2013 IEEE International Conference on Robotics and Automation (ICRA)
Karlsruhe, Germany, May 6-10, 2013

# Perception-Driven Navigation: Active Visual SLAM for Robotic Area Coverage

Ayoung Kim and Ryan M. Eustice

***Abstract*—This paper reports on an integrated navigation algorithm for the visual simultaneous localization and mapping (SLAM) robotic area coverage problem. In the robotic area coverage problem, the goal is to explore and map a given target area in a reasonable amount of time. This goal necessitates the use of minimally redundant overlap trajectories for coverage efficiency; however, visual SLAM's navigation estimate will inevitably drift over time in the absence of loop-closures. Therefore, efficient area coverage and good SLAM navigation performance represent competing objectives. To solve this decision-making problem, we introduce perception-driven navigation (PDN), an integrated navigation algorithm that automatically balances between exploration and revisitation using a reward framework. This framework accounts for vehicle localization uncertainty, area coverage performance, and the identification of good candidate regions in the environment for loop-closure. Results are shown for a hybrid simulation using synthetic and real imagery from an autonomous underwater ship hull inspection application.**

## I. Introduction

To enable robotic autonomous navigation over an area of interest, a robot needs to explore and map the target area while concurrently localizing itself accurately with respect to the map that it builds. This autonomous navigation capability involves three topics, namely simultaneous localization and mapping (SLAM), path planning, and control. There have been some prior efforts on merging SLAM and path planning into an integrated framework—the major difficulty being that path planning typically assumes that a map is known a priori, while SLAM typically assumes that a path is given.

Recently, some approaches have started to evaluate the resulting uncertainty in optimizing the path, such as Belief Roadmaps (BRMs) [1], planning paths on SLAM pose-graphs [2], and calculating possible information gain on a planned path using Rapidly-exploring Random Trees (RRTs) [3]. The BRM approach is closest to our own in that it considers the state's uncertainty when it is computing the objective function, but is different from us as the nodes are sampled from a given map and are not built online during SLAM. Valencia et al. [2] perform SLAM first, then use the resulting pose-graph to plan a path to a goal position considering information gain through the graph. However, and most importantly, in these previous studies online exploration was not considered—optimality was evaluated only by the uncertainty of the robot and the map, not by time or area coverage. This is because exploration was excluded in the evaluation, and the main focus instead was on point-to-point path planning through a given map.

On the other hand, the SLAM community has also made some efforts to add exploration functionality to SLAM, termed "active SLAM". Stemming from the seminal work of *active perception* by Bajcsy [4], which pointed out that control can improve the quality of sensor data, active SLAM is an area in SLAM that tries to find the optimal action that can improve map building and localization performance. Examples of this line of research are found in [5]–[9]. Although these approaches established a basis for combining control with SLAM, they only applied the optimal control input and did not globally solve for path planning.

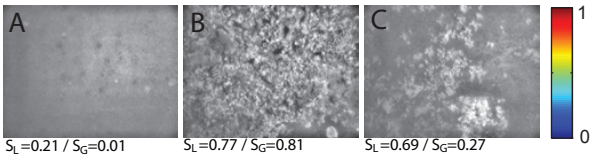
In earlier work on autonomous exploration, Whaite and Ferrie [10] introduced a way to explore considering the uncertainty of the explored model. Although not directly addressing the SLAM problem, their work considered the reduction in model uncertainty through motion as a way to explore. Gonzalez-Banos and Latombe [11] proposed exploration strategies analogous to the next-best-view (NBV) problem [12] in computer vision.

There are some SLAM studies that consider SLAM performance in the exploration phase. These integrated studies try to search for an optimal solution to maximize area coverage and SLAM performance at the same time. Makarenko and Williams [13] presented an integrated exploration scheme based on mutual information. Similarly, Bourgault et al. [14] considered map coverage and localization accuracy in order to generate an adaptive control action. Stachniss et al. [15] pointed out the gist of this unsolved problem between SLAM and exploration. Their SLAM implementation compares the utilities associated with the actions of exploration and revisit in order to determine whether to continue exploring or to revisit a previous location. This formulation is similar to our own in this regard, but additionally we consider the plausibility of measurement availability in evaluating this utility. Recently, Kollar and Roy [16] presented an exploration strategy using Reinforcement learning. Because they assume a priori access to a ground-truth map, their algorithm is trained to learn the trajectory that maximizes the accuracy of the SLAM-derived map. Another learning approach is presented in [17], where a robot starts with initial policy parameters and updates them through active policy learning with consideration of robot motion dynamics.

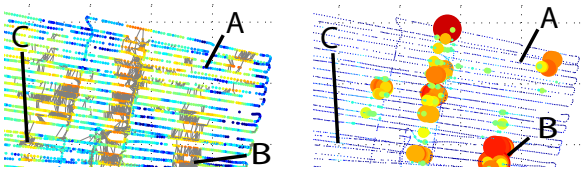
An overarching assumption that our work makes is to start from very little prior information on the area of interest.

This work was supported by the Office of Naval Research under awards N00014-07-1-0791 and N00014-12-1-0092. We would like to thank J. Vaganay and K. Shurn from the Bluefin Robotics Corporation for their excellent support during the course of the experiments.

A. Kim and R. Eustice are with the Department of Naval Architecture & Marine Engineering, University of Michigan, Ann Arbor, MI 48109, USA [ayoungk, eustice]@umich.edu




(a) Sample images




(b) Local saliency ($S_L$) (c) Global saliency ($S_G$)

Fig. 1: Depiction of local/global saliency maps for a underwater hull inspection survey of the *SS Curtiss*. Shown is a top-down view of the $xy$ trajectory. Nodes are color-coded by their saliency level, where 0 indicates non-salient and 1 is salient. A, B and C depict representative sample imagery from the hull. In (c) the node size is scaled with the global saliency level for easier visualization, and in (b) successful pairwise image links are shown as gray edges.

Starting from no prior knowledge on the environment, our algorithm, called perception-driven navigation (PDN), performs SLAM online to build a map for localizing the robot in the target area, while simultaneously planning paths to bound navigation error, subject to efficient area-coverage in terms of the total path length.

