows the AUV to plan exploration missions to find certain habitats (e.g. "find kelp"), while the latter enables image-based queries (e.g. "explore areas that are likely to look similar to this image").

One important consideration for this application is that bathymetry is a much coarser sensor modality: a single 'type' of feature may correspond to many 'types' of visual features. More specifically, the conditional distribution of visual features given bathymetric features may be highly multimodal. In order to predict visual features from bathymetric data, a generative model must be able to select a mode in a principled manner.

In this paper, we propose an approach based on gated feature learning models, which we argue is better equipped to handle the 'one-to-many' relationship between the two modalities. The gated model is equivalent to a mixture of Restricted Boltzmann Machines [1], in which the joint distribution over both modalities is conditioned on a latent indicator variable. This effectively learns multiple Restricted Boltzmann Machine (RBM) components under the same framework, with the indicator variable switching between them on the fly. We propose heuristics to avoid having to specify the number of components. We also present techniques to perform inference when only bathymetry is available, to predict visual features and determine the bathymetry-only mixture probabilities.

Our results with a toy dataset suggest that the model can better capture the conditional distribution of one modality given the other. Experiments with real bathymetry and AUV-based images show that the model can find meaningful clusters from both a visual and bathymetric perspective, and demonstrate the ability to query by image.

The remainder of this paper is structured as follows: Section II summarises related work in underwater classification and multi-modality learning; Section III describes the proposed algorithm; Section IV outlines the datasets used; Section V details the experimental results; and Section VI concludes the paper and proposes areas for future research.

II. RELATED WORK

A. Feature Learning and Multi-modality Learning

The goal of feature learning is to learn a dictionary of basis vectors or features to describe a dataset, in an unsupervised fashion. Many feature learning methods are based on single layer neural networks, including RBMs and Autoencoders, while others include k-means clustering or Gaussian mixture models [2]. The features learned by these methods tend to be similar, with Gabor-like edge filters for natural images, or handwriting "strokes" for the MNIST digits dataset [2].

Feature learning models become much more powerful when they are stacked to form a deep network. Each layer of a deep network learns a progressively more complex feature abstraction, such as an edge, object-part, or whole object [3], and deep learning methods have attained state-of-the-art performance in a range of machine learning tasks [3][4].

Deep learning approaches are particularly well-suited to multi-modality learning problems, because multiple layers

*This work was supported by the Australian Research Council (ARC) through Discovery programme grant numbers DP110101986, DP1093448 and FT110100511, the Australian Government through the SIEF programme, the Australian Centre for Field Robotics at the University of Sydney and the Integrated Marine Observing System (IMOS).

Navid Nourani-Vatani is with MAN Truck and Bus. Email: navid.nourani-vatani@man.edu.

The other authors are with the Australian Centre for Field Robotics, The University of Sydney, NSW, Australia. Emails: f.lastname@acfr.usyd.edu.au.

can capture higher order correlations between two data modalities. Typically, this involves learning a deep network on each modality separately, and training a multi-modality layer on the combined high-level features to capture the correlations between the two. One such method models the relationship between audio and video features [4], and is able to perform tasks such as cross-modality learning (phoneme classification from video features after training an audio-only classifier). Other approaches look at learning the relationship between a large set of images and associated keywords, using Deep Boltzmann Machines [5] or Bayesian co-clustering [6]. Such techniques enable multi-modality queries, such as image keyword tagging, and content-based image retrieval.

B. Learning and Classification for AUVs

AUV dives are frequently used to perform *habitat classification*, or *benthic habitat mapping*, which involve the categorisation of the ocean floor into clusters or classes that summarise its biological and physical properties [7].

Various techniques perform classification of visual imagery, in either a supervised [8] or unsupervised [9] fashion. However, given the limited coverage of in-situ image data, large-scale habitat mapping methods tend to be based on multibeam acoustic bathymetry data, with the visual imagery acting as "ground truth" [10]. One such technique clusters AUV-based benthic imagery, and uses the probabilistic output as training labels for classification of bathymetric features [11]. Our previous work performs multi-modality learning from visual and bathymetric features, with improved habitat mapping accuracy and the ability to sample one modality from the other [12]. Our new method improves on these techniques by enabling image-based queries and unsupervised clustering of the input data.

III. MODEL OVERVIEW

A. Restricted Boltzmann Machines

A Restricted Boltzmann Machine (RBM) is a stochastic generative neural network comprised of a set of binary visible variables $\mathbf{x} \in \{0,1\}^{n_x}$ and binary hidden variables $\mathbf{h} \in \{0,1\}^{n_h}$. The joint distribution $p(\mathbf{x},\mathbf{h})$ is specified by an energy function:

$$\begin{aligned} E(\mathbf{x},\mathbf{h}) &= -\sum_i a_i x_i - \sum_j b_j h_j - \sum_{ij} w_{ij} x_i h_j \\ p(\mathbf{x},\mathbf{h}) &= \frac{e^{-E(\mathbf{x},\mathbf{h})}}{Z} \end{aligned} \tag{1}$$

Here, $\mathbf{W} = [w_{ij}]$ is the weights matrix, $\mathbf{a} = [a_i]$ and $\mathbf{b} = [b_j]$ are the visible and hidden bias vectors respectively, and $Z = \sum_{\mathbf{x},\mathbf{h}} e^{-E(\mathbf{x},\mathbf{h})}$ is the partition function.

