Map Building Fusing Acoustic and Visual Information using Autonomous Underwater Vehicles

Clayton Kunz
Department of Applied Ocean Physics and Engineering
Woods Hole Oceanographic Institution
Woods Hole, MA 02543
ckunz@whoi.edu

Hanumant Singh
Department of Applied Ocean Physics and Engineering
Woods Hole Oceanographic Institution
Woods Hole, MA 02543
hsingh@whoi.edu

Abstract

We present a system for automatically building 3-D maps of underwater terrain fusing visual data from a single camera with range data from multibeam sonar. The six-degree of freedom location of the camera relative to the navigation frame is derived as part of the mapping process, as are the attitude offsets of the multibeam head and the on-board velocity sensor. The system uses pose graph optimization and the square root information smoothing and mapping framework to simultaneously solve for the robot's trajectory, the map, and the camera location in the robot's frame. Matched visual features are treated within the pose graph as images of 3-D landmarks, while multibeam bathymetry submap matches are used to impose relative pose constraints linking robot poses from distinct tracklines of the dive trajectory. The navigation and mapping system presented works under a variety of deployment scenarios, on robots with diverse sensor suites. Results of using the system to map the structure and appearance of a section of coral reef are presented using data acquired by the Seabed autonomous underwater vehicle.

1 Introduction

Autonomous underwater vehicles (AUVs) have seen increasing use in recent years in a number of scientific and industrial applications. They are generally used to measure physical [Dhanak et al., 2001], chemical [Camilli et al., 2010] or acoustic [Sastre-Córdova, 2009] properties of the water through which they move, to build bathymetric maps of the sea floor using acoustic ranging sensors [Yoerger et al., 1999] such as side-scan and multibeam sonar, and to build acoustic and optical images of the sea floor or other targets of interest [Foley et al., 2009] [Eustice et al., 2006] [Clarke et al., 2009]. While in the past mapping AUVs have typically been used to build wide area bathymetric maps from high altitude (on the order of 50 meters), or dense mosaics of photographs taken from low-altitude (on the order of 5 meters), recently there has been more interest in building high-resolution 3-D maps containing visual texture information, either using stereo imagery [Johnson-Roberson et al., 2010], structured lighting [Roman et al., 2010], or "microbathymetric" multibeam sonar [Roman and Singh, 2007]. Regardless of the mapping modality, the desire is to build a self-consistent map, geo-referenced if possible, and the primary problem is that the navigation accuracy of the vehicle is much worse than the resolution of the mapping sensors being used.

In this paper, we focus on the problem of constructing centimeter-scale 3-D bathymetric maps which combine single-camera imagery with multibeam sonar acquired from a low-altitude AUV. We address the difficulties in localization and sensor calibration using the abstraction of a pose graph, which captures the relationships between the estimated trajectory of the robot as it moves through the water and the measurements made by the navigation and mapping sensors in a flexible sparse graphical framework, enabling quick optimization over the trajectory and the map. This solution falls under the broad category of "full" simultaneous localization and mapping (SLAM) [Thrun et al., 2005], because the full robot trajectory is optimized over rather than just the most recent pose. The optimization itself is performed by the square root information Smoothing and Mapping ($\sqrt{\text{SAM}}$) algorithm [Dellaert and Kaess, 2006] which has a publicly available implementation [Kaess et al., 2011]. Our framework is notable in that it enables the calibration of the navigation sensors and the extrinsic camera position and orientation through the use of specialized nodes in the pose graph.

The techniques presented here are related to earlier work in the fields of multibeam bathymetric mapping, visual SLAM, and bundle adjustment, synthesizing these disparate fields into a single consistent framework. In the field of bathymetric range SLAM, the algorithm in [Roman and Singh, 2007] uses the idea of fixed submaps which are registered using the iterative closest point (ICP) algoritihm [Zhang, 1994] to impose relative pose constraints in an delayed-state extended Kalman filter. Our approach reformulates the idea into the pose graph framework, but relies on the same assumption that local improvements in a map will imply a global improvement. We also use a simplified model for submap matching which does not rely on the full ICP algorithm. More distantly related to the approach described here are the efforts in [Barkby et al., 2009] and [Fairfield et al., 2007], both of which rely on particle filters and use an occupancy grid representation to directly optimize over the consistency of the generated terrain map, and the method described in [Walter et al., 2008] which uses a sparse extended information filter. Using the multibeam as a source of relative pose constraints in the optimization, rather than explicitly modeling the multibeam measurements in the pose graph, allows us to use the framework for all constraints – visual, acoustic, and proprioceptive – in a straightforward manner. Moreover, our unified approach allows the resolution of the generated maps to approach the advertised resolution (2 cm at 10 meters range) of our 245 kHz multibeam sensor [Imagenex DeltaT, 2012], exceeding the current common practice of gridding on the order of 50 cm ([Yoerger et al., 2007], [Williams, 2012], [Roman and Singh, 2007]) by an order of magnitude.

From the underwater visual mapping literature, [Kim and Eustice, 2009] use matched features from image pairs to imply five degree of freedom (DOF) relative pose constraints via the essential matrix [Longuet-Higgins, 1981], and then incorporate these constraints into a sparse extended information matrix SLAM system. These kinds of constraints can be incorporated into a pose graph in the same way that multibeam submap matches are, although the constraints must be free of scale. Instead, we use matched features as images of landmarks, and incorporate the observations of these landmarks into the graph, which allows us to use pixel reprojection error directly in the optimization. This is similar to the second phase of the work described in [Pizarro, 2004] and the structure from motion (SFM) approach described in [Nicosevici et al., 2009] and [Escartín et al., 2008], which minimize pixel reprojection error when optimizing for global scene structure and camera motion, after first using the same essential matrix-based scale-free relative pose constraints to fix the navigation. The global optimization undertaken in those approaches is similar to what we describe here, though none explicitly mention the pose graph abstraction, which we rely on to incorporate multibeam-induced constraints and to solve for the extrinsic camera location relative to the robot's navigation frame.

When stereo cameras are available, a pair of images can be used to recover 3-D structure directly. There is a rich and continuing history of stereo vision underwater, both for mapping and for measurement (see e.g. [Li et al., 1997], [Butler et al., 2002], and [Negahdaripour and Madjidi, 2003]). Determining the relationship between local scene strucuture measured by independent stereo views yields constraints that can be directly inserted into a SLAM framework. This is the approach used by [Beall et al., 2010], in a vision-only pose graph formulation which benefits from the high frame-to-frame overlap provided by video cameras. More recently, [Beall et al., 2011] extends this work by removing the need for video-rate imagery and adding vehicle odometry information when visual feature matching fails, but still relies on stereo cameras. The alternative approach in [Johnson-Roberson et al., 2010] integrates meshes produced by independent stereo views into a global visualization, and relies on a stereo SLAM solution (based on a sparse extended information filter similar in structure to the pose graph formulation [Mahon et al., 2008]) to seed the mesh integration. Perhaps closest to our work is that described in [Johnson-Roberson et al., 2009] and in [Singh et al., 2002], both of which integrate independent 3-D maps built using multibeam and stereo vision or SFM for visualization rather than map-building. Our approach, by contrast, does not start with independent 3-D maps, but rather builds one map by fusing information from both sensor modalities in a single optimization.

Finally, there has been recent work on directly coupling a single camera with a multibeam sensor on an AUV. The authors in [Hurtós et al., 2010] describe a method for calibrating a pair of such sensors on an AUV using a target in a test tank. Given the limitations of working in the field and the lack of a suitably large tank, the approach presented in this paper does not rely on an explicit relative sensor calibration, but instead derives the relationship as a side-effect of the smoothing and mapping algorithm. Further examples of tight integration between cameras and acoustic sensors are presented in [Negahdaripour, 2007] and [Negahdaripour et al., 2009], which are concerned with local area imaging, rather than with the creation of large area maps.

Out of the water, there has been a great deal of work in single-camera mapping, both with and without additional navigation information. The work described by [Davison et al., 2007] largely solves the single-camera video-rate SLAM problem using an EKF for indoor scenes; this work is extended in [Civera et al., 2008] to work outdoors, but large frame-to-frame overlap is still required, which can be provided either by a video camera, or by the massive number of photographs of landmarks posted online. In particular, the approach works best when multiple views of the same scene area are acquired from quite distinct points of view, and individual scene points are tracked over multiple video frames. In the deep ocean onboard an AUV, it is normal procedure to maintain constant altitude over a scene and to minimize changes in vehicle roll and pitch, reducing the number of possible vantage points from which a scene can be viewed. Frame-to-frame overlap of less than 50% is also typical, because of limitations in on-board power (and hence lighting), so scene points are usually not matchable across more than two images. Our system only uses pairwise image matches – even in loop closure scenarios, observations are represented in the pose graph by pairwise constraints.

To summarize, the system presented here stands out by

- Treating visual and range mapping measurements in a unified framework,
- Working with a single camera with low frame-to-frame overlap
- Optimizing over the extrinsic camera calibration
- Gridding the bathymetry down to 5 centimeters, or about 1.5% of range.

Our system starts with a graph capturing the robot's trajectory as measured by the onboard navigation sensors, and then adds factor nodes representing visual feature observations and constraints induced by multibeam submap matches, in an incremental fashion. Visual features and the camera extrinsic location are added first, then multibeam constraints are added, so that the final optimization includes data from both mapping sensors and all navigation sensors. This paper proceeds through these steps, starting with a quick review of traditional AUV navigation and the pose graph formulation of the AUV navigation problem, and then stepping through the changes made to the pose graph to incorporate information from the mapping sensors. Field results are described next using data from a dense survey of a 30 meter by 30 meter region of coral reef collected by the Seabed AUV [Singh et al., 2004] (see figure 1), followed by a discussion of future research directions.

2 AUV navigation

Most underwater vehicles carry a diverse suite of navigation sensors which together provide a redundant and often conflicting set of pose estimates. Underwater, a body can move through six degrees of freedom, and on most vehicles four of these six degrees of freedom are directly measurable: depth, roll, pitch, and heading. This is the case on manned submersibles and tethered remotely operated vehicles as well as on free-swimming AUVs. Onboard pose estimators often simply accept measured values as truth, or minimally filter measured values to smooth out noisy signals. Underwater localization is then reduced to estimating the remaining two degrees of freedom corresponding to horizontal position.

We are interested in the deployment scenario requiring the least amount of external infrastructure, in which an underwater vehicle is equipped with a velocity sensor. On any vehicle expected to work within a few hundred meters of the seafloor, the typical sensor will be a doppler velocity log (DVL), which measures 3-D velocity relative to the terrain. These sensors only work within range of the terrain, however, which is dependent on their operating frequency. For 1200 kHz systems, for example, the working range is about 40 meters. Once a 3-D velocity estimate has been determined, a pose estimate can be derived by dead

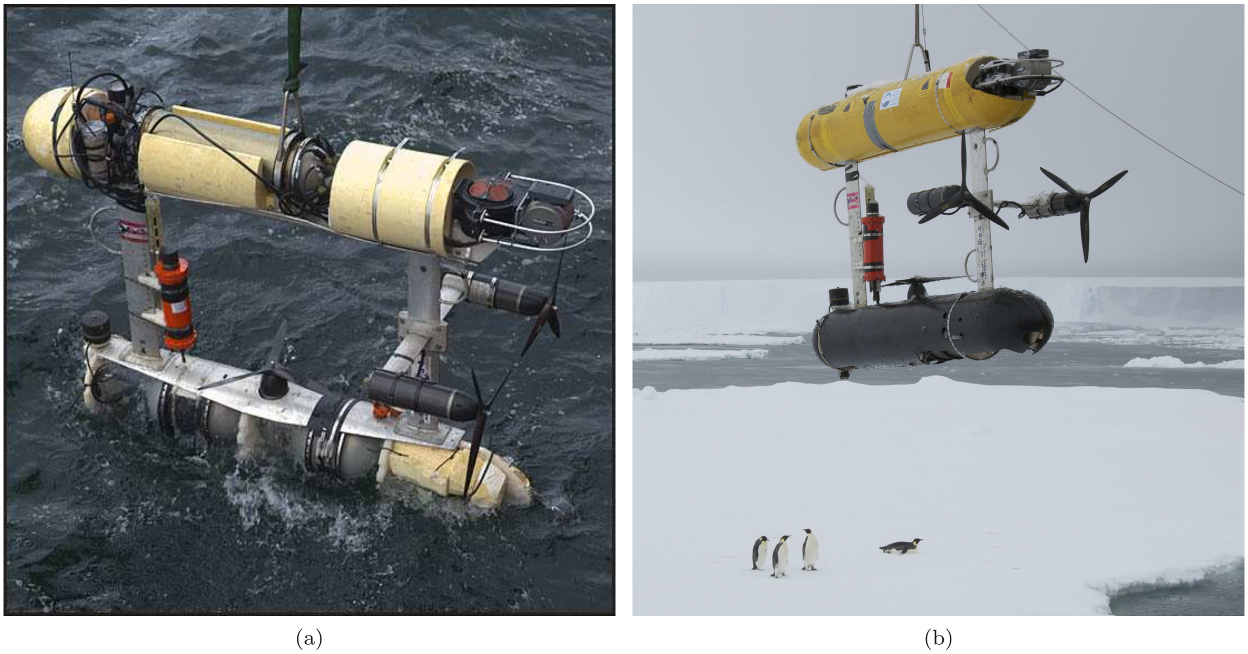
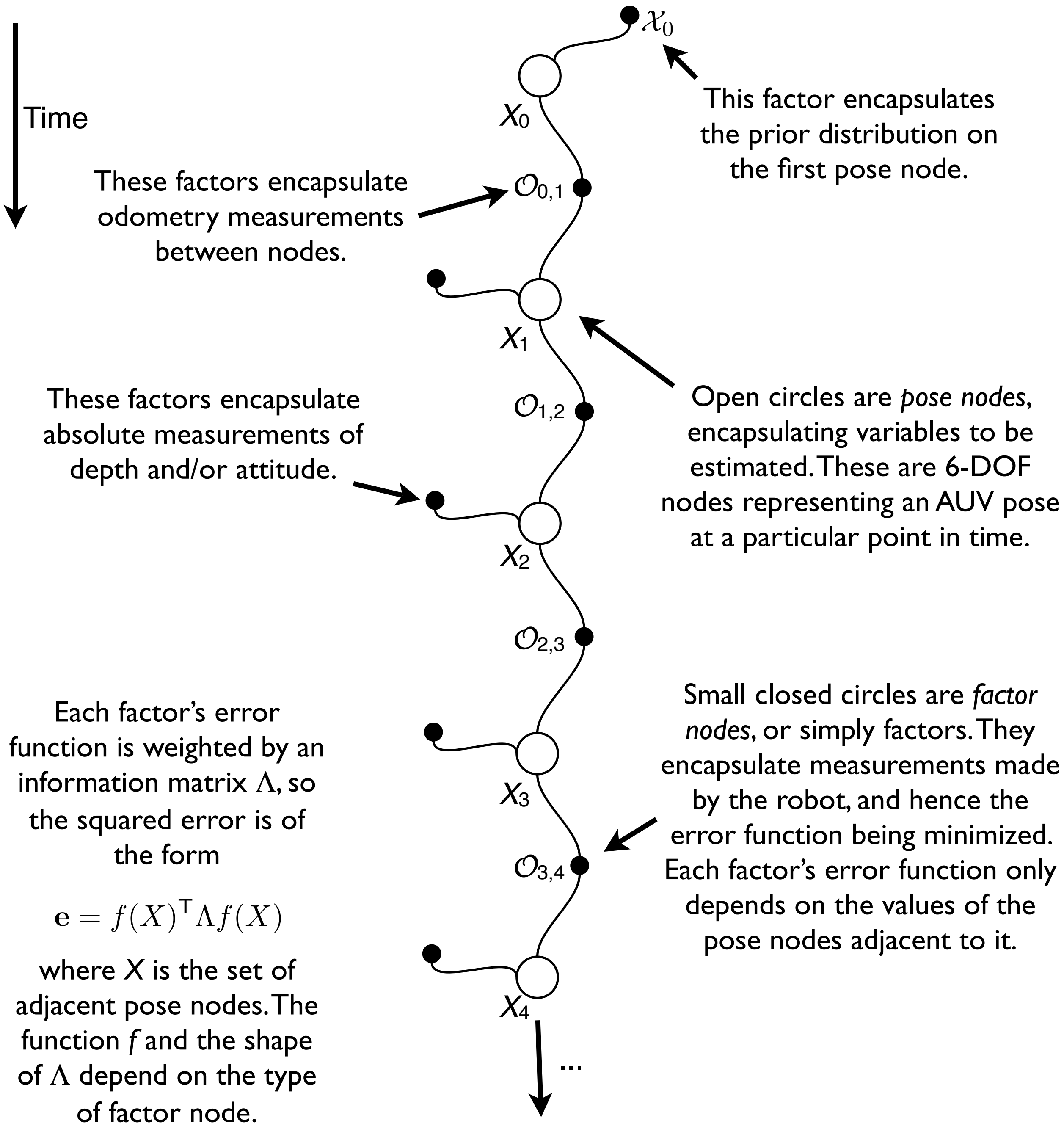


