Visually Navigating the RMS Titanic with SLAM Information Filters

Ryan Eustice and Hanumant Singh
Woods Hole Oceanographic Institution
Woods Hole, MA, USA
{ryan,hanu}@whoi.edu

John Leonard and Matthew Walter
Massachusetts Institute of Technology
Cambridge, MA, USA
{jleonard,mwalter}@mit.edu

Robert Ballard
University of Rhode Island
Narragansett, RI, USA
ballard@gso.uri.edu

***Abstract*— This paper describes a vision-based, large-area, simultaneous localization and mapping (SLAM) algorithm that respects the low-overlap imagery constraints typical of underwater vehicles while exploiting the inertial sensor information that is routinely available on such platforms. We present a novel strategy for efficiently accessing and maintaining consistent covariance bounds within a SLAM information filter, thereby 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 path 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 habitats. 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 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 upon the travel time between the vehicle and transponders.

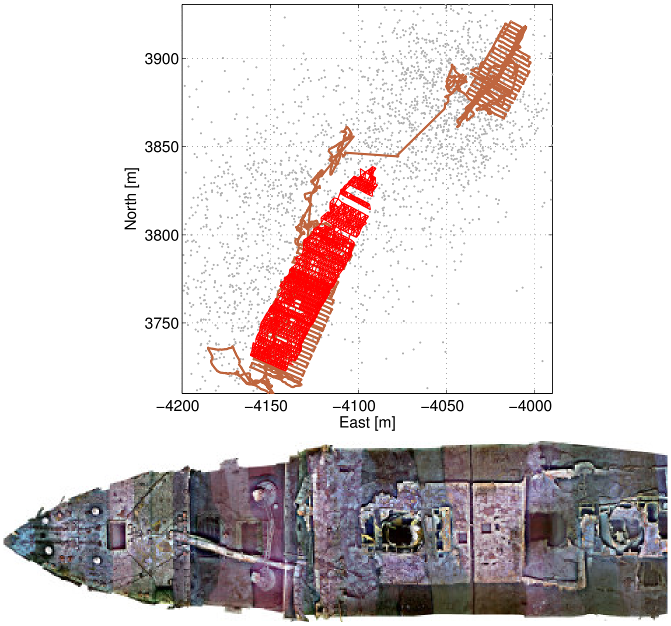


Fig. 1. Mapping results from a ROV survey (boustrophedon pattern) of the RMS Titanic conducted in the summer of 2004. (top) An XY plot comparing the raw dead-reckoned (DR) navigation data (brown), ship-board ultra-short baseline tracking (gray), and reconstructed survey trajectory from a vision-based 6-DOF SLAM information filter (red). Note that the discontinuity in the DR trajectory is the result of navigation sensor dropouts. (bottom) A photomosaic of the RMS Titanic constructed from over 700 digital still images. Note that this photomosaic 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 control points to guide the photomosaicking process. This could be considered as a form of benchmark against which fully autonomous processing can be compared.

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

Within the SLAM community, algorithms exploiting the sparse information representation for SLAM were first proposed by Thrun *et al.* [2], Frese [3], [4], Paskin [5], Eustice [6], and Dellaert [7]. 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 [8]. 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 (overconfident), then 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 (EIF) [2], [6], [7], which is the dual of the extended Kalman filter (EKF) [9]. 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, $\mathcal{N}\big(\boldsymbol{\xi}_t;\boldsymbol{\mu}_t,\Sigma_t\big)$, it is instead parametrized in terms of its information vector and information matrix, $\mathcal{N}^{-1}\big(\boldsymbol{\xi}_t;\boldsymbol{\eta}_t,\Lambda_t\big)$, where [9]

