Autonomous Adaptive Exploration using Realtime Online Spatiotemporal Topic Modeling

Yogesh Girdhar, Philippe Giguère, and Gregory Dudek

1 Introduction

Exploration of dangerous environments, such as underwater coral reefs and shipwrecks, is a difficult and potentially life threatening tasks for humans, which naturally makes the use of an autonomous robotic system very appealing. Exploration through the use of an autonomous agent can find uses in many different scenarios. For example, continuous monitoring of a coral reef could be used to alert a biologist of any harmful changes such as disease or physical damage; surveillance of border regions of a property or nation to detect any security anomalies; and planetary exploration by a rover. In all of these examples, we are essentially interested in identifying surprising observations, and collecting more information about them.

This paper presents such an autonomous system, which is capable of autonomous exploration, and shows its use in a series of experiments to collect image data in challenging underwater marine environments. We presents novel contributions on three fronts. First, we present an online topic-modeling based technique to describe what is being observed using a low dimensional semantic descriptor. This descriptor attempts to be invariant to observations of different corals belonging to the same species, or observations of similar types rocks observed from different viewpoints. Second, we use the topic descriptor to compute the *surprise* score of the current observation. This is done by maintaining an online summary of observations thus far, and then computing the surprise score as the distance of the current observation to the summary, in the topic space. Finally, we present a novel control strategy for

Yogesh Girdhar and Gregory Dudek
Centre for Intelligent Machines, McGill University
318-3480 University Street, Montreal, QC, Canada, H3A 0E9.
e-mail: {yogesh,dudek}@cim.mcgill.ca

Philippe Giguère
Département d'informatique et génie logiciel, Université Laval
1065, Avenue de la Médecine, Quebec, QC, Canada, G1V 0A6.
e-mail: philippe.giguere@ift.ulaval.ca

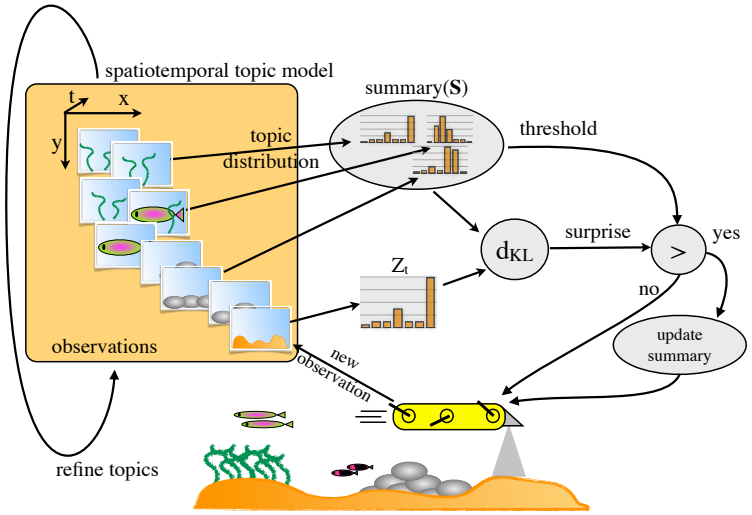


Fig. 1 Each incoming observation made by the robot is a set of visual words, and their corresponding locations. We continuously refine topic labels for each word using an realtime online Gibbs sampler, taking into account the spatiotemporal correlation of topic labels. Observations are hence represented using a distribution over these spatiotemporal topics, which the summarizer then uses to build the summary. Surprise of a new observation is its distance to the closest sample in the summary. If this surprise is above a threshold, we add the new observation to the summary. The surprise score of the current observation is used to control the speed of the robot. Higher surprise score of an observation results in slower speed for the robot.

an underwater robot thats allows for intelligent traversal; hovering over surprising observations, and swimming quickly over previously seen corals and rocks. Fig. 1 shows an overview of the proposed approach.

Exploration, in the context of robotics, has been studied before. Work has been done on autonomous mapping of challenging environments [30], [18], frontier expansion [33], minimizing uncertainty [32], and utility based exploration[14]. Newman et al. [20, 22] have described a system for mapping and semantically labeling urban environments. The semantic labeling framework utilizes a bank of SVM classifiers to label different regions of the scene with one of the pre-defined class labels such as grass, road, wall, or bush. One disadvantage of using such an approach for exploration tasks is that it requires prior knowledge about the kind of labels which the robot might encounter.

As robots collect more and more visual data on their exploration sorties, the task of summarizing the videos collected by the robot so that the any surprising events encountered by the robot are included in the summary, becomes extremely important for the human operator. Several techniques have been studied for such task, such as

use of PCA to ensure visual and geographic coverage[8], and ONSUM[11], a system for identifying most surprising images from a stream of incoming images.

In the underwater robotics domain, there is body of work on autonomous ocean monitoring using robots. Smith et al. [28] have looked at computing robot trajectories which maximize information gained, while minimizing the deviation from the planned path. Das et al. [7] have presented techniques to autonomously observe advecting oceanographic features in the open ocean. Rigby et al. [23] have proposed the use of Gaussian Process to first infer a probabilistic map of benthic features using data from a prior survey, and then plan a path, which results in maximum information gain.

Our focus in this paper is on traversing an environment similar to how a tourist might do so in a new city; stopping and recording any surprising sights, while moving fast when nothing new is in sight. This is similar to the vacation snapshot problem described in [4].

We used an untethered amphibious robot (Aqua [25]), with an in-house designed autopilot, to carry the exploration task. Images were taken with a downward-looking camera, with all computations performed onboard. Its propulsion is based on six flippers that can provide motion in five degrees of freedom. By using a novel combination of gaits, the robot was able to move at various speeds while maintaining its orientation, despite external disturbances. This was necessary in order to complete this exploration task.

In Section 2 we present a realtime online spatiotemporal topic-model (ROST) for describing a stream of word observations in a low dimensional semantic space. In Section 3 we describe the idea of extremum summaries that are used to compute surprise score of an incoming observation. Finally, in Section 4 we describe the control strategy used by Aqua to do surprise-based exploration of the environment.

2 Topic Modeling

To have meaningful summaries, and thus a meaningful surprise score, we must use an image descriptor that is sensitive to thematic changes in the scene, while being immune to low level image changes. We do this via the use of a topic modeling framework, which describes an incoming observation using a low dimensional descriptor, each dimension of which ideally corresponds to absence or presence of different objects or high-level visual patterns in the world.

