*Thesis Summary:*

# Active Visual SLAM with Exploration for Autonomous Underwater Navigation

Ayoung Kim

## 1 Introduction

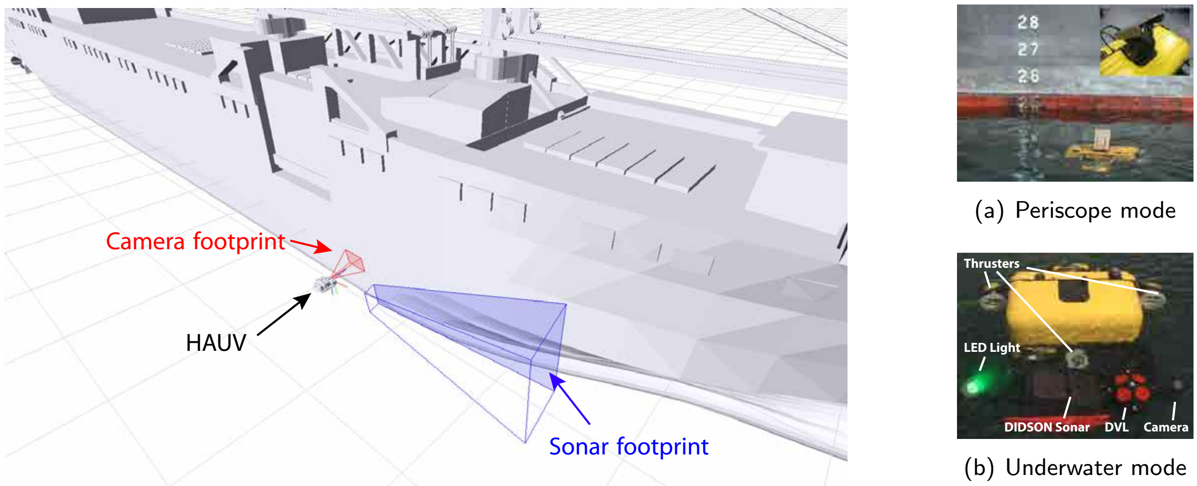
Autonomous underwater vehicles (AUVs) have played an important role in scientific research due to their ability to collect data in underwater environments that are either inaccessible or too dangerous for humans to explore. Other than explorations, underwater structures such as dams, ship hulls, harbors and pipelines also need to be periodically inspected for assessment, maintenance and security reasons. Autonomous vehicles have the potential for better coverage efficiency, improved survey precision and overall reduced need for human intervention.

In comparison to terrestrial navigation, underwater navigation is challenging because the opacity of water to electromagnetic waves precludes the use of the global positioning system (GPS) and other high speed radio communication. Due to the lack of accurate position information from GPS, traditional underwater navigation methods have been used to solve the navigation problem using acoustic signals. Traditionally, navigation limitations can be overcome by using a simultaneous localization and mapping (SLAM) algorithm to fuse sensor measurements derived from vision and/or sonar. Similar to human navigation, in which sight confirms our position when we recognize a previously visited scene, visual measurements significantly reduce the uncertainty when a site is revisited and recognized (i.e., loop-closing). The advantage of visual SLAM arises from these loop-closure camera measurements, which add independent constraints to the pose-graph, and greatly reduce position uncertainty as compared to pure odometry (dead reckoning).

Despite this major contribution in reducing uncertainty, visual measurements may not be uniformly available in an underwater environment where the spatial feature distribution varies greatly. This indicates that successful measurements strongly depend upon two factors: (*i*) the saliency of visual features and (*ii*) their spatial distribution as seen by the robot. The first factor, saliency, is an image measurement that represents distinguishability of a visual feature. The second factor, the observed spatial distribution, is mainly determined by the environment and egomotion of the robot (e.g., path and gaze). A similar interrelation has been proposed by [1] who found that changes in trajectory can result in better navigation. However, we should note that these two factors should be considered simultaneously for better navigation results, especially for underwater images where the possibility of making a valid registration may not be as uniform as in the terrestrial environment. Based upon this motivation, this thesis's goal has been to develop a control scheme for better navigation by providing trajectory perturbations to improve pose observability under the consideration of a visual saliency map.

## 2 Real-time Pose-graph Visual SLAM

The real-time pose-graph visual SLAM presented in this thesis is an extension of visually augmented navigation (VAN) [2] to include geometric model selection for robust image registration, mainly focusing on pose-graph visual SLAM.




(a) Periscope mode

(b) Underwater mode

Figure 1: Illustration of the in-water hull inspection application using an AUV. Depicted is the Hovering Autonomous Underwater Vehicle (HAUV) [3] performing inspection on a hull using camera and sonar sensor modalities. The individual sensor footprints are plotted with red (camera) and blue (sonar) cones. On right column, two camera configurations are provided.

## 2.1 Two-view Camera Registration Engine

This section discusses the core of our visual SLAM perception engine, namely the camera registration engine. Given image features and a proposed overlapping image pair, the process of pairwise motion estimation involves four main steps: 1) image feature extraction; 2) establishing inlier correspondences; 3) fitting a geometric registration model using a robust estimation framework; and 4) optimizing the camera relative-pose constraint. A block-diagram of the overall image registration algorithm is depicted in Figure 2.

**Feature Extraction:** The first step in the two-view camera engine is to extract robust feature points for putative matching. Images are first radially undistorted and contrast corrected using contrast-limited adaptive histogram equalization [4]. This enhances the visual detail and aids in feature extraction. We use a combination of Scale Invariant Feature Transform (SIFT) [5] and Speeded Up Robust Features (SURF) [6] feature detectors/descriptors.

**Pose-Constrained Correspondence Search:** Once the features have been extracted, we use our SLAM pose prior knowledge to guide the pairwise putative matching. This matching is formulated in terms of a probabilistically-driven pose-constrained correspondence search (PCCS) [7, 8]. PCCS allows us to spatially restrict the putative correspondence search region and, thereby, register what would otherwise be feature-sparse underwater imagery.

**Geometric Model Selection:** Once we have established putative correspondences, we can then refine these to determine an inlier correspondence set via a random sample consensus (RANSAC) model-selection framework [9]. One may expect that a locally planar structure assumption is adequate on the open areas of the hull; however, this assumption is not everywhere true because some portions of the hull are highly three-dimensional (e.g., bilge keel, shaft, screw). To accommodate for this structure variability, we employ a geometric model-selection framework to automatically choose the proper registration model, either homography or essential matrix, when robustly determining the inlier set. In this work, we have adopted Torr's Geometric Information Criterion (GIC) [10].