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

A. Constant-Time Measurement Updates

A well known and very attractive property of formulating SLAM in an EIF is that measurement updates are additive and efficient. This is in contrast to the quadratic complexity per update in the EKF. For example, assume the following general measurement function and its first-order linearized form:

$$\begin{aligned}\mathbf{z}_t &= \mathbf{h}(\boldsymbol{\xi}_t) + \mathbf{v}_t\\ &\approx \mathbf{h}(\bar{\boldsymbol{\mu}}_t) + \mathrm{H}(\boldsymbol{\xi}_t - \bar{\boldsymbol{\mu}}_t) + \mathbf{v}_t\end{aligned}$$

where $\boldsymbol{\xi}_t \sim \mathcal{N}\big(\bar{\boldsymbol{\mu}}_t,\bar{\Sigma}_t\big) \equiv \mathcal{N}^{-1}\big(\bar{\boldsymbol{\eta}}_t,\bar{\Lambda}_t\big)$ is the predicted state vector, $\mathbf{v}_t \sim \mathcal{N}\big(\mathbf{0},\mathrm{R}\big)$ is the white measurement noise, 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 [9]:

$$\begin{aligned}\mathrm{K} &= \bar{\Sigma}_t\mathrm{H}^\top\big(\mathrm{H}\bar{\Sigma}_t\mathrm{H}^\top + \mathrm{R}\big)^{-1}\\ \boldsymbol{\mu}_t &= \bar{\boldsymbol{\mu}}_t + \mathrm{K}\big(\mathbf{z}_t - \mathbf{h}(\bar{\boldsymbol{\mu}}_t)\big)\\ \Sigma_t &= \big(\mathrm{I}-\mathrm{KH}\big)\bar{\Sigma}_t\big(\mathrm{I}-\mathrm{KH}\big)^\top + \mathrm{KRK}^\top.\end{aligned} \tag{2}$$

This calculation non-trivially modifies all elements in the covariance matrix resulting in quadratic computational complexity *per* update. In contrast the corresponding EIF update is given by [2]:

$$\begin{aligned}\Lambda_t &= \bar{\Lambda}_t + \mathrm{H}^\top\mathrm{R}^{-1}\mathrm{H}\\ \boldsymbol{\eta}_t &= \bar{\boldsymbol{\eta}}_t + \mathrm{H}^\top\mathrm{R}^{-1}\big(\mathbf{z}_t - \mathbf{h}(\bar{\boldsymbol{\mu}}_t) + \mathrm{H}\bar{\boldsymbol{\mu}}_t\big).\end{aligned} \tag{3}$$

The above equation 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:

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

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., $\bar{\Lambda}_{x_ix_i}$, $\bar{\Lambda}_{x_jx_j}$, and $\bar{\Lambda}_{x_ix_j} = \bar{\Lambda}_{x_jx_i}^\top$). 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 filtered feature-based SLAM information matrix empirically obeys a "close-to-sparse" structure when properly normalized. This observation spawned 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 [4]. These algorithms approximate the SLAM posterior by (effectively) eliminating "small" elements in the corresponding information matrix. The elimination of weak constraints results in a sparse representation allowing the development of efficient filter algorithms that exploit the resulting sparse architecture. This empirical observation of weak inter-landmark constraints has recently been given a solid theoretical foundation by Frese [10] where he mathematically shows that inter-landmark information decays spatially at an exponential rate. This adds some justification for the sparseness approximations utilized in the feature-based SLAM information algorithms mentioned above.

In addition to feature-based techniques, a recent paper by Eustice *et al.* [6] shows that for a view-based SLAM representation the information matrix is *exactly* sparse without any approximation. The implication is that view-based SLAM

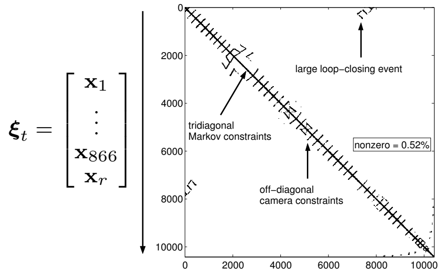


Fig. 2. This figure highlights the *exact* sparsity of the view-based SLAM information matrix using data from a recent ROV survey of the wreck of the RMS Titanic. In all there are 867 robot states where each state is a 12-vector consisting of 6 pose and 6 kinematic components. The resulting information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

systems can take advantage of the sparse information parameterization without incurring any sparse-approximation error. Based upon this insight, we've implemented a view-based SLAM system for our undersea application built around fusing 6-DOF relative-pose camera measurements (using monocular overlapping seafloor imagery) with traditional underwater vehicle DR navigation sensors. As an example, Fig. 2 illustrates the resulting information matrix 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. The wreck was surveyed from midship to stern and then from midship to bow resulting in a large loop-closing event, which is evident from the far off-diagonal elements pointed out in Fig. 2.