2.1 Latent Dirichlet Allocations

Topic modeling methods were originally developed for text analysis. Hoffman [16] introduced the idea of probabilistic Latent Semantic Analysis(pLSA) for text documents, which modeled the probability of observing a word w_i in a given document

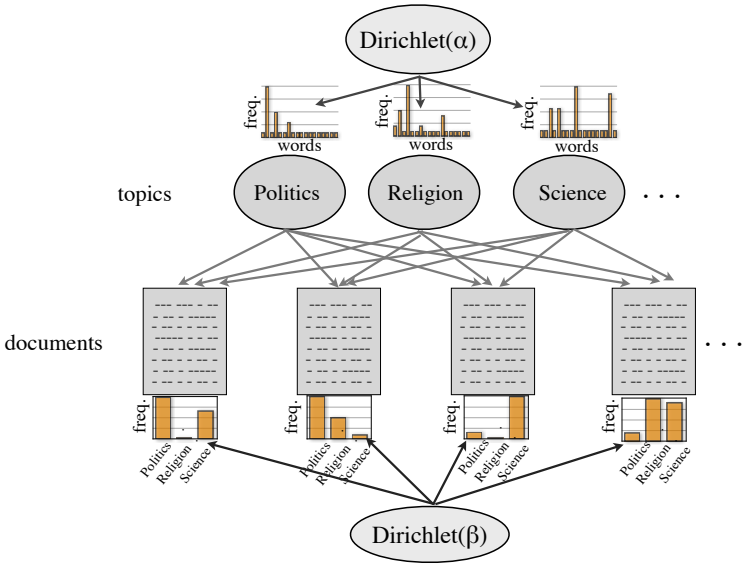


Fig. 2 Latent Dirichlet Allocation (LDA) is a popular technique for describing a set of documents as a mixture of topics, which are themselves a described as distribution over words. Both the topic-word and document-topic distributions have Dirichlet priors which biases these distributions in favor of being sparse.

M as:

$$\mathbf{P}(w_i = v|M) = \sum_{k=1}^{K} \mathbf{P}(w_i = v|z_i = k)\mathbf{P}(z_i = k|M). \tag{1}$$

where w_i takes a value between $1 \ldots V$, and z_i is the hidden variable or topic label for w_i, which takes a value between $1 \ldots K$, where K is a much smaller than V.

The central idea is the introduction of a latent variable z, which models the underlying topic, or the context responsible for generating the word. Each document M in the given corpora is modeled using a distribution $\theta_m(k) = \mathbf{P}(z_i = k|M)$ over topics, and each topic is modeled using a distribution $\phi_k(v) = \mathbf{P}(w_i = v|z_i = k)$ over the set of vocabulary words. During the training phase, these distributions are learned directly using an EM algorithm.

The distribution of topics in a document gives us a low dimensional semantic description of the document, which can be used to compare it with other documents semantically. The problem with this approach is that since the dimensionality of the model is very large, a lot of training data is required. Moreover, it is easy to overtrain for a given data set.

Latent Dirichlet Allocation (LDA), proposed by Blei et al. [2] mitigate the training problem by placing Dirichlet prior on θ and ϕ. Placing Dirichlet priors encourages the distributions to be sparse, which has been shown to give semantically more

relevant topics. Fig. 2 shows the graphical model for LDA. Griffiths et al. [15] subsequently proposed a collapsed Gibbs sampler for LDA, where the state is topic assignments for all the words in all the documents, which is different from original variational approximation based approach proposed by Blei et al.

2.2 Topic Modeling for Visual Observations

The success of LDA-based topic modeling methods for semantic clustering and classification of text documents has led to their use in the computer vision domain. The general idea being that a textual word could be replaced by visual words, such as ones described by Sivic et al. [27]. To generate a visual vocabulary, we first extract visual features such as SIFT[19] or SURF[1] from an unrelated dataset, with high visual diversity. These features are then clustered using the k-means algorithm, with V clusters corresponding to the desired vocabulary size. The cluster centers of these V clusters represent the visual words in the vocabulary. Now, to extract visual words from a given image, first we extract its SURF features, and then map each feature to the index of the closest visual word in the vocabulary.

Both SIFT and SURF features are 128 dimensional floating points vectors, and doing nearest neighbor queries require computing L1 distances between these two vectors, which can be computationally expensive for large number of features. More recently, binary feature descriptors such as Oriented BRIEF (ORB) [24], have been shown to perform well. Distance between two binary feature vectors is typically computed by taking the Hamming distance between the bit strings, which can be implemented very efficiently using XOR operations that are available on all computers. Hence, such binary feature descriptors are much more appropriate for applications requiring realtime operation.

Bosch et al. [3] used pLSA for scene classification and object discovery using such visual words. Works by Fei-Fei et al.[9], and Sivic et al.[26] have demonstrated the use of LDA to model image content, and automatic generation of meaningful object hierarchies.

2.3 Topic Modeling for Robots

Semantic modeling of observation data captured by a mobile robot faces additional challenges compared to semantic modeling of a collection of text documents, or image that are mutually independent.

- Robot observations are generally continuous in space and time, and hence the corresponding semantic descriptor must also be continuous. We must take into account the location of the observed visual words during the refinement, and use it to compute topic priors that are sensitive to changes in time and location of the robot.

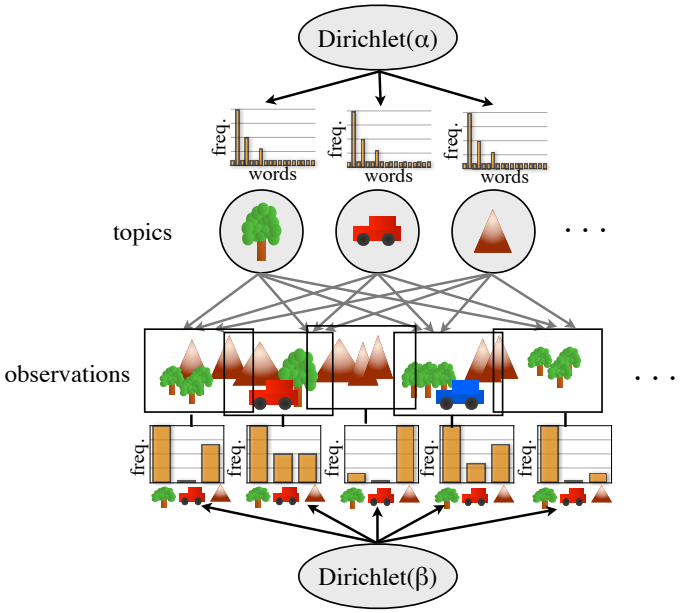


Fig. 3 Spatiotemporal Topics: As a robot observes the world, we would like its observations to be expressed as a mixture of topics with perceptual meaning. We model the topic distribution of all possible overlapping spatiotemporal regions or neighborhoods in the environment, and place a Dirichlet prior on their topic distribution. The topic distribution of the current observation can then be inferred given the topic labels for the neighborhoods in the view. Modeling neighborhoods allows us to use the context in which the current observation is being made to learn its topic labels. To guarantee realtime performance, we only refine a constant number of neighborhoods in each time step, giving higher priority to recently observed neighborhoods.

- The topic model must be updated online and in realtime, since time between two observations is constant. When computing topic labels for a new observation, we must also update topic labels for previous observations in the light on new incoming data.

We address these challenges through the use of a Realtime Online Spatiotemporal Topic modeling (ROST) framework, which we describe in the following sections.

