Visually Navigating the RMS Titanic with SLAM Information Filters

***Abstract*— This paper describes a vision-based large-area simultaneous localization and mapping (SLAM) algorithm that respects the constraints of low-overlap imagery typical of underwater vehicles while exploiting the information associated with the inertial sensors that are routinely available on such platforms. We present a novel strategy for efficiently accessing and maintaining consistent covariance bounds within a SLAM information filter, greatly increasing the reliability of data association. The technique is based upon solving a sparse system of linear equations coupled with the application of constant-time Kalman updates. The method is shown to produce consistent covariance estimates suitable for robot planning and data association. Real-world results are presented for a vision-based 6 DOF SLAM implementation using data from a recent ROV survey of the wreck of the RMS Titanic.**

I. Introduction

This paper addresses the problem of precision navigation and mapping using low-overlap, high resolution image sequences obtained by autonomous undersea vehicles. From a "robotics science" perspective, our primary contribution consists of an efficient algorithm for extracting consistent covariance bounds from SLAM information filters. From a "robotics systems" perspective, we demonstrate automatic visually-augmented navigation processing of a sequence of 866 images of the RMS Titanic (Fig. 1), for a mission with a vehicle track length over 3 km long.

A number of oceanographic applications share the requirement for high resolution imaging of sites extending over hundreds of meters. These include hydrothermal vent sites, cold seep sites, shipwrecks of archaeological significance, coral reefs, and fisheries habitat related regions of interest. One of the significant challenges associated with such tasks is the requirement for precise and accurate navigation to ensure complete repeatable coverage over the site of interest.

Traditionally, the oceanographic community has utilized three different methodologies (by themselves or in combination) to address navigation underwater [1]: (1) transponder networks placed on the seafloor, (2) ship to underwater vehicle bearing (ultra-short baseline) tracking systems, and (3) ranging and inertial sensors on the underwater vehicle. Each of these methodologies trade off different aspects of accuracy, cost, and complexity. For example, transponder networks provide accurate navigation on the seafloor but come at the cost of the overhead required for the deployment and calibration of the individual transponders on the seafloor. These systems are also limited to providing updates every few seconds based on the travel time between the vehicle and the transponder beacon.

In this paper we explore a methodology that utilizes a vision-based SLAM approach to providing high precision accurate navigation measurements when used in concert with inertial measurements made on board by the vehicle. The goal is an algorithm that respects the constraints of low overlap for large-area extended surveys that are typical of imaging from underwater vehicles. Our approach considers this problem from the "information formulation" of SLAM.

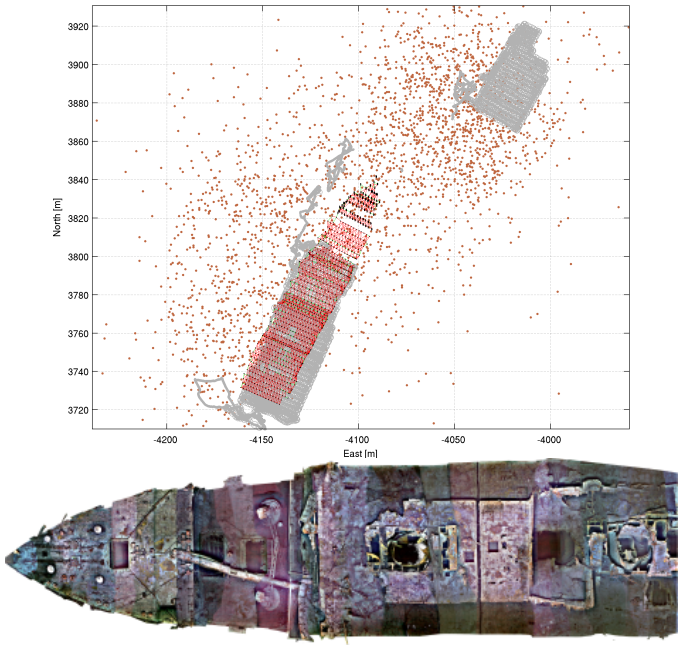


Fig. 1. Mapping results from a summer of 2004 ROV survey of the RMS Titanic. (top) XY plot comparing the raw dead-reckon navigation data (gray), ship-board ultra-short baseline tracking (brown), and reconstructed survey trajectory from a vision-based 6 DOF SLAM information filter (red). (bottom) A photomosaick of the RMS Titanic constructed from over 700 digital still images. Note that this photomosaick is presented for visualization purposes only as a representation of the data that serves as input to our algorithm. It is the result of semi-automatic processing with manual selection of a number of common scene points to guide the photomosaicking process. This could be considered as a form of benchmark against which fully autonomous processing can be compared.

Within the SLAM community, algorithms exploiting the sparse information representation for SLAM were first proposed by Thrun *et al.* [2], Frese [3], [4], and Paskin [5]. These methods exploit the empirical observation that this representation is either sparse or "close to sparse". The sparse information representation allows for linear storage requirements and efficient fusion of sensor measurements. However, the recovery of covariances is a cubic operation if a naive approach is followed.

The key issue on which we focus in this paper is the efficient recovery of *consistent* covariances from the information filter. It is hard to define a single definition of consistency employed uniformly in the prior literature on SLAM. Intuitively, consistency reflects the goal that the error estimates computed by the filter should "match" the actual errors.

In relation to SLAM, consistency of the error estimates is important for data association — determining the correspondences for measurements [6]. This is important both in the context of "local" SLAM (detecting and tracking features), and in a "global" sense (for closing loops). If the SLAM error estimates are too small (over-confident), both of these tasks can become difficult, as will be shown in §IV.

Before describing our approach for efficient recovery of consistent covariances bounds, we first review the basic characteristics of SLAM information filters.