In an RBM, the visible and hidden units form a bipartite graph. This conditional independence property yields the following conditional expressions:

$$\begin{aligned} p(h_j = 1|\mathbf{x}) &= \text{sigm}\Big(b_j + \sum_i w_{ij} x_i\Big) \\ p(x_i = 1|\mathbf{h}) &= \text{sigm}\Big(a_i + \sum_j w_{ij} h_j\Big) \end{aligned} \tag{2}$$

where $\text{sigm}(x) = (1+e^{-x})^{-1}$ is the element-wise logistic sigmoid function.

The probability of an input vector $\mathbf{x}$ can be obtained by marginalising the joint density $p(\mathbf{x},\mathbf{h})$ over the hidden units:

$$\begin{aligned} F(\mathbf{x}) &= -\sum_i a_i x_i - \sum_j \log\Big(1 + e^{b_j + \sum_i w_{ij} x_i}\Big) \\ p(\mathbf{x}) &= \frac{\sum_{\mathbf{h}} e^{-E(\mathbf{x},\mathbf{h})}}{Z} = \frac{e^{-F(\mathbf{x})}}{Z} \end{aligned} \tag{3}$$

where the expression $F(\mathbf{x})$ is known as the *free energy* of a visible vector. Unfortunately, the partition function Z is intractable, which means that the RBM can only compute *unnormalised* probabilities. However, this is not restrictive for our application, and several techniques in the literature can approximate the partition function if necessary [13].

B. Gated Boltzmann Machines and Mixtures of RBMs

In a Gated Boltzmann Machine (GBM), the graphical model becomes a tripartite graph. The joint relationship between $\mathbf{x}$ and $\mathbf{h}$ is conditioned on a vector of 'gating' or conditioning variables $\mathbf{z}$, which modulate both the weights and the biases of the model. As such, the weights matrix becomes a three dimensional parameter tensor $\mathbf{W} \in \mathbb{R}^{n_x \times n_h \times n_z}$ representing the connections between every single visible, hidden, and gating unit. Similarly, the hidden biases $\mathbf{b} \in \mathbb{R}^{n_z \times n_h}$ and visible biases $\mathbf{a} \in \mathbb{R}^{n_z \times n_x}$ are 2D matrices.

If the gating variables are constrained to be a 'one-of-k' (i.e. $\mathbf{z} \in \{0,1\}^{n_z}, \sum_k z_k = 1$), then each possible value for $\mathbf{z}$ indexes a single 2D slice of $\mathbf{W}$ and a 1D slice of each bias matrix. This forms an Implicit Mixture of RBMs model [1], where $\mathbf{z}$ is a mixture indicator variable used to select one of n_z RBM components, each with separate weights and biases.

The conditional expressions for the model are now also conditioned on the mixture indicator $\mathbf{z}$:

$$\begin{aligned} p(h_j = 1|\mathbf{x},\mathbf{z}_k = 1) &= \text{sigm}\Big(b_{jk} + \sum_i w_{ijk} x_i\Big) \\ p(x_i = 1|\mathbf{h},\mathbf{z}_k = 1) &= \text{sigm}\Big(a_{ik} + \sum_j w_{ijk} h_j\Big) \end{aligned} \tag{4}$$

One key difference between the Mixture of RBMs model and other mixture models is that the mixture responsibilities are determined implicitly according to the free energy:

$$p(z_k = 1|\mathbf{x}) = \frac{e^{-F(\mathbf{x},\mathbf{z}_k=1)}}{\sum_k e^{-F(\mathbf{x},\mathbf{z}_k=1)}} \tag{5}$$

where the expression $F(\mathbf{x},\mathbf{z}_k = 1)$ is the free energy of the k^{th} component RBM. Note that the denominator in (5) is tractable, and is linear in the number of mixture components n_z.