2.3.1 Spatiotemporal Topic Smoothing

Let $M_t = (\{w_i\}, \{\mathbf{x}_i\})$ be the observation at time t, consisting of observed visual words $\{w_i\}$ which take a value between $1 \ldots V$, and their associated spatial coordinates $\{\mathbf{x}_i\}$. The neighborhood of word at $(\mathbf{x}_i, t)$, denoted by G_i, is the set of all words observations in its spatiotemporal neighborhood. This neighborhood could either be defined using k nearest neighbors, or using a radius search. Instead of computing topic distributions over documents in a traditional LDA [2], or image windows in

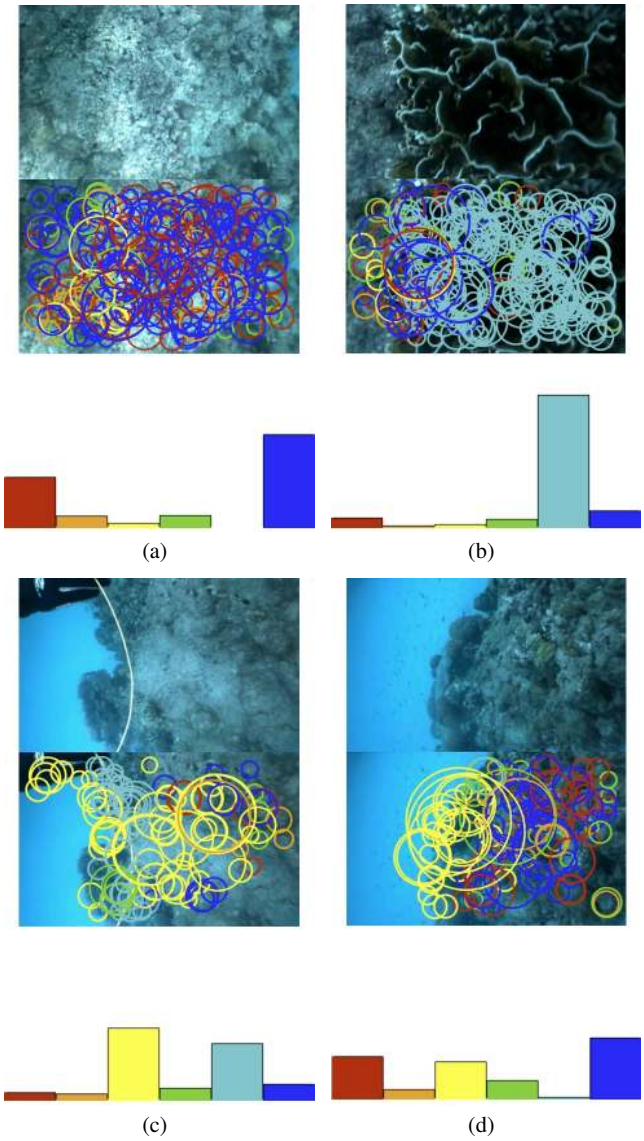


Fig. 4 Example of topics learned on images of the ocean floor taken by the Aqua robot, for a single trajectory. Each visual word is marked by a circle, the size of which corresponds to the size of the visual feature. Histograms depicting the content of each color-coded topic are shown below.

Spatial-LDA [31], we compute topic distributions over these spatiotemporal neighborhoods (Fig. 3). Modeling topic distribution over neighborhoods allows us to use spatiotemporal context in which an observation is being made, which in turn results in much faster convergence as is shown later in our results.

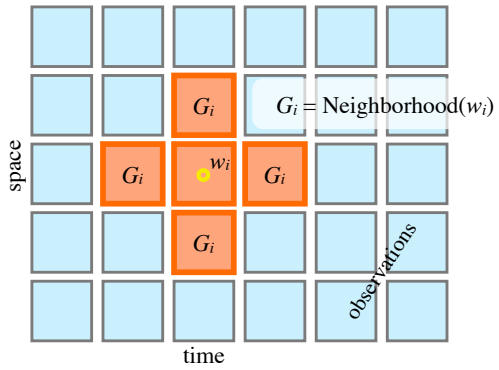


Fig. 5 Each cell shown corresponds to an observation with multiple visual words. We refine the topic label for a word w_i in an observation by taking into account the spatiotemporal context G_i of the observation.

Given a location and time $(\mathbf{x}_i, t)$, we use the following generative model for the observed word w_i:

1. Word distribution for each topic k:

$$\phi_k \sim \text{Dirichlet}(\beta)$$

2. Neighborhood for an observation at $(\mathbf{x}_i, t)$:

$$G_i = \{M_j : M_j \in \text{Neighborhood}(\mathbf{x}_i, t)\}$$

3. topic distribution of the neighborhood G_i :

$$\theta_{G_i} \sim \text{Dirichlet}(\alpha)$$

4. Topic label for a location $(\mathbf{x}_i, t_i)$:

$$z_i \sim \text{Discrete}(\theta_{G_i})$$

5. Word observed at location $(\mathbf{x}_i, t_i)$:

$$w_i \sim \text{Discrete}(\phi_{z_i})$$

where $x \sim Y$ implies that random variable x is sampled from distribution Y.

2.3.2 Gibbs Sampling

Similar to the Gibbs sampler proposed by Griffiths et al.[15], we define the posterior topic distribution of word w_i of observation M_t, with neighborhood G_i:

$$\mathbf{P}(z_i = k | w_i = v, \mathbf{z}_{-i}, \mathbf{w}_{-i}, G_i) \propto \mathbf{P}(w_i = v | z_i = k)\mathbf{P}(z_i = k | G_i) \tag{2}$$

$$= \frac{n^v_{k,-i} + \beta}{\sum_{v=1}^{V}(n^v_{k,-i} + \beta)} \cdot \frac{n^k_{G_i,-i} + \alpha}{\sum_{k=1}^{K}(n^k_{G_i,-i} + \alpha)}, \tag{3}$$

where $n^w_{k,-i}$ counts the number of words of type w in topic k, excluding the current word w_i, and $n^k_{G_i,-i}$ is the number of words with topic label k in neighborhood G_i, excluding the current word w_i, and α, β are the Dirichlet hyper-parameters. Note that for a neighborhood size of 0, $G_i = M_t$, and the above Gibbs sampler is equivalent to the LDA Gibbs sampler.

2.3.3 Realtime Online Gibbs Sampling

Several different strategies exist in the literature to do online refinement of the topic assignment on streaming data [29, 5]. The general idea is to initialize the topic label of the current observation with random labels, and then do a batch refinement of the entire dataset, every time a new document is added. This allows for previous topic assignments to be updated in the light of new observed data. Convergence is guaranteed because in the limit of time going to infinity, the algorithm behaves like a batch Gibbs sampler. However, the problem with such approach is that as the number of observations grow, the model update time grows linearly.

In the context of robotics, the number of refinements between two observation needs to be constant. Hence we randomly sample the observations from a Beta$(a, 1)$ distribution, with $a > 1$, giving higher picking probability to recent observations. This ensures that topic distribution for new observations quickly converge, while older observations are less likely to change their topic assignments. In this work, we set $a = 2$ for all experiments. However, increasing the value of a with time might lead to better results for long experiments. Algorithm 1 shows the realtime topic refinement algorithm.

while *true* **do**
  while *no new observation* **do**
    Randomly sample $r \sim \text{Beta}(a,1)$
    $i \leftarrow \lfloor t * r \rfloor$
    foreach w_j in M_i **do**
      *(*update the topic label for word in the observation *)*
      $G_j \leftarrow \text{Neighborhood}(\mathbf{x}_j, i)$
      $z_j \sim \mathbf{P}(z_j = k | w_j = v, \mathbf{z}_{-j}, \mathbf{w}_{-j}, G_j)$
    end
  end
  $t \leftarrow t+1$
  $\mathbf{M} \leftarrow \mathbf{M} \cup \{M_t\}$
end

Algorithm 1: Refine topic labels, given the current assignment of topics ($\mathbf{z}$) for the set of all observed words($\mathbf{w}$), their locations($\mathbf{X}$), and observation times($\mathbf{t}$).