Figure 1: The Seabed AUV (a) without its fairings, in under-ice configuration. The upper hull contains the control computer, multibeam sonar, doppler velocity log, hyperspectral radiometer, avalanche beacon, and camera. The lower hull contains the vertical thruster, batteries, fibre optic gyro, depth sensor, acoustic modem transducer, fluorometer, and emergency drop weight. A conductivity-temperature-depth sensor and ultra-short baseline transponder are mounted to the forward strut, while the aft strut carries a pair of thrusters that act as a differential drive. With fairings mounted (b), the AUV is ready for deployment under ice.

reckoning: integrating the velocity estimate using the vehicle's attitude sensor to determine the direction of motion in the geo-referenced frame. The pose estimate will be relative to an arbitrary origin, but can be used for relative navigation. For navigation in an Earth-fixed frame, either velocity estimates must be available on the surface, where GPS can anchor the trajectory, or additional infrastructure is required (e.g. using acoustic localization systems). There has been recent work using acoustic doppler current profiler measurements to estimate vehicle drift between the surface and the seafloor [Medagoda et al., 2011]; in this paper, however, we will focus on navigation relative to an arbitrary origin, rather than to an Earth-fixed coordinate system.

2.1 Pose graphs

The use of pose graphs in the robotics literature has greatly increased in recent years as efficient optimizers have become available. We present a brief introduction here, and describe how they can be used for AUV dead reckoning, which will serve as the basis for integrating information from the mapping sensors. Pose graphs are abstract representations of the variables and constraints in an optimization problem which can be easily translated into an efficient implementation. When used to represent a robot's trajectory and set of observations and constraints, a pose graph is usually characterized by a sparse connectivity structure. The pose graph consists of two types of nodes, *pose nodes*, and *factors*, which are connected in a bipartite fashion. The pose nodes represent variables to be estimated, while the factors represent constraints on the variables in the pose nodes to which they are connected. In other words, the pose nodes as a group encapsulate the trajectory, map, and other parameters, and the factors as a group encapsulate the robot's measurements and other indirectly measured constraints. The graph represents an error function capturing the difference between what is measured and what is predicted by a given trajectory and map; each factor therefore encapsulates a small part of the total error function, and carries with it a measurement of some kind and a matrix which weighs the contribution of the factor in the overall error function relative to the others. This weight matrix is called an information matrix, because in the linear case if it is set equal to the inverse of the measurement covariance matrix, then the solution with the least overall squared error will be the best linear unbiased estimator of the parameters. The structure of the pose graph directly mirrors the sparsity structure of the error Jacobian function, as changes in variables only affect the error terms of factors to which the pose nodes representing the variables are connected. See [Dellaert and Kaess, 2006] for an excellent introduction, including a description of how the sparsity of the problem and the structure of SLAM problems in general lead to efficient solutions. An example pose graph for AUV odometry integration is shown in figure 2.



This factor encapsulates the prior distribution on the first pose node.

These factors encapsulate odometry measurements between nodes.

Open circles are *pose nodes*, encapsulating variables to be estimated. These are 6-DOF nodes representing an AUV pose at a particular point in time.

These factors encapsulate absolute measurements of depth and/or attitude.

Small closed circles are *factor nodes*, or simply factors. They encapsulate measurements made by the robot, and hence the error function being minimized. Each factor's error function only depends on the values of the pose nodes adjacent to it.

Each factor's error function is weighted by an information matrix Λ, so the squared error is of the form

$$\mathbf{e} = f(X)^\mathsf{T} \Lambda f(X)$$

where *X* is the set of adjacent pose nodes. The function *f* and the shape of Λ depend on the type of factor node.

Figure 2: The pose graph for integrating velocity measurements and absolute depth and attitude measurements into an AUV trajectory estimate. The trajectory is "anchored" by the first factor node, which encapsulates the prior estimate of the starting AUV position.

2.2 Notation

The six degree of freedom AUV pose at time t is designated X_t and is represented in the graph by a pose node, also designated X_t. Factors are represented by uppercase script letters, so an odometry factor $\mathcal{O}_{t,t+1}$ will thus relate pose nodes X_t and X_{t+1}. An AUV pose X_t comprises six scalars $(x, y, z, \rho, \phi, \theta)$ where (x, y, z) represents meters north, east, and down (toward the center of the Earth) from an arbitrarily defined local origin (at sealevel). The AUV's local frame is defined to have the x axis pointing forward, y axis pointing starboard, and z axis pointing downward, so that (ρ, ϕ, θ) represent roll (rotation about the x axis), pitch (rotation about the y axis), and heading (rotation about the z axis). These frame orientations keep everything right-handed, with positive values for z underwater, and imply that heading angles correspond to what one would expect to read on a true north seeking compass.

In figure 2 the odometry factor $\mathcal{O}_{t,t+1}$ linking poses X_t and X_{t+1} carries with it the 6-DOF measurement of relative motion $O_{t,t+1}$, such that the error will be zero if $X_{t+1} = X_t \oplus O_{t,t+1}$, where the $\oplus$ symbol designates the *compounding* operation from [Smith et al., 1990]. These poses and odometry factors are interchangeable with 4×4 rigid transformation matrices $X_t \leftrightarrow \mathbf{X}_t$ and $\mathcal{O}_{t,t+1} \leftrightarrow \mathbf{O}_{t,t+1}$ so that $\mathbf{X}_t$ transforms points from the AUV's local reference frame at time step t to the world frame, and $\mathbf{O}_{t,t+1}$ transforms points from the AUV's reference frame at time $t + 1$ to the AUV's reference frame at time t.

The error function for an odometry node is then derived from the difference between the measured odometry $O_{t,t+1}$ and the odometry implied by the relative transformation between X_t and X_{t+1} for a given set of values for these pose nodes. The error can be expressed compactly as $(X_{t+1} \ominus X_t) - O_{t,t+1}$, where $A \ominus B$ is defined as $B^{-1} \oplus A$, and the inversion can be derived from the matrix representation of $\mathbf{X}_t$. This error vector has six elements, corresponding to the six degrees of freedom for the relative pose, so three of the terms capture angle differences and must be normalized to lie between $-\pi$ and π. This amounts to comparing 3-D rotations by subtracting Euler angles, which is only valid for small differences; fortunately for odometry factors the angles involved are very small. Other formulations for comparing 3-D rotations include using the axis of a normalized quaternion as in [Kümmerle et al., 2011], or using the Rodrigues vector (axis of rotation multiplied by rotation magnitude), equivalent to members of the Lie algebra $\mathfrak{se}(3)$ as in [Strasdat et al., 2010]. Throughout this paper, we assume small angles and simply use angular differences, which has been well justified by several data sets collected by the AUV.

The goal of the $\sqrt{\text{SAM}}$ algorithm is to find values for all the variables in the pose nodes, such that the overall error function

$$\sum \mathbf{e}_{\mathcal{F}}^{\mathsf{T}} \Lambda_{\mathcal{F}} \mathbf{e}_{\mathcal{F}} \tag{1}$$

is minimized, where the sum is over all factors in the graph, and $\Lambda_{\mathcal{F}}$ is the information matrix for factor $\mathcal{F}$. This is a weighted nonlinear least squares problem, because the error term $\mathbf{e}_{\mathcal{F}}$ for each factor is in general the result of a nonlinear function, particularly because of the conversion from rotation matrices to Euler angles. The optimization can be solved with a good initialization point by code publicly available [Kaess et al., 2011] from the authors of [Kaess et al., 2008]. The overall translation ambiguity of the problem is addressed by either forcing the first pose to lie at the origin, or by "anchoring" it with a *prior factor* $\mathcal{X}_0$, as shown in the figure, which has error function $\mathbf{e} = X_0 - \mathcal{X}_0$, so that the error is zero if the pose X_0 matches the "measurement" contained in the factor.

2.3 AUV pose graph navigation

In the absence of other constraints or absolute measurements, a solution to dead-reckoning integration can be obtained by "walking down the chain," compounding the previous pose by the odometry measurement to obtain the next pose. Such a solution will have zero error in the pose graph, because it will not be overconstrained – the linearized error matrix will be square and full rank. Because some of the degrees of freedom of the AUV can be measured directly, however, each pose node is potentially attached to an absolute measurement (depth or attitude) factor in addition to the odometry factors connecting it to its neighbors. In that case, the linearized error matrix becomes rectangular, and the pose graph optimizer smooths over the trajectory to minimize the overall squared error in the discrepancy between the absolute and relative measurements. In fact, the redundancy provided by the navigation sensors makes it possible to estimate roll and pitch biases in the velocity sensor, using the same framework [Kunz, 2011].

Each kind of measurement that can be made by the AUV must correspond in the graph to a type of factor node; each type of factor node has its own error function and rules about the kind and number of pose nodes to which it must be connected. The on-board depth and attitude sensors measure four degrees of freedom, and are represented by the factor nodes $\mathcal{X}_t$, with the exception of the prior on the first pose, $\mathcal{X}_0$, which includes a prior on x and y to anchor the trajectory in space. Measurements from the depth sensor can either be bundled with attitude measurements into a single observation factor, or kept distinct with 1-DOF observations for depth and 3-DOF observations for attitude. Although the navigation sensors are not synchronized, the update rates are high enough that they can be interpolated to a fixed time base, or to the rate of a reasonably fast sensor. For this paper we bundle the depth and attitude measurements together, and use the DVL as the "base" sensor, so there is one pose node in the graph for each DVL ping (about 7 per second), though we have also used the multibeam (which runs at about 10 Hz) as the base sensor in some cases. The error function for these 4-DOF measurements $\mathcal{X}_t = \begin{bmatrix} \hat{z}_t & \hat{\rho}_t & \hat{\phi}_t & \hat{\theta}_t \end{bmatrix}$ is simply

$$\mathbf{e}_{\mathcal{X}_t} = \begin{bmatrix} z_t & \rho_t & \phi_t & \theta_t \end{bmatrix}^{\mathsf{T}} - \begin{bmatrix} \hat{z}_t & \hat{\rho}_t & \hat{\phi}_t & \hat{\theta}_t \end{bmatrix}^{\mathsf{T}} \tag{2}$$

i.e. the difference between what is predicted by the pose and what is measured by the sensors, again with the angle differences fixed to be between $-\pi$ and π radians. The associated 4×4 information matrix is diagonal, with values equal to σ^{-2} for each measured term based on the expected sensor measurement noise. For the set of sensors on the Seabed AUV, we set these values to $\begin{bmatrix} 16 & 3249 & 3249 & 3249 \end{bmatrix}$, corresponding to variances of 0.0625 meters squared in depth, and 1 degree squared in each of roll, pitch, and heading. These variances are higher than those advertised by the instrument manufacturers: the fibre optic gyro ([IXSEA Octans, 2012]) claims a standard deviation of 0.01 degree in attitude, and the depth sensor ([Paroscientific, 2012]) claims 0.01% accuracy, but we have found these claims (particularly with the depth sensor) to be unrealistic, and the presence of latency between sensor measurements smears the signals even more. Also, the gyro is mounted far from the camera and both are far from the DVL on the AUV, and while the frame is reasonably rigid, small deflections are unavoidable which increase the effective noisiness of the gyro.