II. SLAM Information Filters

A number of recent SLAM algorithms have explored reformulating the estimation problem within the context of an extended information filter [2], [5], [7], [8]. The information form is often called the canonical or natural representation of the Gaussian distribution because it stems from expanding the quadratic in the exponential. The result is that rather than parameterizing the normal distribution in terms of its mean and covariance as in $\mathcal{N}\big(\boldsymbol{\xi}_t; \boldsymbol{\mu}_t, \Sigma_t\big)$, it is instead parametrized in terms of its information vector and information matrix, $\mathcal{N}^{-1}\big(\boldsymbol{\xi}_t; \boldsymbol{\eta}_t, \Lambda_t\big)$ [9]. The two forms are related via (1).

$$\Lambda_t = \Sigma_t^{-1} \qquad \boldsymbol{\eta}_t = \Lambda_t \boldsymbol{\mu}_t \tag{1}$$

A. Constant-Time Measurement Updates

A well known and very attractive property of formulating SLAM in the information form is that measurement updates are an additive and efficient operation. This is in contrast to the quadratic complexity per update in the covariance form. For example, assume the following general nonlinear measurement function (2) and its first order linearized form (3)

$$\mathbf{z}_t = \mathbf{h}(\boldsymbol{\xi}_t) + \mathbf{v}_t \tag{2}$$

$$\approx \mathbf{h}(\bar{\boldsymbol{\mu}}_t) + \mathrm{H}(\boldsymbol{\xi}_t - \bar{\boldsymbol{\mu}}_t) + \mathbf{v}_t \tag{3}$$

where $\boldsymbol{\xi}_t$ is the predicted state vector distributed according to $\boldsymbol{\xi}_t \sim \mathcal{N}\big(\bar{\boldsymbol{\mu}}_t, \bar{\Sigma}_t\big) = \mathcal{N}^{-1}\big(\bar{\boldsymbol{\eta}}_t, \bar{\Lambda}_t\big)$, $\mathbf{v}_t$ is the white measurement noise $\mathbf{v}_t \sim \mathcal{N}\big(\mathbf{0}, \mathrm{R}\big)$, and H is the Jacobian evaluated at $\bar{\boldsymbol{\mu}}_t$. The EKF covariance update requires computing the Kalman gain and updating $\bar{\boldsymbol{\mu}}_t$ and $\bar{\Sigma}_t$ via (4) [9]. This calculation non-trivially modifies all elements in the covariance matrix resulting in quadratic computational complexity *per* update [10].

$$\begin{aligned} \mathrm{K} &= \bar{\Sigma}_t \mathrm{H}^\top \big(\mathrm{H}\bar{\Sigma}_t \mathrm{H}^\top + \mathrm{R}\big)^{-1} \\ \boldsymbol{\mu}_t &= \bar{\boldsymbol{\mu}}_t + \mathrm{K}\big(\mathbf{z}_t - \mathbf{h}(\bar{\boldsymbol{\mu}}_t)\big) \\ \Sigma_t &= \big(\mathrm{I} - \mathrm{KH}\big)\bar{\Sigma}_t \big(\mathrm{I} - \mathrm{KH}\big)^\top + \mathrm{KRK}^\top \end{aligned} \tag{4}$$

In contrast the corresponding EIF update is given by (5) [2].

$$\begin{aligned} \Lambda_t &= \bar{\Lambda}_t + \mathrm{H}^\top \mathrm{R}^{-1} \mathrm{H} \\ \boldsymbol{\eta}_t &= \bar{\boldsymbol{\eta}}_t + \mathrm{H}^\top \mathrm{R}^{-1} \big(\mathbf{z}_t - h(\bar{\boldsymbol{\mu}}_t) + \mathrm{H}\bar{\boldsymbol{\mu}}_t\big) \end{aligned} \tag{5}$$

Equation (5) shows that the information matrix is additively updated by the outer product term $\mathrm{H}^\top \mathrm{R}^{-1} \mathrm{H}$. In general, this outer product modifies all elements of the predicted information matrix $\bar{\Lambda}_t$, however a key observation is that the SLAM Jacobian H is always sparse [2]. For example, in our application we use a view-based SLAM implementation built around using a camera to extract relative pose measurements from pairwise registration of overlapping images of the environment. Given a pair of images I_i and I_j, image registration provides a relative pose measurement between states $\mathbf{x}_i$ and $\mathbf{x}_j$ resulting in a sparse Jacobian of the form

$$\mathrm{H} = \begin{bmatrix} 0 \cdots & \frac{\partial \mathbf{h}}{\partial \mathbf{x}_i} & \cdots 0 \cdots & \frac{\partial \mathbf{h}}{\partial \mathbf{x}_j} & \cdots 0 \end{bmatrix} \tag{6}$$

As a result only the four-block elements corresponding to $\mathbf{x}_i$ and $\mathbf{x}_j$ of the information matrix need to be modified (i.e., $\bar{\Lambda}_{x_i x_i}$, $\bar{\Lambda}_{x_j x_j}$, and $\bar{\Lambda}_{x_i x_j} = \bar{\Lambda}_{x_j x_i}^\top$). Since measurements only ever involve a fixed portion of the SLAM state vector updates can be performed in constant time.

B. Sparse Representation

Thrun *et al.* [2] originally showed that the feature-based SLAM filter information matrix empirically obeys a "sparse" structure when properly normalized. This observation has lead to the development of a number of computationally efficient feature-based SLAM algorithms such as Sparse Extended Information Filters (SEIFs) [2], Thin-Junction Tree Filters (TJTFs) [5], and Tree-Map filters [7]. These algorithms approximate the SLAM posterior by eliminating "small" elements in the information matrix. The elimination of weak constraints results in a sparse information matrix allowing the development of efficient filter algorithms that exploit the resulting sparse information matrix architecture. This empirical observation of weak inter-landmark constraints has recently been given a solid theoretical foundation by Frese [4] where he mathematically shows that information between landmark features decays spatially at an exponential rate. This adds some justification for the sparseness approximations in the feature-based SLAM information form.

In addition to feature-based techniques, a recent paper by Eustice *el al.* [8] shows that for a view-based representation the SLAM information matrix is *exactly* sparse without having to make any approximations. The implication of this result is that view-based SLAM systems can take advantage of the sparse information parameterization without incurring any approximation error. Based upon this insight, for our undersea application we've implemented a view-based SLAM system built around fusing 6 DOF relative pose camera measurements from monocular overlapping seafloor imagery with traditional underwater vehicle dead-reckon navigation sensors. The result shown in Fig. 2 illustrates the information matrix representation associated with registering 866 images and fusing them with navigation data from a grid-based ROV survey of the wreck of the RMS Titanic. The off-diagonal elements correspond to spatial relative pose measurements made by the camera while the block-tridiagonal elements arise from the Markov process model and temporal camera measurements.

The wreck was surveyed from midship to stern and then from midship to bow resulting in a large loop-closing event which can be seen in the information matrix as pointed out in Fig. 2.