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 shown in (1).

1) Recovering the mean: Naïve 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 following sparse, symmetric, positive-definite, linear system of equations:

$$\Lambda_t \boldsymbol{\mu}_t = \boldsymbol{\eta}_t. \tag{4}$$

Such systems can be iteratively solved via the classic 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 multilevel relaxation 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 4.5 MB of double precision storage for the nonzero elements while its inverse requires over 865 MB.

Fortunately, recovering the entire covariance matrix usually is not necessary for SLAM as many of the data association and robotic planning decisions often 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). For example, marginalization is easy in the covariance form since it corresponds to extracting the appropriate sub-block from the covariance matrix while in the 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 the 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 the state vector resulting in cubic complexity due to matrix inversion in the Schur complement.

To get around this dilemma, Thrun *et al.* proposed 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 the sub-block over desired variables), their strategy is to choose an appropriate sub-block from the information matrix such that its inverse approximates the actual covariance for the subset of

TABLE I

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

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

	MARGINALIZATION	CONDITIONING
	$p(\boldsymbol{\alpha}) = \int p(\boldsymbol{\alpha},\boldsymbol{\beta})d\boldsymbol{\beta}$	$p(\boldsymbol{\alpha}\|\boldsymbol{\beta}) = p(\boldsymbol{\alpha},\boldsymbol{\beta})/p(\boldsymbol{\beta})$
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	$\boldsymbol{\eta} = \boldsymbol{\eta}_\alpha - \Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\boldsymbol{\eta}_\beta$ $\Lambda = \Lambda_{\alpha\alpha} - \Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\Lambda_{\beta\alpha}$	$\boldsymbol{\eta}' = \boldsymbol{\eta}_\alpha - \Lambda_{\alpha\beta}\boldsymbol{\beta}$ $\Lambda' = \Lambda_{\alpha\alpha}$

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 inverts this sub-block to obtain an covariance matrix for $\mathbf{x}_i$ and $\mathbf{x}_j$ conditioned on all other variables that have an indirect influence. They note that empirical testing indicates that their approximation seems to work well in practice for their application [16] despite the fact that using conditional covariances should result in an overconfident 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 that 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 that 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 (4), 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 robot covariance-column is accessible as well. Note that by definition $\Lambda_t\Sigma_t = \mathrm{I}$, 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 as $\Lambda_t\Sigma_{*i} = \mathbf{e}_i$. To obtain the robot's covariance at any time step we simply augment our original linear system (4) to include an appropriate set of basis vectors, $\mathrm{E}_r = \{\mathbf{e}_r\}$, such that the solution to (5) provides access to our current state and robot covariance-column.

$$\Lambda_t \begin{bmatrix}\boldsymbol{\mu}_\mathbf{t} & \Sigma_{*r}\end{bmatrix} = \begin{bmatrix}\boldsymbol{\eta}_t & \mathrm{E}_r\end{bmatrix} \tag{5}$$

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 filter. What we present in the following section is a way of maintaining *covariance bounds* that are consistent with respect to the inverse of the information matrix. 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 (5) 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 [6], [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 $\Sigma_{ii} \equiv \Sigma_{rr}$ and can be obtained by solving (5). 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 [17], 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.[1]

To obtain the actual joint-covariance $\bar{\Sigma}_{joint}$ from the information matrix requires marginalizing out all other elements in our map except for $\mathbf{x}_r$ and $\mathbf{x}_i$ leading to cubic complexity in

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

the number of eliminated variables. However, we can obtain a bounded approximation to $\bar{\Sigma}_{joint}$ at any time-step by using the solution from (5) 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

$$\tilde{\bar{\Sigma}}_{joint} = \begin{bmatrix} \bar{\Sigma}_{rr} & \bar{\Sigma}_{ir}^\top \\ \bar{\Sigma}_{ir} & \tilde{\Sigma}_{ii} \end{bmatrix} \tag{6}$$