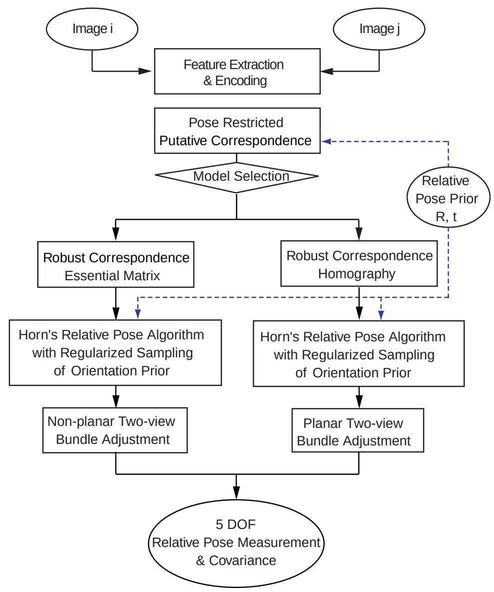


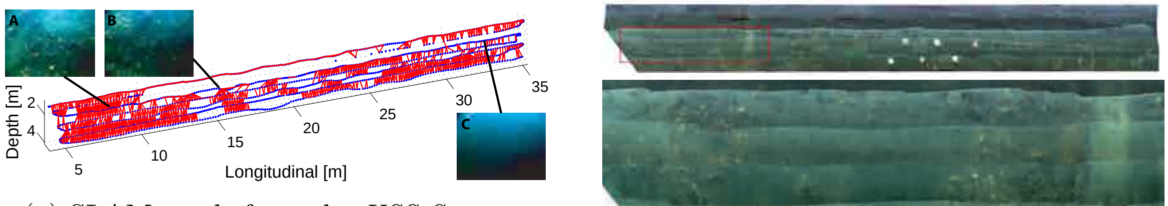

Figure 2: Block-diagram of the core image registration engine.

**Two-view Bundle Adjustment:** After selecting the proper model via geometric model selection, the model and set of correspondences are optimized within a two-view bundle adjustment. Horn's relative-pose algorithm [11] is applied prior to the bundle adjustment to refine the inliers set and provide an initial guess of the relative-pose. Then, a two-view bundle adjustment estimates the optimal relative-pose constraint between the two cameras. In either case (homography or essential matrix), the two-view bundle adjustment results in a 5-degree of freedom (DOF) bearing-only camera measurement, and a first-order estimate of its covariance [12], which is then published as a constraint for SLAM back-end.

## 2.2 Implementation and Results

We present real-time visual SLAM implementation for in-water hull inspection of the aircraft carrier *USS Saratoga*. The experiment consisted of seven 30 m legs of a boustrophedon survey, each spaced 0.5 m apart in depth. The camera and Doppler velocity log (DVL) were mounted on tilt actuators so that they approximately maintained a nadir view to the hull. The standoff position of the robot was controlled at 1.5 m from the hull throughout the experiment with a horizontal trajectory speed of 0.5 m/s. In this experiment, the incremental smoothing and mapping (iSAM) algorithm was selected as the SLAM back-end.

The final trajectory resulting from visual SLAM with model selection is shown in Figure 3(a). The red lines indicate pose constraints derived from image pairs. The camera was restarted while the robot was returning to the bow in the second leg of the trajectory. This resulted in a blank section in the middle of the second track-line during which the uncertainty ellipsoid inflated as the robot moved without camera measurements over this section. However, once the camera was restarted at the end of the second leg, the SLAM algorithm re-localized the vehicle by adding additional pose constraints to the previous track (first leg). The distribution of visual features strongly affects the camera measurement distribution. As the robot hovers and moves from the feature-less region (C in Figure 3(a) toward stern right) to the feature-rich region (A and B in Figure 3(a) toward bow left), we see a strong increase in the number of camera measurements.




(a) SLAM result from the *USS Saratoga*

(b) Texture mapped reconstruction

Figure 3: SLAM and 3D photomosaic result from the *USS Saratoga*. (a) The camera-derived SLAM pose constraints for inspection of the *USS Saratoga* are shown with successful camera links in red. Each vertex represents a node in the pose-graph where the red lines indicate the 5-DOF camera measurements. Images A and B show sample images from the feature-rich region and image C presents an example image from the feature-less region. (b) 3D photomosaic of *USS Saratoga* with a zoomed view.

As a by-product of the SLAM trajectory estimation, the 3D structure of the ship hull can be reconstructed using the final trajectory estimate and the pairwise image correspondences. Surface fitting with texture mapping was conducted to generate a 3D photomosaic (Figure 3(b)). Note that the major purpose of having the 3D photomosaic is to have qualitative verification of the SLAM result. Because the photomosaic was generated using the final SLAM estimate and back projected image points (without additional blending at the image processing level), the continuity in the image seams between tracks illustrates that the resulting pose-graph is self-consistent. During the mission, six artificial targets, pre-installed on the hull to verify the SLAM navigation performance, are all found as indicated by the six white circles that appear in the texture-mapped reconstruction.

# 3 Visual Saliency for SLAM

We develop two novel metrics to measure an image's visual saliency for SLAM. The first is local saliency, which is capable of measuring image registrability correlates well with successful camera links. The second is global saliency, which identify visually distinct (i.e., rare) scenes with respect to the rest of the hull. Both are computed using a bag-of-words (BoW) model for image representation. Registrability refers to the intrinsic feature richness of an image (i.e., the amount of texture an image has). The lack of image texture, as in the case of mapping an underwater environment with feature-poor regions, prevents image registration from being able to measure the relative-pose constraint. However, texture is not the only factor that defines saliency—an easy counterexample is an image of a checkerboard pattern or a brick wall. Images of these type of scenes have high texture, but likely will fail registration due to spatial aliasing of common features.