## II. Visual Saliency

In visual SLAM, not all images are equal in terms of their utility for keyframe registration. This is especially true in the underwater environment, where the spatial distribution of good visual features is not uniformly abundant. In this paper, we adopt the visual saliency metrics defined earlier by Kim and Eustice [18], [19]: *local saliency* ($S_L$) and *global saliency* ($S_G$). Each measure provides a normalized score from 0 to 1, where 1 indicates highly salient imagery and 0 indicates non-salient imagery. These metrics are computed from a bag-of-words (BoW) representation of the imagery where the vocabulary is built online.

Fig. 1 depicts local and global saliency maps as applied to a portion of a SLAM survey taken from an underwater hull inspection mission. As shown, local saliency measures the intra-image texture richness of the scene, which is highly related to the ability to make successful pairwise keyframe registrations, while global saliency measures the inter-image rarity of a keyframe. As depicted, 'A' represents a non-salient image, 'B' is a locally and globally salient image, and 'C' is a locally but not globally salient image. In our application, the robot measures these two saliency levels for every keyframe it inserts into the SLAM pose-graph; as it proceeds on a mission, it uses them in PDN's reward calculation, discussed next.

## III. Perception-Driven Navigation

Typically, SLAM is formulated as a passive process that localizes and builds a map using whatever data sequence and

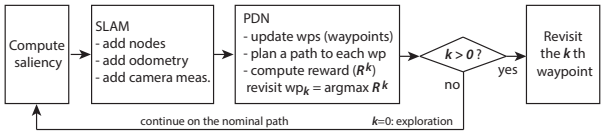



Fig. 2: PDN's block-diagram. Given access to the current SLAM pose-graph and saliency map, PDN selects a set of candidate waypoints, plans a path to these waypoints, and compute rewards for each of these waypoints. The reward, $\mathcal{R}^k$, is computed for each waypoint $k$, where $k = 0$ corresponds to the reward from exploration (i.e., $\mathcal{R}_{exp} = \mathcal{R}^0$). Lastly, either a revisiting or exploration action is executed to yield the maximum reward.

exploration trajectory it is provided. PDN is designed to sit one layer above SLAM in that it represents an integrated framework to evaluate rewards and execute actions to guide the robot for better SLAM navigation and area coverage performance. In this work, we have adopted the incremental smoothing and mapping (iSAM) algorithm by Kaess et al. [20], [21] as the SLAM back-end. In our application, constraints from odometry, monocular camera, attitude, and pressure depth are fused within iSAM [22].

PDN consists of three modules (Fig. 2) that 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. If the revisit action is selected to be the next best control, the robot will revisit the selected waypoint then return to the release point to resume the nominal mission. By comparing the maximum reward for revisiting or exploring, the robot is able to choose the next best control step.

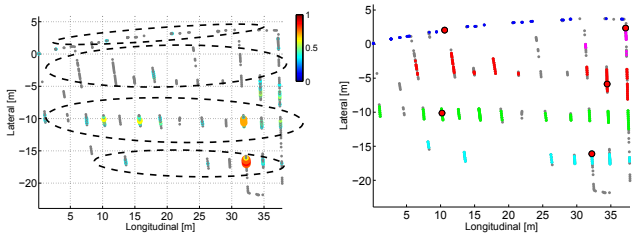
For the derivation of PDN, we start with the following assumptions and problem definitions:

1) The boundaries of the target coverage area are given.
2) An initial reference trajectory is preplanned using this information and covers the target area in efficient time.
3) The maximum allowable navigation uncertainty is defined by the user, and will be used to trigger PDN's revisit decision.
4) No other prior information on the environment is provided. Planning and evaluation will be performed online while the SLAM pose-graph is built.

Given the above assumptions, PDN solves for an intelligent solution to the area coverage planning problem while considering SLAM's navigation performance.

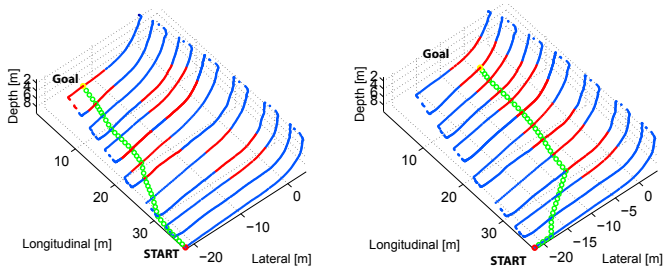
### A. Waypoint Generation

Although all nodes in the pose-graph could be considered as possible 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




(a) Cluster formation (b) Resulting waypoints

Fig. 3: A depiction of waypoint generation. (a) Globally salient nodes are shown overlaid on the thresholded locally salient nodes (gray). Each cluster is roughly marked with a dotted ellipse for visualization. Globally salient nodes are color-coded by their global saliency level, and enlarged in size for visualization. (b) The resulting waypoints for each cluster are shown (red dots). Each cluster is denoted by a different color (red, blue, green, magenta and cyan).




(a) Distributed saliency regions (b) Concentrated saliency regions

Fig. 4: Point-to-point path planning example using a synthetic saliency map for illustration. Blue dots represent non-salient nodes in the graph and salient regions are marked red. The planned paths are depicted with a sequence of green circles linking the start and goal positions. (a) and (b) are sample point-to-point paths for two different local saliency distributions. Note that saliency weighted A* results in paths biased toward the salient regions in the environment.

nodes are visually plausible for loop-closure. Therefore, PDN computes expected rewards for only a subset of candidate nodes selected for their visual saliency level, resulting in only a subset of locally and globally salient nodes being identified as candidate waypoints.

First, we threshold acquired keyframes based upon their local saliency level to generate a set of candidate nodes with strong local saliency scores. In PDN, a local saliency threshold of $S_L^{\mathsf{wp}} = 0.5$ is used. Next, an online clustering algorithm, Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [23], [24], groups locally salient nodes into local neighborhoods, forming clusters. Finally, within each cluster, we select a representative waypoint node by considering both its visual uniqueness (i.e., high global saliency level) and usefulness for loop-closure (i.e., lowest pose uncertainty). This process allows us to compute the $N_{wp}$ best candidate waypoints for revisitation.