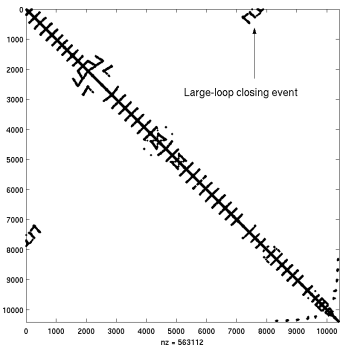


Fig. 2. This figure highlights the *exact* sparsity of the view-based SLAM information matrix using data from a recent ROV survey of the wreck of the RMS Titanic. In all there are 867 robot states where each state is a 12-vector consisting of 6 pose and 6 kinematic components. The resulting information matrix is a $10,404 \times 10,404$ matrix with only 0.52% nonzero elements.

C. State Recovery

While the insight of "sparseness" has lead to the development of computationally efficient SLAM algorithms such as the ones previously mentioned, an issue countering the information filter is the question of how to gain efficient access to the state estimate and its uncertainty. Referring back to (1) we see that the information parameterization embeds the state mean and covariance within the information vector and information matrix respectively. State recovery implies that whenever we want to actually recover our state estimate for the purposes of motion planning, data association, map recovery, linearizing our process or observation models, etc., we must invert the relationship given in (1).

1) Recovering the mean: Naive recovery of our state estimate through matrix inversion results in cubic complexity and destroys any efficiency gained over the EKF. Fortunately, closer inspection shows that recovery of the state mean $\boldsymbol{\mu}_t$ can be posed more efficiently as solving the sparse, symmetric, positive-definite, linear system of equations shown in (7).

$$\Lambda_t \boldsymbol{\mu}_t = \boldsymbol{\eta}_t \tag{7}$$

Such systems can be solved via the classic iterative method of conjugate gradients (CG) [11]. In general, CG can solve this system in n iterations (with $\mathcal{O}(n)$ cost per iteration where n is the size of the state vector) and typically in many fewer iterations if the initialization is good [12]. In addition, since the state mean $\boldsymbol{\mu_t}$ typically does not change significantly with each measurement update (excluding key events like loop-closure) this relaxation can take place over multiple time steps using a fixed number of iterations per update [2], [13]. Also, recently proposed multigrid SLAM algorithms such as [12], [14] appear capable of solving this system with linear asymptotic complexity. This is achieved by sub-sampling poses and performing the relaxation over multiple spatial resolutions which has the effect of improving convergence rates.

2) Recovering covariance: The covariance matrix corresponds to the inverse of the information matrix, however, actually recovering the covariance via (1) is not practical since matrix inversion is a cubic operation. Additionally, while the information matrix can be a sparse representation for storage, in general, its inverse results in a *fully dense* covariance matrix despite any sparsity in the information form [3]. This means that calculating the covariance matrix requires quadratic memory storage which may become prohibitively large for very large maps (e.g., maps $\geq \mathcal{O}(10^5)$ state elements). To illustrate this point, for the $10,404 \times 10,404$ information matrix shown in Fig. 2, storing it in memory only requires 4.5MB of double precision storage for the nonzero elements while its inverse requires over 865MB.

Fortunately, recovering the entire covariance matrix usually isn't necessary for SLAM as many of the data association and robotic planning decisions typically do not require the full covariance matrix but only the covariance over subsets of state variables [15]. Unfortunately, accessing only subsets of state variables in the information form is not an easy task. The covariance and information representations of the Gaussian distribution lead to very different computational characteristics with respect to the fundamental probabilistic operations of marginalization and conditioning. Table I summarizes these operations where we see that the covariance and information representations exhibit a dual relationship. For example, marginalization is easy in covariance form since it corresponds to extracting the appropriate sub-block from the covariance matrix while in information form it is hard because it involves calculating the Schur complement over the variables we wish to keep (note that the opposite relation holds true for conditioning which is easy in the information form and hard in covariance form). Therefore, even though we may only need access to covariances over subsets of the state elements [15] (and thus only have to invert a small information matrix related to the subset of variables we are interested in), accessing them in the information form requires marginalizing out most of our state vector resulting in cubic complexity due to matrix inversion in the Schur complement.

TABLE I

SUMMARY OF MARGINALIZATION AND CONDITIONING OPERATIONS ON A GAUSSIAN DISTRIBUTION EXPRESSED IN COVARIANCE AND INFORMATION FORM

$p(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \mathcal{N}\left(\left[\begin{smallmatrix}\boldsymbol{\mu}_\alpha \\ \boldsymbol{\mu}_\beta\end{smallmatrix}\right], \left[\begin{smallmatrix}\Sigma_{\alpha\alpha} & \Sigma_{\alpha\beta} \\ \Sigma_{\beta\alpha} & \Sigma_{\beta\beta}\end{smallmatrix}\right]\right) = \mathcal{N}^{-1}\left(\left[\begin{smallmatrix}\boldsymbol{\eta}_\alpha \\ \boldsymbol{\eta}_\beta\end{smallmatrix}\right], \left[\begin{smallmatrix}\Lambda_{\alpha\alpha} & \Lambda_{\alpha\beta} \\ \Lambda_{\beta\alpha} & \Lambda_{\beta\beta}\end{smallmatrix}\right]\right)$

	MARGINALIZATION $p(\boldsymbol{\alpha}) = \int p(\boldsymbol{\alpha}, \boldsymbol{\beta})\, d\boldsymbol{\beta}$	CONDITIONING $p(\boldsymbol{\alpha} \mid \boldsymbol{\beta}) = p(\boldsymbol{\alpha}, \boldsymbol{\beta}) / p(\boldsymbol{\beta})$
COV. FORM	$\boldsymbol{\mu} = \boldsymbol{\mu}_\alpha$ $\Sigma = \Sigma_{\alpha\alpha}$	$\boldsymbol{\mu}' = \boldsymbol{\mu}_\alpha + \Sigma_{\alpha\beta}\Sigma_{\beta\beta}^{-1}(\boldsymbol{\beta} - \boldsymbol{\mu}_\beta)$ $\Sigma' = \Sigma_{\alpha\alpha} - \Sigma_{\alpha\beta}\Sigma_{\beta\beta}^{-1}\Sigma_{\beta\alpha}$
INFO. FORM	$\boldsymbol{\eta} = \boldsymbol{\eta}_\alpha - \Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\boldsymbol{\eta}_\beta$ $\Lambda = \Lambda_{\alpha\alpha} - \Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\Lambda_{\beta\alpha}$	$\boldsymbol{\eta}' = \boldsymbol{\eta}_\alpha - \Lambda_{\alpha\beta}\boldsymbol{\beta}$ $\Lambda' = \Lambda_{\alpha\alpha}$