C. Architecture

Our model uses a deep multi-modality architecture similar to previous work [4][12], with feature learning performed on the visual and bathymetric data modalities separately, and then a final multi-modality layer to capture the correlations between the two (Fig. 1) The main difference is in the use of a gated model for the multi-modality layer, which enables

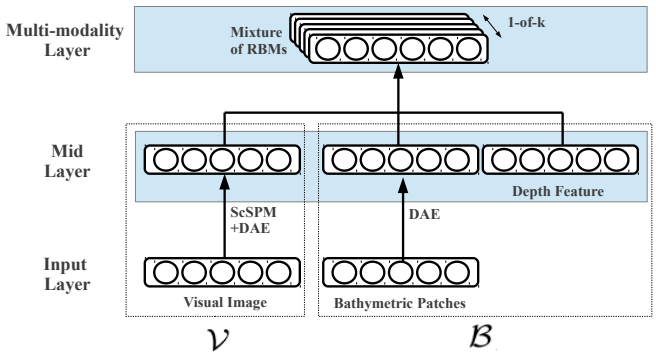


Fig. 1. Schematic showing the model architecture. Features from both modalities are concatenated in the mid layer, and then passed into the multi-modality layer. The one-of-k indicator variable indexes a single RBM component from the Mixture of RBMs model.

additional inference capabilities, such as clustering, visual prediction, and image-based queries.

We use the same mid layer features as in [12]. The mid layer visual features are extracted using a Sparse coding Spatial Pyramid Matching (ScSPM) algorithm followed by Random Projections for dimensionality reduction [9], proceeded by a single Denoising Autoencoder (DAE) layer [14]. The bathymetric patch is split into two components: a zero-meaned patch representing the local shape or texture, and the mean ocean depth value. A single DAE layer is learned on the patches to capture textural features, and the hidden activations of the DAE comprise the mid layer features. The ocean depth is incorporated directly into the mid layer as a "histogram" feature. Similar to a one-of-k encoding, the depth range of the dataset is divided into 1m bins. The bin containing the estimated depth value is set to a value of 1, and the surrounding bins are encoded according to a Gaussian-like falloff.

D. Training

Given a set of training vectors $\{\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(N)}\}$, RBM models are usually trained to maximise the mean log probability of the data, $L = \frac{1}{N}\sum_{n=1}^{N} \log p(\mathbf{x}^{(n)})$, using Stochastic Gradient Descent. The gradient of L with respect to the parameters Θ is given by:

$$\frac{\partial L}{\partial \Theta} = N\mathbb{E}\left[\frac{\partial E(\mathbf{x},\mathbf{h},\mathbf{z})}{\partial \Theta}\right] - \sum_{n=1}^{N} \mathbb{E}\left[\frac{\partial E(\mathbf{x}^{(n)},\mathbf{h},\mathbf{z})}{\partial \Theta}\middle|\mathbf{x}^{(n)}\right] \quad (6)$$

The second expectation can be estimated using Gibbs sampling to draw unbiased samples from the conditional distribution $p(\mathbf{h}, \mathbf{z}|\mathbf{x}^{(n)})$, but the first term is intractable. As a result, the Maximum Likelihood gradients are approximated using the Contrastive Divergence (CD) algorithm [1], commonly used for a variety of energy-based models.

To prevent overfitting, a regularisation penalty ("weight decay") is often added to the learning objective. Furthermore, past work shows that selectively activated hidden units usually lead to better discriminative performance [2]. In this work, the sparsity cost is given by the cross entropy between the average activation of each unit ($\hat{\rho}_j$) and a user-defined sparsity (ρ), and the a weight decay term is the square of the Frobenius norm of the weights tensor, $\|\mathbf{W}\|_F{}^2$.

Algorithm 1 Predicting visual features from bathymetry

1: **for** $k = 1$ to n_z **do**
2: Initialise the mid layer feature vector with zeros for the visual features, $\mathbf{x} = [\mathbf{x}_\mathcal{B}; \mathbf{x}^*_\mathcal{V}] = [\mathbf{x}_\mathcal{B}; 0; 0; ...; 0]$.
3: **while** not converged **do**
4: Compute $\mathbb{E}_k[\mathbf{h}|\mathbf{x}] = p(\mathbf{h}|\,\mathbf{x}, \mathbf{z}_k = 1)$.
5: Compute $\mathbb{E}_k[\mathbf{x}_\mathcal{V}|\mathbf{h}] = p(\mathbf{x}_\mathcal{V}|\,\mathbf{h}{=}\mathbb{E}_k[\mathbf{h}|\mathbf{x}], \mathbf{z}_k = 1)$
6: **if** $\|\mathbf{x}^*_\mathcal{V} - \mathbb{E}_k[\mathbf{x}_\mathcal{V}|\mathbf{h}]\| < \epsilon$ **then**
7: converged
8: **else**
9: $\mathbf{x}^*_\mathcal{V} \Leftarrow \mathbb{E}_k[\mathbf{x}_\mathcal{V}|\mathbf{h}]$, $\mathbf{x} \Leftarrow [\mathbf{x}_\mathcal{B}; \mathbf{x}^*_\mathcal{V}]$
10: **end if**
11: **end while**
12: $\mathbb{E}_k[\mathbf{x}_\mathcal{V}|\mathbf{x}_\mathcal{B}] \Leftarrow \mathbf{x}^*_\mathcal{V}$.
13: **end for**

E. Cluster Heuristics

To avoid having to specify the number of mixture components required, we use heuristics to add and remove components on-the-fly during training. During learning, the mixture responsibility $p(\mathbf{z}_k = 1 \mid \mathbf{x})$ of a cluster k is monitored, and a cluster is removed, when the mean value over the entire dataset exceeds or drops below a threshold, respectively. When splitting a cluster, the new cluster parameters are copied directly from the existing cluster. Our experiments show that after a few parameter updates, the two identical clusters diverge to capture different parts of the input dataset.