Fig. 3 shows a typical result. Plot (a) depicts globally salient nodes overlaid on the thresholded locally salient nodes—a dotted ellipse denotes the extent of each cluster. Within each cluster, the node with the lowest pose uncertainty among the globally salient nodes is selected as the representative waypoint. The clustered waypoints, plot (b), are then sorted by time and assigned with a waypoint number from 1 to $N_{wp}$, where waypoint number 0 is reserved for the exploration action.

### B. Path Generation

Using this set of waypoints, the robot evaluates the reward that can be obtained by revisiting versus exploring. In this procedure, prior to the reward evaluation the robot computes a shortest path from its current pose to each waypoint in order to evaluate the expected reward along that path. In our application, finding the shortest path can be viewed as a traditional point-to-point path planning problem. We use the global A* algorithm [25], but with the heuristic function weighted by local saliency:

$$d(\mathbf{x}_i, \mathbf{x}_k) = w(S_L^k) \cdot \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2 + (z_i - z_k)^2}.$$

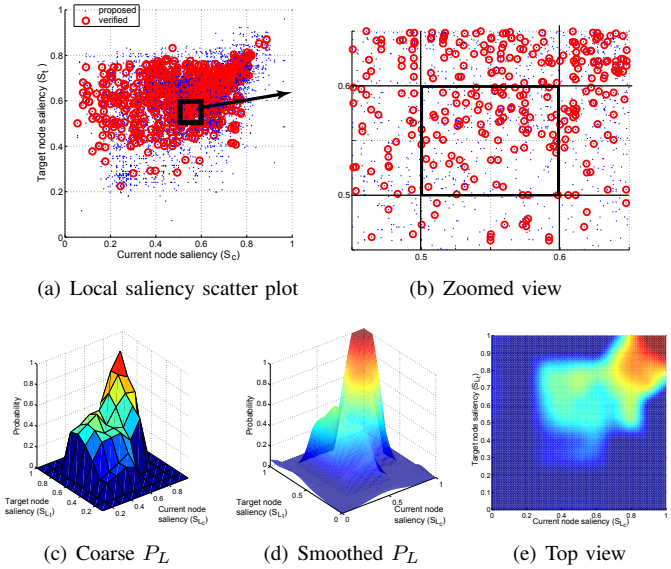
The weight term, $w(S_L^k)$, is modeled in such a way as to double the Euclidean distance to nodes with zero saliency and to preserve the original distance to nodes with full saliency, $w(S_L^k) = 2 - S_L^k$.

Repeated bisection of sample nodes in the pose-graph yields a sequence of nodes, called milestones, denoted $\{\mathbf{x}_m\}$. During the generation of these milestones, we interpolate between milestones if necessary to complete a path, $\mathcal{P} = \{\mathbf{x}_p\}$, that a vehicle can follow during a revisit action. Fig. 4 shows sample point-to-point paths for two different local saliency maps. Using the current robot node as the starting point, the computed paths reach the waypoint as the goal node. Note that the resulting paths are biased toward passing through salient regions in the environment due to the saliency weighted heuristic function. Because local minima may occur with the saliency weighting heuristic, we detect and avoid these using a perturbation action [26] that tests using a pure Euclidean distance heuristic.

### C. Reward for a Path

Reward for a path is defined in terms of the robot's navigation uncertainty and achieved area coverage. For the robot uncertainty, we use the terminating pose covariance, and for the area coverage, we use the area coverage ratio as the performance measure.

*1) Saliency-based Measurement Probability:* For each point-to-point candidate path, we solve for the estimated final robot uncertainty using an exactly sparse delayed-state filter (ESDF) [27]. Using expected odometry and camera measurements along the path, the robot can estimate the final terminating covariance along that trajectory. However, estimating the amount of information gained from future camera measurements is not exact, and we need to develop a way of approximating the camera measurement likelihood. Camera measurements are binary, either success (1) or failure (0), with the camera link event, $L$, being a Bernoulli random variable. When adding a set of expected camera measurements, we use local saliency to empirically model the likelihood




(a) Local saliency scatter plot (b) Zoomed view

(c) Coarse $P_L$ (d) Smoothed $P_L$ (e) Top view

Fig. 5: Construction of empirical probability of link success, $P_L$, using prior data. The model is generated as a function of pairwise saliency levels ($S_{L_c}$ and $S_{L_t}$). To model this, we use a scatter plot of link trials, (a) and (b), using data from previous missions from three different vessels (the *R/V Oceanus*, the *USCGC Venturous*, and the *USCGC Seneca*). A coarse distribution for $P_L(l = 1; S_{L_c}, S_{L_t})$ is then built by calculating the ratio of verified links to the number proposed links in each bin of the scatter plot (c). Then, surface-fitting to this coarse result yields the final smooth distribution (d) and (e).

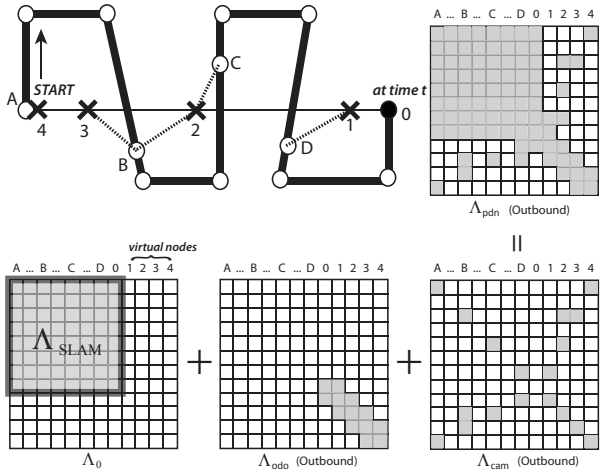



Fig. 6: Robot pose uncertainty propagation in PDN. Only the outbound revisit action is illustrated for simplicity (PDN computes the information for the round-trip). The robot starts from node A moving along the thick line, and reaches the current node 0 at time $t$. This illustration shows construction of the PDN information matrix when the robot executes a revisit action from the current node 0 to a revisit point A. The revisit action is marked with a thin line linking 0 and A with virtual nodes 1,2,3, and 4 along the revisit path to A. Nodes A,B,C, and D are existing nodes in the pose-graph, and also are the candidate nodes that these virtual nodes make camera measurements with. The expected camera measurements are marked with a dotted line between 1–4 and A–D.

of successful registration in order to compute the expected information gain along the path. The observation is that we can model this probability using statistics from prior SLAM and saliency results.