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 (6) represents a valid positive-semidefinite and, therefore, consistent approximation satisfying

$$\tilde{\bar{\Sigma}}_{joint} - \bar{\Sigma}_{joint} = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{\Sigma}_{ii} - \Sigma_{ii} \end{bmatrix} \geq 0, \tag{7}$$

since $\tilde{\Sigma}_{ii} - \Sigma_{ii} \geq 0$. Given that (6) provides a consistent approximation to the true covariance, we can use it to compute conservative first-order probabilities of relative-pose in the usual way, $\mathbf{x}_{ri} = \ominus\mathbf{x}_r \oplus \mathbf{x}_i$ [18], 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 (5) is efficient only when we have a good starting point [12], [13]. In other words, when we solve (5) for the latest state and robot covariance-column, our estimates $\boldsymbol{\mu}_t$ and Σ_{*r} from the 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_{*j}, 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 (6) and perform a simple constant-time Kalman filter update on a *per* re-observation basis. In other words, we only update the 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 (2) on the fixed-size state vector $\mathbf{y} = \left[\mathbf{x}_r^\top, \mathbf{x}_i^\top\right]^\top$ to 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

$$p(\mathbf{y}) = \int_{\mathbf{x}_j \neq \{\mathbf{x}_r, \mathbf{x}_i\}} \mathcal{N}^{-1}\big(\bar{\boldsymbol{\eta}}_t, \bar{\Lambda}_t\big) d\mathbf{x}_j = \int_{\mathbf{x}_j \neq \{\mathbf{x}_r, \mathbf{x}_i\}} \mathcal{N}\big(\bar{\boldsymbol{\mu}}_t, \bar{\Sigma}_t\big) d\mathbf{x}_j, \tag{8}$$

which results in the distribution

$$p(\mathbf{y}) = \mathcal{N}\Big(\begin{bmatrix} \bar{\boldsymbol{\mu}}_r \\ \bar{\boldsymbol{\mu}}_i \end{bmatrix}, \begin{bmatrix} \bar{\Sigma}_{rr} & \bar{\Sigma}_{ir}^\top \\ \bar{\Sigma}_{ir} & \Sigma_{ii} \end{bmatrix}\Big). \tag{9}$$

Noting that (6) already provides us with a consistent approximation to this distribution we have

$$\tilde{p}(\mathbf{y}) = \mathcal{N}\Big(\begin{bmatrix} \bar{\boldsymbol{\mu}}_r \\ \bar{\boldsymbol{\mu}}_i \end{bmatrix}, \begin{bmatrix} \bar{\Sigma}_{rr} & \bar{\Sigma}_{ir}^\top \\ \bar{\Sigma}_{ir} & \tilde{\Sigma}_{ii} \end{bmatrix}\Big) \tag{10}$$