The 6-DOF odometry factors $\mathcal{O}_{t,t+1}$ incorporate relative measurements made by the DVL. Each DVL sensor reading includes velocity measurements along the three orthogonal axes in the sensor's reference frame, relative to the sea floor; these measurements are rotated into the vehicle's reference frame as forward, starboard, and downward velocities $\begin{bmatrix} \alpha & \beta & \gamma \end{bmatrix}^{\mathsf{T}}$. The origin of the AUV's frame is set at the DVL, rotated to make the frame "level," accounting for roll and pitch biases in the mount. It is important to note that these velocity measurements are terrain-relative, so if the terrain is not flat (relative to gravity) there will be a nonzero γ component to the velocity vector even if the AUV is not ascending or descending. Changes in attitude, on the other hand, are provided by the fibre-optic gyro, and are independent of the local terrain. An odometry factor, therefore, contains a measurement capturing the local translation

$$\begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix} = \delta t \mathbf{R} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \tag{3}$$

and a measurement capturing the local change in rotation, which is computed by converting the rotation rates into axis-angle form, and then multiplying the angle by δt and converting back to Euler angles. The rotation matrix $\mathbf{R}$ in equation 3 captures the permutation from the DVL frame to the vehicle frame, which includes roll and pitch bias terms which can be estimated from the data.

Since the system is overconstrained it is important to consider the starting estimate for the trajectory, because without a good starting point the iterative optimization process can converge to a local minimum, or even fail to converge. For this case, it is sufficient to initialize each pose node with a modification of the "forward chaining" procedure described above, integrating the velocity at each node to produce a pose estimate, but forcing the local attitude and depth to match interpolated values provided by the depth sensor and fibre-optic gyro. The $\sqrt{\text{SAM}}$ framework then smooths out the trajectory based on a combination of the odometry and the absolute sensors. Because the disagreement between these sensors is generally small, initializing the graph this way is sufficient to yield a good solution.

To this point, the pose graph has provided a way to smoothly integrate relative odometry measurements with absolute position measurements. Measurements provided by the camera will be added next, followed by constraints between poses induced from matched multibeam submaps.

3 Visual landmarks in the graph

The use of cameras for navigation in robotics can be loosely divided into two applications: as odometers, and as a source of landmark observations. The former case uses image matches made over small displacements to provide an independent estimate of local robot motion, while the latter case treats images as containing information that will allow the robot to recognize its position at a later point in time. A pair of cameras calibrated together can yield metric 6-DOF relative pose estimates, called stereo visual odometry, a technique widely used in terrestrial robotics [Howard, 2008] [Olson et al., 2003], and underwater in [Beall et al., 2010], for example. Underwater, we have found that a DVL generally provides more reliable odometry information over short distances than does a pair of cameras, without any calibration requirements [Kunz and Singh, 2010]. Given this and the fact that 3-D information can be recovered both from multibeam and from a single camera, we do not generally use a stereo pair for mapping.

3.1 Image matching

Establishing correspondences between images of the same point in the world is a necessary first step for determining visual landmarks. Because power limitations prevent the capture of video from AUVs, instead of dense optical flow-based methods we use the familiar technique of detecting salient image features and describing them in a way that is invariant to the expected changes in view. For underwater mapping, lighting and viewpoint invariance are important, but because the AUV attempts to maintain constant distance from the terrain, we don't expect large changes in scale nor large image changes that cannot be reasonably well-modeled with translation and rotation about the optical axis. While the work described here takes place in post-processing, so that computational complexity is not a pressing concern, there is increasing interest in on-board mapping, which will benefit from video-rate feature detection, matching algorithms, and focused search techniques such as those used in [Davison et al., 2007]. Imagery from the Seabed AUV is preprocessed to account for the illumination pattern of the strobe, and color corrected. There is now a full menu of very effective feature detectors and descriptors to choose from (see for example [Mikolajczyk et al., 2005]); we select interest points using a scaled Harris corner detector, and describe them using coefficients of Zernike polynomials. This combination is simple, robust to image rotation, and effective at finding features in images with low frame-to-frame overlap [Pizarro and Singh, 2003]. Image preprocessing, and color correction in particular, is an ongoing research problem; see [Kaeli et al., 2011] for recent developments, and [Vasilescu et al., 2010] for a formulation based on adaptive lighting. A typical image pair with matched features is shown in figure 3.

Detected features are then matched image to image – each descriptor in one image is compared to all the descriptors in a candidate matching image, using Euclidean distance to compare descriptor vectors. Only definitive matches are retained, and outliers in the set of matches between a pair of images are rejected using either RANSAC [Fischler and Bolles, 1981] or least median of squares [Meer et al., 1991]. For outlier rejection, we use consistency with an image-to-image affine transformation as the model if we expect the scene to be approximately planar, and consistency with a fundamental matrix (i.e. with epipolar geometry) as the model if we expect the scene to have good 3-D structure – the choice is a parameter to the algorithm. While epipolar geometry is valid for any pair of images of the same scene, recovery of the fundamental matrix is only possible if the scene is sufficiently non-planar. Over terrain with rich texture, we have found that using an affine model is generally good enough to find many consistent matches, in spite of the fact that it is more restrictive than the epipolar model. We explicitly choose not to use a fully projective model – the additional two degrees of freedom over the affine model require more point correspondences, and we have

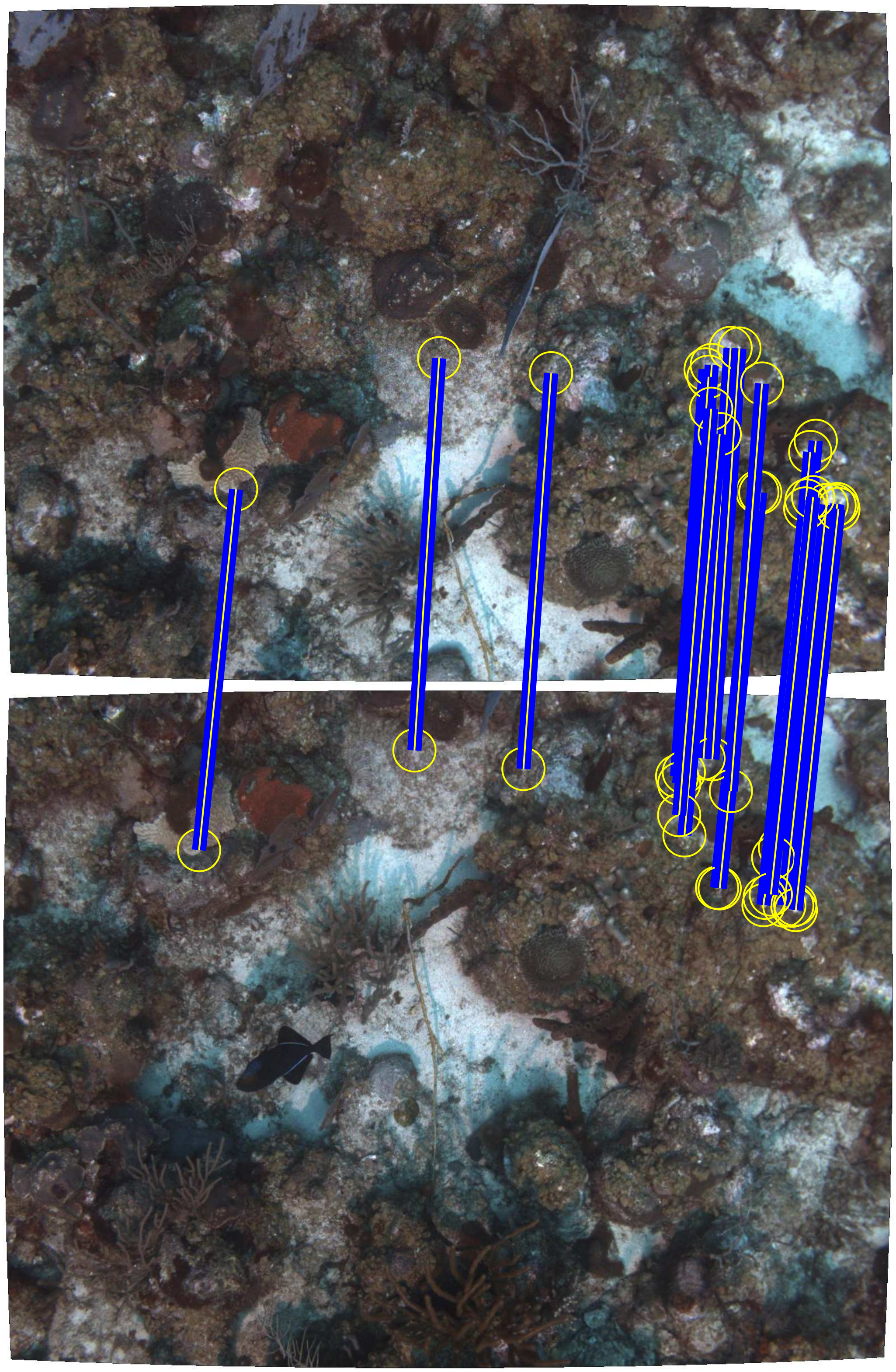


Figure 3: Two typical images captured by Seabed over a coral reef, with matched features shown. The images generally have about 50% overlap, so each point on the seafloor is usually seen in only one or two images.

found that using the projective model does not produce better quality image matches. This is not surprising given the relative lack of perspective foreshortening induced by "flyover" style imaging.

3.2 Constraints induced by matched image pairs

We use the common pinhole model of image formation, which captures perspective projection in the matrix equation $\mathbf{p} = \mathbf{K}\mathbf{X}$, where $\mathbf{p} = \lambda \begin{bmatrix} u & v & 1 \end{bmatrix}^\intercal$ is a homogeneous image coordinate (i.e. defined up to the scale factor λ[1]), and $\mathbf{X} = \begin{bmatrix} X & Y & Z \end{bmatrix}^\intercal$ is a point in the world, relative to the coordinate frame defined with the origin at the camera's center of projection, x-axis parallel to the image's horizontal axis, y-axis parallel to the image's vertical axis, and z-axis pointing from the image center out toward the scene. The 3×3 camera calibration matrix $\mathbf{K}$ contains information about the camera which does not change from image to image: the focal length, pixel aspect ratio, and location of the center of the image. Matters are complicated underwater by the fact that light bends as it passes from water into the (air-filled) pressure housing containing the camera, effectively narrowing the camera's field of view and introducing nonlinear distortion. We have found that the distortion in 3-D reconstruction due to refraction is largely eliminated by typical radial lens distortion models when the scene is viewed from a close distance; see [Treibitz et al., 2012] for a full analysis. Because of this and in particular with planar optical windows, calibrating the intrinsic properties of the camera in water (the $\mathbf{K}$ matrix and the distortion parameters) is necessary for mapping.

Given a calibrated camera and a set of matched features over a pair of images, there are two possible ways to constrain the pose graph. The $\mathbf{K}$ matrix allows us to use normalized image coordinates, and gives rise to the essential matrix, which specializes the epipolar geometry to the normalized case. The essential matrix can be estimated from five matched pixels, and it can be decomposed into a 3-D rotation and a translation vector [Nistér, 2004]. This means that even without incorporating any prior estimate of the camera motion from AUV odometry, a pair of image matches in principle can inform five of the six degrees of freedom of the camera motion between two points in time – only the scale of the motion is not recoverable. A 5-DOF relative pose constraint factor can then be introduced into the pose graph, connected to the two poses from which the two images were taken. Each matched image will thus introduce five scalar constraints into the pose graph, and potentially only six additional variables (the camera's location relative to the vehicle navigation frame) will have to be estimated over the whole trajectory.

An alternative approach, used here and in photogrammetry as "bundle adjustment," is to treat each matched

[1] We can assume $\lambda \neq 0$ as a camera cannot see to infinity underwater.

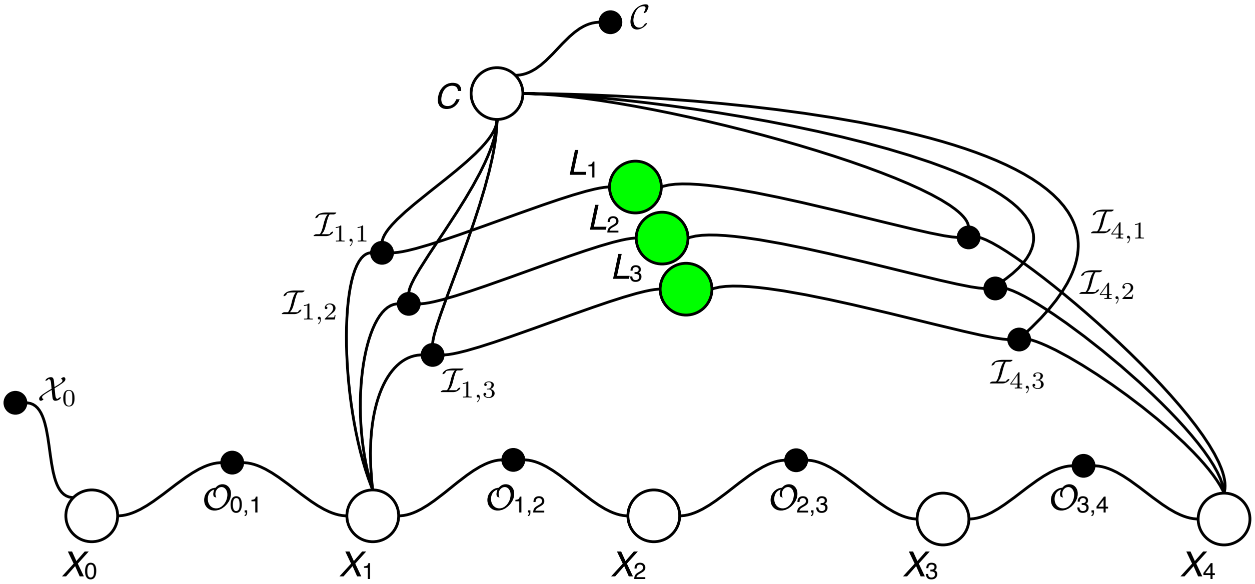


Figure 4: The bundle adjustment, or visual SLAM pose graph. Each matched feature is a single 3-D landmark L_s, and the factors $\mathcal{I}_{i,s}$ capture the pixel reprojection error. An additional pose node C captures the 6-DOF relationship between the camera and the DVL, with prior $\mathcal{C}$. Absolute observation factors provided by depth and attitude sensors are not shown.

image feature as a proper landmark, with a 3-D location which must be estimated. This formulation adds four scalar constraints to the graph for each matched feature: the (x, y) pixel coordinate of the imaged pixel in each of the two images in which it is seen, but also adds three parameters that must be estimated, namely the 3-D location of the imaged point. Each matched feature thus has a net effect of introducing one more scalar constraint into the graph. Since each image pair must have at least five matched features to determine an essential matrix, it makes sense to use this approach rather than the essential matrix-based approach mentioned above, even though the size of the overall problem will be larger. This also allows pixel reprojection error (i.e. consistency with what is actually measured) to be used directly in the pose graph, rather than deviation from a 5-DOF pose constraint. The resulting pose graph is shown in figure 4, which also includes a pose node for the camera's location on the AUV. A prior factor is linked to the camera's extrinsic pose, allowing for an "eyeball" calibration to weakly constrain the optimization.