Figure 4 shows examples of topics which were learnt by running the above topic model on an underwater image sequence containing 2000 images.

3 Summaries and Surprises

Summarizing observations made by a robot has recently gained popularity in robotics[21, 12]. Our goal, however, is to compute a summary which assists in evaluating the novelty of a new observation. We do this by maintaing a summary that is representative of all of observations made thus far, and then compute the surprise score as the distance to the this summary.

3.1 Cost Function

Imagine each observation to be a point in a high dimensional Euclidean space. We then pose the summarization problem as a sampling problem, where we would like to identify observations belonging to the summary set, which minimizes the maximum distance of any observation to its closest observation in the summary.

Let $\mathbf{M}^t = \{M_1, \ldots, M_t\}$ be the set of all observations till time t. We maintain a subset of k observations as the summary $\mathbf{S} = \{S_1, \ldots, S_k\}$, $\mathbf{S} \subseteq \mathbf{M}^t$, such that the maximum distance of an observation to its closest summary sample is minimized. The cost function is thus defined as:

$$Cost(\mathbf{S}|\mathbf{M}^t) = \max_i \min_j d(M_i, S_j), \tag{4}$$

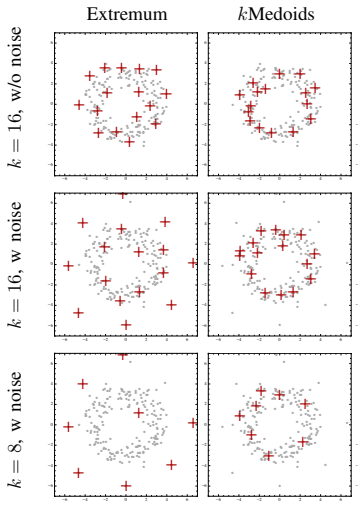


Fig. 6 Extremum vs *k*-medoids Summaries. The dataset consists of 200 points generated randomly around a circle in $\mathbb{R}^2$. The summaries generated by the two algorithms are shown in the first row. Since there are no outliers in the dataset, the summaries seem similar. In the second row, we add 8 extra samples from a different distribution, which are all outliers in the context of the other points. Adding these outliers highlights the differences between the two strategies. We see that extremum summary favors picking the outliers, whereas the *k*-medoids summary ignores these outliers completely. In the last row, we reduce the summary size and see the differences exaggerated even more. The extremum summary is almost entirely made up of the outliers, whereas the *k*-medoids summary is only representative of the mean.

where d is the distance function, which measures distance as the symmetric KL divergence between the corresponding topic distributions, which we describe in Sec 2. Such a summary is sometimes called as an *Extremum Summary* [12], because minimizing the above cost function is essentially minimizing the distance of the worst outlier to the summary. This is different from a more typical a *k*-medoids clustering based summary, which tries to minimize the mean distance of an observation to the closest summary.

If the distance function obeys the triangle inequality, which is true in our case, then not only is this problem NP-hard, but Huse and Nemhauser [17] showed that any α-approximation of this problem is also NP-hard for $\alpha < 2$. Gonzalez [13] proved that the simple greedy solution of recursively picking the farthest samples, has an approximation ratio of 2, which is likely the best we can do unless P=NP.

Figure 6 highlights the characteristic difference in summaries generated by the extremum summary algorithm, and the *k*-medoids algorithm. The dataset consists of 200 points generated randomly around a circle in $\mathbb{R}^2$. The summaries generated by the two algorithms are shown in the first row. Since there are no outliers in the dataset, the summaries seem similar. In the second row of Fig. 6, we add 8 extra samples from a different distribution, which are all outliers in the context of

the other points. Adding these outliers highlights the differences between the two strategies. We see that extremum summary favors picking the outliers, whereas the k-medoids summary ignores these outliers completely. In the last row of Fig. 6, we reduce the summary size and see the differences amplified. The extremum summary is almost entirely made up of the outliers, whereas the k-medoid summary is still only representative of the mean.

Although a k-medoids summary might be useful when we want to model the mean properties of an environment, if however, we are interested in identifying the range of what was observed, then an extremum summary is more useful since its objective function ensures that each observations is close to at least one of the summary samples.

The novelty or surprise of a new observation $\xi(M_t|\mathbf{S})$ is defined as its Hausdorff distance to the summary [11].

$$\xi(M_t|\mathbf{S}) = \min_j d(M_t, S_j). \tag{5}$$

In other words, the summary is chosen to minimize the maximum distance an observation has to the closest observation in the summary. Now, if a new observation is farther still, it is considered as surprising.

3.2 Online Summarization

In the online case, Charikar et al. [6] have proposed a simple strategy where after each pick, the picking threshold is doubled. This leads to a summary which is guaranteed to have a cost less than $8\times$'optimal'. However, since the topic assignment of samples in the summary are continuously being refined, we instead set the threshold dynamically to $2\times$'minimum inter-sample distance in $\mathbf{S}$', as illustrated in Fig. 7.

To control the summary size, we simply use the greedy offline summarization algorithm on the summary to keep the summary of desired size. In our prior work[12], we have studied the rate of growth of the summary, when the threshold is set to the mean distance of a summary sample to the remaining summary. This is useful in the case when we want the summary size to grow with the data.

4 Robot Control

Given the surprise score of the current location, we would like the robot to change its exploration speed such that it spends more time at locations with high surprise score. Achieving this goal has two challenges. First, smoothly mapping the surprise score of the observation to the speed of the vehicle, and second, maintaining the depth and attitude of the vehicle at different speeds.

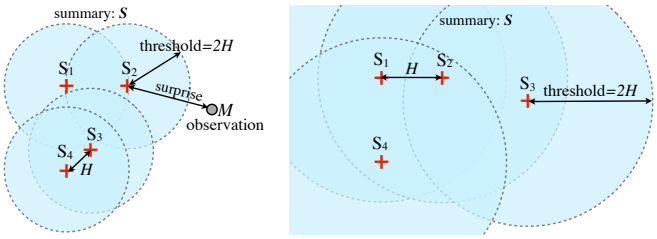


Fig. 7 (a) Given a summary, represented here by the '+' sign, we define the threshold score for updating the summary as twice the smallest inter-sample distance H. When a new observation M arrives, we compute its surprise: the distance to the closest summary sample. If the surprise exceeds the threshold $2H$, then the summary is updated to include the new observation. The updated summary and the threshold are shown in (b).

4.1 Surprise based Speed Control

Let $q_t = \xi(M_t|\mathbf{S})/2H$ be the normalized surprise score of an incoming observation at time t. We then set the speed(v) of the robot by mapping the surprise score through a sigmoid function:

$$v(t) = \frac{1}{1+e^{\gamma(q_t-0.5)}}, \tag{6}$$

where γ controls the *responsiveness* of the robot. A higher γ made the scheduling of the forward velocity v more aggressive. The use of this sigmoid function allows the robot to smoothly transition between being completely still when the observation is is surprising enough to update the summary, to moving at full speed when traversing over a previously seen environment. We calculated γ empirically, and found $\gamma = 10$ to perform well during our sea trials. A plot of the sigmoid function with $\gamma = 10$ is shown in Fig. 8.