where the only difference between the actual distribution (9) and the approximation (10) is the conservative marginal, $\tilde{\Sigma}_{ii}$. Using the measurement, $\mathbf{z}_{ri}$, we now perform a constant-time Kalman update (2) on (10) 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 (8) 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: $\Sigma_{*r}$ {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 $\tilde{\Sigma}_{ii} \leftarrow \tilde{\Sigma}_{ii}^+$
      end if
    end if
  end for
```

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

IV. Results

This section presents experimental results validating our covariance recovery strategy from the information matrix 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

TABLE II
POSE SENSOR CHARACTERISTICS.

Measurement	Sensor	Precision
Roll/Pitch	Tilt Sensor	±0.1°
Heading	North-Seeking FOG	±0.1°
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

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. However, note that the results being presented here were produced using imagery from *one* camera only — the purpose of this self-imposed restriction to a monocular sequence is to demonstrate the general applicability of our visually augmented navigation strategy.

Fig. 5 summarizes our mapping results using an exactly sparse view-based SLAM information filter as proposed by [6]. 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 $3100\ \mathrm{m}^2$ 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 [19] founded on:

- Extracting a combination of both Harris [20] and SIFT [21] 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 [22] 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 [19].

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 re-localized 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.

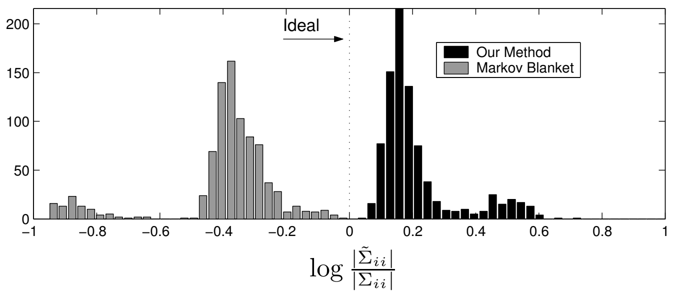
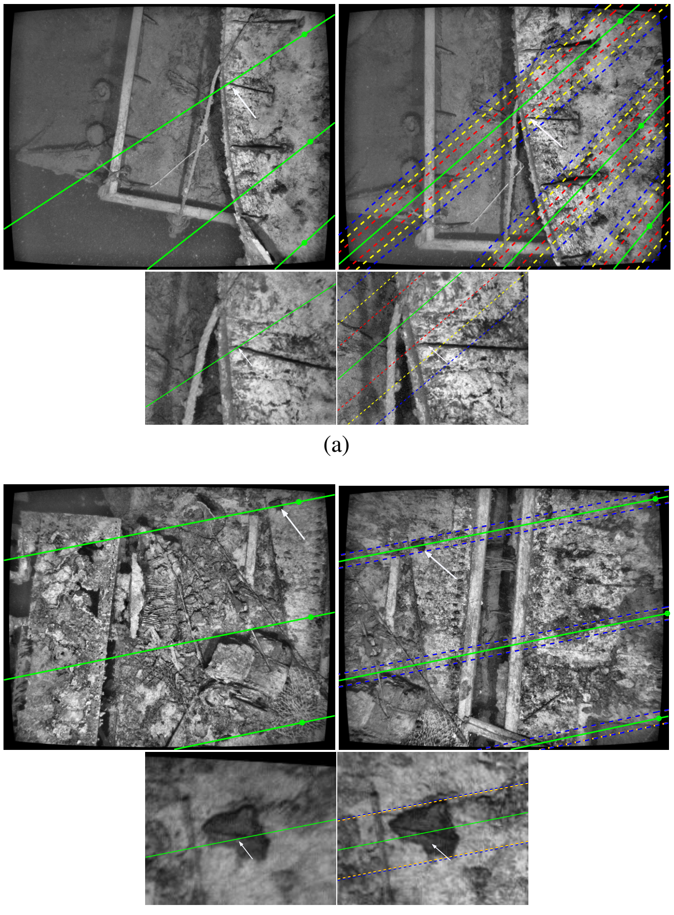


Fig. 3. This figure compares the Markov Blanket covariance approximation method and the one presented in this paper to the actual covariance obtained by inverting the information matrix; the results are computed for the information matrix shown in Fig. 2. For each method we compute the relative-pose of each state entry, $\mathbf{x}_i$, to the robot (i.e., $\mathbf{x}_{ri} = \ominus\mathbf{x}_r \oplus \mathbf{x}_i$) and associated first-order covariance. We then plot as a histogram 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) versus overconfidence (negative values) (a value of zero is ideal as this would indicate a ratio of one). Note that the Markov Blanket method is overconfident while ours is conservative.