3.3 Landmark discovery and photomosaicking

While the visual SLAM pose graph in figure 4 can recover both sparse 3-D scene structure and the AUV's trajectory, it begs the question of how the image feature matches are determined in the first place. Certainly a brute-force approach of attempting to match every image from a dive with every other image is not only computationally undesirable, but also at risk of adding false links to the graph. The approach in

[Eustice et al., 2008], particularly appropriate to incremental or in-situ mapping, is to use the most recent estimate of the AUV's position to forge hypotheses about potentially matching images. We compromise between the two approaches, taking advantage of the photomosaicking capability we already use for building 2-D visualizations [Pizarro and Singh, 2003]. This is a purely image-based approach, which assumes that photographs are available in the order in which they were taken, and attempts to match images taken sequentially in time, iteratively refining the link topology based on the set of affine transformations which best align the images. After a few iterations, a graph with many loops and track-to-track links describes the topology; for the coral reef data the topology is shown in figure 5 and a rendering of the corresponding mosaic is shown in figure 6. Navigation information can be used to determine additional hypothesized links, either before or after pose graph optimization. For situations in which speed is of paramount importance, the approach used by [Davison et al., 2007] restricts not only the image to compare, but also the image regions to search for feature matches. The approach in [Nicosevici et al., 2009] matches features from an incrementally rendered 3-D model to each new image, also reducing the number of comparisons which must be made. That system and ours share the advantage of using a 2-D photomosaicking system for determining image features and landmarks, namely that the resulting mosaics can be used as texture maps for ultimately visualizing the full 3-D structure, which we will discuss below.

3.4 Bundle adjustment

Moving from 2-D photomosaicking to 3-D reconstruction, the link topology determined in the mosaicking process is added into the pose graph, with each matched feature contributing one landmark with two observations, as shown in figure 4. Each observation factor is linked to the pose node X_i corresponding to the AUV's pose at the time the image was acquired, and the node representing the camera's extrinsic location relative to the AUV. Using the equivalent 4×4 matrix $\mathbf{X}_i$ to represent the AUV's pose at time i, and $\mathbf{C}$ to represent the the camera's pose in the AUV's frame, and the 3×1 vector $\mathbf{L}_s$ for a landmark imaged at homogeneous pixel location $\mathbf{p}_{i,s} = \lambda \begin{bmatrix} u_{i,s} & v_{i,s} & 1 \end{bmatrix}^\mathsf{T}$, then the error in the observation factor is

$$\mathcal{I}_{i,s} = (\mathbf{X}_i\mathbf{C})^{-1}\mathbf{L}_s - \mathbf{K}^{-1}\mathbf{p}_{i,s} \tag{4}$$

where both vectors in the subtraction have been "regularized" so that their third coordinate is equal to one – the error function measures actual pixel reprojection error, in normalized coordinates. Note that all three factors in the first term of the equation are being solved for in the pose graph optimization. For the 2×2 information matrix, we experimentally chose a matrix with the calibrated focal length on the diagonal,

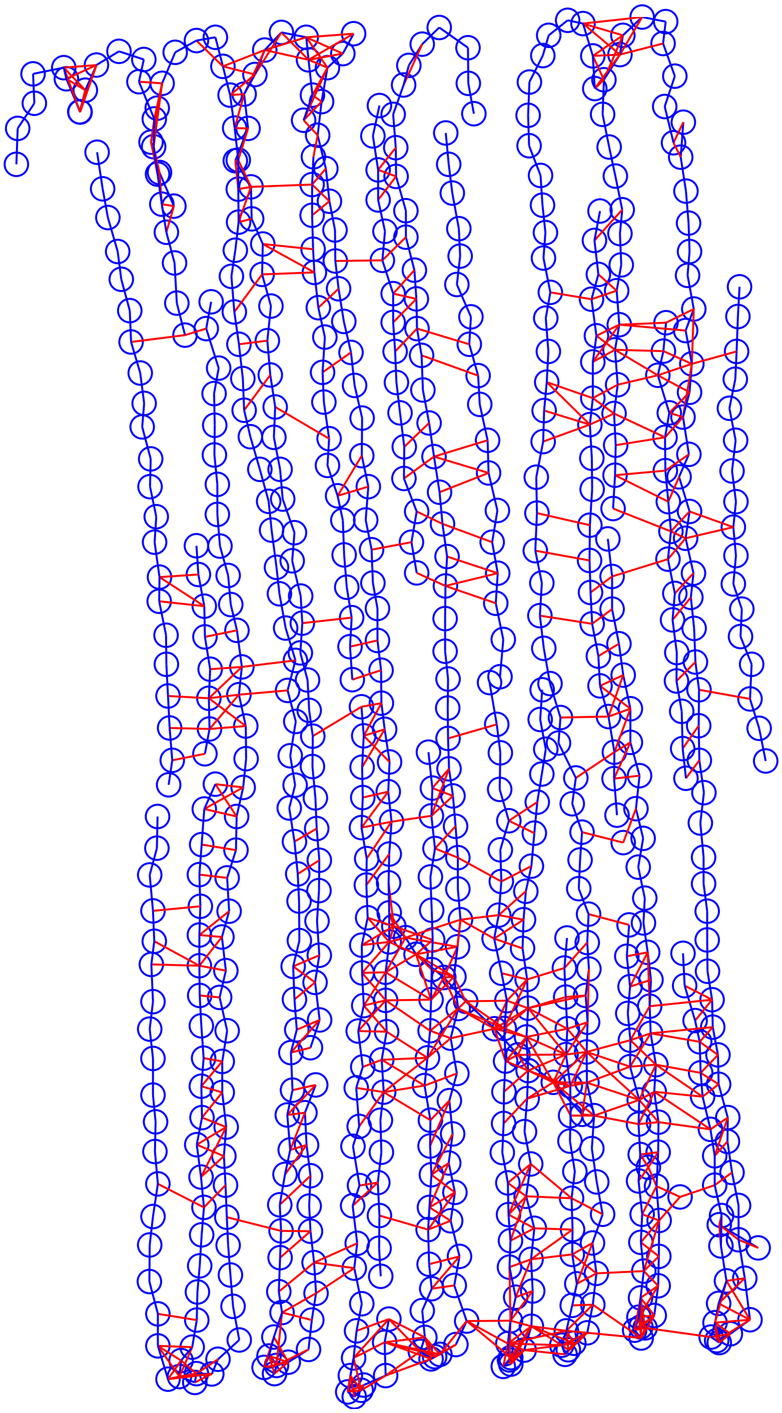


Figure 5: Link topology of about half of the coral reef dive. This shows the connectivity of one connected component of the image link graph. Of the 1476 images acquired during the dive, these 744 images (represented by circles) were matched automatically into the largest connected component. The location of each image in the graph is estimated from the photomosaic rendering, not from the AUV trajectory as recovered by the pose graph optimization. Matched images that were acquired sequentially in time are connected by dark (blue) lines, while cross-links are connected by light (red) lines. The remaining 732 images are split among 51 smaller mosaics, including 20 singletons.

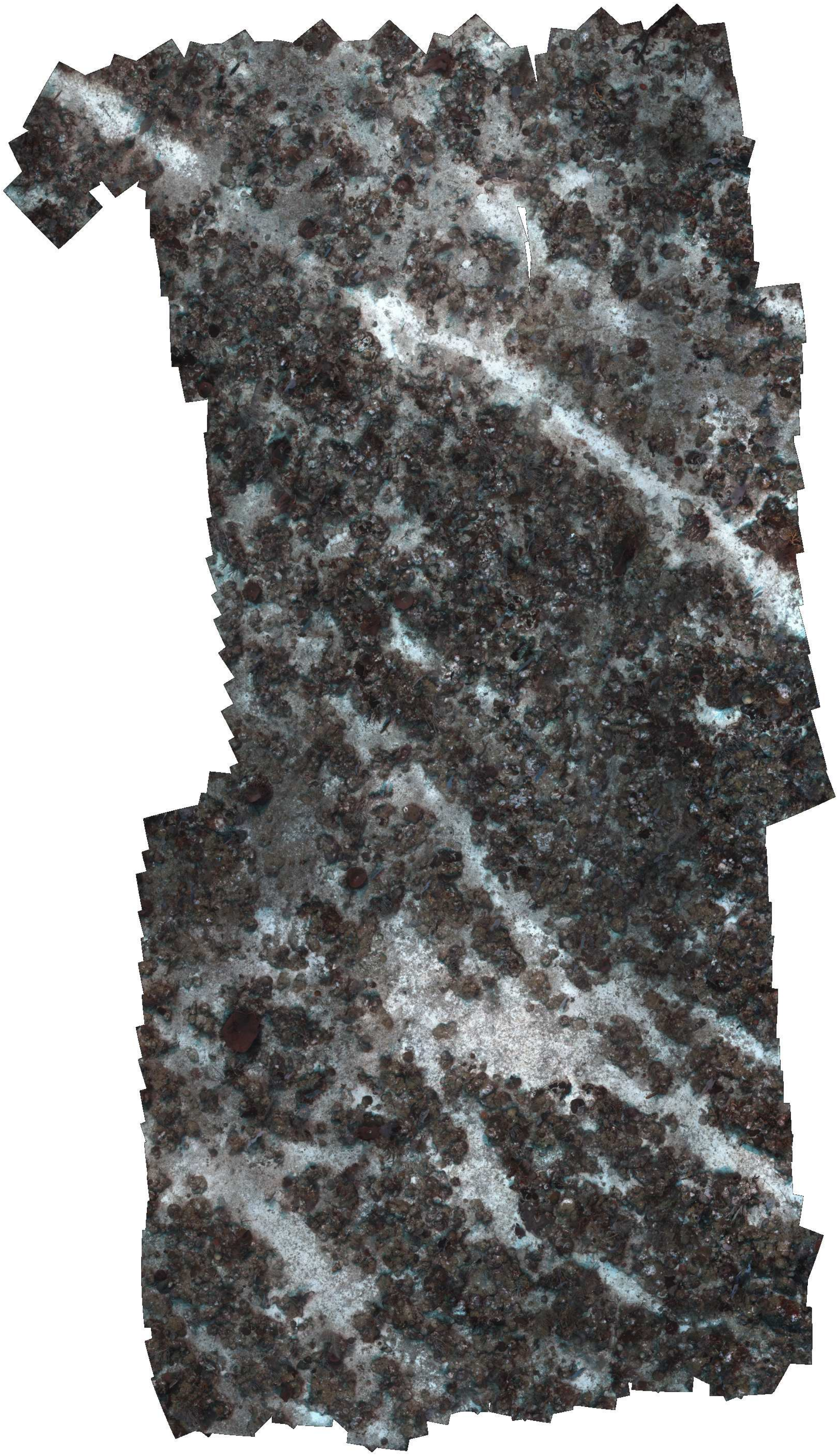
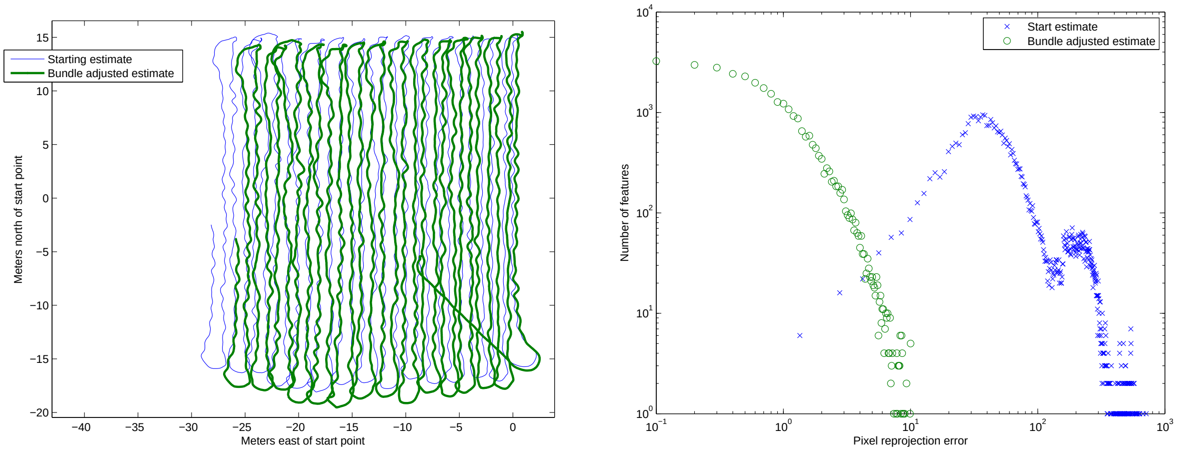


Figure 6: One mosaic from the coral reef data set, corresponding to the link topology shown in figure 5.

corresponding to a variance of one pixel. This means the observed visual features are not individually weighed strongly enough to overwhelm the influence of the AUV odometry, but that as a group the visual features are ultimately able to “pull” tracklines together and close loops.

Each pose node in the graph must be initialized for the optimization. The AUV pose nodes are initialized using odometry, and the extrinsic camera location is initialized using the “eyeball” calibration or nominal mounting parameters – typically the camera is mounted about 1.4 meters forward of the DVL, and is rotated 90 degrees, so that the camera x axis points to vehicle starboard, and the camera y axis points aft. The landmark locations are initialized by triangulation from the AUV pose and camera location estimates, which is a rather noisy guess at the 3-D scene structure, particularly when the AUV has moved a long distance between acquiring matched images. Since it is unlikely that a perfect triangulation will exist for a given feature, the location of a landmark is initialized to the 3-D point closest to the two camera rays under the starting navigation estimate. No weight is given to this initial estimate (there is no prior factor on the landmark locations), but in order to reduce false matches landmarks with ranges differing by more than 20% of the vehicle’s altitude as estimated by the DVL are not used. Once the bundle adjustment has been performed, fewer visual landmarks are rejected in the next stage, when multibeam constraints are added to the pose graph.