4.2 Depth and Attitude Control

A major difficulty in operating underwater robots, especially at low speed, is the fact that they move with six degrees of freedom (x, y, z and the three Euler angles roll ϕ, pitch θ, and yaw ψ) in a dynamic fluid. Their motion relies on the dynamic pressures induced by the moving water impacting the different surfaces. Some of these surfaces are specifically designed to facilitate the control of the robot: in our case, these are the 6 flippers. A unique characteristic of our robot is that these six surfaces serve both as propulsion mechanism as well as control surfaces. Thrust is generated by oscillating these flippers rapidly around an offset angle. At the same

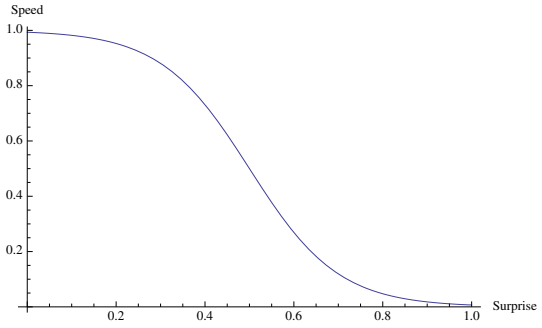


Fig. 8 Mapping a surprise score to robot speed using a sigmoid function results in smooth transition between being completely when something surprising has been observed, to moving at full speed when observing what has been observed before.

(a) (b) (c)

Fig. 9 Pictures showing the flippers' angle due to the action of the autopilot system, during one of the sea trials. (a) the robot is performing a heave-up maneuver to maintain depth and attitude at zero forward speed, corresponding to strategy S_2. (b) the robot is executing a combined heave up, pitch up and slow forward speed maneuver. (c) the robot is performing a pitch-up maneuver at high speed, corresponding to strategy S_1.

time, this offset angle means that, on average, a lift force will be generated by the dynamic pressure impinging on them [10]. Thus, the dynamics and controllability of the robot will be heavily dependent on the forward velocity.

We employed two different strategies to maintain depth. At higher forward velocities ($v > 0.2$), depth was maintained via pitch angle changes, and the robot executing such maneuver can be seen in Fig. 9(c). When the robot had no forward velocity ($v = 0$), maintaining depth required the use of a heaving thrust. This motion was accomplished by having the 6 flippers pointing upward or downward, as shown by the robot in Fig. 9(a). This way, the net thrust produced by the oscillating flippers does not induce forward motion. Attitude stabilization was still possible with this leg configuration, by means of a forward/aft differential thrust for pitch corrections and left/right differential thrust for roll corrections. For low velocities ($v < 0.2$), the robot flippers were placed so as to generate both heaving and forward motion (Fig. 9(b)). All of these pictures were taken from a single trial, demonstrating the need to adapt the locomotion strategy in order to satisfy motion requirements.

In Extension 1(part 3), we show a live demonstration of the underwater vehicle, as it traverses an underwater environment.

5 Experiments

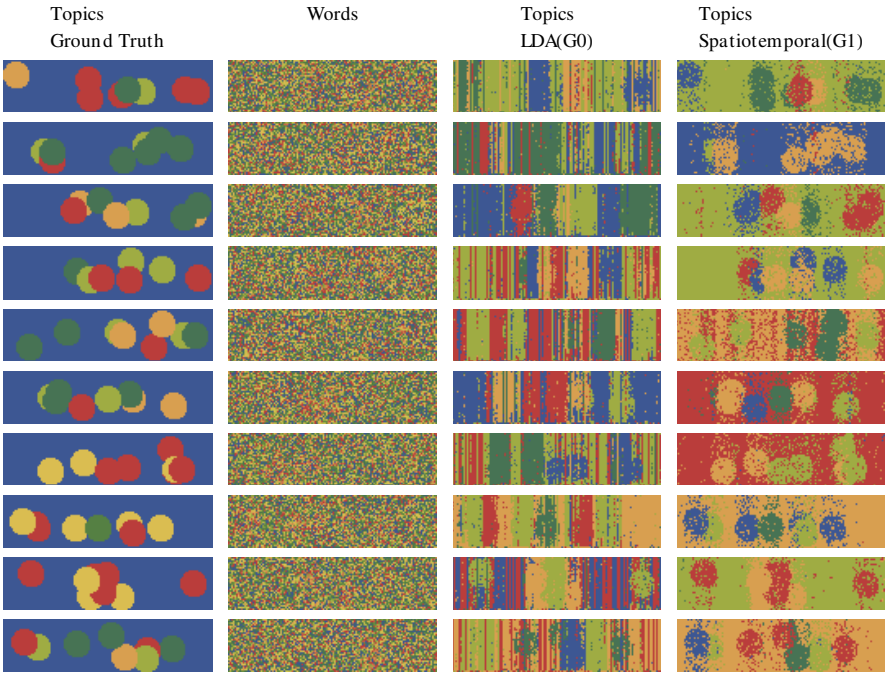
5.1 Evaluation of Spatiotemporal Topics

To test the effectiveness of the proposed spatiotemporal topic modeling, we used artificially generated datasets, for which the ground truth is known. Each dataset was generated in the following manner. As shown in Fig. 10(a) column 1, we generated ground truth topic labels such that each column corresponds to an observation, and each pixel corresponds to a word. Each dataset has 8 objects(disks), spanning multiple observations, and are placed randomly. Each object has a topic label $z \in [0,K)$ that is chosen randomly from one of the $K = 5$ topics. Topic labels are shown by the pixel color. Given the topic label of a word $z = k$, we generated word labels by randomly picking $v \in [0,V)$ from distribution $\mathbf{P}(v|z)$. We used vocabulary size $V = 100$. The vocabulary was equally divided between the K topics such that V/K words were preferred exclusively by each topic. To add noise, we used a non-zero probability of emitting a word not related to the given topic.

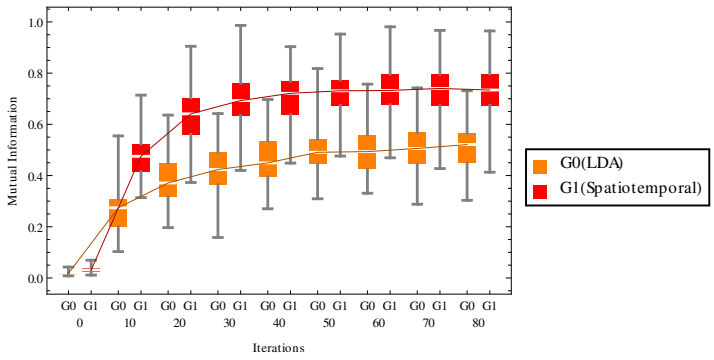
$$\mathbf{P}(v|z) \propto \begin{cases} V/K + \eta & \text{if v is related to } z \\ \eta & \text{otherwise} \end{cases} \tag{7}$$

We used $\eta = 0.1$ for all our experiments. As a result, the probability that the sampled word was preferred by the given topic was 0.71, and with probability 0.29 the sampled word was preferred by another topic. Examples of the resulting words are shown in Fig. 10(a) column 2. We can see that original pattern is invisible to human eyes. Finally, we used these generated words as input to the proposed spatiotemporally smoothed topic model, and refined the topics in batch mode for 80 iterations. A neighborhood size of 1 implies that while refining each observation, represented as a column in the dataset image, we take into account the two adjacent columns. A neighborhood size of 0 corresponds to the traditional LDA, where each column is refined independently. Fig. 10(a) columns 3,4 show the resulting topic labels. We used hyper-parameter values $\alpha = 0.1$ and $\beta = 1.0$ for all our experiments. We experimented with 100 randomly generated datasets, out of which the first 10 are shown in Fig. 10(a). We see that in most cases, use of the proposed technique results in much more accurate topic labels.