**Local Saliency:** We define local saliency as an intra-image measure of feature diversity. In defining the diversity, entropy has been widely used as a measure of randomness, though in other works it has been primarily used on image channels. We assess the diversity of words occurring within image $I_i$ by examining the entropy of its BoW histogram, which works regardless of color channel availability, $H_i = -\sum_{k=1}^{W(t)} p(w_k) \log_2 p(w_k)$. Here, $p(w)$ is the empirical BoW probability distribution within the image computed over the set of vocabulary words $\mathcal{W}(t) = \{w_k\}_{k=1}^{W(t)}$ where $W(t)$ is the size of the vocabulary, which grows with time since we build the vocabulary online. Hence, to normalize our entropy measure, we use the ratio of $H_i$ to the maximum possible entropy to yield a normalized entropy measure $S_{L_i} \in [0, 1]$:

$$S_{L_i} = \frac{H_i}{\log_2 W(t)}. \tag{1}$$

This entropy-derived measure captures the diversity of words (descriptors) appearing within an image.
**Global Saliency:** We define global saliency as an inter-image measure of the uniqueness or rarity of features occurring within an image. To tackle this problem, we use inverse document frequency (idf) [13, 14, 15] to detect rare words and expect a high idf for words (descriptors) that are rare in the dataset. $\mathcal{G}_i(t) = \sum_{k \in \mathcal{W}_i} \log_2 \frac{N(t)}{n_{w_k}(t)}$. Here, $\mathcal{W}_i \subset \mathcal{W}(t)$ represents the subset of vocabulary words occurring within image $I_i$, $n_{w_k}(t)$ is the current number of documents in the database containing word $w_k$, and $N(t)$ is the current number of documents in the database. To guarantee independent sample statistics used in our idf calculation, only spatially distinct (i.e., non-overlapping) images are used to update $n_{w_k}(t)$ and $N(t)$. Lastly, as was the case with our local saliency measure, we normalize the rarity measure for image $I_i$ to have a normalized global saliency score $S_{G_i} \in [0, 1]$:

$$S_{G_i}(t) = \frac{\mathcal{G}_i(t)}{\mathcal{G}_{\mathsf{max}}}, \tag{2}$$

where the normalizer, $\mathcal{G}_{\mathsf{max}}$, is the maximum summed idf score encountered thus far.

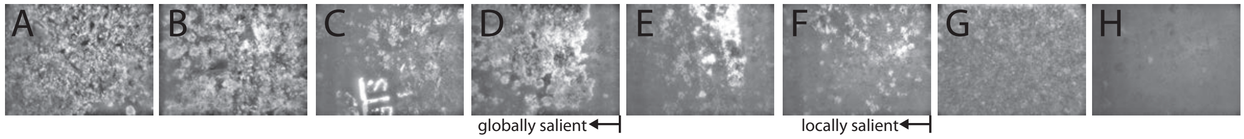
### 3.1 Saliency-informed Visual SLAM

Using our previously defined local saliency measure, we can improve the performance of visual SLAM in two ways:

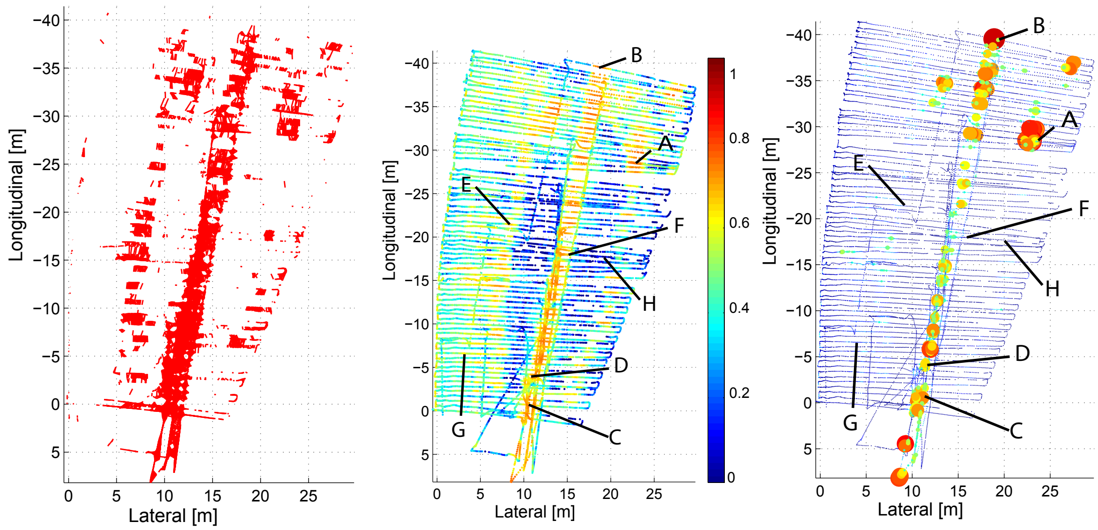
1. We can sparsify the pose-graph by retaining only visually salient keyframes;
2. We can make link proposals within the graph more efficient and robust by combining visual saliency with measures of geometric information gain.

In the first step, we can decide whether or not a keyframe should be added at all to the graph by evaluating its local saliency level—this allows us to cull visually homogeneous imagery, which results in a graph that is more sparse and visually informative. This improves the overall efficiency of graph inference and eliminates nodes that would otherwise have low utility in underwater visual perception.

In the second step, we can improve the efficiency of link proposal by making it "salient-aware". For efficient link proposal, [16] used expected information gain to prioritize which edges to add to the graph—thereby retaining only informative links. However, when considering the case of visual perception, not all camera-derived measurements are equally obtainable. Pairwise registration of low local saliency images will fail unless there is a strong prior to guide the putative correspondence search, whereas pairwise registration of highly salient image pairs often succeeds even with a weak or uninformative prior. Hence, when evaluating the expected information gain of proposed links, we should take into account their visual saliency, as this is an overall good indicator of whether or not the expected information gain (i.e., image registration) is actually obtainable. By doing so, we can propose the addition of links that are not only geometrically informative, but also visually plausible.




(a) Sample images




(b) Camera constraints (c) Local saliency $S_L$ (d) Global saliency $S_G$

Figure 4: Local and global saliency maps on the *SS Curtiss*. (a) Sample images of *SS Curtiss*. (b) A top-down view of the pose-graph depicting where the successful pairwise camera-derived edges occur. (c) A top-down view of the pose-graph with our local saliency metric overlaid. (d) A top-down view of the pose-graph with our global saliency metric, $S_G$, overlaid. The node size has been scaled by its saliency level for visual clarity.