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 that does not 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 (overconfident) are negative. We note that Fig. 3 highlights that our method is conservative while the Markov Blanket is overconfident. Furthermore, for this dataset the histogram plot shows that our method tends to be conservative by a smaller margin than the Markov Blanket is overconfident.

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 [19]. However, when the 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 overconfident 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



(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 (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 search bands correspond to the conservative covariance recovery method presented in this paper (blue), the actual covariance based upon inverting the information matrix (yellow), and the Markov Blanket covariance recovery method (red). The bottom two images show "zoomed" views. Closer inspection reveals that the red search region does not contain the true correspondence area while the yellow and blue regions 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 overconfidence associated with the Markov Blanket approximation.

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 overconfidence in the Markov Blanket approximation.

V. Conclusion

In conclusion, we have presented a novel algorithm for efficiently 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 constant-time 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 including large-area scalable mapping, 6-DOF vehicle motion, 3D underwater environments, and visual perception.

VI. Acknowledgements

The lead author wishes to thank Seth Teller for his motivating discussions regarding the pursuit of a large-area, scalable estimation framework. This work was funded in part by the CenSSIS ERC of the NSF under grant EEC-9986821. This paper is WHOI contribution number 11391.

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 *Proc. Intl. Symp. Robotics Research*, Springer-Verlag, London, 2000.

[2] S. Thrun, Y. Liu, D. Koller, A. Ng, Z. Ghahramani, and H. Durrant-Whyte, "Simultaneous localization and mapping with sparse extended information filters," *Intl. J. Robotics Research*, vol. 23, no. 7-8, pp. 693–716, July-Aug. 2004.

[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, "Treemap: An O(Log N) algorithm for simultaneous localization and mapping," in *Spatial Cognition IV*, C. Freksa, Ed. Springer Verlag, 2004.

[5] M. Paskin, "Thin junction tree filters for simultaneous localization and mapping," in *Intl. Joint Conf. on A.I.*, San Francisco, CA, 2003, pp. 1157–1164.

[6] R. Eustice, H. Singh, and J. Leonard, "Exactly sparse delayed-state filters," in *Proc. IEEE Intl. Conf. Robotics & Auto.*, Barcelona, Spain, 2005, pp. 2428–2435.

[7] F. Dellaert, "Square root SAM," in *Proc. Robotics Science & Systems*, 2005.