Even with landmark pruning, there are still more than enough features to improve the map; for the coral reef data set, 18900 features of 35593 were rejected, and 1917 matched image pairs were used involving 1255 images of 1476 total images acquired during the course of the dive. Figure 7 shows the estimate of the AUV’s trajectory before and after the bundle adjustment step, and the pixel reprojection error in actual (non-normalized) pixels using the start estimates for the camera pose and AUV trajectory, and the improvement in this error after the optimization. The error at the beginning of the optimization is quite large, and bimodal, because the triangulation is based on incorrect navigation and camera extrinsic calibration information. Picking the 3-D point closest to the two rays corresponding to the matched features does not minimize the reprojection error equally in both images, since an average of 3-D positions is being used, rather than an average 2-D displacement from the observed feature locations. Alternative initialization methods, like the inverse depth parameterization suggested in [Civera et al., 2008], might work better here, though the successive reduction in uncertainty provided by that method depends on repeated views (more than two) of the same scene point from progressively distinct vantage points, which we do not have. In any case, the pose graph optimization minimizes the reprojection error of all of the included visual landmarks, so that afterwards the distribution of errors is unimodal and the bulk of the errors are less than one pixel.



(a) (b)

Figure 7: The change in the estimated AUV trajectory using only visual landmarks in the pose graph. This is the result of pose-graph visual SLAM, or bundle adjustment with odometry. The change in trajectory is shown in (a), and the corresponding change in pixel reprojection error is shown in (b), which is a pair of histograms of the distribution of errors. The bimodal nature of the starting reprojection error histogram is due to the method used to initialize landmark positions, as explained in the text.

The incorporation of matched visual features into the pose graph not only gives us an improved estimate of the AUV's trajectory and a sparse set of 3-D scene points, but it also provides an improvement on the 2-D photomosaic, by providing a means to link disconnected subgraphs in the link topology together. Without navigation data, any missing link in the starting chain of image matches irrevocably separates a potential global view into disassociated components – rather than assuming the AUV's navigation estimate and the extrinsic camera calibration are perfect and attempting to render a composite mosaic from pieces, bringing both the image features and the navigation information together in a single framework allows for an improvement of both.

4 Adding multibeam constraints

We proceed now to further constraining the pose graph and hence the SLAM solution by incorporating relative pose constraints provided by the multibeam sensor. Multibeam echo sounders are the underwater analog to the laser scanners used in terrestrial robots. They produce a set of ranges in a fan-shaped pattern with a field of view of 90 to 120 degrees. For seafloor mapping, the sensors are oriented in "profiling" mode, pointing straight down from the vehicle to which they are attached, so that the field of view is perpendicular to the direction of travel. Because of the finite speed of sound (about 1500 meters per second in sea water), the principles of operation of multibeam sonars are different than those of laser scanners. Laser scanners measure ranges effectively instantaneously, and line scans are effected by a spinning platform in the sensor. Multibeam sonars, on the other hand, measure a whole set of ranges with a single pulse of sound, by "beam forming" the emitted pulse into a fan shape, and then measuring the difference in time of arrival (or the phase differential) of the echoes at several different transducer positions within the sonar head. This allows a ship or an underwater vehicle to create a dense bathymetric map even if it is moving faster than a meter per second through the water.

One pervasive problem in underwater mapping is lack of available ground truth. With terrestrial robots, it is often possible to check a map against a benchmark, which can be produced using surveying equipment incorporating GPS and fixed laser scanners. Underwater neither GPS nor laser range scanners work, and the sensors available for validation are the same as those used to build maps in the first place. In the ocean, then, the best metric we can use for measuring map "goodness" is self-consistency. The easiest way to measure self-consistency in a multibeam map is to grid, or bin the map down to a fixed resolution, assigning a single depth value to each grid cell based on the mean of the ranges measured to that cell, and then to check the variance of the distribution of ranges for each cell – small variances indicate good consistency. While the consistency of a map can be visually estimated by examining the binning variances in regions seen by multiple tracklines, a quantitative measure of the goodness of the entire map is the average variance over all the cells, or (equivalently, if the number of cells does not change from map to map) by the sum of the variances over all the cells.

4.1 Incrementally building a multibeam map

Building a map with multibeam data reduces to a question of geometry – if the pose S_t of the multibeam sonar head is known at the time t of each ping, then the range from each beam can be placed into a globally-referenced point cloud, which can eventually be gridded into a map. Determining S_t requires knowing the AUV's position at time t, and the 6-DOF pose offset P between the AUV's navigation frame and the multibeam head. If R_j is the rotation describing the beam angle for beam j, and $b_{t,j}$ is the range for beam j at time t, then the corresponding point $p_{t,j}$ is

$$p_{t,j} = X_t \oplus P \oplus R_j \oplus b_{t,j} \tag{5}$$

and the mapping problem requires estimating X_t and P. The first of these parameters is computed in the pose graph optimization, assuming good knowledge of the synchronization between the navigation sensors and the multibeam. The second requires extrinsic calibration of the multibeam head, which we perform automatically by minimizing the variance of gridded maps built with hypothesized offsets, outside of the pose graph framework. See [Singh et al., 2000] and [Kunz, 2011] for details.

4.2 Constraints induced by multibeam matches

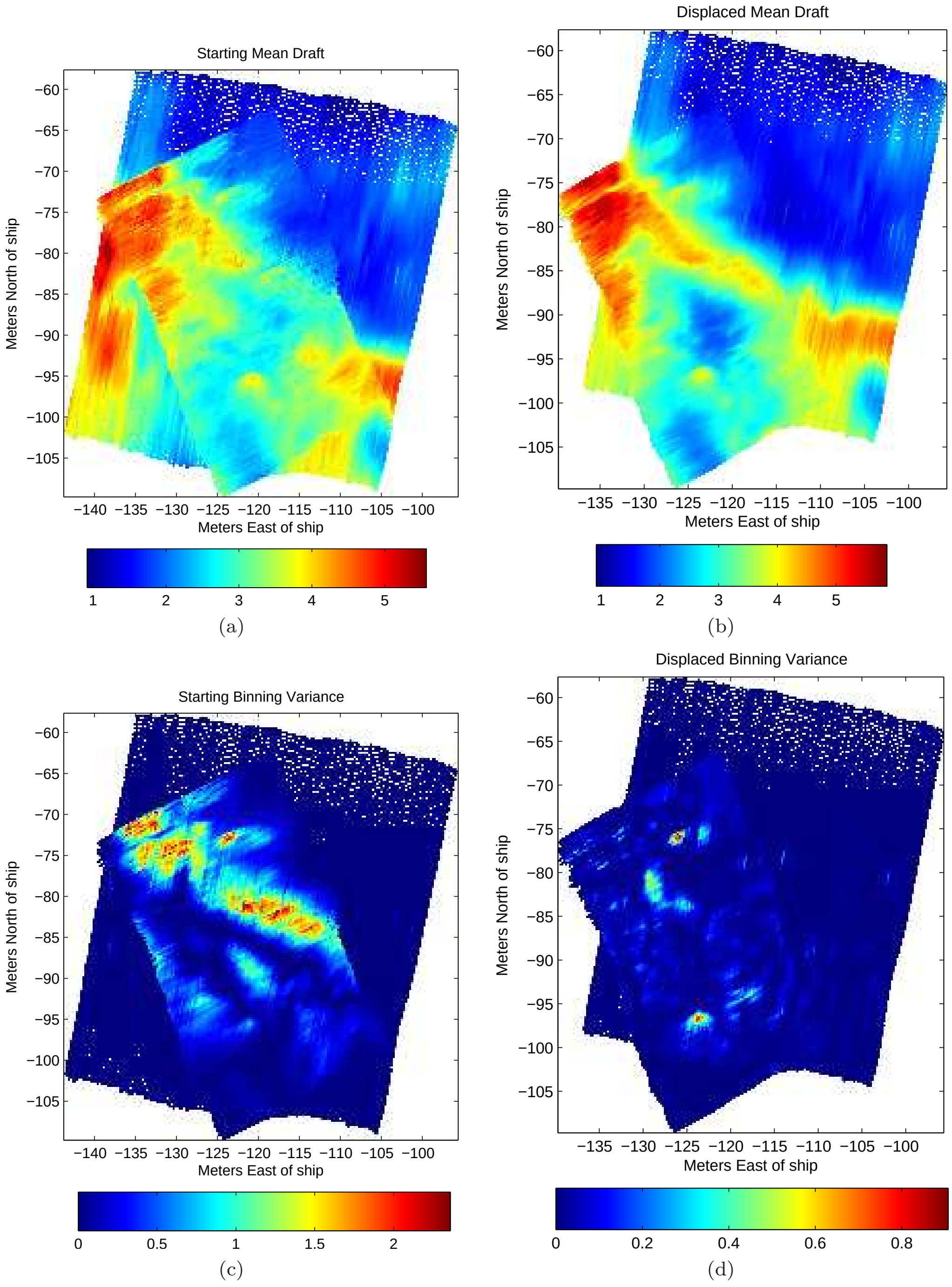
Multibeam sonars are used to build maps incrementally, one ping at a time, and successive pings are accumulated into a range map. Because each ping has relatively low resolution (relative to laser scanners, for example), distinctive landmarks are difficult to identify and differentiate from one another in single scans. For this reason, multibeam mapping efforts using SLAM such as [Roman and Singh, 2007] and [Barkby et al., 2009] do not rely on landmarks, but instead either make use of relative pose constraints (in the former case), or directly use an occupancy grid representation which dispenses with the need for the explicit connection of mapping regions (in the latter case). When distinct features can be expected to be seen, such as in man-made environments, more traditional SLAM approaches such as EKFs can be made to work quite well, such as in [Ribas et al., 2008]. While our method is closely related to the former approach, as the pose graph framework is much better suited for handling relative pose constraints than it is full occupancy grids, we have started experimenting with the incorporation of occupancy grids into the same scheme – see section 6.1 below. We build small submaps where the AUV's trajectory crosses over itself and find the transformation that best aligns them, and then determine a relative pose constraint induced by the match. In the pose graph, these constraints are very similar to the 6-DOF odometry constraints linking poses at adjacent time steps, but they link two poses that are arbitrarily far from each other in time. Because the constraint only links two poses in the graph, the smoothing step does not force the individual submaps to remain static – rather the whole trajectory (and hence the map) is warped to minimize the overall error described by all of the constraints.

The submaps themselves are produced using the starting trajectory estimate from navigation aided by visual bundle adjustment described above, taking into account the footprint of the sensor on the terrain, as well as the shape of the planned mission. Crossover points are planned for, and tracklines are designed to be close enough to each other to allow significant sensor footprint overlap from leg to leg. While both "intersections" in the trajectory and parallel tracklines can be used to produce relative pose constraints, the data used for the discussion in this paper do not include intersecting tracklines, because the AUV surfaced prematurely before reaching this part of its preprogrammed mission. Once two gridded submaps are built, they are aligned by scanning over a horizontal displacement between the two of them, and finding the (x, y) displacement which minimizes the average difference in depths between the two maps, namely

$$\frac{1}{N_{\text{overlap}}} \sum_{(x,y)\in M_1 \wedge (x+\Delta x, y+\Delta y)\in M_2} \left(\bar{z}_{(1,x,y)} - \bar{z}_{(2,x+\Delta x,y+\Delta y)}\right)^2 \tag{6}$$

where the sum is over all cells that contain data in both maps, and $\bar{z}_{k,x,y}$ is the mean depth for the cell at location (x, y) in map k. We do not take the additional step of matching submaps using ICP (taken by [Roman and Singh, 2007] for example) for several reasons. By searching only for a horizontal displacement which aligns the maps, we are assuming that the attitude and depth of the AUV are well-known by this point in the mapping process, which should be the case given the refinement of the trajectory using visual features. Furthermore, ICP is minimizing a different local error function, typically the sum of distances between points and locally computed planes, rather than displacement in meshes in the z direction. We have found experimentally that the ICP-optimized transformation matching two submaps will often result in higher overall depth variance once the combined point clouds are binned. An example submap alignment is shown in figure 8, which shows gridded maps containing the two submaps both before and after they have been aligned. The submaps shown are from a distinct dive, in which the AUV was measuring sea ice draft rather than bathymetry. These maps are shown because the reduction in variance with submap matching is easier to visualize with this data than with the coral reef data. The left column of the figure shows the initial configuration of submaps, and the right column shows the result, with one map shifted relative to the other. The top row shows the binned map themselves (mean draft), while the bottom row shows per-bin variance in the measured draft. The reduction of variance from the left to the right column shows that the alignment has improved the local self-consistency of the combined map.

A confidence estimate of the depth image correlation is provided by fitting a 2-D quadratic locally to the error surface being minimized. Equation 6 provides an error term for each (x, y) offset estimate, to which we then fit a quadratic using linear least squares. The shape of the quadratic around the minimum is governed by the quadratic's Hessian matrix, which is in turn used as the information matrix to weight the relative pose constraint's influence on the pose graph. This is because the Hessian describes how quickly the error function changes as the 2-D displacement is changed, which indicates how confident we should be in the match. If the determinant of the Hessian is too close to zero (or is negative), then the match is unreliable and the link is not added to the pose graph. The use of the Hessian here means that matching submaps of flat terrain will only weakly influence the pose graph, because the correlation error score at the best displacement will have a small second derivative. More importantly, linear features such as troughs and ridges will have strong minimum errors along one dimension only, which is reflected by the magnitude of the Eigenvalues of the



(a)

(b)

(c)

(d)

Figure 8: Two submaps from different portions of the AUV's trajectory during a dive to measure sea ice draft, combined into a single gridded map. The mean bin values, representing draft, are shown in (a) and (b). The corresponding per-bin variances are in (c) and (d), using the same color scale – the maximum variance in (d) is one third of the maximum variance in (c). The maps in (a) and (c) show the starting configuration, before one map has been shifted to match the other. Once the best displacement has been found, the resulting combined maps are shown in (b) and (d).