These cluster heuristics are essential in order to learn a useful model, as they allow useless mixture components to be removed, freeing up parameters to allow dominant mixture components to be split. In our experiments without them, the model often uses a single mixture component for a large proportion of the data.

F. Predicting Visual Features

We can predict the midlayer visual features $\mathbf{x}_\mathcal{V}$, conditioned on the midlayer bathymetric features $\mathbf{x}_\mathcal{B}$, using a mean field approximation (Algorithm 1). For each mixture component (indexed by k), we use the input values to compute the mean hidden activations $\mathbb{E}_k[h_j|\mathbf{x}] = p\,(h_j = 1|\mathbf{x})$, which are then in turn used to compute the conditional expectations $\mathbb{E}_k[x_\mathcal{V}|\mathbf{h}]$. This process can be iterated until convergence, yielding $\mathbb{E}_k[\mathbf{x}_\mathcal{V}|\mathbf{x}_\mathcal{B}]$, the conditional expection of $\mathbf{x}_\mathcal{V}$ under the k^{th} mixture component. In our experiments, a single iteration is enough to yield a good conditional estimate.

We then approximate the bathymetry-only mixture responsibilities according to:

$$p\,(z_k = 1|\mathbf{x}_\mathcal{B}) \approx \frac{e^{-F(\mathbf{x}_\mathcal{B}, \mathbf{x}_\mathcal{V}=\mathbb{E}_k[\mathbf{x}_\mathcal{V}|\mathbf{x}_\mathcal{B}], \mathbf{z}_k=1)}}{\sum_k e^{-F(\mathbf{x}_\mathcal{B}, \mathbf{x}_\mathcal{V}=\mathbb{E}_k[\mathbf{x}_\mathcal{V}|\mathbf{x}_\mathcal{B}], \mathbf{z}_k=1)}} \quad (7)$$

That is, use each component RBM to fill in missing dimensions with their conditional expectations, compute the free energies given these 'best-case' scenarios, and then normalise the probabilities over all mixture components.

G. Image-based queries

Given a region of interest, visual features can be predicted from the bathymetry and compared to a query image to

produce a utility map over the whole region. This can then be used by a planning algorithm to explore areas where similar images are likely to be observed.

The query-by-image procedure is as follows. First, for each point in the region of interest, we predict the visual features from the local bathymetry (i.e. compute the conditional expectation $\mathbb{E}_k[\mathbf{x}_\mathcal{V}|\mathbf{x}_\mathcal{B}]$ according to each mixture component k), and compute the marginal mixture responsibilities $p(\mathbf{z}|\mathbf{x}_\mathcal{B})$. We then define a utility function $\mathcal{U}$, which acts as a proxy for the likelihood of observing the query image given the bathymetry at a particular location. The utility at a particular location is based on the similarity between the query image and each of the n_z predicted images, scaled by the associated mixture probabilities:

$$\mathcal{U} = \sum_{k=1}^{n_z} p(z_k = 1|\mathbf{x}_\mathcal{B})\ \mathcal{S}(\mathbf{x}_{\mathcal{V}q}, \mathbb{E}_k[\mathbf{x}_\mathcal{V}|\mathbf{x}_\mathcal{B}]) \tag{8}$$

where $\mathbf{x}_{\mathcal{V}q}$ is the midlayer visual feature vector for the query image, and $\mathcal{S}(\mathbf{u}, \mathbf{v})$ is a metric computing the similarity between $\mathbf{u}$ and $\mathbf{v}$. In this work, we use the normalised cross-correlation metric, given by $\mathcal{S}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u}\cdot\mathbf{v}}{\|\mathbf{u}\|\|\mathbf{v}\|}$.

H. Classification

With a gated model, there are a number of options for features that can be extracted for classification. The mixture responsibilities are themselves a good low-dimensional feature set, since the model naturally uses different mixture components for different parts of the input space. Additionally, for a given data vector, we can obtain hidden unit activations for all mixture components and stack them into a single vector. The latter option usually provides more useful information for a classification task, since it captures the similarity of the input data to the learned features in all component RBMs.

Features obtained by unsupervised feature learning or deep learning models are usually passed into a linear classifier. In this paper we use a Logistic Regression (LR) classifier for all experiments.

IV. DATASETS

We introduce a two-dimensional toy dataset (Fig. 2) to illustrate the operation of our algorithm and gauge its effectiveness. While it is highly simplified compared to our real multi-modality dataset, it is designed to share one key characteristic: the fact that the conditional distribution of visual features (represented by dimension $x_\mathcal{V}$) given bathymetric features (dimension $x_\mathcal{B}$) can be highly multimodal. The toy dataset was created by generating polynomial curve segments from random coefficient values with additive Gaussian noise.