To quantitatively evaluate these results, we computed the Mutual Information between the ground truth topic labels and the proposed methods. Mutual Information $I(X,Y)$ essentially measures the reduction in Entropy of a random variable X, after observing random variable Y.



(a) Artificially generated datasets along with the topics computed by the topic models.



(b) Mutual Information between the computer topic labels and the ground truth.

Fig. 10 We generated 100 artificial datasets out of which we show the first 10 in (a). Each column corresponds to an observation, and each pixel corresponds to a word. The color of the pixel corresponds to the topic/word label. In (b) we show the Mutual Information between the results and the ground truth. We see that topic model which takes into account its two adjacent neighbors (G1), consistently outperforms the topic model which does not use the neighborhood information (G0).

$$I(X,Y) = H(X) - H(X|Y) \tag{8}$$

$$= \sum_{x,y} \mathbf{P}(x,y) \log \frac{\mathbf{P}(x,y)}{\mathbf{P}(x)\mathbf{P}(y)} \tag{9}$$

To compute Mutual Information between the ground truth labels, and the topic model generated labels, we set x to topic label from the ground truth, y to the topic labels produced by the topic models. Figure 10(b) shows the box-whisker plot of the mutual information between the ground truth, and the two topic models, for 100 datasets. The box corresponds to 75% and 25% quantiles, and the line in the middle is the median mutual information across all datasets. The whiskers show the minimum and maximum range of the Mutual Information score. We see that the proposed spatiotemporal topic model clearly outperforms traditional LDA, as it has higher Mutual Information score.

5.2 Perplexity Convergence

Computing perplexity is a common way of evaluating language models. Per word perplexity of a document could be intuitively interpreted as the uncertainty in recreating the word labels from the topic label distribution being used by the model to describe a document.

Per word perplexity of an observation M_t is defined as:

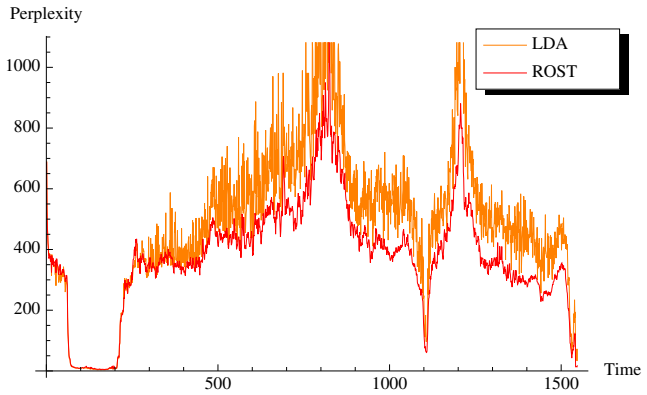
$$\text{Perplexity}(M_t) = exp\left(-\frac{\sum_i^W \log p(w_i|M_t)}{W}\right), \tag{10}$$

where W is the number of words in observation M_t, and w_i is the ith word in the observation. The log likelihood of a word in an observation can be computed using an expression similar to Eq. 3, with the difference that instead of considering $\mathbf{P}(z|G)$, the probability of a topic given its neighborhood, we use $\mathbf{P}(z|M)$, the probability of a topic given the observation to which it belongs. Computing perplexity of an observation, instead of a neighborhood, allows us to compare the performance of ROST, with normal LDA, which ignores the spatiotemporal neighborhood.

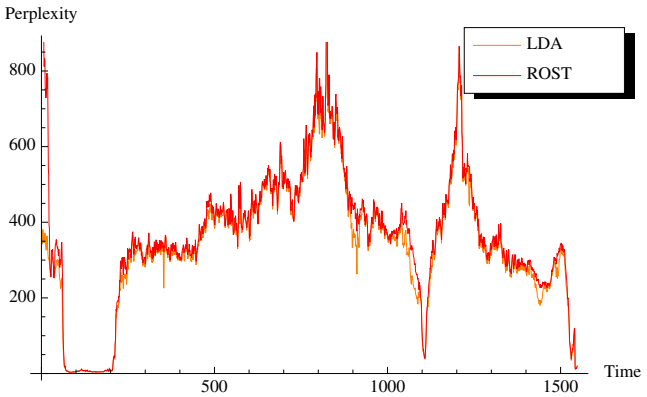
In Fig. 11(a), we plot the perplexity of an observation, immediately before we receive the next observation; i.e., we refine the model until the next observation is available, and then we output the perplexity. We see that ROST consistently converges to a lower perplexity, when compared to a simple LDA based refinement. In the latter, we do not bias the refinement towards the present time, and do not take into account the spatiotemporal neighborhood of the observations. To make LDA work in realtime, we sample the observations to be refined uniformly from all observations thus far.

Even though ROST uses temporally biased refinement, in Fig. 11(b) we see that the final observation perplexity, measured after all observations have been process, is almost indistinguishable from LDA.

The dataset used in Fig. 11 consists of 1500 images observed in an underwater environment. Each image was of size 1024x640, and was split into non-overlapping windows of size 160x160. Each of these sub-windows was considered as an observation, for a total of 42000 observations. Each observation had maximum of 6



(a) Online Perplexity



(b) Final Perplexity

Fig. 11 Per image perplexity for a dataset consisting of 42000 observations at 1500 time step, in an underwater environment. (a) shows perplexity of a new observation computed online, immediately before the next observation was processed. We see that ROST consistently has lower perplexity, compared to LDA. (b) shows perplexity of the observations in the dataset, after the last observation was made. We see that final perplexity of LDA and ROST is similar.

neighbors, 4 spatial and 2 temporal, forming its neighborhood. We used a vocabulary size of 5180, of which 5000 were ORB features, and 180 were words describing hue of a region. The dataset had a total of 1.63M words. The observations were made at the rate of 5Hz.

5.3 Evolution of Topics in an Indoor Environment

Topics learned by ROST evolve over time as more data is observed. Fig. 12 shows an example of this topic evolution in and indoor environment. The images (a)-(d) are in

temporal sequential order, and show the growth of the topic model as time elapses. The circles correspond to extracted visual words, and the color corresponds to the different topic labels. Initially, we see that the entire scene is labeled with the same topic label, as is shown in Fig. 12(a). After a few time steps, we see that ROST is successfully able to segment the scene using two topics corresponding to the upper and lower half of the room (Fig. 12(b)). Eventually, we see that many more topic labels are being used to represent different parts of the room (Fig. 12(c,d)).

In Extension 1(part 1), we show a video demonstration of ROST topic labels being used to summarize an indoor environment. The video shows different topics quickly converging to different parts of the room, such as: the bookshelf, the ceiling, the windows, and the pin-up board. The video also show the summarizer using these topic distributions to pick images from different parts of the scene. An orange box around a summary image identifies the summary image closest to the current observation.