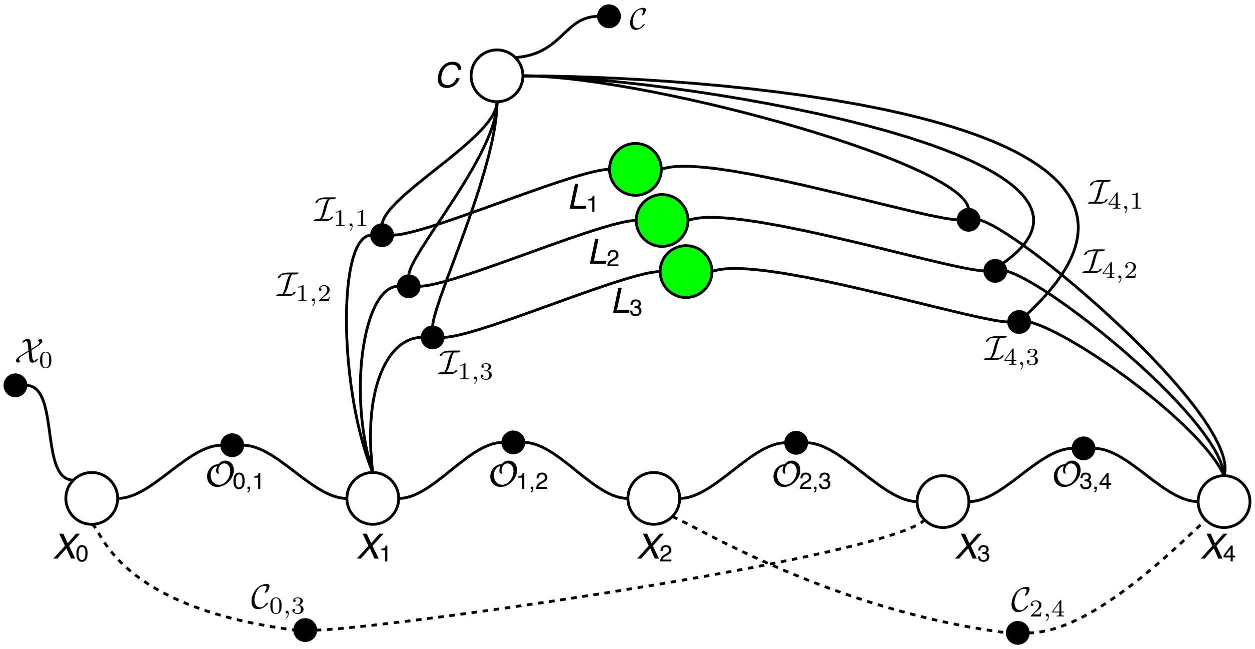


Figure 9: Adding multibeam constraints to the bundle adjustment pose graph allows both types of mapping sensors to be incorporated into the problem at the same time. Absolute observation factors are not shown. The dashed lines show relative pose constraints induced by multibeam submap matches. A full graph will have dozens of submap matches and thousands of visual landmarks.

Hessian: the resulting information matrix will constrain the AUV poses only in the directions in which the match is well-conditioned.

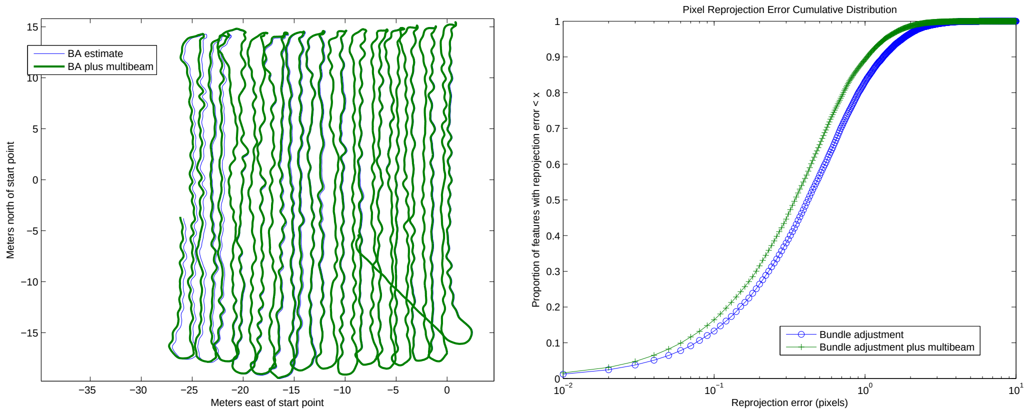
Once the local submap match has been found and the information matrix determined, the displacement and information matrix are transformed into a relative pose constraint in the graph, by orienting the displacement and Hessian relative to the first of the two AUV poses being compared. For each pair of matched submaps, a single relative pose constraint of the type $\mathcal{C}_{i,j}$ is added to the graph in this way, using the multibeam pings nearest the center of each submap to “anchor” the constraint – if the multibeam ping times do not correspond exactly to the times of the pose nodes in the graph (e.g. when using the DVL as the base sensor instead of the multibeam), then the constraint is adjusted slightly so that it correctly constrains the relationship between the two closest pose nodes to these times. The number of factors added to the graph depends on the trajectory of the dive, but as long as the number of these cross-constraints does not increase more than linearly with the size of the graph, the sparsity of the overall system remains intact. A portion of a pose graph incorporating both visual and multibeam constraints is shown in figure 9.

5 Results

5.1 Two sensors are better than either alone

At each step in the process, the influence of the additional information on the quality of the map must be considered. We can consider each sensor modality independently, by applying our method incrementally, omitting the influence of visual landmarks or multibeam constraints, and examining how the addition of constraints from one mapping sensor affects the quality of the map produced by the other mapping sensor in isolation. Consistency in the multibeam map is measured by the variance in the binning step and reduction in the weighted error of cross links in the pose graph. At the same time, pixel reprojection error is used to check consistency in the use of visual features in the map. Although the rest of the discussion uses data acquired by the Seabed AUV in May 2010 from a coral reef in Puerto Rico, we have also applied this technique to the measurement of sea ice draft in Antarctica with only multibeam data, and to a Pacific sponge reef survey including both visual and multibeam data, but in which many of the acquired images contained only mud. For the coral reef data set, 53 multibeam submap matches were combined with 16470 visual landmarks over 26010 AUV poses. These constraints were derived from 1355 individual images and 35495 multibeam pings (each with 480 beams) over a 30 meter square area of seafloor; each multibeam submap includes 70 pings on average. The change in trajectory after incorporating multibeam data into the bundle adjustment result is shown in figure 10, together with the change in pixel reprojection error. Although the change in trajectory is quite small, the shift in the distribution of pixel reprojection errors (shown in figure 10b as a pair of cumulative distributions, rather than as histograms) is apparent: the addition of constraints from multibeam into the pose graph improves the reprojection error for visual landmarks.

The reverse perspective is shown in figure 11, which depicts histograms of multibeam bin variances at various stages of the optimization. While the distributions of binning variances are close to each other (with the exception of the completely uninitialized starting condition), there are differences in the underlying data which are worth pointing out. Firstly, the final result of running a multibeam-only optimization is comparable to the result of using both multibeam and visual landmarks for this data set. This means that we are close to the upper limit in map consistency that can be achieved with this combination of bin size (5 cm square) and terrain roughness. This is also apparent from figure 10a: the shift in AUV trajectory induced by the incorporation of both kinds of constraints is small enough that most of the multibeam rays do not change from one bin to another, and so the overall consistency of the bathymetric map does not change much.



(a)

(b)

Figure 10: After adding multibeam constraints to the bundle adjustment (BA) problem, the AUV's trajectory is further refined to the final estimate shown in (a). The adjustment in trajectory is minimal, but it does improve the pixel reprojection error for visual landmarks, as seen in the cumulative distribution in (b).

Secondly, the error values of the individual multibeam pose constraints are improved by the addition of visual landmarks into the pose graph. This is not apparent from the figure, which only shows overall binning variance, but it is significant because it indicates that visual landmarks improve the overall consistency of the AUV trajectory relative to the multibeam constraints, just as the addition of multibeam constraints improve the pixel reprojection error. This is in spite of the fact that the starting error values of the multibeam constraints are in general *higher* when the trajectory has been initialized by the bundle adjustment result than when the starting map is made only using a traditional multibeam "patch test" ([Singh et al., 2000]). This is explained by the fact that the change in the estimate of the AUV trajectory induced by the visual landmarks leads to a different estimate of the multibeam head attitude – once a new attitude has been estimated, the result of the combined optimization is that the relative pose constraints have a lower weighted error than the final result when visual landmarks are not used. The visual features, in other words, make the overall estimate of the trajectory and bathymetry less consistent to begin with, but afford a better overall solution once the optimization is performed. The final binned bathymetry map is shown in figure 12.

In this data set, the multibeam map is sufficiently consistent to allow binning at 5 cm horizontal resolution from 3 meters vertical range, or 6 meters slant range (assuming a 120 degree multibeam field of view). We have used the same system without the addition of visual features to bin multibeam data of ice floe draft at 25 cm from 20 meters vertical range. These values approach the advertised resolution of 0.2% of range. While the consistency of binned maps can be quantified by the variances of bin depths, without ground truth the judgement of bathymetric maps is ultimately qualitative. Still, we have seen that the addition of visual features into the navigation system has allowed for a reduction in bin size over standard practice.

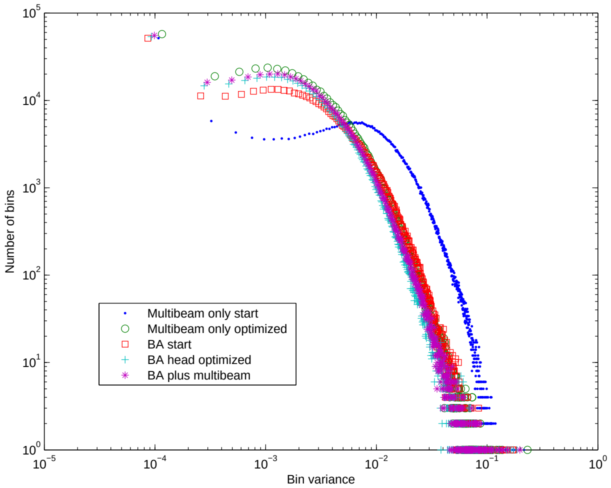


Figure 11: Histograms of multibeam binning variances. Each symbol in the graph corresponds to the variance histogram at a particular point in the optimization. Lower variance implies better self-consistency.

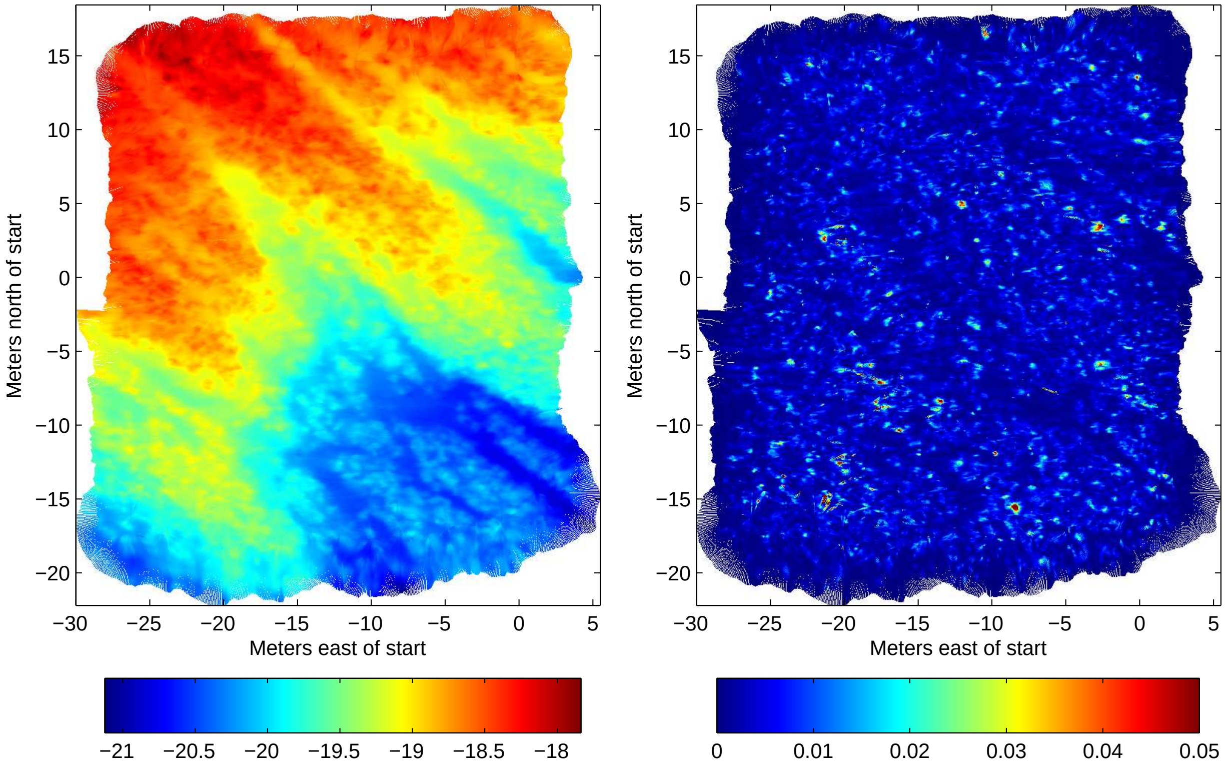


Figure 12: The bathymetry map incorporating both multibeam cross links and visual landmarks. On the left is the mean depth in each bin in meters, and on the right is the variance in each bin in meters squared. Each grid cell is five centimeters on a side.