The real-world bathymetry data is in the form of 15×15 pixel patches of gridded data from Geoscience Australia [15]. The uniform grid has a separation of $1.6\,\mathrm{m}$ between points, so that the patches represent an area of $22.4 \times 22.4\,\mathrm{m}^2$.

We also utilise 1360×1024 pixel visual images taken by our AUV *Sirius* off the Eastern coast of Tasmania, Australia [16]. Matched multi-modality data was obtained by extracting a bathymetry patch centred at the AUV position

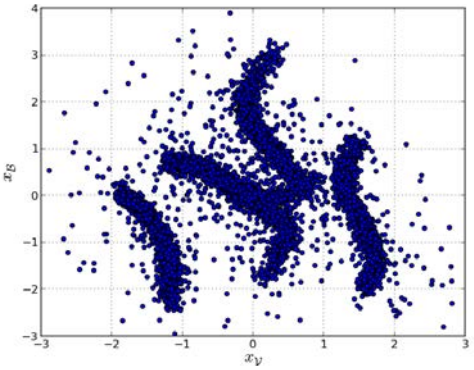


Fig. 2. The 2D toy dataset generated for this problem. The dataset is designed such that the conditional distribution $p(x_\mathcal{V}|x_\mathcal{B})$ can be highly multimodal.

corresponding to each image. For this work, we ignore any potential misregistration due to errors in AUV localisation; this is a reasonable assumption because the navigation accuracy is similar to the bathymetric grid spacing and much smaller than the scale of spatial variation of benthic habitats.

For classification tasks, we utilise expert annotations of the AUV image data, consolidated into 5 habitat classes characterised by keywords "sand", "screw shell rubble", "reef / sand interface", "reef", and "kelp" (*Ecklonia Radiata*). The labelled dataset contains $75,400$ visual images, each paired with a bathymetric patch, and is split equally into a training and test set.

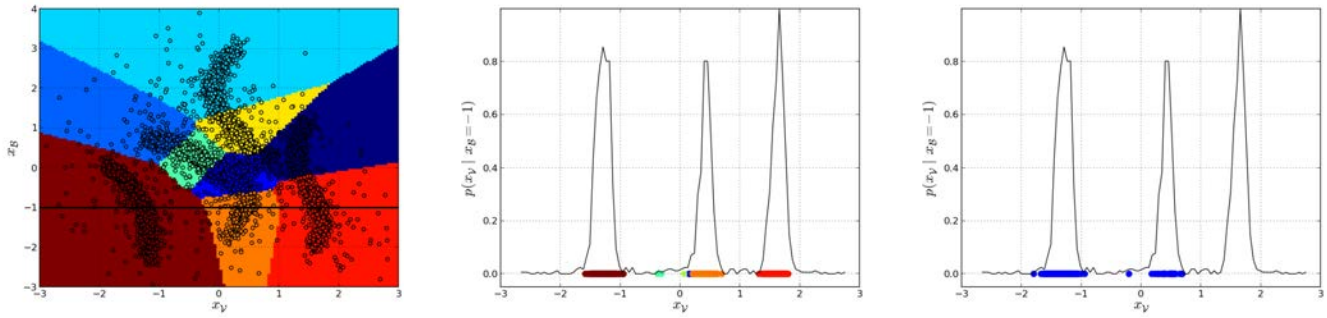
V. RESULTS

We present results with simulated and real-world marine data. All models are developed in Python using the pylearn2 library [17] and are trained on a NVIDIA GeForce GTX 590 GPU.

A. Toy results

The toy dataset results are shown in Fig. 3. As demonstrated by Fig. 3(a), different component RBMs are used to model different parts of the dataset, which means that the data can be clustered in an unsupervised fashion. Fig. 3(b) shows the result of sampling from the conditional distribution $p(x_\mathcal{V} \,|\, x_\mathcal{B} = -1)$ (the line marked in Fig. 3(a)) using a MixRBM and the method outlined in Section III-F. Even with a highly multimodal conditional distribution, the model can produce reasonable samples, and each mode is represented by a different component RBM. In contrast, Fig. 3(c) shows the same result with a standard RBM, by initialising the missing $x_\mathcal{V}$ value to zero and performing a number of iterations of Gibbs sampling [12]. With this approach, the Gibbs chain is not always able to mix between modes of the conditional distribution. This could be alleviated by initialising the missing dimension randomly and repeating a number of times, but this process scales exponentially with the number of missing dimensions, whereas the corresponding Mixture-of-RBMs method is linear in the number of mixture components.