To get around this dilemma, Thrun *et al.* propose a data association strategy based upon using *conditional* covariances [2], [16]. Since conditional information matrices are easy to obtain in the information form (simply extract a sub-block over the desired variables) their strategy is to choose an appropriate sub-block from the information matrix such that it's inverse approximates the actual covariance for the subset of variables they are interested in. In particular, given two state variables of interest, $\mathbf{x}_i$ and $\mathbf{x}_j$, their approximation selects the joint-Markov blanket $\mathbf{M}_i^+ \cup \mathbf{M}_j^+$ (i.e., $\mathbf{M}_k^+$ represents state variables *directly* connected to $\mathbf{x}_k$ in a graph theoretic sense within the information matrix) and additionally if the intersection is null (i.e., $\mathbf{M}_i^+ \cap \mathbf{M}_j^+ = \emptyset$) variables along a path connecting $\mathbf{x}_i$ and $\mathbf{x}_j$ topologically. Their method then extracts and inverts this sub-block to obtain an covariance matrix for $\mathbf{x}_i$ and $\mathbf{x}_j$ conditioned on all other variables which have an indirect influence. They note that empirical testing shows that their approximation method seems to work well in practice for their application [16] despite the fact that using conditional covariances should result in an over-confident approximation.

III. Consistent Covariance Recovery

Our strategy for approximate covariance recovery from the information form is formulated upon gaining efficient access to meaningful values of covariance which are consistent with respect to the actual covariance obtained by matrix inversion. The motivation for a consistent approximation is that we guard against under-representing the uncertainty associated with our state estimates which otherwise could lead to data association and robot planning errors. It is the access to meaningful values of joint-covariance for robot interaction, data association, and decision making in the information form which motivates our discussion. In this section we describe our strategy for obtaining covariance bounds within the context of our view-based SLAM application.

A. Efficiently Accessing The Robot's Covariance

We begin by noting that recovery of our state estimate $\boldsymbol{\mu}_t$ from the information form already requires that we solve the sparse, symmetric, positive-definite system of equations (7) and moreover that this system can be solved in linear time using the iterative techniques outlined in §II-C.1 (i.e., [12], [14]). Our covariance recovery strategy for the information form is based upon augmenting this linear system of equations so that the current robot pose covariance is accessible as well. Note that by definition (8) holds and therefore by picking the i^{th} basis vector $\mathbf{e}_i$ from the identity matrix we can use it to selectively solve for a column of the covariance matrix denoted as Σ_{*i} (9).

$$\Lambda_t \Sigma_t = \mathrm{I} \tag{8}$$

$$\Lambda_t \Sigma_{*i} = \mathbf{e}_i \tag{9}$$

To obtain the robot's covariance at any time step we simply augment our original linear system (7) to include an appropriate set of basis vectors $\mathrm{E}_r = \{\mathbf{e}_r\}$ such that the solution to (10) provides access to our current state and the robot's covariance-column.

$$\Lambda_t \begin{bmatrix} \boldsymbol{\mu}_\mathbf{t} & \Sigma_{*r} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\eta}_t & \mathrm{E}_r \end{bmatrix} \tag{10}$$

B. Consistent Covariances for Data Association

In this section we outline our strategy for recovering approximate joint-covariances useful for *data association*. Before we begin we want it to be clear to the reader that our technique for obtaining and maintaining these covariances should not be confused with the actual updating and mechanics of the information parameterization. What we present in the following section is a way of maintaining *covariance bounds* that are consistent with respect to the information parameterization. Furthermore, these covariances are used for data association *only* and are not in any way involved in the actual update and maintenance of the information filter representation. With that being said we now present our algorithm.

1) Inserting a new map element: Given that (10) provides a mechanism for efficient access to the robot's covariance-column Σ_{*r}, we exploit it to obtain useful covariance bounds for other map elements. For example, whenever we insert a new image I_i into our view-based map we correspondingly must add a new element $\mathbf{x}_i$ into our view-based SLAM state vector [8], [17]. This new state element $\mathbf{x}_i$ corresponds to a sampling of our robot state at time t_i (i.e., $\mathbf{x}_i = \mathbf{x}_r(t_i)$) and represents our estimate of where the robot was when it took that image. Since the two states are coincident at time t_i the covariance for $\mathbf{x}_i$ is $\Sigma_{ii} = \Sigma_{rr}$ and can be obtained by solving (10). A well-known property of SLAM is that over time the covariance for $\mathbf{x}_i$ will *decrease* as new sensor measurements are incorporated and all map elements become fully correlated [15]. Therefore, storing $\tilde{\Sigma}_{ii} = \Sigma_{ii}$ as our initial approximate covariance estimate for $\mathbf{x}_i$ serves as a *conservative* bound to the actual marginal covariance for all time, (i.e., $\tilde{\Sigma}_{ii} \geq \Sigma_{ii}(t)$).

2) Data association: In our application, the joint-covariance between the time-projected robot pose $\mathbf{x}_r$ and any other map entry $\mathbf{x}_i$, (i.e., $\bar{\Sigma}_{joint} = \left[\begin{smallmatrix} \bar{\Sigma}_{rr} & \bar{\Sigma}_{ri} \\ \bar{\Sigma}_{ri} & \Sigma_{ii} \end{smallmatrix}\right]$) is needed for two operations: link proposal and pose-constrained correspondence searches. Link proposal corresponds to hypothesizing which images in our view-based map could potentially share common overlap with the current image being viewed by the robot, denoted I_r, and therefore could potentially be registered to generate a relative pose measurement. The second operation, pose-constrained correspondence searches, uses the relative pose estimate between candidate images I_i and I_r to restrict the image-based correspondence search to probable regions based upon a two-view point transfer relation [17], [18]. [1]

To obtain the actual joint-covariance $\bar{\Sigma}_{joint}$ from the information form requires marginalizing out all other elements in our map except for $\mathbf{x}_r$ and $\mathbf{x}_i$ leading to cubic complexity in the number of eliminated variables. However, we can obtain a bounded approximation to $\bar{\Sigma}_{joint}$ at any time-step by using the

[1] Note that the standard maximum likelihood data association technique for feature-based SLAM also only depends on extracting $\bar{\Sigma}_{joint}$ [15].