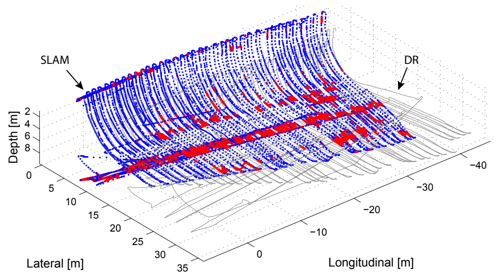
## 3.2 Implementation and Results

### 3.2.1 Local and Global Saliency Maps

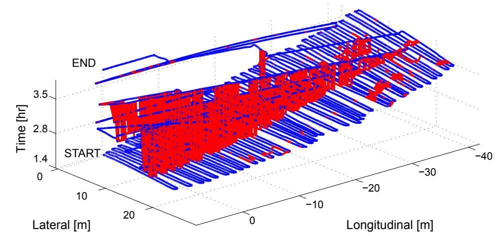
Overlaying the local saliency map on the SLAM result on surveying *SS Curtiss* shows the coincidence of successful camera registrations and areas with a high local saliency score. To have a successful pairwise camera measurement, both images need to be locally salient (i.e., texture rich). Note that successful measurements (red lines in local maps) have been made only when both of the images have a high local saliency score. Unlike the local saliency metric, the global saliency metric reacts to rare features. As a validation, the global saliency map is shown overlaid on the SLAM results. This global saliency score, $S_G$, which also spans from 0 to 1, is overlaid on top of the SLAM trajectory and nodes with $S_G > 0.4$ are enlarged for easier visualization. Note that global saliency can be used to identify visually distinct (i.e., rare) scenes with respect to the rest of the hull. These visually distinctive regions, for example, could serve as useful locations for guiding where the robot should revisit for attempting loop-closure.

### 3.2.2 Saliency-informed SLAM

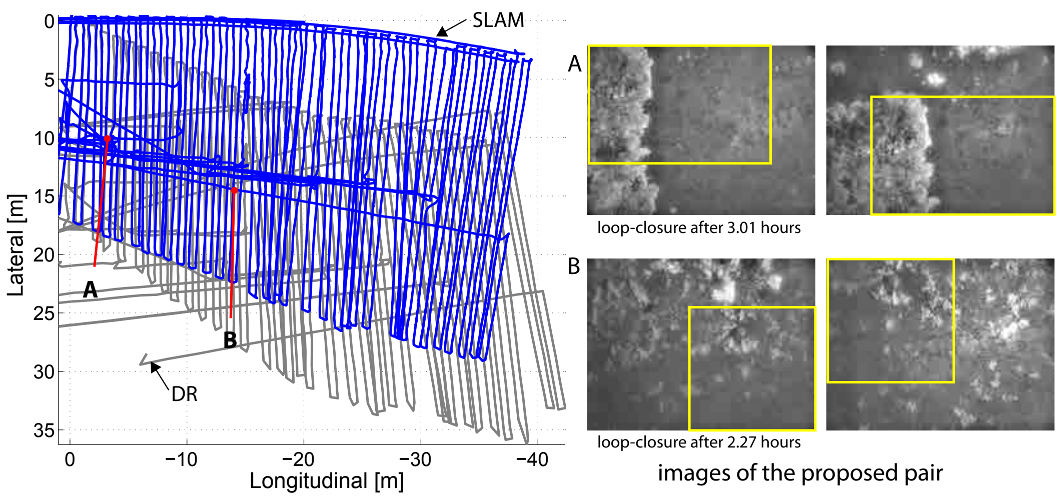
For this experiment, we first ran the visual SLAM algorithm in a "perceptually naive" mode to benchmark its performance in the absence of knowledge on visual saliency. For this test we added




(a) Visual SLAM result with only salient nodes




(b) Time elevation graph depicting loop-closure measurements




(c) Top-down view of SLAM vs. DVL dead-reckoning graph

Figure 5: Saliency-informed SLAM result for the *SS Curtiss*. (a) The blue dotted trajectory represents the iSAM estimate with camera constraints depicted as red edges, while the gray trajectory represents dead-reckoned (DR) trajectory. (b) The $xy$ component of the SLAM trajectory estimate is plotted versus time, where the vertical axis represents mission time. (c) A top-down view of the SLAM estimate versus DR. The positions marked $A$ and $B$ are two examples of where large loop-closure events take place.

keyframes at a fixed spatial sample rate resulting in approximately 70% sequential image overlap, and used geometric information gain only for link hypothesis, while trying excessive number of measurement candidates ($n_{plink}$=30). We refer to this baseline result as "the exhaustive SLAM graph", as all nominal nodes were added and all geometrically informative links where tried. This baseline result contains 17,207 camera nodes, 29,426 5-DOF camera constraints, and it required a cumulative processing time of 10.70 hours (this includes image registration and iSAM inference). Next, we re-ran the visual SLAM algorithm, but this time with the saliency-based keyframe selection and saliency informed information gain link hypothesis enabled, proposing 3 candidates per node ($n_{plink}$=3). Using the saliency-based front-end, we reduced the total number of keyframes from 17,207 (in the exhaustive set), to only 8,728—a 49.3% reduction by culling visually uninformative nodes from the graph. The SLAM result can be computed in less than 1.31 hours, which is 2.6 times faster than the actual mission duration time of 3.4 hours. In the top-down view (Figure 5(c)), the images on the right depict the keyframes and registered loop-closure event, verifying the overall consistency of the metric SLAM solution. The maximum difference between saliency incorporated and exhaustive SLAM is 1.10 meters, whereas the DR trajectory shows significantly larger error due to the navigation drift.

In terms of saliency's affect on SLAM performance, we note that even with far fewer nodes in the graph, we were still able to achieve almost the same total number of camera measurements in the graph. Using only half (50.7%) of the exhaustive graph nodes and a significantly smaller number of link proposals (3.4%), we achieved important cross-track camera measurements. For easier loop-closure visualization, Figure 5(b) shows a time elevation graph of camera registration constraints—here the vertical axis indicates elapsed mission time. While saliency ignored SLAM also shows reduced error over DR, saliency informed SLAM substantially outperforms it by resulting in more (112.4%) verified links and, thus, less error (15%) relative to the baseline exhaustive SLAM result, even with a smaller number of link proposals. This is because the saliency-ignored SLAM result failed to obtain critical, cross-track camera measurements that the saliency-informed SLAM successfully achieved.

# 4 Perception-driven Navigation

As the final step, we introduce robotic autonomous navigation over an area of interest using these saliency metrics. For this type of area coverage mission, a robot needs to explore and map the area, while localizing itself accurately on the map that it builds. This autonomous navigation capability involves three topics, namely simultaneous localization and mapping (SLAM), path planning, and control. In particular, this thesis considers the task of area coverage (i.e., to cover a certain area of interest) under the constraint of bounded navigation error. Specifically, our area coverage objective seeks a balanced control between exploration and revisiting in order to achieve better SLAM performance. Without loop-closure, SLAM will inevitably accumulate navigation drift over time; thus, we need to revisit portions of the map to bound error growth. At the same time, we need to pursue exploration, which is a competing objective that requires mapping the entire area in a reasonable time. Furthermore, and more importantly, SLAM, path planning, and control are interwoven and, thus, inseparable problems.