For the Bernoulli random variable, $L$, we seek to model its probability of success, $P_L$. Because each link is associated with two local saliency levels—the current node saliency, $S_{L_c}$, and the target node saliency, $S_{L_t}$—we can build a probability of link success as a function of these two saliency levels:

$$P_L = P_L(l = 1; S_{L_c}, S_{L_t}) \sim \text{Bernoulli}. \tag{1}$$

To empirically measure this probability, we generate a scatter plot from prior data and divide it into a set of bins with bin size of $\delta = 0.1$ (Fig. 5(a)–(b)). The empirical probability of link success is then calculated by counting the number of proposed links versus the number of verified links in each bin, which builds up a coarse model (Fig. 5(c)) as a function of the two associated saliency values (the current node saliency and the target node saliency). Then, this coarse model is smoothed using surface fitting (Fig. 5(d)–(e)).

*2) Robot Uncertainty ($\mathcal{U}_{robot}$):* For PDN, we use Fisher information for evaluating the resulting covariance matrix for integrated SLAM and path planning. Because the camera measurement is not certain, we compute the expected information gain from a path, and evaluate the expected terminating covariance matrix. We use the determinant of the covariance matrix as a measure of navigation uncertainty.

The process of constructing the information matrix using an ESDF is illustrated in Fig. 6. Note that only the outbound portion of the revisit action is shown for visual clarity (PDN computes the information for the round-trip). The ESDF-based approach is to construct a small extended information filter (EIF) problem by adding a set of odometry constraints and a set of expected camera measurements in the form of delta information to the current information matrix, $\Lambda_0$. In the toy example of Fig. 6, a robot starts from node A and moves along the thick line, reaching the current node, denoted 0, at time $t$. To evaluate the terminating covariance from revisiting the A node, two sources of delta information are added: one from odometry, $\Lambda_{\text{odo}}$, and the other representing camera constraints, $\Lambda_{\text{cam}}$. Summing these three information matrices builds PDN's expected information matrix:

$$\Lambda_{\text{pdn}} = \Lambda_0 + \Lambda_{\text{odo}} + \Lambda_{\text{cam}}. \tag{2}$$

The expected delta information from odometry measurements, $\Lambda_{\text{odo}}$, is built from a sequence of virtual nodes. Starting from the current node, $\mathbf{x}_0$, the odometry noise covariance, Q, and path, $\mathcal{P} = \{\mathbf{x}_i, \cdots, \mathbf{x}_p\}$, are summed for all expected odometry measurements for the round-trip travel to the waypoint along the revisit path.

$$\begin{aligned} \Lambda_{\text{odo}} &= \sum_{i=0}^{p-1} \mathrm{H}_{\text{odo}_{i,i+1}}^{\top} \cdot \mathrm{Q}_{i,i+1}^{-1} \cdot \mathrm{H}_{\text{odo}_{i,i+1}} \quad \text{Outbound} \\ &+ \sum_{i=p}^{1} \mathrm{H}_{\text{odo}_{i,i-1}}^{\top} \cdot \mathrm{Q}_{i,i-1}^{-1} \cdot \mathrm{H}_{\text{odo}_{i,i-1}} \quad \text{Inbound} \end{aligned} \tag{3}$$

The noise for the odometry constraint, $\mathrm{Q}_{i,i+1}$, is scaled with the travel distance between nodes $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$. The odometry measurement model is the relative-pose between two sequential nodes ($\mathbf{x}_i$ and $\mathbf{x}_{i+1}$) and can be represented using the tail-to-tail operation by Smith et al. [28]. The resulting Jacobian, $\mathrm{H}_{\mathtt{odo}_{i,i+1}}$, is sparse with nonzero block matrices on the $i^{\text{th}}$ and $(i+1)^{\text{th}}$ element. Hence, summing all odometry information results in a block-tridiagonal matrix (Fig. 6).

For the camera measurements, we similarly add all expected camera measurements along the revisit path. Because PDN proposes the same number of link hypotheses, $n_{plink}$, as in the normal SLAM process, there are multiple expected camera measurements per each virtual node along the path. When a virtual node is $\mathbf{x}_i$ and the candidate paired for camera measurement is node $\mathbf{x}_c$, the first-order expected information gain from camera measurements is calculated as

$$\begin{aligned}\Lambda_{\mathtt{cam}} &= \sum_{i=0}^{p-1}\sum_{c\in\mathcal{L}_i} P_L \cdot \mathrm{H}_{\mathtt{cam}_{c,i}}^{\top}\mathrm{R}^{-1}\mathrm{H}_{\mathtt{cam}_{c,i}} \quad \text{Outbound} \\ &+ \sum_{i=p}^{1}\sum_{c\in\mathcal{L}_i} P_L \cdot \mathrm{H}_{\mathtt{cam}_{c,i}}^{\top}\mathrm{R}^{-1}\mathrm{H}_{\mathtt{cam}_{c,i}} \quad \text{Inbound}, \quad (4)\end{aligned}$$

where $\mathrm{H}_{\mathtt{cam}_{c,i}}$ is the camera measurement Jacobian [29], $\mathrm{R}$ is the fixed camera measurement noise covariance, $\mathcal{L}_i$ is the index set of camera measurement candidates associated with virtual node $\mathbf{x}_i$, and $P_L = P_L(l=1; S_{L_c}, S_{L_t})$ is the empirical probability of the link to be successful. Unlike odometry measurements, not all expected camera measurements result in registration success, which depends greatly upon the visual feature distribution in the environment.