solution from (10) to provide us with the current covariance-column representing the joint-covariances between the time-projected robot and all other map entries $\bar{\Sigma}_{*r}$ (note that this solution is equivalent to what could be obtained by full matrix inversion of $\bar{\Lambda}_t$). Using this result we can construct a conservative joint-covariance approximation to $\bar{\Sigma}_{joint}$ as

$$\tilde{\bar{\Sigma}}_{joint} = \begin{bmatrix} \bar{\Sigma}_{rr} & \bar{\Sigma}_{ir}^\top \\ \bar{\Sigma}_{ir} & \tilde{\Sigma}_{ii} \end{bmatrix} \tag{11}$$

where $\bar{\Sigma}_{rr}$ and $\bar{\Sigma}_{ir}$ are extracted from $\bar{\Sigma}_{*r}$, and $\tilde{\Sigma}_{ii}$ is our conservative covariance bound for $\mathbf{x}_i$ as described in §III-B.1. Note that (11) represents a valid positive-semidefinite, and therefore consistent, approximation satisfying

$$\tilde{\bar{\Sigma}}_{joint} - \bar{\Sigma}_{joint} = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{\Sigma}_{ii} - \Sigma_{ii} \end{bmatrix} \geq 0 \tag{12}$$

since $\tilde{\Sigma}_{ii} - \Sigma_{ii} \geq 0$. Given that (11) provides a consistent approximation to the true covariance, we can use it to compute conservative 1st-order probabilities of relative poses $\mathbf{x}_{ri} = \ominus\mathbf{x}_r \oplus \mathbf{x}_i$ in the usual way [10] for link hypothesis and correspondence searches.

3) Updating our covariance bounds: Since $\tilde{\Sigma}_{ii}$ serves as a *conservative* approximation to the actual covariance Σ_{ii} for map element $\mathbf{x}_i$, we would like to be able to place tighter bounds on it as we gather more measurement information. In fact, the careful reader will recognize that our SLAM information filter *is implicitly already doing this* for us, however the issue is that extracting the actual filter bound Σ_{ii} from the information matrix representation is not particularly convenient. Note that while we could access Σ_{ii} by solving for the covariance-column Σ_{*i} using an appropriately chosen set of basis vectors, the reason for not doing this is that iteratively solving systems like (10) is efficient only when we have a good starting point [12], [13]. In other words, when we solve (10) for the latest state and robot covariance-column, our estimates $\boldsymbol{\mu}_t$ and Σ_{*r} from that last time-step serve as good seed points and therefore typically only require a small number of iterations per time-step to update (excluding loop-closing events). In the case of solving for an arbitrary column Σ_{*i} we do not have a good *a priori* starting point and therefore convergence will be slower.

Our approach for tightening the bound $\tilde{\Sigma}_{ii}$ is to use our joint-covariance approximation (11) and perform a simple constant-time Kalman filter update on a *per* re-observation basis. In other words, we only update our covariance bound $\tilde{\Sigma}_{ii}$ when the robot re-observes $\mathbf{x}_i$ and successfully generates a relative pose measurement $\mathbf{z}_{ri}$ by registering images I_i and I_r. We then use that relative pose measurement to perform a Kalman update (4) on the fixed size state vector $\mathbf{y} = \left[\mathbf{x}_r^\top, \mathbf{x}_i^\top\right]^\top$ and obtain the new conservative bound $\tilde{\Sigma}_{ii}^+$.

Mathematically, the distribution over $\mathbf{y}$ corresponds to marginalizing out all elements in our state vector except for $\mathbf{x}_r$ and $\mathbf{x}_i$ as

$$p(\mathbf{y}) = \int_{\mathbf{x}_j \neq \{\mathbf{x}_r, \mathbf{x}_i\}} \mathcal{N}^{-1}\big(\bar{\boldsymbol{\eta}}_t, \bar{\Lambda}_t\big) d\mathbf{x}_j = \int_{\mathbf{x}_j \neq \{\mathbf{x}_r, \mathbf{x}_i\}} \mathcal{N}\big(\bar{\boldsymbol{\mu}}_t, \bar{\Sigma}_t\big) d\mathbf{x}_j \tag{13}$$

which results in the distribution

$$p(\mathbf{y}) = \mathcal{N}\Big(\begin{bmatrix} \bar{\boldsymbol{\mu}}_r \\ \bar{\boldsymbol{\mu}}_i \end{bmatrix}, \begin{bmatrix} \bar{\Sigma}_{rr} & \bar{\Sigma}_{ir}^\top \\ \bar{\Sigma}_{ir} & \Sigma_{ii} \end{bmatrix}\Big) \tag{14}$$

Noting that (11) already provides us with a consistent approximation to this distribution we have

$$\tilde{p}(\mathbf{y}) = \mathcal{N}\Big(\begin{bmatrix} \bar{\boldsymbol{\mu}}_r \\ \bar{\boldsymbol{\mu}}_i \end{bmatrix}, \begin{bmatrix} \bar{\Sigma}_{rr} & \bar{\Sigma}_{ir}^\top \\ \bar{\Sigma}_{ir} & \tilde{\Sigma}_{ii} \end{bmatrix}\Big) \tag{15}$$

where the only difference between the actual distribution (14) and the approximation (15) is the conservative marginal $\tilde{\Sigma}_{ii}$. Using the measurement $\mathbf{z}_{ri}$ we now perform a constant-time Kalman update (4) on (15) yielding the conditional distribution $\tilde{p}(\mathbf{y}|\mathbf{z}_{ri})$ from which we retain only the updated marginal bound $\tilde{\Sigma}_{ii}^+$ for element $\mathbf{x}_i$. This update is computed in constant-time for each re-observed feature.

Note that by abstractly performing the marginalization step of (13) before computing the Kalman update, we have avoided any inconsistency issues associated with only storing the marginal bounds $\tilde{\Sigma}_{ii}$ and not representing the intra-map correlations. This ensures that our update step will result in a consistent marginal bound for data association that will improve over time as we re-observe map elements.

Require: Σ_{*r} {initialize bound}
  if $\mathbf{x}_i$ = new map element **then**
    store $\tilde{\Sigma}_{ii} \leftarrow \Sigma_{rr}$
  end if

Require: $\bar{\mu}_t, \bar{\Sigma}_{*r}$ {data association and bound update}
  for all $\mathbf{x}_i$ **do**
    $\tilde{\bar{\Sigma}}_{joint} \leftarrow \begin{bmatrix} \bar{\Sigma}_{rr} & \bar{\Sigma}_{ri} \\ \bar{\Sigma}_{ri} & \tilde{\Sigma}_{ii} \end{bmatrix}$
    compute link hypothesis
    if candidate link **then**
      do constrained correspondence search on I_i and I_r
      if image registration success **then**
        do Kalman update on $\tilde{\Sigma}_{joint}$ using measurement $\mathbf{z}_{ri}$
        store $\tilde{\Sigma}_{ii} \leftarrow \Sigma_{ii}^+$
      end if
    end if
  end for

Algorithm 1: Calculation of marginal covariance bounds used for data association.