## 4.1 Overview

perception-driven navigation (PDN) consists of three modules: clustering, planning, and reward evaluation (Figure 6), which will be presented in detail in the following subsections. While the normal SLAM process passively localizes itself and builds a map, PDN (*i*) clusters salient nodes into a set of candidate revisit waypoints, (*ii*) plans a point-to-point path for each candidate revisit waypoint, and (*iii*) computes a reward for revisiting each waypoint candidate. The calculated reward measures the utility of revisiting that waypoint (i.e., loop-closure) versus continuing exploration for area coverage. By comparing the maximum reward for revisiting or exploring, the robot is able to choose the next best control step.

Given the desired target area coverage ($\mathcal{A}_{\text{target}}$) and user defined allowable navigation uncertainty ($\Sigma_{\text{allow}}$), PDN solves for an intelligent solution to the area coverage planning problem while considering SLAM's performance. In PDN's derivation, we address three issues. First, as our approach considers the resulting SLAM performance, the current robot uncertainty should play a role in the path planning. Thus, the current robot uncertainty is a control parameter that triggers the re-planning for better localization and mapping. Second, because the robot should complete the area coverage mission in a timely manner, the map uncertainty in terms of uncovered area needs to be considered. Finally, PDN needs to be computationally scalable for real-time, real-world per-

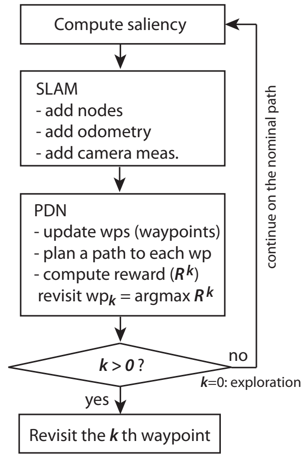



Figure 6: Illustration of PDN's flow diagram. Provided with a SLAM pose-graph and saliency map, PDN updates a set of waypoints, plans a path to these waypoints, and computes rewards for each of the waypoints. The reward $\mathcal{R}^k$ is computed for each waypoint $k$ where the reward from exploration $\mathcal{R}_{exp} = \mathcal{R}^0$ corresponds to the $0^{\text{th}}$ waypoint ($k = 0$). Lastly, either a revisiting or exploration action is executed to yield the maximum reward.

formance. Since the complexity of the algorithm scales linearly with the number of revisit points, maintaining a feasible number of candidate revisit nodes is our focus.

**Waypoint Generation:** Although all nodes in the pose-graph could be considered as waypoints, evaluating the outcome for all possible revisits is impractical. Moreover, due to the uneven spatial distribution of feature-rich areas in the environment, not all nodes are visually plausible for loop-closure. Therefore, PDN computes expected rewards for only a subset of candidate nodes selected for their visual saliency levels. This waypoint generation consists of two parts: salient node clustering and waypoint selection for each cluster. First, we threshold keyframes based upon their local saliency level to generate a set of 3D points with strong local saliency. Then, we cluster locally salient nodes into local neighborhoods, forming clusters, from which we select a representative waypoint node by considering both its visual uniqueness (i.e., global saliency level) and usefulness for loop-closure.

**Path Generation:** For this set of waypoints, the robot computes a shortest path from its current pose to each waypoint to evaluate the expected reward along that path. We found that, although the surface is curved, applying A* [17] globally on the existing nodes results in a fast and globally optimal path on the sample nodes because the nodes are continuously distributed without obstacles.

**Reward for a Path:** Reward for a path is defined in terms of the robot's navigation uncertainty and achieved area coverage. We introduce a weighted sum that determines the balance between pose uncertainty and area coverage. Although we maximize the reward, the formulation can be more intuitively understood when we consider each term as a penalty. The navigation uncertainty term corresponds to the penalty for SLAM, where the action with minimal uncertainty increase is preferred. The area coverage metric is the penalty in area coverage when performing an action. By taking a weighted sum of these two costs, we can evaluate the total penalty, $\mathcal{C}^k$, for a waypoint $k$. The reward is the minus of this penalty, and PDN selects an action with the largest reward, or in other words, with the minimal penalty.

$$\mathcal{C}^k = \alpha \cdot \mathcal{U}^k_{robot} + (1-\alpha) \cdot \mathcal{A}^k_{map} \quad (3)$$

$$\mathcal{R}^k = -\mathcal{C}^k \quad (4)$$

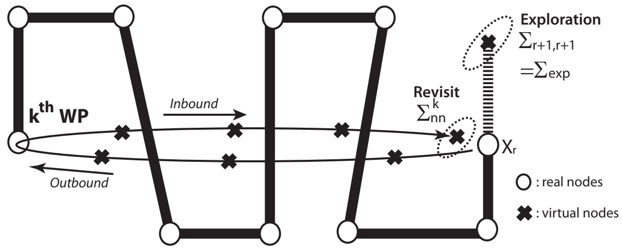



Figure 7: Robot pose uncertainty from revisiting versus exploration. Two terminating node uncertainties from revisiting and exploration are compared. Real nodes on the graph are shown as circles whereas virtual nodes are marked with 'X'.

Then, the revisiting waypoint $k^*$ is determined by maximizing the reward,

$$k^* = \text{argmax } \mathcal{R}^k = \text{argmin } \mathcal{C}^k, \tag{5}$$

where $k \in \{0, 1, 2, \cdots N_{wp}\}$ and $k = 0$ corresponds to the exploration action.