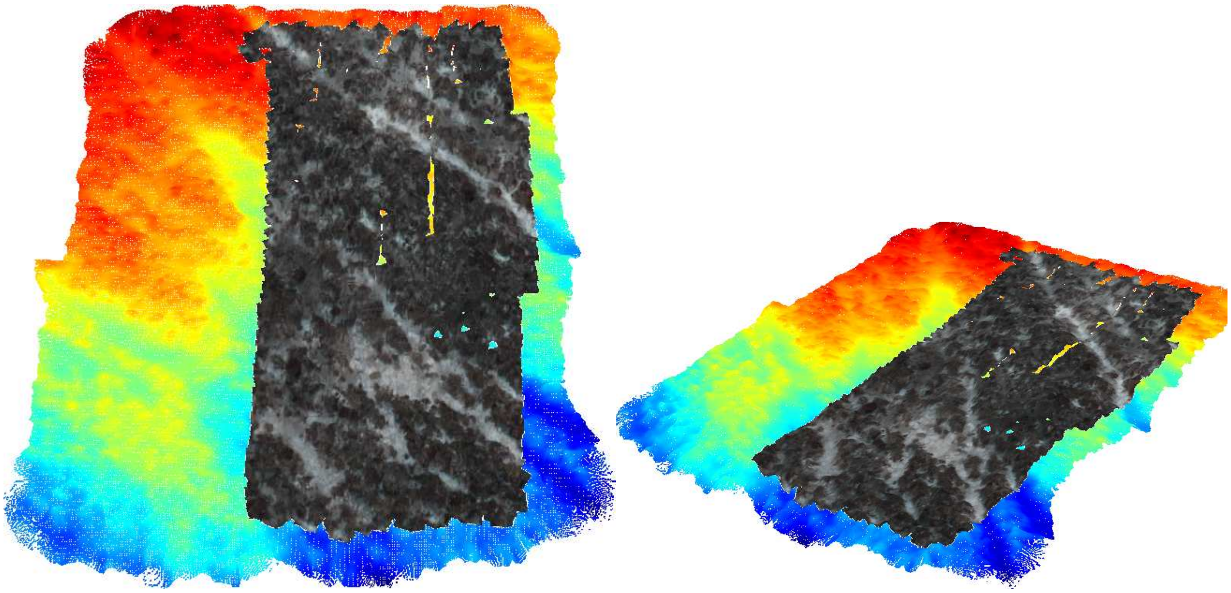


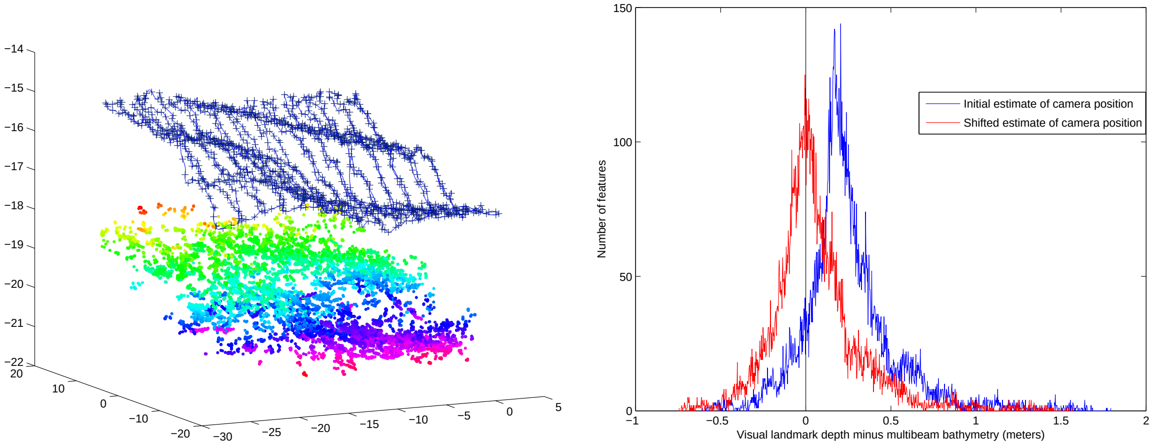
Figure 13: Rendering of the multibeam bathymetry, partially texture-mapped using the photomosaic from figure 6. Intensity on the non-texture-mapped portion of the bathymetry corresponds to depth, with white corresponding to 17.75 meters and black to 21 meters.

So if adding multibeam data to the bundle adjusted estimate only marginally improves the vision-only result, and adding visual data to the multibeam estimate only marginally improves the multibeam-only result, why use both sensors? There are two answers: first of all, using both sensors can be a big help in environments where the data are not as rich as on a coral reef, where there is a large amount of visual texture and 3-D topography (i.e. both sensors perform well in this situation). There are many places of interest underwater characterized by sparsely distributed regions of high visual texture surrounded by regions with very little texture. Secondly, even in texture-rich scenarios, fusing the two sensors together provides a new way in which to visualize the environment: the imagery can be texture-mapped onto the multibeam bathymetry, because the optimization produces the 6-DOF AUV pose at the time of every ping and image capture, as well as the 6-DOF offsets to both mapping sensors. Such a view is shown in figure 13. While the figure only shows a single mosaic texture-mapped onto the bathymetry mesh, texture mapping does not require all the images to be mosaicked together. The mapping is determined by projecting the center of each bathymetry cell onto each of the image planes, determining which viewpoint “sees” the point closest to the image center, and then using the projected pixel location as the texture coordinate for that vertex in the mesh. This can either be done using mosaics as the texture images, or the individual images themselves – it makes some sense to prefer the source imagery, since mosaics will have distortions induced by the 3-D structure which will likely not be completely reversed by the texture mapping process.

5.2 Comparing 3-D structure from vision and multibeam

We have focused to this point on the consistency of the sensors individually. It is also worth asking about the consistency in the two different 3-D estimations of scene structure that are built by the mapping sensors. The multibeam gives us relatively dense bathymetry directly, and the bundle adjustment produces a sparser set of 3-D points. How well to they compare? Figure 14 shows the visual landmarks with the AUV trajectory, and an outline of two histograms of the vertical distance in meters between visual landmarks and the corresponding grid cell in the bathymetry. Both histograms clearly show a rather wide unimodal distribution – this width is explained by the vastly different resolutions of the mapping sensors, as each pixel images a region in space that is much smaller than the 25 square centimeters captured by a single multibeam grid cell. The two distributions are shifted relative to each other: the dark (blue) line, representing the result of the initial combined optimization, shows a bias of about 0.2 meters between the bathymetry as measured by the two sensors. The light (red) line in the figure shows the same comparison after changing the initial camera location in the optimization by moving it vertically by this bias. At first glance, this change might not be surprising: the camera has been moved by a distance approximately equal to the discrepancy between the ranges measured by the two sensors, and this has seemingly removed the bias. But it is important to realize that the bias has been removed without affecting the final overall optimization error, and the pixel reprojection error histograms have also remained essentially unchanged – in fact the histograms are so similar that it is not useful to show plots of them. This means that the camera can be moved vertically in the AUV's frame at least by this small amount without any penalty in the optimization error.

The fact that such a large shift in the scene structure can take place without significantly affecting the optimization error or the pixel reprojection errors indicates that the error surface being minimized over is wide and flat at the bottom, at least with respect to the z position of the camera relative to the AUV. There are two explanations for this. The first is that the AUV is attempting to maintain a constant altitude over the seafloor while capturing images, and the lack of relative vertical motion, and especially the lack of any significant change in roll or pitch, together mean that this region of the parameter space is not being exercised. The second is that the pose graph as it has been defined does not directly compare the 3-D structure of the scene as measured by the two sensors. Only the depths of visual features are computed in the optimization, while the constraints in the map that are contributed by multibeam are of the form of relative AUV poses. Thus while information from two sensors is simultaneously used to determine the AUV's trajectory, the estimation of the scene structure from the multibeam is somewhat indirect – multibeam binning happens outside of the optimization framework. Tightening the variances on the feature points, or



(a) (b)

Figure 14: The scene structure as a collection of 3-D visual landmarks colored by depth is shown in (a) with the recovered AUV trajectory. In (b) is the disparity in scene depth computed from multibeam and from camera features. The dark (blue) line shows the difference when starting the combined optimization with an “eyeball” estimate of the camera location, while the light (red) line shows the difference after re-starting the optimization with a camera location that has been shifted by the peak in the blue line. The black line is at zero, for reference. The final error in the two optimization runs is approximately equal, as are the pixel reprojection error histograms.

optimizing over the intrinsic camera calibration parameters, will not cause this dimension of the parameter space to be excited. Instead, future dives can be planned to include increased vertical AUV motion, ideally including rolling and/or pitching motion. Although a parameter has been found which (for reasonable values) does not influence the global optimization error in a significant way, the two estimates of scene structure generally only differ by this vertical offset, which is easily recovered. Another way to view this is that the smoothing and mapping optimization recovers 6-DOF AUV motion, the full scene structure, and the 5-DOF extrinsic camera location, allowing the one last degree of freedom to be measured directly from the output. Similarly, as mentioned in section 4.1 the 3-DOF multibeam attitude offset is recovered independently of the smoothing and mapping optimization, though in that case the recovery happens before the optimization, rather than afterwards.

5.3 Tunable parameters and performance

As with many SLAM systems, the performance of the pose graph approach presented here is subject to the values of tunable parameters. Included in these parameters in order of importance are the values in the information matrices for navigation measurements (depth, attitude, velocity) and pixel reprojection error; the parameters used for outlier rejection in establishing feature correspondence between images; and the threshold used for rejection of multibeam submap correspondences. The information matrix values work as weights in the global error function, and changing them has the expected effect, causing one set of measurements to dominate the solution. Our choice of using standard deviations of one pixel for reprojection errors, and the values mentioned in section 2.3 for absolute sensor measurements balances the influence of each, allowing image measurements to change the pose graph without introducing any abrupt jumps. The outlier rejection parameters are much more straightforward: it makes sense to aggressively reject mismatches in both image-to-image and multibeam submap matching to prevent incorrect factors from being introduced into the graph, even at the expense of valid links not being used.

6 Conclusions and future work

6.1 Future research

While there are several directions still to be pursued in this work, perhaps the most interesting next step to take involves more directly linking the scene structure as computed by the two mapping sensors. Our current approach with the multibeam sensor attempts to improve the overall consistency of the map by adding constraints on individual map subsections – the assumption is that improving local consistency will also improve global consistency.

In contrast, the landmark-based approach used to incorporate visual features into the map directly aims to minimize pixel reprojection error for every matched feature, so the scene structure is solved for along with the AUV trajectory in the optimization, without the need for any “local-to-global” consistency assumption. To take the same approach with range data, the overall map consistency should be used as an error metric in the optimization. Adding such a global consistency metric would likely require a change in problem representation or at the very least a large reorganization of the pose graph to make use of something like an occupancy grid, so that the binned output is treated directly by the optimizer: the depth information from multibeam needs to be brought into the pose graph directly, rather than through the indirect method of relative pose contraints. The most straightforward approach would be to represent each cell of the grid with a pose node estimating bathymetry, and to connect each such node to AUV pose nodes which measure range to that cell, via a measurement factor. Visual landmark nodes would also be connected to grid cell nodes, via a factor which measures the error between the bathymetry as represented by the grid cell, and the estimated depth of the visual landmark. The problem is that as the optimizer runs, the connectivity of the graph would change, as changes in the estimate of the AUV’s trajectory would yield measurements of different areas of the terrain, and hence different grid cells. While current graph optimization engines allow for incremental updates to the graph structure to be made [Kaess et al., 2012], these are generally in the form of new landmarks being introduced, new poses being added, or new measurements connecting poses and landmarks being made. We are suggesting a change that would be akin to a wholesale reassignment of several observations from one landmark to another, which would require column reordering optimizations at each step in the minimization as the sparsity structure of the Jacobian would radically change from iteration to iteration. This would require only a linear increase in complexity, as the batch optimizer already takes this step once at the beginning; however it would require more work to build such a system that could operate on-board an AUV as the data are being collected. In any case, such a change would both allow the direct optimization over the full multibeam map, and provide a means for the bathymetry as measured by the multibeam to be directly compared to the scene structure as measured by the camera, pulling the two sets of depth estimates together and improving the flatness of the error surface relative to the z position of the camera.

Finally, an incremental implementation of this system that could run on-board a vehicle while it is underwater would be helpful for improving the pose estimate used by the AUV for navigation. The $\sqrt{\text{SAM}}$ framework can be used incrementally and efficiently, but the implementation is not yet optimized for use on an embedded system. Visual feature matching is already done on smart phones, though additional pre-processing is necessary on underwater imagery to account for lighting conditions. While multibeam submap matching is a fast 2-D correlation, the beam-forming software currently requires a second computer. In short, the current implementation is not fast enough to run on-board, but this is due to hardware limitations rather than to computational complexity issues.

6.2 Conclusion

We have presented a technique for building 3-D maps of the seafloor incorporating range information from acoustic sensors with visual texture from a single camera using a uniform optimization framework relying on the abstraction of the pose graph. The system is generalizable to several different deployment scenarios, with varying sensor configurations and degrees of control stability – in short the post-processing steps described here are independent of the particular hardware used to collect the data, except for the easy-to-meet assumptions of measurability of depth, attitude and velocity. While this work has revealed additional lines of research to be pursued, the system as it stands robustly and autonomously creates accurate and useful high-resolution 3-D visual maps of underwater terrain, as we have demonstrated with data collected by an AUV over coarse terrain.

Acknowledgements

The work described herein was funded by the National Science Foundation Censsis ERC under grant number EEC-9986821, and by the National Oceanic and Atmospheric Administration under grant number NA090AR4320129. The authors gratefully acknowledge this support. We also thank the anonymous reviewers for their thoughtful comments and insights.

References

[Barkby et al., 2009] Barkby, S., Williams, S. B., Pizarro, O., and Jakuba, M. (2009). An efficient approach to bathymetric SLAM. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 219–224. doi:10.1109/IROS.2009.5354248.

[Beall et al., 2011] Beall, C., Dellaert, F., Mahon, I., and Williams, S. B. (2011). Bundle adjustment in large-scale 3D reconstructions based on underwater robotic surveys. In *OCEANS, 2011 IEEE - Spain*, pages 1–6. doi:10.1109/Oceans-Spain.2011.6003631.

[Beall et al., 2010] Beall, C., Lawrence, B. J., Ila, V., and Dellaert, F. (2010). 3D reconstruction of underwater structures. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4418–4423. doi:10.1109/IROS.2010.5649213.

[Butler et al., 2002] Butler, J., Lane, S., Chandler, J., and Porfiri, E. (2002). Through-water close range digital photogrammetry in flume and field environments. *The Photogrammetric Record*, 17(99):419–439. doi:10.1111/0031-868X.00196.