These results illustrate the key benefits of the gated MixRBM model as compared to a standard RBM. In addition



(a) Input data coloured according to most probable mixture component

(b) Conditional samples generated by Mixture of RBMs, overlaid on the conditional distribution, coloured by mixture component

(c) Conditional samples generated by standard RBM, overlaid on the conditional distribution

Fig. 3. Results on the toy dataset. Since neither the RBM or MixRBM can analytically determine the conditional distribution, it is crudely approximated for (b) and (c) using the histogram of all points within some δ of the setpoint value $x_{\mathcal{B}} = -1$.

to unsupervised clustering of the input data, the model can be used to tractably generate conditional samples from explicit regions of our highly multimodal distribution. In contrast with previous work [12], the model can map a bathymetric feature to multiple options simultaneously rather than a single mode / label.

B. Classification

A key requirement of the model is the ability to extract useful features for classification tasks. Classification results are shown in Table I for different modality combinations, along with comparisons to other approaches. In the "baseline" scenario, the midlayer features are passed directly into the classifier, and the "DAE" scenario uses our previous approach [12]. In the $\mathbf{z}$ (one hot) scenario, each input is encoded as a feature vector with a one for the cluster with highest probability and zeros for the remaining dimensions. The other two feature scenarios refer to the mixture responsibilities $p(\mathbf{z}|\mathbf{x})$ and the mean hidden activations $p(\mathbf{h}|\mathbf{x})$ for all mixture components. In the $\mathcal{B}$ or $\mathcal{V}$ modality only cases, the marginal mixture responsibilities $p(\mathbf{z}|\mathbf{x}_{\mathcal{B}})$ and $p(\mathbf{z}|\mathbf{x}_{\mathcal{V}})$ are computed according to Section III-F.

From the results with one hot $\mathbf{z}$ features, we can observe that the most probable cluster component itself holds a lot of information about the habitat label, with 77% accuracy with both modalities. Converting the one-hot vector to a vector of mixture probabilities yields a small improvement in performance for all scenario combinations, and by using the full set of hidden features, the classification accuracy is considerably higher. Both the DAE and MixRBM techniques offer a 12% improvement when only bathymetry is available, and also yield a greater accuracy with the other modality options.

We also report the area under the Receiver Operating Characteristic curve (AUROC) for each of the classifiers (Table II), to assess how the results may vary as the discrimination threshold is changed. Since the standard ROC curve is only applicable to binary classification problems, we generate micro-averaged ROC curves, where each label

TABLE I

CLASSIFICATION ACCURACY (%) FOR VARIOUS INPUT MODALITIES

		Modalities		
Model	Features	$\mathcal{B}$ and $\mathcal{V}$	$\mathcal{B}$ only	$\mathcal{V}$ only
Baseline	Midlayer	82.24	67.62	79.98
DAE + LR [12]	$p(\mathbf{h}\|\mathbf{x})$	87.23	79.05	81.42
MixRBM + LR	$p(\mathbf{h}\|\mathbf{x})$ (all)	87.89	79.24	81.30
	$p(\mathbf{z}\|\mathbf{x})$	78.09	59.09	72.96
	$\mathbf{z}$ (one hot)	77.66	54.56	70.42

TABLE II

AREA UNDER RECEIVER OPERATING CHARACTERISTIC CURVE (AUROC) FOR VARIOUS INPUT MODALITIES

	Modalities		
Model	$\mathcal{B}$ and $\mathcal{V}$	$\mathcal{B}$ only	$\mathcal{V}$ only
Baseline	0.9630	0.9103	0.9614
DAE + LR [12]	0.9785	0.9520	0.9647
MixRBM + LR	0.9833	0.9540	0.9641

is represented as a one-of-k binary vector, and each of the class dimensions is considered as a single binary classifier. The results from Table II are consistent with the classification accuracies, and demonstrate that the proposed method provides a considerable improvement over the baseline, and is nearly identical in performance to the DAE approach [12].

C. Clustering

In addition to improving classification performance, the model is able to cluster the data in an unsupervised fashion, unlike the model presented in [12]. When applied to our real-world multi-modality dataset, the model used 17 mixture components. The 10 clusters with the greatest number of input samples are shown in Fig. 4. It is important to note that the technique is clustering the data jointly over *both* visual and bathymetric inputs. Thus, while most images within each cluster are visually similar, some may be assigned according to bathymetric similarity.