Finally, adding these three information matrices (2) yields the expected information matrix for pursuing a virtual path to the waypoint. For the reward calculation, we efficiently obtain the terminating covariance, $\Sigma_{nn}^k$, from the expected information matrix by computing the $n^{\text{th}}$ block-column of the covariance matrix, $\Sigma_{*n}^k$, as per [30],

$$\Lambda_{\mathtt{pdn}}\Sigma_{*n}^k = \mathrm{I}_{*n}, \quad (5)$$

where $\mathrm{I}_{*n}$ is the $n^{\text{th}}$ block-column of the $n \times n$ block identity matrix. This formulation is computationally efficient and avoids inverting the entire information matrix to recover the round-trip pose covariance for the $k$ candidate waypoints.

Next, the terminating covariance for exploration is computed by propagating the current SLAM pose covariance one step forward. From the current SLAM node, we compute the resulting covariance assuming that the previous odometry holds for this one-step propagation too. Index $r$ refers to the current robot node, which is also the last node in the existing pose-graph, and all nodes later than $r$ are virtual.

$$\Sigma_{\text{exp}} = \Sigma_{r+1,r+1} = \mathrm{H}_{\mathtt{odo}_{r,r+1}}\Sigma_{rr}\mathrm{H}_{\mathtt{odo}_{r,r+1}}^{\top} \quad (6)$$

$$\mathbf{x}_{r,r+1} = \mathbf{x}_r \oplus \mathbf{x}_{r-1,r} \quad (7)$$

$$\mathrm{H}_{\mathtt{odo}_{r,r+1}} = \left[0,\cdots,0,\frac{\partial \mathbf{x}_{r,r+1}}{\mathbf{x}_r},0,\cdots,0\right] \quad (8)$$

Lastly, the reward term for robot uncertainty, $\mathcal{U}_{robot}^k$, 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, the robot uncertainty is defined as

$$\begin{aligned}\mathcal{U}_{robot}^{k=0} &= \begin{cases} 0, & \text{if } \frac{|\Sigma_{\text{exp}}|}{|\Sigma_{\text{allow}}|} < 1 \\ \frac{|\Sigma_{\text{exp}}|^{\frac{1}{6}}}{|\Sigma_{\text{allow}}|^{\frac{1}{6}}}, & \text{otherwise} \end{cases} \\ \mathcal{U}_{robot}^{k>0} &= \frac{|\Sigma_{nn}^k|^{\frac{1}{6}}}{|\Sigma_{\text{allow}}|^{\frac{1}{6}}}, \quad k = 1,\cdots,N_{wp}\end{aligned}, \quad (9)$$

where the sixth root of the 6-degree of freedom (DOF) pose determinant is used [31], [32] to yield a measure with SI units of $\mathrm{m}\cdot\mathrm{rad}$.

Essentially, PDN compares the two propagated uncertainties from revisiting and exploring, and then chooses the smaller one whenever the exploration uncertainty exceeds the reference allowable uncertainty. When the revisit action has the same or less value than pursuing exploration, the revisit does not produce enough loop-closures to overcome the increased navigation uncertainty from detouring. Note that in the previous studies by Bourgault et al. [14], Makarenko et al. [13], and Stachniss et al. [15], revisiting is always expected to be beneficial since there is no consideration for the actual likelihood of obtaining the loop-closure. In our approach, however, PDN has a realistic expectation for the likelihood of camera loop-closures based upon visual saliency.

*3) Area Coverage ($\mathcal{A}_{map}$):* As a final step in the reward calculation, we add a bias term for area coverage. Our purpose 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 localization uncertainty 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 coverage area is provided by the user,

$$\mathcal{A}_{map}^k = \frac{\mathcal{A}_{\text{to\_cover}}}{\mathcal{A}_{\text{target}}} = \frac{\mathcal{A}_{\text{target}} - \mathcal{A}_{\text{covered}} + \mathcal{A}_{\text{redundant}}^k}{\mathcal{A}_{\text{target}}}. \quad (10)$$

Here, $\mathcal{A}_{\text{target}}$ is the target coverage area, $\mathcal{A}_{\text{covered}}$ is the survey area already explored, 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 length with the sensor field of view width and has nonzero value to penalize the revisit action—it is zero for exploration.

*4) Combined Reward:* To fuse the two reward terms, we introduce a weight, $\alpha$, that controls the balance between the pose uncertainty and area coverage terms. Although we seek to maximize the reward, the formulation can be more intuitively understood when we consider each term as a penalty. The uncertainty increase 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 each waypoint $k$. The reward is the minus of this penalty, and PDN selects an action with the largest reward, or in other words, the one with minimal penalty:

$$\mathcal{C}^k = \alpha \cdot \mathcal{U}^k_{robot} + (1-\alpha) \cdot \mathcal{A}^k_{map}, \quad (11)$$

$$\mathcal{R}^k = -\mathcal{C}^k. \quad (12)$$

By adjusting $\alpha$, we can change the emphasis on robot navigation uncertainty versus area coverage performance in the reward evaluation. When $\alpha = 0$ no weight is imposed on the pose uncertainty and the algorithm tries to cover the area as fast as possible. This corresponds to an open-loop survey over the target area. On the other hand, when $\alpha = 1$ full weight is on the pose uncertainty and the robot will revisit whenever it exceeds the allowable navigation uncertainty. Our approach allows intuitive selection of weight [33] as it balances between two normalized terms, whereas weighting factors are experimentally determined in other works [34].

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

$$k^* = \operatorname{argmax} \mathcal{R}^k = \operatorname{argmin} \mathcal{C}^k, \quad (13)$$

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

## IV. Results

In this section, we present an evaluation of PDN as applied to a hybrid simulation generated from real ship hull inspection data. We surveyed the *SS Curtiss* in February 2011 using the Hovering Autonomous Underwater Vehicle (HAUV) [19] and built a dense SLAM result as shown in Fig. 7. Using this densely sampled SLAM result as a baseline, we plan a simulated mission by subsampling from it a set of nominal trajectory nodes—unused nodes are reserved for PDN to simulate revisit actions. In the first set of tests, we provide a synthetic saliency distribution on the mapping area to evaluate the performance of PDN using known ground-truth. In the second set of tests, we present hybrid simulation results from PDN using the real underwater hull imagery. In all cases of evaluation the PDN result is compared against two typical survey patterns: "open-loop survey" and "exhaustive revisit". Open-loop survey follows the preplanned nominal trajectory without any revisiting, while exhaustive revisit uses deterministic revisit actions to achieve loop-closures.