IV. Results

This section presents experimental results validating our covariance recovery strategy from the information form using data gathered during a recent survey of the RMS Titanic. The wreck was surveyed during the summer of 2004 by the deep-sea ROV *Hercules* operated by the Institute for Exploration of the Mystic Aquarium. The ROV was equipped with a standard suite of oceanographic dead-reckon navigation sensors capable of measuring heading, attitude, altitude, XYZ bottom-referenced Doppler velocities, and a pressure sensor for depth; Table II summarizes the sensor capabilities. In

addition, the vehicle was also equipped with a calibrated stereo rig consisting of two downward-looking 12-bit digital-still cameras that collected imagery at a rate of 1 frame every 8 seconds. Furthermore, note that the results being presented were produced using imagery from *one* camera only — the purpose of this self-imposed restriction to a monocular sequence of images is to demonstrate the general applicability of our visually augmented navigation strategy.

TABLE II
POSE SENSOR CHARACTERISTICS.

Measurement	Sensor	Precision
Roll/Pitch	Tilt Sensor	$\pm 0.1^\circ$
Heading	North-Seeking FOG	$\pm 0.1^\circ$
Body Frame Velocities	Acoustic Doppler	± 0.01 m/s
Depth	Pressure Sensor	± 0.01 m
Altitude	Acoustic Altimeter	± 0.1 m
Downlooking Imagery	Calibrated 12-bit CCD	1 frame every 8 s

Fig. 5 summarizes our mapping results using an exactly sparse view-based SLAM information filter as proposed by [8]. During the course of the grid-based survey the vehicle traversed a 2D path length of 3.1 km and a 3D XYZ path length of 3.4 km maneuvering to maintain a safe altitude off the deck of the wreck. The convex hull of the final mapped region encompasses an area over 3100m^2 and in all a total of 866 images were used to provide 3494 camera-generated relative-pose constraints. These constraints were generated using a state-of-the-art feature-based image registration approach [18] founded on:

- Extracting a combination of both Harris [19] and SIFT [20] interest points from each image.
- Establishing putative correspondences between overlapping candidate image pairs using a constrained correspondence search [17].
- Employing a statistically robust Least-Median-of-Squares [21] registration methodology to find the corresponding Essential matrix.
- Two-view maximum likelihood refinement to extract the 5 DOF relative pose constraint (i.e., azimuth, elevation, Euler roll, Euler pitch, Euler yaw) based upon minimizing the reprojection error [18].

In Fig. 5(a) we see a time progression of the camera constraints and vehicle pose estimation result. In particular, the third figure from the left shows the closing of a large loop where the vehicle meandered its way from the stern of the ship back towards the bow with it's camera turned off and then successfully relocalized based upon correctly registering 4 image pairs out of 64 hypothesized candidates. Fig. 5(b) shows the final resulting pose-constraint network and Fig. 5(c) a "zoomed in" view of the boxed region to facilitate comparison of the marginal covariance bounds estimated by our algorithm to the actual bounds obtained by matrix inversion. Note that all estimated bounds were verified to be consistent with the actual bounds by performing Cholesky decomposition on their difference to establish positive definiteness.

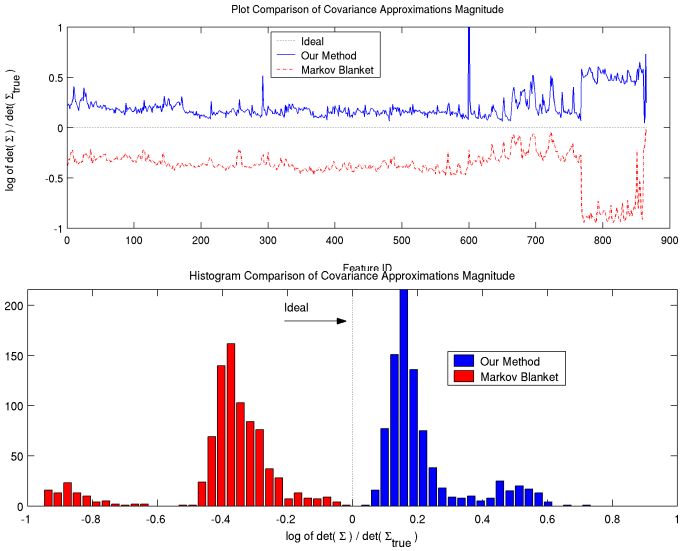


Fig. 3. This figure compares the Markov Blanket covariance approximation method and the one presented in this paper to the actual covariance obtained by inverting the information matrix; the results are computed for the information matrix shown in Fig. 2. For each state entry $\mathbf{x}_i$, we compute its relative pose to the robot $\mathbf{x}_r$ (i.e., $\mathbf{x}_{ri} = \ominus\mathbf{x}_r \oplus \mathbf{x}_i$) and associated first-order covariance for each of the methods. We then plot the log of the ratio of the determinant of the approximated covariance to the determinant of the actual covariance to facilitate comparison of conservativeness (positive values) / over-confidence (negative values). (top) Plot of the log ratio verses feature id for all $\mathbf{x}_i$. Note that a value of zero is ideal as this would indicate a ratio of one. (bottom) Same data as above but presented in histogram form. Both plots show that the method presented in this paper is conservative while the Markov Blanket method is over-confident.