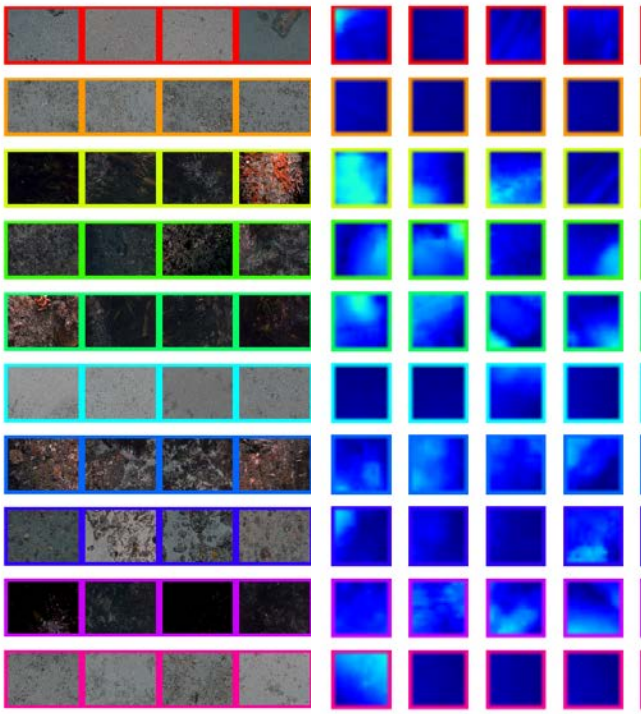


Fig. 4. Examples from the 10 largest clusters (each row). Each image (left) is matched with its corresponding bathymetric patch (right)

D. Prediction of Visual Features and Image-based Queries

By predicting visual features in unseen areas, the model can handle image-based queries, which can aid survey planning. We present query-by-image results for a region in the Tasmanian shelf known as O'Hara Bluff, using the procedure in Section III-G. The bathymetry map for O'Hara Bluff is shown in Fig. 5, with the AUV trajectory overlaid, and Fig. 6 shows query images from different habitat classes and their resulting utility maps.

The results are consistent with what we would expect for each habitat class. Sand images may be observed anywhere, but are more likely in the deep, flat-bottomed areas towards the East, while reef images are usually found in rugose (rugged terrain) regions. Images containing both sand and reef are likely to occur at the interface between the two, while kelp forests are restricted to shallower waters.

The results demonstrate that, without any supervision, the model can handle image-based queries and produce a utility map consistent with known class-based predictions. Despite the fact that the AUV transect covers a small fraction of O'Hara Bluff, the image-based queries can predict the utility over the larger region.

VI. CONCLUSIONS

In this paper, we have proposed a model based on a mixture of Restricted Boltzmann Machines, to perform multi-modality learning using visual images and remotely sensed bathymetry data from shipborne multibeam SONAR. Unlike past approaches, this method explicitly captures the one-to-many relationship between the two modalities: any bathymetric feature may correspond to a number of different

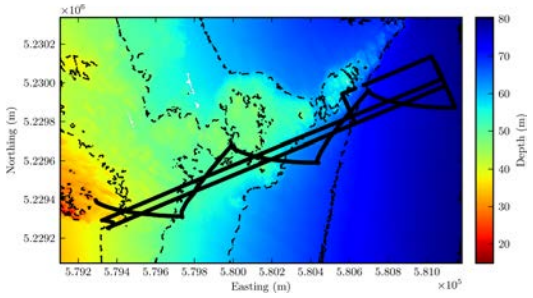
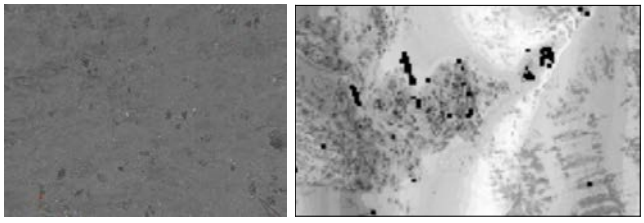
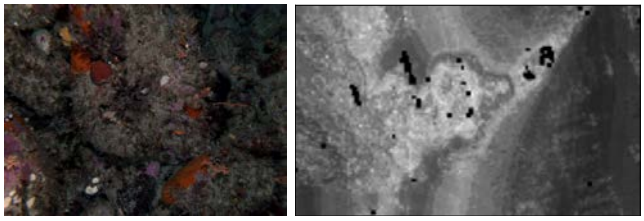


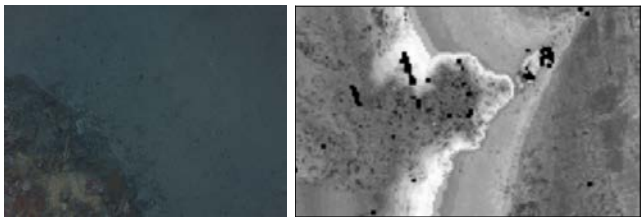
Fig. 5. Bathymetry map for O'Hara Bluff region, overlaid with the trajectory traversed by the AUV



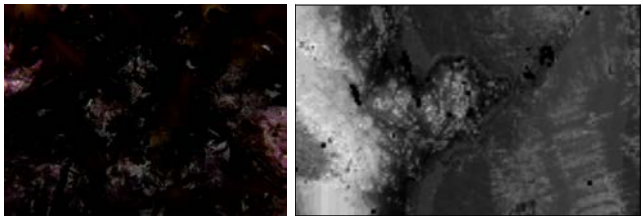
(a) Sand



(b) Reef



(c) Reef / Sand Interface



(d) Kelp

Fig. 6. Some image-based query results for images from different habitat classes. Left: Query images. Right: Corresponding utility maps over the whole O'Hara Bluff region, with white regions indicating higher utility.

visual features. We have demonstrated empirically that the model can cluster the modalities in an unsupervised fashion, both singularly and jointly, and can be used to predict visual features in unseen areas using bathymetry alone. This can be useful for "query-by-image" tasks, where the goal is to explore areas that are likely to look similar to a given input

image. Such queries are particularly useful because they do not require any supervised training or manual annotation of the visual imagery.