In our application, the SLAM navigation uncertainty is dominated by $xy$ positional uncertainty because depth is bounded with absolute pressure-depth measurements. For a given desired positional uncertainty bound, $\sigma_{xy,allow}$, the overall allowable navigation covariance bound is computed as

$$|\Sigma_{\text{allow}}| = \sigma^2_{xy,allow} \cdot \sigma^2_{xy,allow} \cdot \sigma^2_d \cdot \sigma^2_r \cdot \sigma^2_p \cdot \sigma^2_h,$$

where depth uncertainty $\sigma_d = 0.01$ m and attitude uncertainty $\sigma_r = \sigma_p = \sigma_h = 0.1^\circ$ (roll, pitch and heading) are used. In this evaluation, we specify $\sigma_{xy,allow} = 0.25$ m. The target coverage area is computed using the vessel's longitudinal length $\ell$, width (half-beam) $w$, and draft $h$. In this evaluation, the target coverage area is $\mathcal{A}_{target} = \ell \times (w + h) = 40$ m $\times$ $(20$ m $+ 10$ m$) = 1200$ m$^2$.

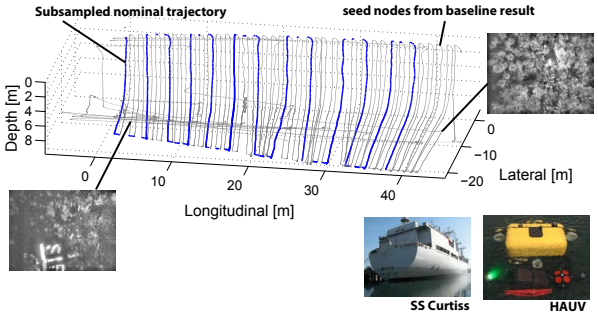



Fig. 7: Setup of the hybrid simulation for PDN's evaluation. Gray dots are nodes from the baseline SLAM result. The sampled nominal trajectory in blue mimics a simulated mission by sub-sampling from the baseline SLAM result. The nodes not used in the nominal trajectory planning will be used as seed nodes in the simulated control phase. Note that each seed node is associated with a real underwater image. Thumbnails show the HAUV, the vessel we surveyed (*SS Curtiss*), and sample imagery from its hull.

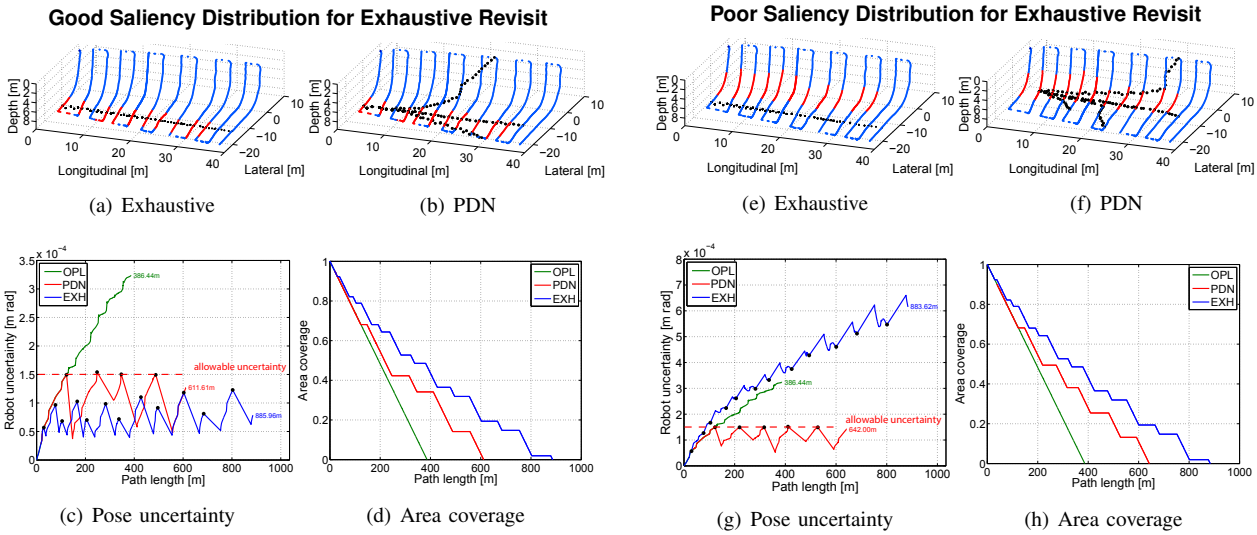
### A. PDN with Synthetic Saliency Map

The first set of tests are with a synthetic saliency map imposed over the area. We set $\alpha = 1$ so that full weight is on the pose uncertainty. The PDN action is verified for two different types of saliency distributions—distributed and concentrated, which we compare to exhaustive revisit control and open-loop survey.

For the exhaustive revisit, the robot is commanded to revisit a point on the first trackline in every other survey leg. In this setup, the exhaustive revisit happens on a line along the bottom of the hull (i.e., the keel). Because in practice this revisit path is typically preplanned without knowledge of the actual feature distribution in the environment, we use the same exhaustive revisit path for both the distributed and concentrated saliency simulations. Hence, when we have a biased feature distribution, as in our simulation, the exhaustive revisit path can either be always on the salient regions (Fig. 8(a)) or never pass through the salient regions (Fig. 8(e)).