Part of the reward function needs to represent the robot's navigation uncertainty. For each waypoint, we compute the expected covariance propagation along the point-to-point path generated. The resulting robot uncertainty ($\mathcal{U}_{robot}$) from a revisit action is computed as the expected terminating covariance from the round-trip travel to the waypoint. The exactly sparse delayed-state filter (ESDF)-based approach is to construct a small extended information filter (EIF) by adding a set of odometry constraints and a set of expected camera measurements in the form of delta information to the current SLAM information matrix, $\Lambda_0$. To evaluate the terminating covariance from revisiting a certain node, two sources of delta information are added: one from odometry ($\Lambda_{\texttt{odo}}$) and the other representing camera constraints ($\Lambda_{\texttt{cam}}$). Lastly, summing up these three information matrices ($\Lambda_0, \Lambda_{\texttt{odo}}, \Lambda_{\texttt{cam}}$) builds the expected information matrix from PDN ($\Lambda_{\texttt{pdn}}$),

$$\Lambda_{\texttt{pdn}} = \Lambda_0 + \Lambda_{\texttt{odo}} + \Lambda_{\texttt{cam}}. \tag{6}$$

Adding these three information matrices yields the expected information matrix for pursuing a virtual path to the waypoint. For the reward calculation, we are interested in the final pose uncertainty in order to evaluate the usefulness of this action. This is the terminating covariance for the $k^{\text{th}}$ waypoint, $\Sigma^k_{nn}$, and is computed for all $N_{wp}$ waypoints. This round-trip pose covariance is calculated for each waypoint (i.e., for each revisit action), and is compared to the covariance from exploration, where the terminating covariance for exploration is computed by propagating the current covariance one step forward. The reward term for robot uncertainty, $\mathcal{U}^k_{robot}$, is computed as the ratio of the localization uncertainty for the next-best-action to the user-defined allowable navigation uncertainty, $\Sigma_{\text{allow}}$. For the $k^{\text{th}}$ waypoint,

$$\begin{aligned} \mathcal{U}^{k=0}_{robot} &= \begin{cases} 0, & \text{if } \frac{|\Sigma_{\text{exp}}|}{|\Sigma_{\text{allow}}|} < 1 \\ \frac{|\Sigma_{\text{exp}}|^{\frac{1}{6}}}{|\Sigma_{\text{allow}}|^{\frac{1}{6}}}, & \text{otherwise} \end{cases} \\ \mathcal{U}^{k>0}_{robot} &= \frac{|\Sigma^k_{nn}|^{\frac{1}{6}}}{|\Sigma_{\text{allow}}|^{\frac{1}{6}}}, \quad k = 1, \cdots, N_{wp} \end{aligned} \tag{7}$$

where $k = 0$ is the candidate exploration action, $k > 0$ are the $1, \cdots, N_{wp}$ candidate revisit waypoints[1]. Basically, PDN compares the two propagated uncertainties from revisiting and exploring,

[1]We have taken the $6^{\text{th}}$ root of the 6-DOF pose determinant in the numerator and denominator terms so that individually their SI units are $\text{m} \cdot \text{rad}$, which provides a more physically meaningful length scale for taking ratios.

and then chooses the smaller one as in Figure 7. When the expected exploration covariance is below the allowable covariance, the cost in the robot pose uncertainty term, $\mathcal{U}^0_{robot}$, is zero, leading the robot to pursue exploration. On the other hand, when the exploration covariance exceeds the allowable covariance, then the robot pose uncertainty term for exploration, $\mathcal{U}^0_{robot}$, is compared against all candidate revisit actions, $\mathcal{U}^{k>0}_{robot}$, which will be smaller when revisiting is likely to obtain enough loop-closures to overcome the increased navigation uncertainty from detouring. Unlike previous studies in active exploration of [18], [19] and [20], where the authors did not consider the actual likelihood of obtaining perceptual loop-closures, our approach introduces a realistic expectation in the reward calculation for the likelihood of camera loop-closures based upon visual saliency.

Secondly, we add a bias term for area coverage. The purpose of the mission is to cover a target area in a timely manner while considering SLAM's navigation performance. In other words, without an area coverage term, there will be a trivial solution to this problem—to repeatedly revisit to keep the uncertainty very small. To prevent this, the area coverage term for the $k^{\text{th}}$ waypoint is defined as the ratio of area-to-cover with respect to the target-coverage-area, where the target area is provided by the user,

$$\mathcal{A}^k_{map} = \frac{\mathcal{A}_{\text{to\_cover}}}{\mathcal{A}_{\text{target}}} = \frac{\mathcal{A}_{\text{target}} - \mathcal{A}_{\text{covered}} + \mathcal{A}^k_{\text{redundant}}}{\mathcal{A}_{\text{target}}}, \tag{8}$$

$$\mathcal{A}^k_{\text{redundant}} = \text{redundant coverage by revisiting} \tag{9}$$

$$\begin{cases} = 0, & \text{for exploration} \\ = l(\mathcal{P}^k) \cdot D > 0 & \text{for revisiting} \end{cases}. \tag{10}$$

Here, $l(\mathcal{P}^k)$ is the expected path length added by revisiting the $k^{\text{th}}$ waypoint, $D$ is the sensor field of view width, $\mathcal{A}_{\text{target}}$ is the pre-defined target coverage area as set in the mission planning phase, and $\mathcal{A}_{\text{redundant}}$ is the expected redundant area coverage produced by a revisiting action. This additional area is the result of multiplication of the revisit path with the sensor field of view width and has nonzero value, $\mathcal{A}^k_{\text{redundant}} = l(\mathcal{P}^k) \cdot D$.

## 4.2 Implementation and Results

For validation, we present an evaluation of PDN as applied to a hybrid simulation trajectory generated from real ship hull inspection data. Since there is no ground-truth available for our underwater missions, we use the baseline exhaustive SLAM result from previous chapter to generate a hybrid simulation with preplanned nominal trajectory. In all cases of evaluation, PDN results are compared with other traditional preplanned survey schemes, in terms of robot uncertainty (as a measure of SLAM performance) and area coverage rate (as a measure of coverage performance). One pattern is an open-loop control that follows the given nominal trajectory without any revisiting. The other survey pattern is to preplan some deterministic revisit actions during the preplanning phase, which are aimed at achieving any possible loop-closures. This deterministic revisit strategy is typical of underwater vehicle operations, and is passively preplanned or executed by a human pilot. In this work, we call this preplanned regular revisit "exhaustive revisit". In the exhaustive revisit scenario, the vehicle is controlled to come back to a waypoint in every other track-line for possible loop-closure, regardless of the actual feature distribution in the environment.