Future work will focus on using the model for autonomous planning in unseen areas. Another interesting direction will be to extend the approach to perform multi-modality learning for other platforms, such as for ground vehicles mounted with velodyne LIDAR and cameras.

ACKNOWLEDGMENTS

The authors thank Asher Bender, Ariell Friedman, and Daniel Steinberg, for access to some of the datasets used for this research. This work was supported by the Australian Research Council (ARC) and the New South Wales and Tasmanian State Governments, and the Integrated Marine Observing System (IMOS) through the DIISR National Collaborative Research Infrastructure Scheme. The authors would like to thank the Captain and crew of the R/V Challenger. Their sustained efforts were instrumental in facilitating successful deployment and recovery of the AUV. Thanks to Justin Hulls and Jan Seiler for help and support on-board the ship and for providing the supervised image labels. The ship-borne multibeam sonar data were collected, processed and gridded by Geoscience Australia. We also acknowledge the help of all those who have contributed to the development and operation of the IMOS AUV Facility.

REFERENCES

[1] V. Nair and G. E. Hinton, "Implicit mixtures of Restricted Boltzmann Machines," in *Advances in neural information processing systems*, 2009, pp. 1145–52.

[2] A. Coates, H. Lee, and A. Y. Ng, "An analysis of single-layer networks in unsupervised feature learning," *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2010.

[3] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proceedings of the 26th Annual Int. Conf. on Machine Learning*, 2009, pp. 609–616.

[4] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proceedings of the 28th Annual Int. Conf. on Machine Learning*, 2011, pp. 689–696.

[5] N. Srivastava and R. Salakhutdinov, "Multimodal learning with Deep Boltzmann Machines," in *Advances in Neural Information Processing Systems*, vol. 25, 2012, pp. 2231–2239.

[6] G. Irie, D. Liu, Z. Li, and S.-F. Chang, "A bayesian approach to multimodal visual dictionary learning," in *IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 329 – 336.

[7] S. B. Williams, O. R. Pizarro, M. V. Jakuba, C. R. Johnson, N. S. Barrett, R. C. Babcock, G. A. Kendrick, P. D. Steinberg, A. J. Heyward, P. J. Doherty *et al.*, "Monitoring of benthic reference sites: using an autonomous underwater vehicle," *Robotics & Automation Magazine, IEEE*, vol. 19, no. 1, pp. 73–84, 2012.

[8] O. Beijbom, P. J. Edmunds, D. I. Klinez, B. G. Mitchellz, and D. Kriegman, "Automated annotation of coral reef survey images," in *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2012.

[9] D. Steinberg, "An unsupervised approach to modelling visual data," Ph.D. dissertation, Univ. of Sydney, 2013.

[10] C. J. Brown, S. J. Smith, P. Lawton, and J. T. Anderson, "Benthic habitat mapping: A review of progress towards improved understanding of spatial ecology of the seafloor using acoustic techniques," *Estuarine, Coastal & Shelf Science*, vol. 92, no. 3, pp. 502–20, 2011.

[11] A. Bender, S. B. Williams, and O. Pizarro, "Classification with probabilistic targets," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2012, pp. 1780–86.

[12] D. Rao, M. De Deuge, N. Nourani-Vatani, B. Douillard, S. B. Williams, and O. Pizarro, "Multimodal learning for autonomous underwater vehicles from visual and bathymetric data," in *IEEE Int. Conf. on Robotics and Automation*, 2014, pp. 3819–25.

[13] R. Salakhutdinov and I. Murray, "On the quantitative analysis of Deep Belief Networks," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 872–879.

[14] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *The Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.

[15] M. Spinoccia, "Bathymetry grids of south east Tasmania shelf," *Geosciences Australia*, 2011. [Online]. Available: http://www.ga.gov.au/marine/bathymetry.html

[16] S. Williams, O. Pizarro, M. Jakuba, and N. Barrett, "AUV benthic habitat mapping in South Eastern Tasmania," in *Field and Service Robotics*, 2010, pp. 275–284.

[17] I. J. Goodfellow, D. Warde-Farley, P. Lamblin, V. Dumoulin, M. Mirza, R. Pascanu, J. Bergstra, F. Bastien, and Y. Bengio, "Pylearn2: a machine learning research library," *arXiv preprint arXiv:1308.4214*, 2013. [Online]. Available: http://arxiv.org/abs/1308.4214