[Camilli et al., 2010] Camilli, R., Reddy, C. M., Yoerger, D. R., Van Mooy, B. A. S., Jakuba, M. V., Kinsey, J. C., McIntyre, C. P., Sylva, S. P., and Maloney, J. V. (2010). Tracking hydrocarbon plume transport and biodegradation at Deepwater Horizon. *Science*, 330(6001):201–204. doi:10.1126/science.1195223.

[Civera et al., 2008] Civera, J., Davison, A. J., and Martínez Montiel, J. M. (2008). Inverse depth parametrization for monocular SLAM. *IEEE Transactions on Robotics*, 24(5):932–945. doi:10.1109/TRO.2008.2003276.

[Clarke et al., 2009] Clarke, M. E., Tolimieri, N., and Singh, H. (2009). Using the Seabed AUV to assess populations of groundfish in untrawlable areas. In Beamish, R. J. and Rothschild, B. J., editors, *The Future of Fisheries Science in North America*, volume 31 of *Fish and Fisheries*, pages 357–372. Springer Science + Business Media B.V.

[Davison et al., 2007] Davison, A. J., Reid, I. D., Molton, N. D., and Stasse, O. (2007). MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067. doi:10.1109/TPAMI.2007.1049.

[Dellaert and Kaess, 2006] Dellaert, F. and Kaess, M. (2006). Square root SAM: Simultaneous localization and mapping via square root information smoothing. *International Journal of Robotics Research*, 25(12):1181–1203. doi:10.1177/0278364906072768.

[Dhanak et al., 2001] Dhanak, M. R., An, P. E., and Holappa, K. (2001). An AUV survey in the littoral zone: Small-scale subsurface variability accompanying synoptic observations of surface currents. *IEEE Journal of Oceanic Engineering*, 26(4):752–768. doi:10.1109/48.972117.

[Escartín et al., 2008] Escartín, J., García, R., Delaunoy, O., Ferrer, J., Gracias, N., Elibol, A., Cufi, X., Neumann, L., Fornari, D. J., Humphris, S. E., and Renard, J. (2008). Globally aligned photomosaic of the Lucky Strike hydrothermal vent field (Mid-Atlantic ridge, 37 °18.5′N): Release of georeferenced data, mosaic construction, and viewing software. *Geochemistry, Geophysics, Geosystems*, 9:12009. doi:10.1029/2008GC002204.

[Eustice et al., 2008] Eustice, R. M., Pizarro, O., and Singh, H. (2008). Visually augmented navigation for autonomous underwater vehicles. *IEEE Journal of Oceanic Engineering*, 33(2):103–122.

[Eustice et al., 2006] Eustice, R. M., Singh, H., Leonard, J. J., and Walter, M. R. (2006). Visually mapping the RMS Titanic: Conservative covariance estimates for slam information filters. *International Journal of Robotics Research*, 25(12):1223–1242. doi:10.1177/0278364906072512.

[Fairfield et al., 2007] Fairfield, N., Kantor, G., and Wettergreen, D. (2007). Real-time SLAM with octree evidence grids for exploration in underwater tunnels. *Journal of Field Robotics*, 24(1-2):3–21. doi:10.1002/rob.20165.

[Fischler and Bolles, 1981] Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395.

[Foley et al., 2009] Foley, B. P., Dellaporta, K., Sakellariou, D., Bingham, B. S., Camilli, R., Eustice, R. M., Evagelistis, D., Ferrini, V., Katsaros, K., Kourkoumelis, D., Mallios, A., Micha, P., Mindell, D., Roman, C., Singh, H., Switzer, D., and Theodoulou, T. (2009). The 2005 Chios ancient shipwreck survey: New methods for underwater archaeology. *Hesperia*, 78(2):269–305.

[Howard, 2008] Howard, A. (2008). Real-time stereo visual odometry for autonomous ground vehicles. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3946–3952. doi:10.1109/IROS.2008.4651147.

[Hurtós et al., 2010] Hurtós, N., Cufí, X., and Salvi, J. (2010). Calibration of optical camera coupled to acoustic multibeam for underwater 3D scene reconstruction. In *OCEANS 2010 IEEE - Sydney*, pages 1–7. doi:10.1109/OCEANSSYD.2010.5603907.

[Imagenex DeltaT, 2012] Imagenex DeltaT spec sheet [online]. (2012). URL: http://www.imagenex.com/Delta_T_6000_Profiling_Specs_rev4.pdf [cited 2012-06-04].

[IXSEA Octans, 2012] IXSEA Octans spec sheet [online]. (2012). URL: http://www.ixsea.com/pdf/br-octans-2011-08-web.pdf [cited 2012-01-20].

[Johnson-Roberson et al., 2009] Johnson-Roberson, M., Pizarro, O., and Willams, S. B. (2009). Towards large scale optical and acoustic sensor integration for visualization. In *OCEANS 2009 - EUROPE*, pages 1–4. doi:10.1109/OCEANSE.2009.5278237.

[Johnson-Roberson et al., 2010] Johnson-Roberson, M., Pizarro, O., Williams, S. B., and Mahon, I. (2010). Generation and visualization of large-scale three-dimensional reconstructions from underwater robotic surveys. *Journal of Field Robotics*, 27(1):21–51. doi:10.1002/rob.20324.

[Kaeli et al., 2011] Kaeli, J. W., Singh, H., Murphy, C., and Kunz, C. (2011). Improving color correction for underwater image surveys. In *Proc. IEEE/MTS Oceans Conference and Exhibition*.

[Kaess et al., 2011] Kaess, M., Johannsson, H., and Leonard, J. iSAM: Incremental smoothing and mapping [online]. (2011). URL: http://people.csail.mit.edu/kaess/isam/ [cited 6/6/2011].

[Kaess et al., 2012] Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J., and Dellaert, F. (2012). iSAM2: Incremental smoothing and mapping using the Bayes tree. *International Journal of Robotics Research*, 31:217–236.

[Kaess et al., 2008] Kaess, M., Ranganathan, A., and Dellaert, F. (2008). iSAM: Incremental smoothing and mapping. *IEEE Transactions on Robotics*, 24(6):1365–1378. doi:10.1109/TRO.2008.2006706.

[Kim and Eustice, 2009] Kim, A. and Eustice, R. (2009). Pose-graph visual SLAM with geometric model selection for autonomous underwater ship hull inspection. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1559–1565. doi:10.1109/IROS.2009.5354132.

[Kümmerle et al., 2011] Kümmerle, R., Grisetti, G., Strasdat, H., Konolige, K., and Burgard, W. (2011). g^2o : A general framework for graph optimization. In *Proc. IEEE International Conference on Robotics and Automation*, Shanghai.

[Kunz, 2011] Kunz, C. (2011). *AUV Navigation and Mapping in Dynamic, Unstructured Environments*. PhD thesis, Massachusetts Institute of Technology and Woods Hole Oceanographic Institution.

[Kunz and Singh, 2010] Kunz, C. and Singh, H. (2010). Stereo self-calibration for seafloor mapping using AUVs. In *Autonomous Underwater Vehicles (AUV), 2010 IEEE/OES*, pages 1–7. doi:10.1109/AUV.2010.5779655.

[Li et al., 1997] Li, R., Li, H., Zou, W., Smith, R., and Curran, T. (1997). Quantitative photogrammetric analysis of digital underwater video imagery. *IEEE Journal of Oceanic Engineering*, 22(2):364–375. doi:10.1109/48.585955.

[Longuet-Higgins, 1981] Longuet-Higgins, H. C. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(5828):133–135. URL: http://dx.doi.org/10.1038/293133a0.

[Mahon et al., 2008] Mahon, I., Williams, S. B., Pizarro, O., and Johnson-Roberson, M. (2008). Efficient view-based SLAM using visual loop closures. *IEEE Transactions on Robotics*, 24(5):1002–1014. doi:10.1109/TRO.2008.2004888.

[Medagoda et al., 2011] Medagoda, L., Williams, S. B., Pizarro, O., and Jakuba, M. V. (2011). Water column current profile aided localisation combined with view-based SLAM for autonomous underwater vehicle navigation. In *Proc. IEEE International Conference on Robotics and Automation*, pages 3048–3055. doi:10.1109/ICRA.2011.5980141.

[Meer et al., 1991] Meer, P., Mintz, D., Rosenfeld, A., and Kim, D. Y. (1991). Robust regression methods for computer vision: A review. *International Journal of Computer Vision*, 6(1):59–70.

[Mikolajczyk et al., 2005] Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., and Van Gool, L. (2005). A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1/2):43–72. URL: http://hal.inria.fr/inria-00548528, doi:10.1007/s11263-005-3848-x.

[Negahdaripour, 2007] Negahdaripour, S. (2007). Epipolar geometry of opti-acoustic stereo imaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(10):1776–1788. doi:10.1109/TPAMI.2007.1092.

[Negahdaripour and Madjidi, 2003] Negahdaripour, S. and Madjidi, H. (2003). Stereovision imaging on submersible platforms for 3-D mapping of benthic habitats and sea-floor structures. *IEEE Journal of Oceanic Engineering*, 28(4):625–650.

[Negahdaripour et al., 2009] Negahdaripour, S., Sekkati, H., and Pirsiavash, H. (2009). Opti-acoustic stereo imaging: On system calibration and 3-D target reconstruction. *IEEE Transactions on Image Processing*, 18(6):1203–1214. doi:10.1109/TIP.2009.2013081.

[Nicosevici et al., 2009] Nicosevici, T., Gracias, N., Negahdaripour, S., and Garcia, R. (2009). Efficient three-dimensional scene modeling and mosaicing. *Journal of Field Robotics*, 26(10):759–788.

[Nistér, 2004] Nistér, D. (2004). An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–770. doi:10.1109/TPAMI.2004.17.

[Olson et al., 2003] Olson, C. F., Matthies, L. H., Schoppers, M., and Maimone, M. W. (2003). Rover navigation using stereo ego-motion. *Robotics and Autonomous Systems*, 43(4):215–229.

[Paroscientific, 2012] Paroscientific Digiquartz spec sheet [online]. (2012). URL: http://www.paroscientific.com/pdf/3000&4000.pdf [cited 2012-01-20].

[Pizarro, 2004] Pizarro, O. (2004). *Large Scale Structure from Motion for Autonomous Underwater Vehicle Surveys*. PhD thesis, MIT / Woods Hole Oceanographic Institution.

[Pizarro and Singh, 2003] Pizarro, O. and Singh, H. (2003). Toward large-area mosaicing for underwater scientific applications. *IEEE Journal of Oceanic Engineering*, 28(4).

[Ribas et al., 2008] Ribas, D., Ridao, P., Tardós, J. D., and Neira, J. (2008). Underwater SLAM in man-made structured environments. *Journal of Field Robotics*, 25(11-12):898–921.

[Roman et al., 2010] Roman, C., Inglis, G., and Rutter, J. (2010). Application of structured light imaging for high resolution mapping of underwater archaeological sites. In *OCEANS 2010 IEEE - Sydney*, pages 1–9. doi:10.1109/OCEANSSYD.2010.5603672.

[Roman and Singh, 2007] Roman, C. and Singh, H. (2007). A self-consistent bathymetric mapping algorithm. *Journal of Field Robotics*, 24(1-2):23–50. doi:10.1002/rob.20164.

[Sastre-Córdova, 2009] Sastre-Córdova, M. M. (2009). Backscattering of sound from salinity fluctuations: Measurements off a coastal river estuary. In *OCEANS 2009 - EUROPE*, pages 1–6. doi:10.1109/OCEANSE.2009.5278339.

[Singh et al., 2004] Singh, H., Can, A., Eustice, R., Lerner, S., McPhee, N., Pizarro, O., and Roman, C. (2004). Seabed AUV offers new platform for high-resolution imaging. *EOS, Transactions of the AGU*, 85(31):289,294–295.

[Singh et al., 2002] Singh, H., Salgian, G., Eustice, R., and Mandelbaum, R. (2002). Sensor fusion of structure-from-motion, bathymetric 3D, and beacon-based navigation modalities. In *Proc. IEEE International Conference on Robotics and Automation*, volume 4, pages 4024–4031. doi:10.1109/ROBOT.2002.1014366.

[Singh et al., 2000] Singh, H., Whitcomb, L., Yoerger, D., and Pizarro, O. (2000). Microbathymetric mapping from underwater vehicles in the deep ocean. *Computer Vision and Image Understanding*, 79(1):143–161.

[Smith et al., 1990] Smith, R., Self, M., and Cheeseman, P. (1990). *Estimating uncertain spatial relationships in robotics*, pages 167–193. Springer-Verlag New York, Inc., New York, NY, USA.

[Strasdat et al., 2010] Strasdat, H., Montiel, J. M. M., and Davison, A. J. (2010). Scale drift-aware large scale monocular SLAM. In *Proceedings of Robotics: Science and Systems*, Zaragoza, Spain.

[Thrun et al., 2005] Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics*. Intelligent Robotics and Autonomous Agents Series. The MIT Press.

[Treibitz et al., 2012] Treibitz, T., Schechner, Y., Kunz, C., and Singh, H. (2012). Flat refractive geometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(1):51–65. doi:10.1109/TPAMI.2011.105.

[Vasilescu et al., 2010] Vasilescu, I., Detweiler, C., and Rus, D. (2010). Color-accurate underwater imaging using perceptual adaptive illumination. In *Proceedings of Robotics: Science and Systems*, Zaragoza, Spain.

[Walter et al., 2008] Walter, M., Hover, F., and Leonard, J. (2008). SLAM for ship hull inspection using exactly sparse extended information filters. In *Proc. IEEE International Conference on Robotics and Automation*, pages 1463–1470. doi:10.1109/ROBOT.2008.4543408.

[Williams, 2012] Williams, S. B. (2012). personal communication.

[Yoerger et al., 1999] Yoerger, D., Bradley, A., Cormier, M., Ryan, W., and Walden, B. (1999). High resolution mapping of a fast spreading mid ocean ridge with the autonomous benthic explorer. In *Proc. 11th International Symposium on Unmanned Untethered Submersible Technology*.

[Yoerger et al., 2007] Yoerger, D. R., Jakuba, M., Bradley, A. M., and Bingham, B. (2007). Techniques for deep sea near bottom survey using an autonomous underwater vehicle. *International Journal of Robotics Research*, 26(1):41–54. doi:10.1177/0278364907073773.

[Zhang, 1994] Zhang, Z. (1994). Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152.