### 4.2.1 PDN with Synthetic Saliency Map

The first set of tests are with a synthetic saliency map imposed over the area with full weight on the pose uncertainty ($\alpha = 1.0$). For the exhaustive revisit, the robot is commanded to revisit a

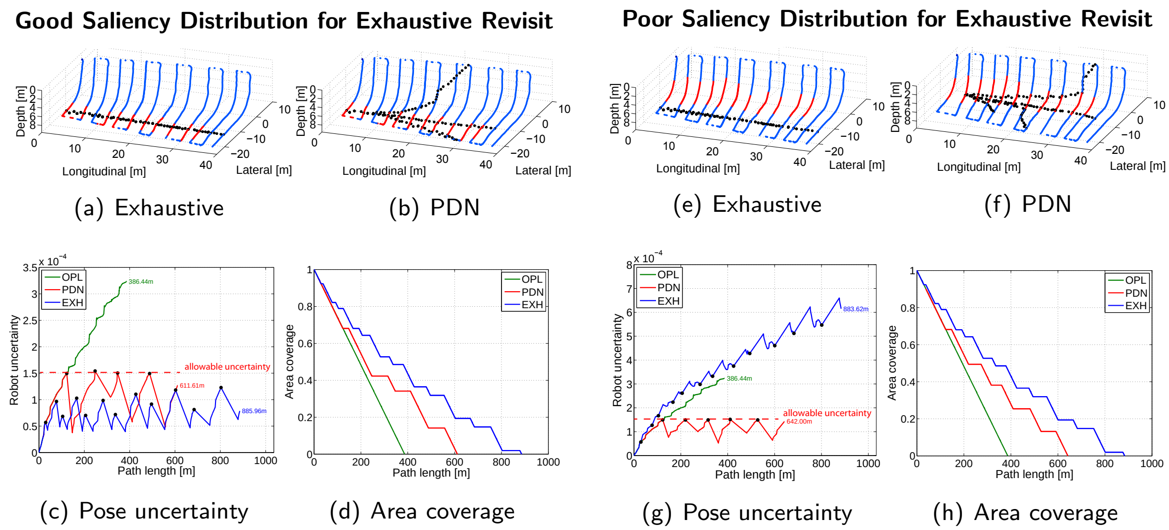


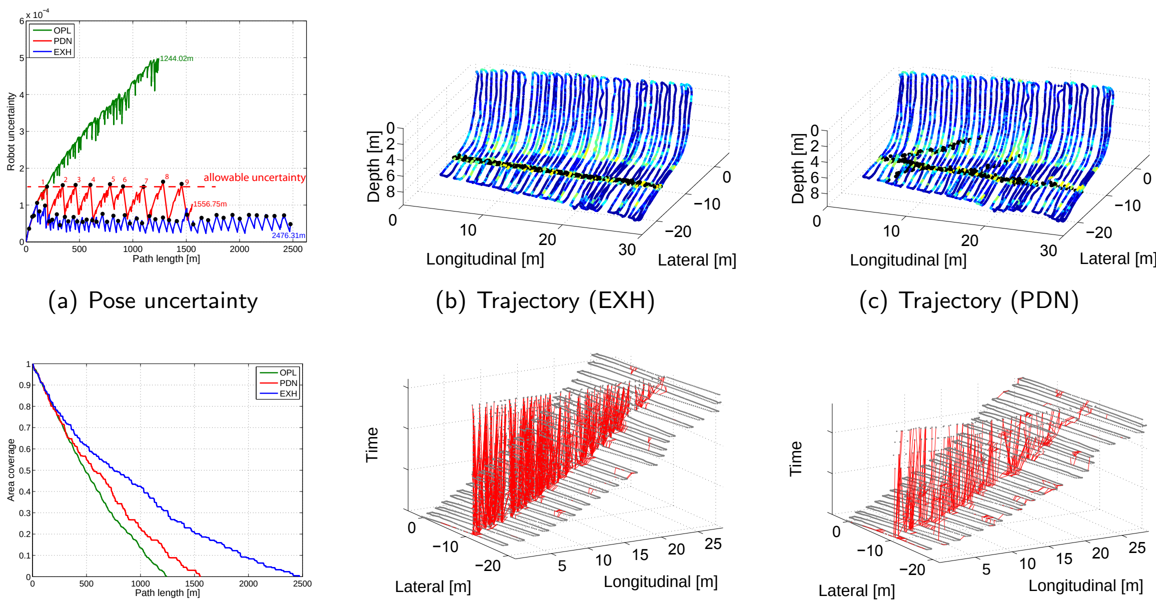

Figure 8: PDN results for biased saliency maps. Similar to the case of uniform distribution, pose uncertainty and area coverage graph are compared for two biased saliency regions among open-loop (green), exhaustive revisit (blue) and PDN (red), where the black dots indicate points when revisit occurred. Open-loop and PDN perform the same as in the evenly distributed salient region case. However, exhaustive revisit strongly depends on the spatial distribution of feature-rich areas.

point on the first track-line in every other track-line it travels. In this test, the exhaustive revisit happens on a line on the bottom of the hull. Because this repeated visit is preplanned without knowing the actual visual feature distribution in the environment, we assign the same exhaustive revisit control for all cases.

In this test, we show cases where the feature-rich distribution is biased to show how the preplanned exhaustive revisit succeeds and fails depending on the saliency distribution. When all of the exhaustive revisit paths land on the salient region (Figure 8(a)), the likelihood of obtaining loop-closure during the revisit is higher, and the exhaustive revisit achieves tightly bounded uncertainty for the robot pose (Figure 8(c)). On the other hand, when none of the revisit paths are on salient regions, as in the case of Figure 8(e), the algorithm basically performs worse than open-loop. Without meaningful loop-closures on the revisit, the control just increases the overall path length and slows coverage rate, as can be seen in Figure 8(g) and Figure 8(h). Unfortunately, the salient region distribution cannot be known a priori when the preplanning takes place. Note that for both cases, the total path length and the area coverage rate stays the same for the exhaustive revisit since it is preplanned. On the other hand, from PDN's point of view, there is no difference between all three cases, resulting in a consistent performance on uncertainty bounding and area coverage.

#### 4.2.2 PDN with Real Image Data

We now evaluate PDN for saliency-informed SLAM using real underwater images for a camera mission profile. The saliency map is generated and updated online from the real underwater images that are available from the baseline result. Using the pair of real-world images, the saliency score and camera registration engine are applied as in the normal saliency-informed SLAM process. A weight factor of $\alpha = 0.75$ is selected in PDN to impose a biased weight on the pose-uncertainty rather than area coverage. Similar to the synthetic saliency case, the uncertainty and area coverage graph for PDN is compared with open-loop (OPL) and exhaustive (EXH) revisit. Based upon the




(a) Pose uncertainty (b) Trajectory (EXH) (c) Trajectory (PDN)

(d) Area coverage (e) Time elevation (EXH) (f) Time elevation (PDN)

Figure 9: PDN for saliency-informed SLAM on a camera mission. Similar to the synthetic saliency case, (a) and (d) show the pose uncertainty and area coverage with respect to the path length. In trajectories (b) and (c), nodes are color-coded by their saliency level from the real images. The exhaustive revisit was preplanned over the salient band, however, PDN is able to find this same optimal path to follow as in (c). In the time elevation graphs ((e) and (f)), PDN shows a comparable number of successful loop-closures to the exhaustive revisit.

knowledge of the saliency distribution in the baseline result, we preplanned the exhaustive revisit path to be laid over the salient band to provide the best possible case to be compared with PDN. Because the exhaustive revisit is intentionally planned over the salient region, the resulting graph of exhaustive revisit shows the maximum SLAM performance—maintaining low uncertainty, but producing an exceeding number of revisits and longer path length.