Fig. 8 shows results for where the visual feature distribution is biased, this is to show how the preplanned exhaustive revisit either succeeds and fails depending on the saliency distribution. A measure of the robot's pose uncertainty, $\sqrt[6]{|\Sigma_{rr}|}$, is plotted in Fig. 8(c) and (g). Fig. 8(d) and (h) plot the ratio of the remaining area to cover with respect to the path length, where the black dots indicate points when revisit occurred.

When all of the exhaustive revisit paths land on the salient region, the likelihood of obtaining loop-closure during the revisit is higher, and the exhaustive revisit achieves tightly bounded uncertainty for the robot pose. On the other hand, when none of the revisit paths are on salient regions, as in the case of Fig. 8(e), exhaustive revisit basically performs worse than open-loop. Without meaningful loop-closures on the revisit, the control just increases path length and slows coverage rate, as can be seen in Fig. 8(g) and (h). Unfortunately, the




(a) Exhaustive (b) PDN (e) Exhaustive (f) PDN

(c) Pose uncertainty (d) Area coverage (g) Pose uncertainty (h) Area coverage

Fig. 8: PDN results for synthetic saliency maps. (a), (b), (e), (f) Trajectory of the robot with nodes color-coded by their saliency level. The same color scheme as in Fig. 4 is used, while the black dots indicate points when revisit occurred. (c), (d), (g), (h) Pose uncertainty and area coverage performance are compared for open-loop (green), exhaustive revisit (blue), and PDN (red). The performance of exhaustive revisit strongly depends upon the spatial distribution of feature-rich regions in the target area and their intersection with the preplanned revisit path, while for PDN it is able to automatically adapt to the saliency distribution in an intelligent way to maintain consistent navigation performance.

salient region distribution cannot be known a priori when the survey 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 the two saliency cases, since PDN is able to automatically adapt its revisit actions to the environment, resulting in consistent navigation performance and area coverage.

### *B. PDN with Real Image Data*

We now evaluate PDN's performance using real underwater images from the same dataset. This time a weight factor of $\alpha = 0.75$ is used so that navigation uncertainty is only given a mild preference over area coverage performance.

Similar to the synthetic images case, the uncertainty and area coverage graph for PDN are compared with open-loop and exhaustive revisit. Based upon our knowledge of the actual resulting saliency distribution obtained in the baseline SLAM result, we preplanned the exhaustive revisit path to be over a visually salient region to provide the best possible case for comparison with PDN. Because the exhaustive revisit is intentionally planned over the salient region, the resulting graph for exhaustive revisit shows the maximum obtainable SLAM performance—maintaining low uncertainty, but producing an exceeding number of revisits (12) and longer path length (866.28 m).

Fig. 9(a)-(b) show the uncertainty change and area coverage rate for open-loop, exhaustive revisit, and PDN. As shown in Fig. 9(e), PDN adapted its trajectories to obtain visual loop-closures to reduce the uncertainty whenever it exceeds the allowable covariance bound. Note that the number of revisits by PDN (five) is substantially smaller than the exhaustive revisit case (twelve). PDN's result uses a less number of revisits while still maintaining full control over the navigation uncertainty level. The loop-closing camera measurements are clearly illustrated in the time elevation graph of Fig. 9(d)-(f). The red lines in the graph depict the camera measurements made by the loop-closures. Because the trackline spacing is wide, there is no image overlap between adjacent tracklines, and all of the camera measurements in the graph are from revisit actions. As can be seen in the time elevation graphs, PDN obtained a similar number of loop-closures as compared to the exhaustive revisit case. A video showing the PDN process on this hybrid simulation is available as multimedia attachment `pnd.mp4`.

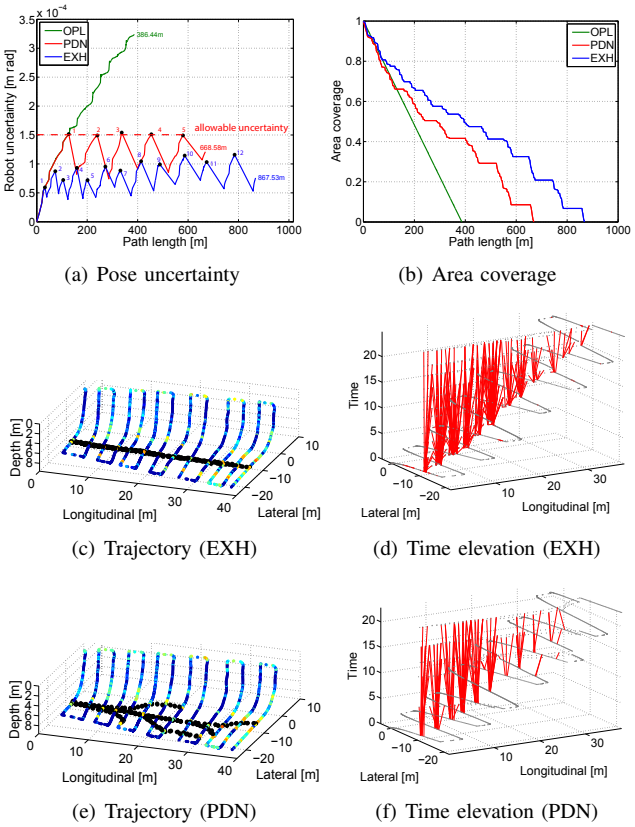
## V. Conclusion

This paper presented perception-driven navigation, an active SLAM approach that takes into account area coverage and navigation uncertainty performance to efficiently explore a target area of interest. A weighting factor, $\alpha$, provides control over this balance. A hybrid simulation using trajectories with both synthetic and real underwater images were tested to evaluate PDN's performance. Results show PDN's ability to plan visually plausible revisit paths for loop-closure while controlling the navigation uncertainty level and achieving efficient area coverage rates.

## References

[1] S. Prentice and N. Roy, "The belief roadmap: Efficient planning in linear POMDPs by factoring the covariance," *Int. J. Robot. Res.*, vol. 28, no. 11-12, pp. 1448–1465, Dec. 2009.

[2] R. Valencia, J. Andrade-Cetto, and J. Porta, "Path planning in belief space with pose SLAM," in *Proc. IEEE Int. Conf. Robot. and Automation*, May. 2011, pp. 78–83.