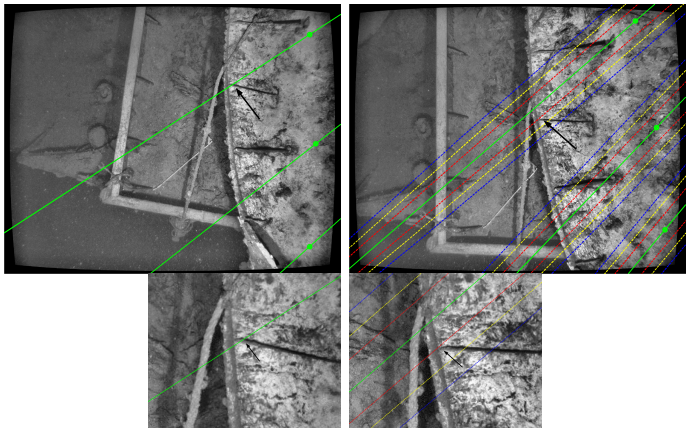
Fig. 3 provides a quantitative assessment comparing the bounds obtained by our algorithm to the bounds obtained by inverting only the Markov Blanket as proposed in [2], [16]. To provide a fair assessment, we choose to evaluate the *relative* uncertainty between the robot $\mathbf{x}_r$ and any other map element $\mathbf{x}_i$. Our justification for this metric is that the Markov Blanket method results in a conditional covariance which doesn't accurately reflect *global* map uncertainty but rather *relative* map uncertainty. Using the information matrix of Fig. 2, for each map element $\mathbf{x}_i$ we computed the first-order relative-pose covariance matrix between it and the robot. For our metric we chose to compute the log of the determinant of the approximation covariance to the determinant of the actual obtained by matrix inversion. Therefore, ratios greater than one (conservative) are positive and ratios less than one (over-confident) are negative. We note that Fig. 3 highlights that our method is conservative while the Markov Blanket is over-confident. Furthermore, for this dataset the histogram plot shows that our method tends to be conservative by a smaller margin than the Markov Blanket is over-confident.

Finally, Fig. 4 demonstrates the actual value of this conservative approximation within the context of pose constrained correspondence searches. Here we see two pairs of images and their predicted epipolar geometry based upon our state estimates. For a calibrated camera, the epipolar geometry is defined by the relative camera poses and defines a 1D search

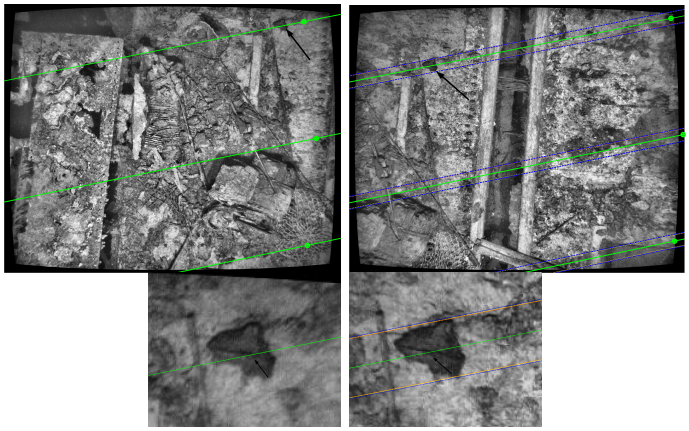
constraint. However, when our relative pose estimates are uncertain this 1D constraint becomes a search *region* [17]. Fig. 4(a) shows that the Markov Blanket approximation of the relative pose uncertainty is too over-confident for this image pair such that the 99.9% confidence search region does not contain the true correspondence causing image registration to fail. However, the true correspondence does lie within the search bounds associated with the actual and conservative approximation allowing image registration to succeed. Fig. 4(b) shows that for another image pair, the two methods produce equivalent results highlighting the unpredictability of the over-confidence in the Markov Blanket approximation.

V. Conclusion

In conclusion, we have presented a novel algorithm for extracting consistent covariance bounds useful for data association in SLAM information filters. We showed that our method provides a conservative approximation useful for real-world tasks such a image link hypothesis and constrained correspondence searches. The method's complexity scales asymptotically linear with map size as measured by solving for the robot's covariance-column coupled with fixed state size Kalman updates for re-observed map elements. Our results were presented within the context of an actual robotic mapping survey of the RMS Titanic embodying several challenging SLAM research tasks such as: large-area scalable mapping, 6 DOF, an unstructured underwater environment, and visual perception.



(a)



(b)

Fig. 4. This figure illustrates using the approximate covariance recovery technique presented in the paper for data association within the context of constraining image-based correspondence searches. (a) The top two images have spatial overlap and are candidates for image registration. The image on the left shows the predicted epipolar geometry in green for the camera pair and is instantiated based upon our state estimates. The image on the right shows the corresponding epipolar lines and their associated 99.9% confidence-bound search regions based upon the uncertainty in our state estimates. The different ellipse colors are Blue: conservative covariance recovery method presented in this paper, Yellow: actual covariance based upon inverting the information matrix, Red: Markov Blanket covariance recovery method. The bottom two images show "zoomed" segments along a corresponding epipolar line. Closer inspection reveals that the red ellipse search region does not contain the true correspondence area while the yellow and blue ellipses do. (b) A demonstration of the same correspondence test but for a different camera pair. Here we see that both covariance recovery methods yield nearly identical results to the actual covariance obtained by matrix inversion. This highlights the unpredictable level of over-confidence associated with the Markov Blanket approximation.

References

[1] L. Whitcomb, D. Yoerger, H. Singh, and J. Howland, "Advances in Underwater Robot Vehicles for Deep Ocean Exploration: Navigation, Control and Survery Operations," in *The Ninth International Symposium on Robotics Research*, Springer-Verlag, London, 2000, p. to appear.

[2] S. Thrun, Y. Liu, D. Koller, A. Ng, Z. Ghahramani, and H. Durrant-Whyte, "Simultaneous Localization and Mapping with Sparse Extended Information Filters," *International Journal of Robotics Research*, Accepted, To Appear.

[3] U. Frese and G. Hirzinger, "Simultaneous Localization and Mapping - a Discussion," in *Proceedings of the IJCAI Workshop Reasoning with Uncertainty in Robotics*, Seattle, WA, 2001, pp. 17–26.

[4] U. Frese, "A Proof for the Approximate Sparsity of Slam Information Matrices," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, Accepted, To Appear.

[5] M. Paskin, "Thin Junction Tree Filters for Simultaneous Localization and Mapping," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, San Francisco, CA, 2003, pp. 1157–1164.

[6] J. Neira and J. Tardos, "Data Association in Stochastic Mapping Using the Joint Compatibility Test," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 890–897, December 2001.

[7] U. Frese, "Treemap: An O(Log N) Algorithm for Simultaneous Localization and Mapping," in *Spatial Cognition IV*, C. Freksa, Ed. Springer Verlag, 2004.

[8] R. Eustice, H. Singh, and J. Leonard, "Exactly Sparse Delayed State Filters," in *Proceedings of the International Conference on Robotics and Automation*, Barcelona, Spain, Submitted, Under Review.

[9] Y. Bar-Shalom, X. Rong Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: John Wiley & Sons, Inc., 2001.

[10] R. Smith, M. Self, and P. Cheeseman, *Estimating Uncertain Spatial Relationships in Robotics*, ser. Autonomous Robot Vehicles. Springer-Verlag, 1990.