5.4 Underwater Enviroments

Some examples of the topics learned in an underwater environment are shown in Fig 4, and also in Video Extension 1(part 2). We see that the topics are representative of underlying physical phenomenon, and do well in describing scenes where a mixture of these exist. Red and blue topics are being used to represent rocks in the dataset, yellow for the sand-rock boundary, and cyan for the fire coral and the white rope. We set both summary size and topic size to 6 for our experiments. The hyper-parameters for the LDA were determined empirically.

Fig. 13(a) shows an example of the final summary generated by our online topics based summarizer from a sample trajectory. The corresponding histograms show the distribution of topics in the image. Fig. 13(b) shows uniformly sampled images over the same trajectory, presented here for comparison. We see that the proposed algorithm was able to recognize different species of corals (images 2 and 3), and the accidental inclusion of a diver's hand with a rope (image 6). When these images were observed, the robot evaluated them as surprising, and as a result, slowed down to a halt. Once these images were added to the summary, the surprise score falls instantly, and the robot continued forward in search for new surprises.

In Video Extension 1(part 3), we show a live demonstration of an AUV as it traverses an underwater environment. Its speed is controlled by the surprise score, and as a result we see the robot stopping over different, previously unobserved visual features, and then moving on at higher speeds when there is nothing of surprise.

6 Conclusion

We have demonstrated a novel autonomous robotic system that can be used to assist in semantic exploration of an environment. This required the development of a novel exploration technique, which first semantically models the scene, and then controls the speed of exploration based on the surprise score by the semantic model.

We presented ROST, a realtime online spatiotemporal topic modeling framework, which attempts to model semantics of the streaming observed visual data. Using ROST, we compute a surprise score of incoming observations, which is sensitive to presence of high-level patterns in the scene, such as different coral species, rocks, and sand. We showed that the proposed technique results in significantly better topic labels, when compared with Latent Dirichlet Allocation based topic model. Moreover, the topic labels computed using ROST converge quickly, and are suitable for use in autonomous agents working with realtime constraints.

Given a fixed trajectory, we demonstrated the robot traversing it with a non-uniform speed, stopping at locations containing surprising observations, and moving at high speeds over seemingly boring or previously observed regions. The resulting summaries produced by our system were able to capture the visual diversity of the underwater environment.

Our ongoing future work is focused on developing better realtime online topic modeling techniques, such as the use of nonparametric hierarchical Dirichlet processes, and their use in control of different robotic platforms for exploration tasks.

Acknowledgment

The authors would like to thank Ioannis Rekleitis for his underwater videography, and Doina Precup and Luc Devroye for their helpful discussions.

Appendix A: Index to Multimedia Extensions

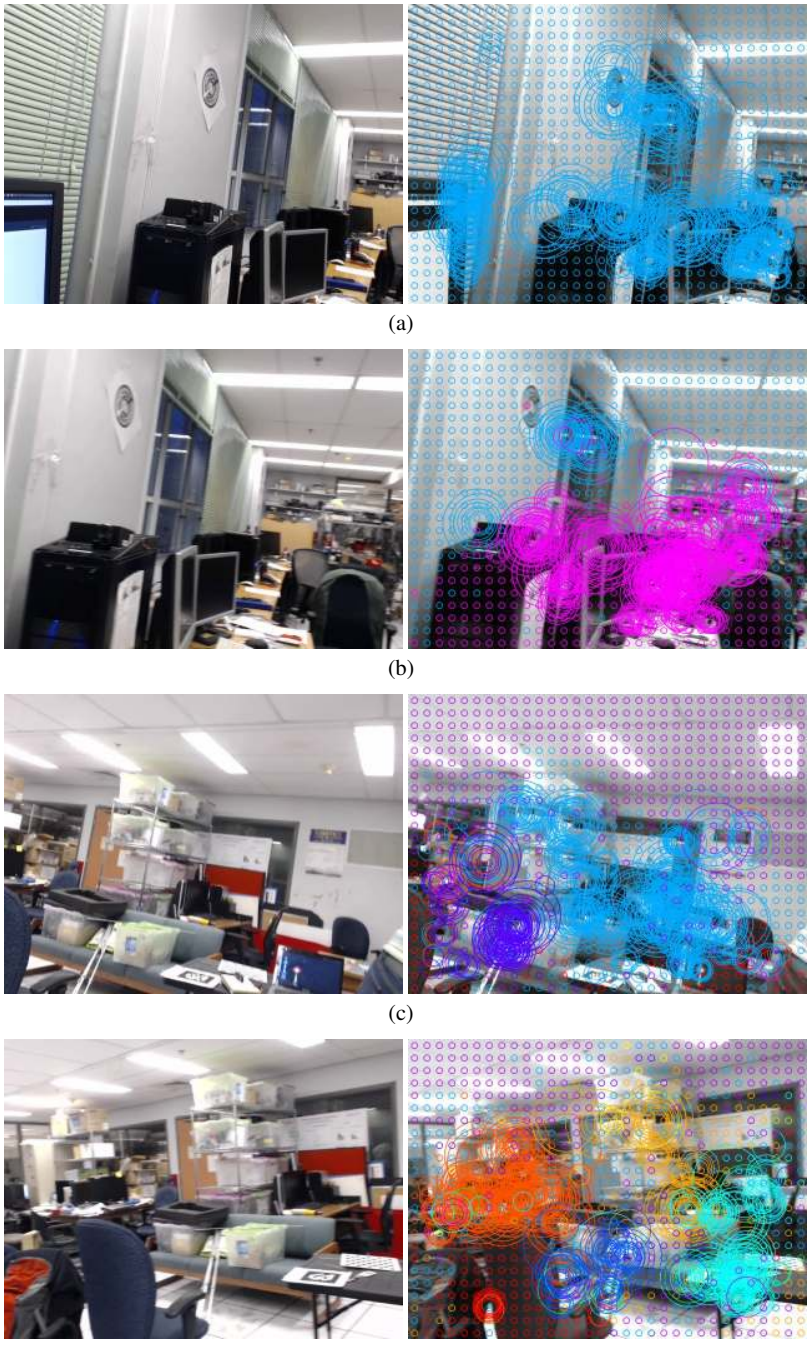
Extension	Media Type	Description
1	Video	Demonstration of ROST topic modeling framework, and its use in an underwater robot for autonomous exploration.

References

1. H. Bay, T. Tuytelaars, and L. V. Gool. SURF: Speeded Up Robust Features. *European Conference on Computer Vision*, 2006.

2. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.
3. A. Bosch, A. Zisserman, and X. Muñoz. Scene Classification Via pLSA. In A. Leonardis, H. Bischof, and A. Pinz, editors, *Computer Vision – ECCV 2006*, volume 3954 of *Lecture Notes in Computer Science*, pages 517–530. Springer Berlin / Heidelberg, 2006.
4. E. Bourque and G. Dudek. Automated Image-Based Mapping. In *Proceedings of the Workshop on Perception of Mobile Agents, Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 61–70, June 1998.
5. K. R. Canini, L. Shi, and T. L. Griffiths. Online Inference of Topics with Latent Dirichlet Allocation. *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 5(1999):65–72, 2009.
6. M. Charikar, C. Chekuri, T. Feder, and R. Motwani. Incremental clustering and dynamic information retrieval. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing - STOC '97*, pages 626–635, New York, New York, USA, May 1997. ACM Press.
7. J. Das, F. Py, T. Maughan, T. O'Reilly, M. Messie, J. Ryan, G. S. Sukhatme, and K. Rajan. Coordinated sampling of dynamic oceanographic features with underwater vehicles and drifters. *The International Journal of Robotics Research*, 31(5):626–646, Apr. 2012.
8. G. Dudek and J.-P. Lobos. Towards Navigation Summaries: Automated Production of a Synopsis of a Robot Trajectories. *2009 Canadian Conference on Computer and Robot Vision*, pages 93–100, May 2009.
9. L. Fei-Fei and P. Perona. A Bayesian hierarchical model for learning natural scene categories. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 524 – 531 vol. 2, June 2005.
10. P. Giguere, G. Dudek, and C. Prahacs. Characterization and modeling of rotational responses for an oscillating foil underwater robot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3000–3005, Oct. 2006.
11. Y. Girdhar and G. Dudek. ONSUM: A System for Generating Online Navigation Summaries. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS)*, pages 746–751, Oct. 2010.
12. Y. Girdhar and G. Dudek. Efficient on-line data summarization using extremum summaries. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3490–3496, 2012.
13. T. F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985.
14. R. Grabowski, P. Khosla, and H. Choset. Autonomous exploration via regions of interest. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, volume 2, pages 1691–1696. IEEE, 2003.
15. T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5228–5235, 2004.
16. T. Hofmann. Unsupervised Learning by Probabilistic Latent Semantic Analysis. *Machine Learning*, 42(1):177–196, 2001.
17. W.-L. Hsu and G. L. Nemhauser. Easy and hard bottleneck location problems. *Discrete Applied Mathematics*, 1(3):209–215, 1979.
18. M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert. iSAM2: Incremental smoothing and mapping using the Bayes tree. *The International Journal of Robotics Research*, 31(2):216–235, Dec. 2011.
19. D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 2004.
20. P. Newman, G. Sibley, M. Smith, M. Cummins, A. Harrison, C. Mei, I. Posner, R. Shade, D. Schroeter, L. Murphy, W. Churchill, D. Cole, and I. Reid. Navigating, Recognizing and Describing Urban Spaces With Vision and Lasers. *The International Journal of Robotics Research*, 28(11-12):1406–1433, July 2009.

21. R. Paul, D. Rus, and P. Newman. How was your day? Online Visual Workspace Summaries using Incremental Clustering in Topic Space. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2012.
22. I. Posner, D. Schroeter, and P. Newman. Online generation of scene descriptions in urban environments. *Robotics and Autonomous Systems*, 56(11):901–914, Nov. 2008.
23. P. Rigby, O. Pizarro, and S. Williams. Toward Adaptive Benthic Habitat Mapping Using Gaussian Process Classication. *Journal of Field Robotics*, 27(6):741–758, 2010.
24. E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB : an efficient alternative to SIFT or SURF. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2564—-2571, 2011.
25. J. Sattar, G. Dudek, O. Chiu, I. Rekleitis, P. Giguère, A. Mills, N. Plamondon, C. Prahacs, Y. Girdhar, M. Nahon, and J.-P. Lobos. Enabling Autonomous Capabilities in Underwater Robotics. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS)*, pages 3628–3634, Nice, France, Sept. 2008.
26. J. Sivic, B. C. Russell, A. Zisserman, W. T. Freeman, and A. A. Efros. Unsupervised discovery of visual object class hierarchies. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, 2008.
27. J. Sivic and A. Zisserman. Video Google: Efficient Visual Search of Videos. In J. Ponce, M. Hebert, C. Schmid, and A. Zisserman, editors, *Toward Category-Level Object Recognition*, volume 4170 of *Lecture Notes in Computer Science*, pages 127–144. Springer Berlin / Heidelberg, 2006.
28. R. N. Smith, M. Schwager, S. L. Smith, B. H. Jones, D. Rus, and G. S. Sukhatme. Persistent Ocean Monitoring with Underwater Gliders: Adapting Sampling Resolution. *Journal of Field Robotics*, 28(5):714–741, 2011.
29. X. Song, C.-Y. Lin, B. L. Tseng, and M.-T. Sun. Modeling and predicting personal information dissemination behavior. In *Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining - KDD '05*, page 479, New York, New York, USA, Aug. 2005. ACM Press.
30. S. Thrun, S. Thayer, W. Whittaker, C. Baker, W. Burgard, D. Ferguson, D. Hannel, M. Montemerlo, A. Morris, Z. Omohundro, and C. Reverte. Autonomous exploration and mapping of abandoned mines. *IEEE Robotics & Automation Magazine*, 11(4):79–91, Dec. 2004.
31. X. Wang and E. Grimson. Spatial Latent Dirichlet Allocation. In *Advances in Neural Information Processing Systems*, volume 20, pages 1577–1584, 2007.
32. P. Whaite and F. Ferrie. Autonomous exploration: driven by uncertainty. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(3):193–205, Mar. 1997.
33. B. Yamauchi. A frontier-based approach for autonomous exploration. In *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97. 'Towards New Computational Principles for Robotics and Automation'*, pages 146–151. IEEE Comput. Soc. Press, 1997.



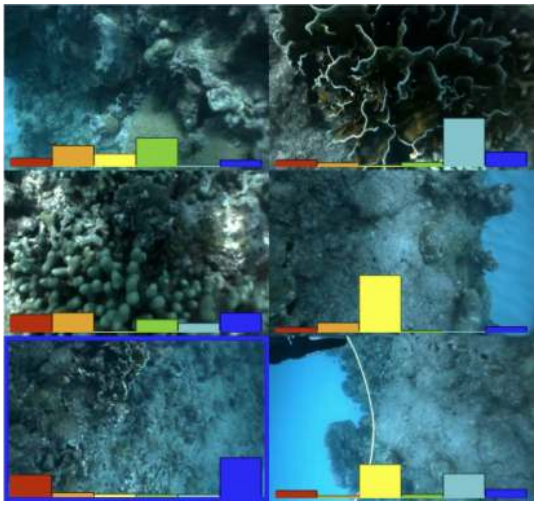
(a)

(b)

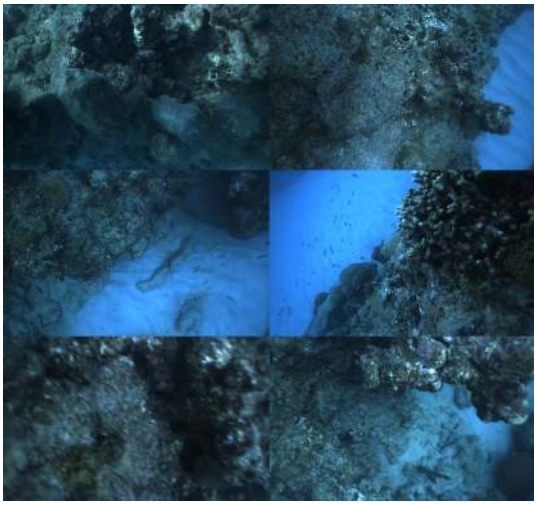
(c)

(d)

Fig. 12 Topics learned by ROST evolve over time to capture the growing complexity of the scene. The figures are temporally sequenced, and show more topics being used to label different parts of the scene as more time elapses.



(a) Topics



(b) Uniform

Fig. 13 (a) A summary of six images generated online by the system. The histogram shows the distribution of visual topics in an image, each color corresponding to a different topic. (b) For comparison we show images sampled uniformly over the robot trajectory.