(a) Pose uncertainty (b) Area coverage

(c) Trajectory (EXH) (d) Time elevation (EXH)

(e) Trajectory (PDN) (f) Time elevation (PDN)

Fig. 9: PDN evaluation for real imagery. Similar to the synthetic saliency case, (a) and (b) show the pose uncertainty and area coverage performance with respect to the path length—annotated black dots indicate when revisits occurred. In trajectories (c) and (e), nodes are color-coded by their measured visual saliency from the real imagery; for visual clarity, nodes included in the revisit path are shown as black dots. The exhaustive revisit was purposely preplanned over the salient region, note that PDN is able to find this same optimal path to follow as in (e). In the time elevation graphs (d) and (f), PDN shows a similar number of successful loop-closures to exhaustive revisit.


[3] D. S. Levine, "Information-rich path planning under general constraints using rapidly-exploring random trees," Master's thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, Cambridge MA, Jun. 2010.

[4] R. Bajcsy, "Active perception," *Proc. of the IEEE*, vol. 76, no. 8, pp. 996–1005, 1988.

[5] H. J. S. Feder, J. J. Leonard, and C. M. Smith, "Adaptive mobile robot navigation and mapping," *Int. J. Robot. Res.*, vol. 18, no. 7, pp. 650–668, 1999.

[6] R. Sim and N. Roy, "Global a-optimal robot exploration in SLAM," in *Proc. IEEE Int. Conf. Robot. and Automation*, Barcelona, Spain, 2005, pp. 661–666.

[7] R. Sim, "Stable exploration for bearings-only SLAM," in *Proc. IEEE Int. Conf. Robot. and Automation*, Apr. 2005, pp. 2411–2416.

[8] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 1052–1067, 2007.

[9] M. Bryson and S. Sukkarieh, "An information-theoretic approach to autonomous navigation and guidance of an uninhabited aerial vehicle in unknown environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, Aug. 2005, pp. 3770–3775.

[10] P. Whaite and F. Ferrie, "Autonomous exploration: Driven by uncertainty," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 3, pp. 193–205, mar 1997.

[11] H. H. Gonzalez-Banos and J.-C. Latombe, "Navigation strategies for exploring indoor environments," *Int. J. Robot. Res.*, vol. 21, no. 10–11, pp. 829–848, 2002.

[12] C. Connolly, "The determination of next best views," in *Proc. IEEE Int. Conf. Robot. and Automation*, vol. 2, Mar. 1985, pp. 432–435.

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

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

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

[16] T. Kollar and N. Roy, "Trajectory optimization using reinforcement learning for map exploration," *Int. J. Robot. Res.*, vol. 27, no. 2, pp. 175–196, 2008.

[17] R. Martinez-cantin, O. D. Freitas, A. Doucet, and J. A. Castellanos, "Active policy learning for robot planning and exploration under uncertainty," in *Proc. Robot.: Sci. & Syst. Conf.*, 2007.

[18] A. Kim and R. M. Eustice, "Combined visually and geometrically informative link hypothesis for pose-graph visual SLAM using bag-of-words," in *Proc. of the IEEE/RSJ Int. Conf. on Intell. Robots and Syst.*, San Francisco, CA, Sep. 2011, pp. 1647–1654.

[19] ——, "Real-time visual SLAM for autonomous underwater hull inspection using visual saliency," *IEEE Trans. Robot.*, 2013, in Press.

[20] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental smoothing and mapping," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1365–1378, Dec. 2008.

[21] M. Kaess, H. Johannsson, and J. Leonard, "Open source implementation of iSAM," http://people.csail.mit.edu/kaess/isam, 2010.

[22] F. S. Hover, R. M. Eustice, A. Kim, B. Englot, H. Johannsson, M. Kaess, and J. J. Leonard, "Advanced perception, navigation and planning for autonomous in-water ship hull inspection," *Int. J. Robot. Res.*, vol. 31, no. 12, pp. 1445–1464, Oct. 2012.

[23] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Int. Conf. on Knowledge Discovery and Data Mining*, 1996, pp. 226–231.

[24] M. Daszykowski, B. Walczak, and D. L. Massart, "Looking for natural patterns in data part 1: Density-based approach," *Chemometrics and Intelligent Laboratory Systems*, vol. 56, no. 2, pp. 83–92, 2001.

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

[26] L. E. Kavraki and S. M. LaValle, "Motion planning," in *Springer Handbook of Robotics*. Springer, 2008, pp. 109–131.

[27] R. M. Eustice, H. Singh, and J. J. Leonard, "Exactly sparse delayed-state filters for view-based SLAM," *IEEE Trans. Robot.*, vol. 22, no. 6, pp. 1100–1114, Dec. 2006.

[28] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous Robot Vehicles*, I. Cox and G. Wilfong, Eds. Springer-Verlag, 1990, pp. 167–193.

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

[30] R. M. Eustice, H. Singh, J. J. Leonard, and M. R. Walter, "Visually mapping the RMS Titanic: Conservative covariance estimates for SLAM information filters," *Int. J. Robot. Res.*, vol. 25, no. 12, pp. 1223–1242, 2006.

[31] H. Carrillo, I. Reid, and J. Castellanos, "On the comparison of uncertainty criteria for active SLAM," in *Proc. IEEE Int. Conf. Robot. and Automation*, May. 2012, pp. 2080–2087.

[32] J. Kiefer, "General equivalence theory for optimum designs (approximate theory)," *The Annals of Statistics*, vol. 2, no. 5, pp. 849–879, 1974.

[33] A. Kim and R. M. Eustice, "Next-best-view visual SLAM for bounded-error area coverage," in *IROS Workshop on Active Semantic Perception*, Vilamoura, Portugal, Oct. 2012.

[34] J. Du, L. Carlone, M. Kaouk Ng, B. Bona, and M. Indri, "A comparative study on active SLAM and autonomous exploration with particle filters," in *IEEE/ASME Int. Conf. on Advanced Intell. Mechatronics*, Jul. 2011, pp. 916–923.