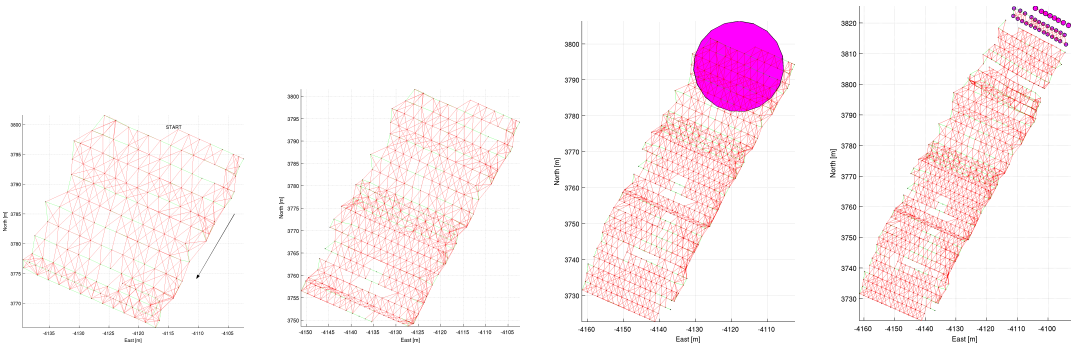
[11] J. Shewchuk, "An Introduction to the Conjugate Gradient Method without the Agonizing Pain," Carnegie Mellon Univerisity, Technical Report CMU-CS-94-125, August 1994.

[12] K. Konolige, "Large-Scale Map-Making," in *Proceedings of the AAAI*, San Jose, CA, 2004, pp. 457–463.

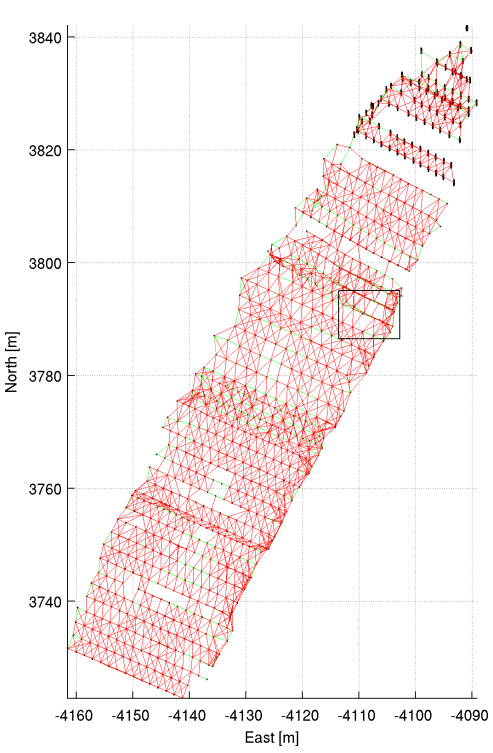
[13] T. Duckett, S. Marsland, and J. Shapiro, "Learning Globally Consistent Maps by Relaxation," in *IEEE International Conference on Robotics & Automation*, San Francisco, CA, April 2000, pp. 3841–3846.

[14] U. Frese, P. Larsson, and T. Duckett, "A Multigrid Algorithm for Simultaneous Localization and Mapping," *IEEE Transactions on Robotics*, Accepted, To Appear.

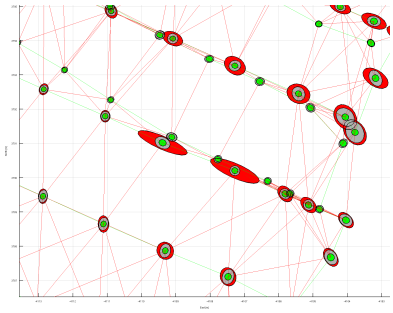
[15] M. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba, "A Solution to the Simultaneous Localization and Map Building (Slam) Problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, June 2001.



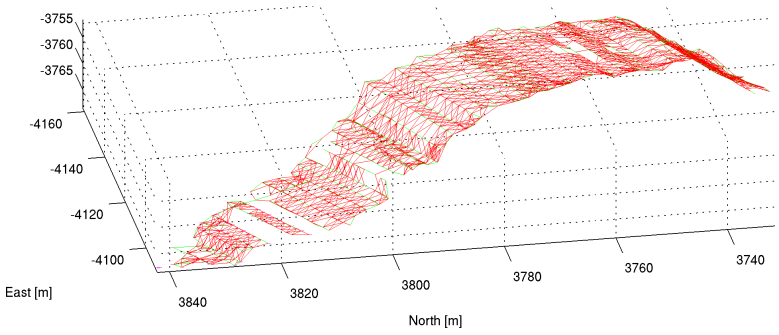
(a)



(b)



(c)



(d)

Fig. 5. This figure summarizes the results of our visually-based navigation of the RMS Titanic. (a) Time progression of our camera constraint network shown with 3-sigma bounds, from left to right: images 1–200, 1–400, 1–600, 1–800. Green links represent temporally consecutive registered image pairs while red links represent spatially registered image pairs. Note the large loop-closing event which occurred in third plot from left. (b) Final pose-constraint network associated with using 866 images to provide 3494 camera constraints, 3-sigma bounds are shown. (c) Inset of final result illustrating the consistency of the data association bounds generated using our algorithm. Note, 3-sigma bounds have been inflated by a factor of 30 for interpretation. Red: initial covariance bound associated with pose insertion into map, Gray: current estimate of marginal covariance bound based upon using a constant-time Kalman update per re-observation, Green: actual marginal covariance bound obtained by inverting the information matrix. (d) XYZ view of recovered pose-constraint network. Note that the recovered vehicle poses and correspondences can be used as direct inputs to a standard bundle adjustment step for structure recovery.

[16] Y. Liu and S. Thrun, "Results for Outdoor-Slam Using Sparse Extended Information Filters," in *IEEE International Conference on Robotics and Automation*, vol. 1, September 2003, pp. 1227–1233.

[17] R. Eustice, O. Pizarro, and H. Singh, "Visually Augmented Navigation in an Unstructured Environment Using a Delayed State History," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, vol. 1, New Orleans, USA, April 2004, pp. 25–32.

[18] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[19] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," in *Proceedings of the 4th Alvey Vision Conference*, Manchester, U.K., 1988, pp. 147–151.

[20] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[21] P. Rousseeuw and A. Leroy, *Robust Regression and Outlier Detection*. New York: John Wiley and Sons, 1987.