Uncertainty change and the area coverage rate are presented in Figure 9(a) and (d) for both types of missions. As shown in Figure 9(c), PDN followed trajectories to obtain expected visual loop-closures to reduce the uncertainty whenever it exceeded the allowable covariance bound. Specifically, exhaustive revisits in the camera mission result in 47 revisit actions with twice the total path length of the nominal trajectory. Assuming a constant speed for the vehicle, this exhaustive revisit strategy would double the overall mission time. In this camera mission, note that the number of revisits by PDN (9) is substantially smaller than the exhaustive revisit case. PDN presents a result with less number of revisits while maintaining full control on the uncertainty level, and still achieving the important loop-closures. The loop-closing camera measurements are clearly illustrated in the time elevation graph of Figure 9(e) and (f). The red lines in the graph depict the camera measurements made by the loop-closures. As can be seen in the time elevation graphs, PDN obtained a similar number of loop-closures as compared to the exhaustive revisit case.

## 5 Conclusion

This thesis proposed an integrated approach toward robotic navigation and exploration for autonomous robot missions. SLAM and path planning have traditionally been considered as two separate problems, each assuming some prior information from the other. This thesis started from the viewpoint of SLAM and presented a metric for visual saliency that could be used with geometric information to improve loop-closure performance. This saliency-informed SLAM result was then combined with planning to lead the robot autonomously along trajectories that yielded better SLAM results and survey area coverage. Experiments using real underwater mission data and simulations were provided from several different vessels to evaluate the reported algorithms.

Contributions of this thesis includes $(i)$ Real-time visual SLAM is developed and successfully applied to a real-world AUV ship hull inspection application. Specifically, a monocular camera image registration engine is developed and integrated into a real-time SLAM implementation. $(ii)$ Developed two novel measures of visual saliency that improve underwater visual SLAM. Local saliency measures texture richness of a scene and aids keyframe selection. Global saliency detects rarity of a scene. $(iii)$ A novel solution for concurrent SLAM and planning for the robotic area coverage problem, called PDN is developed. PDN is an integrated navigation algorithm that automatically achieves efficient target area coverage while maintaining good visual SLAM navigation performance. PDN provides an intelligent and fully autonomous online control scheme for efficient bounded-error area coverage that strikes a balance between revisit and exploration actions in a decision theoretic way.

## References


[1] R. Sim and N. Roy, "Global a-optimal robot exploration in SLAM," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation*, Barcelona, Spain, 2005, pp. 661–666.

[2] R. M. Eustice, H. Singh, J. J. Leonard, and M. R. Walter, "Visually mapping the RMS Titanic: Conservative covariance estimates for SLAM information filters," *Intl. Journal of Robotics Research*, vol. 25, no. 12, pp. 1223–1242, 2006.

[3] J. Vaganay, M. Elkins, D. Esposito, W. O'Halloran, F. Hover, and M. Kokko, "Ship hull inspection with the HAUV: U.S. Navy and NATO demonstrations results," in *Proc. of the IEEE/MTS OCEANS Conf. and Exhibition*, Boston, MA, 2006, pp. 1–6.

[4] K. Zuiderveld, "Contrast limited adaptive histogram equalization," in *Graphics Gems IV*, P. Heckbert, Ed. Boston: Academic Press, 1994, vol. IV, pp. 474–485.

[5] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Intl. Journal of Comp. Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[6] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comp. Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[7] R. M. Eustice, O. Pizarro, and H. Singh, "Visually augmented navigation for autonomous underwater vehicles," *IEEE Journal of Oceanic Engineering*, vol. 33, no. 2, pp. 103–122, Apr. 2008.

[8] N. Carlevaris-Bianco and R. M. Eustice, "Multi-view registration for feature-poor underwater imagery," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation*, Shanghai, China, May. 2011, pp. 423–430.

[9] A. Kim and R. M. Eustice, "Pose-graph visual SLAM with geometric model selection for autonomous underwater ship hull inspection," in *Proc. of the IEEE/RSJ Intl. Conf. on Intell. Robots and Syst.*, St. Louis, MO, Oct. 2009, pp. 1559–1565.

[10] P. Torr, "Model selection for two view geometry: A review," in *Shape, Contour and Grouping in Comp. Vision.* Springer, 1999, pp. 277–301.

[11] B. Horn, "Relative orientation revisited," *Journal of the Optical Society of America A*, vol. 8, no. 10, pp. 1630–1638, Oct 1991.

[12] R. Haralick, "Propagating covariance in computer vision," in *Proc. of the Intl. Conf. Pattern Recognition*, vol. 1, Jerusalem, Israel, Oct. 1994, pp. 493–498.

[13] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of Documentation*, vol. 28, pp. 11–21, 1972.

[14] G. Salton and C. S. Yang, "On the specification of term values in automatic indexing," *Journal of Documentation*, vol. 29, pp. 351–372, 1973.

[15] S. Robertson, "Understanding inverse document frequency: On theoretical arguments for idf," *Journal of Documentation*, vol. 60, pp. 503–520, 2004.

[16] V. Ila, J. Porta, and J. Andrade-Cetto, "Information-based compact pose SLAM," *IEEE Transaction on Robotics*, vol. 26, no. 1, pp. 78–93, Feb. 2010.

[17] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach.* Pearson Education, 2003.

[18] F. Bourgault, A. A. Makarenko, S. B. Williams, B. Grocholsky, and H. F. Durrant-Whyte, "Information based adaptive robotic exploration," in *Proc. of the IEEE/RSJ Intl. Conf. on Intell. Robots and Syst.*, 2002, pp. 540–545.

[19] A. A. Makarenko, S. B. Williams, F. Bourgault, and H. F. Durrant-Whyte, "An experiment in integrated exploration," in *Proc. of the IEEE/RSJ Intl. Conf. on Intell. Robots and Syst.*, 2002, pp. 534–539.

[20] C. Stachniss, G. Grisetti, and W. Burgard, "Information gain-based exploration using rao-blackwellized particle filters," in *Proc. of the Robotics: Science & Syst. Conf.*, Cambridge, MA, USA, 2005.