[8] J. Neira and J. Tardos, "Data association in stochastic mapping using the joint compatibility test," *IEEE Trans. Robotics and Automation*, vol. 17, no. 6, pp. 890–897, Dec. 2001.

[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] U. Frese, "A proof for the approximate sparsity of SLAM information matrices," in *Proc. IEEE Intl. Conf. Robotics & Auto.*, Barcelona, Spain, 2005, pp. 331–337.

[11] J. Shewchuk, "An introduction to the conjugate gradient method without the agonizing pain," Carnegie Mellon University, Technical Report CMU-CS-94-125, Aug. 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 *Proc. IEEE Intl. Conf. Robotics & Auto.*, San Francisco, CA, Apr. 2000, pp. 3841–3846.

[14] U. Frese, P. Larsson, and T. Duckett, "A multilevel relaxation algorithm for simultaneous localisation and mapping," *IEEE Trans. Robotics*, vol. 21, no. 2, pp. 1–12, 2005.

[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 Trans. Robotics and Automation*, vol. 17, no. 3, pp. 229–241, June 2001.

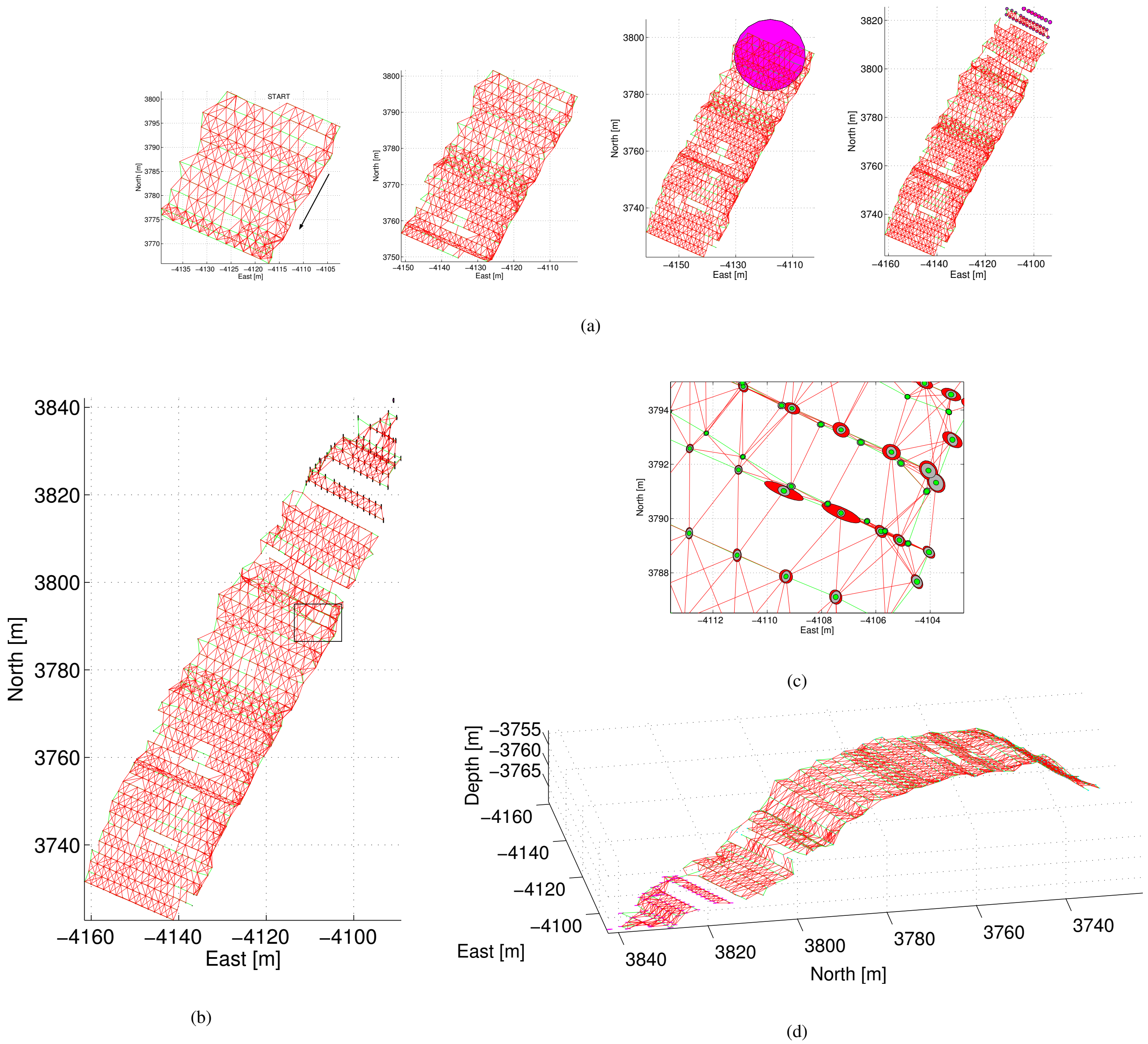


Fig. 5. This figure summarizes the results of our visually-based navigation of the RMS Titanic. (a) The time progression of our pose-constraint network shown with 3-sigma bounds, from left to right: images 1–200, 1–400, 1–600, 1–800. Green links represent temporally consecutive registered image pairs while red links represent spatially registered image pairs. Note the large loop-closing event that occurred in the third plot from left. (b) The final pose-constraint network associated with using 866 images to provide 3494 camera constraints, 3-sigma bounds are shown. (c) An inset of the 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. Shown are the initial covariance bounds associated with pose insertion into the map (red), the current estimate of marginal covariance bounds based upon using constant-time Kalman updates (gray), and the actual marginal covariance bounds obtained by inverting the information matrix (green). (d) An XYZ view of the recovered pose-constraint network. Note that the recovered vehicle poses and image correspondences can be used as direct inputs into a standard bundle adjustment step for structure recovery.

[16] Y. Liu and S. Thrun, “Results for outdoor-SLAM using sparse extended information filters,” in *Proc. IEEE Intl. Conf. Robotics & Auto.*, vol. 1, Sept. 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 *Proc. IEEE Intl. Conf. Robotics & Auto.*, vol. 1, New Orleans, USA, Apr. 2004, pp. 25–32.

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

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

[20] 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.

[21] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *Intl. J. Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

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