Chapter 8
Simultaneous Localization and Mapping in Marine Environments

Maurice F. Fallon*, Hordur Johannsson, Michael Kaess, John Folkesson, Hunter McClelland, Brendan J. Englot, Franz S. Hover and John J. Leonard

Abstract Accurate navigation is a fundamental requirement for robotic systems—marine and terrestrial. For an intelligent autonomous system to interact effectively and safely with its environment, it needs to accurately perceive its surroundings. While traditional dead-reckoning filtering can achieve extremely high performance, the localization accuracy decays monotonically with distance traveled. Other approaches (such as external beacons) can help; nonetheless, the typical prerogative is to remain at a safe distance and to avoid engaging with the environment. In this chapter we discuss alternative approaches which utilize onboard sensors so that the robot can estimate the location of sensed objects and use these observations to improve its own navigation as well its perception of the environment. This approach allows for meaningful interaction and autonomy. Three motivating autonomous underwater vehicle (AUV) applications are outlined herein. The first fuses external range sensing with relative sonar measurements. The second application localizes relative to a prior map so as to revisit a specific feature, while the third builds an accurate model of an underwater structure which is consistent and complete. In particular we demonstrate that each approach can be abstracted to a core problem of incremental estimation within a sparse graph of the AUV's trajectory and the locations of features of interest which can be updated and optimized in real time on board the AUV.

Maurice F. Fallon · Hordur Johannsson · Michael Kaess · John Folkesson · Hunter McClelland · Brendan J. Englot · Franz S. Hover · John J. Leonard
Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA.
e-mail: mfallon@mit.edu,kaess@mit.edu,hordurj@mit.edu,johnfolk@mit.edu,huntermc@mit.edu,benglot@mit.edu,hover@mit.edu,jleonard@mit.edu

Mae L. Seto et al., *Marine Robot Autonomy*, DOI 10.1007/978-1-4614-5658-2_6,

8.1 Introduction

In this chapter we consider the problem of simultaneous localization and mapping (SLAM) from a marine perspective. Through three motivating applications, we demonstrate that a large class of autonomous underwater vehicle (AUV) missions can be generalized to an underlying set of measurement constraints which can then be solved using a core pose graph SLAM optimization algorithm known as incremental smoothing and mapping (iSAM) [40].

Good positioning information is essential for the safe execution of an AUV mission and for effective interpretation of the data acquired by the AUV [26, 47]. Traditional methods for AUV navigation suffer several shortcomings. Dead reckoning and inertial navigation systems (INS) are subject to external disturbances and uncorrectable drift. Measurements from Doppler velocity loggers can be used to achieve higher precision, but position error still grows without bound. To achieve bounded errors, current AUV systems rely on networks of acoustic transponders or surfacing for GPS resets, which can be impractical or undesirable for many missions of interest.

The goal of SLAM is to enable an AUV to build a map of an unknown environment and concurrently use that map for positioning. SLAM has the potential to enable long-term missions with bounded navigation errors without reliance on acoustic beacons, *a priori* maps, or surfacing for GPS resets. Autonomous mapping and navigation is difficult in the marine environment because of the combination of sensor noise, data association ambiguity, navigation error, and modeling uncertainty. Considerable progress has been made in the past 10 years, with new insights into the structure of the problem and new approaches that have provided compelling experimental demonstrations.

To perform many AUV missions of interest, such as mine neutralization and ship hull inspection, it is not sufficient to determine the vehicle's trajectory in ~~post processing~~ post-processing after the mission has been completed. Instead, mission requirements dictate that a solution is computed in ~~real-time,~~ real time to enable closed-loop position control of the vehicle. This requires solving an ~~ever growing~~ ever-growing optimization problem incrementally by only updating quantities that actually change instead of recomputing the full solution—a task for which iSAM is well suited.

Each application presents a different aspect of smoothing-based SLAM:

- Smoothing as an alternative to filtering: the use of ~~non-traditional~~ nontraditional acoustic range measurements to improve AUV navigation [18] ~~;~~
- ~~Re-localizing~~ Relocalizing in an existing map: localizing and controlling an AUV using natural features using a forward looking sonar [24] ~~; and~~
- Loop closure used to bound error and uncertainty: combining AUV motion estimates with observations of features on a ship's hull to produce accurate hull reconstructions [35] ~~.~~

A common theme for all three applications is the use of pose graph representations and associated estimation algorithms that exploit the graphical model structure of the underlying problem.

First we will overview the evolution of the SLAM problem in the following section.

8.2 Simultaneous Localization and Mapping

The earliest work which envisaged robotic mapping within a probabilistic framework was the seminal paper by Smith *et al.* [69]. This work proposed using an extended Kalman filter (EKF) to estimate the first and second moments of the probability distribution of spatial relations derived from sensor measurements. Moutarlier and Chatila provided the first implementation of this type of algorithm with real data [55], using data from a scanning laser range finder mounted on a wheeled mobile robot operating indoors. The authors noted that the size of the state vector would need to grow linearly with the number of landmarks and that it was necessary to maintain the full correlation between all the variables being estimated~~, thus~~; thus, the algorithm scales quadratically with the number of landmarks [11].

The scalability problem was addressed by a number of authors. The sparse extended information filter (SEIF) by Thrun *et al.* [74] uses the information form of the EKF in combination with a sparsification method. One of the downfalls of that approach was that it resulted in ~~over-confident~~ overconfident estimates. These issues were addressed in the exactly sparse delayed-state filters (ESDFs) by Eustice et al. [14, 15] and later with the exactly sparse extended information filter (ESEIF) by Walter et al. [79].

Particle filters have also been used to address both the complexity and the data association problem. The estimates of the landmark locations become independent when conditioned on the vehicle trajectory. This fact was used by Montemerlo *et al.* [54] to implement FastSLAM. The main drawback of particle filters applied to the ~~high dimensional~~ high-dimensional trajectory estimation is particle depletion. In particular, when a robot completes a large exploration loop and decides upon a loop closure, only a small number of particles with independent tracks will be retained after any subsequent ~~re-sampling~~ resampling step.

In purely localization tasks (with static prior maps) particle filters have been successful. Monte Carlo localization allowed the Minerva robotic museum guide to operate for 44 km over 2 weeks [72]. More recently it has been used by Nuske *et al.* to localize an AUV relative to a marine structure using a camera [60], exploiting GPU-accelerated image formation to facilitate large particle sets.

Filtering approaches have some inherent disadvantages when applied to the SLAM problem: ~~Measurements~~ measurements are linearized only once based on the current state estimate—at the time the measurement is added. Further, it is difficult to apply delayed measurements or to revert a measurement once it has been applied to the filter. The Atlas framework by Bosse et al. [6] addresses these issues by

combining local ~~sub-maps~~ submaps and a nonlinear optimization to globally align the ~~sub-maps. Each sub-map~~ submaps. Each submap has its own local coordinate frame, so the linearization point cannot deviate as far from the true value as in the case of global parameterization.

8.2.1 *Pose Graph Optimization ~~using~~ Using Smoothing and Mapping*

As the field has evolved, full SLAM solutions [48, 73] have been explored to overcome the linearization errors that are the major source of ~~sub-optimality~~ suboptimality of filtering-based approaches. Full SLAM includes the complete trajectory into the estimation problem rather than just the most recent state. This has led to the SLAM problem being modeled as a graph where the nodes represent the vehicle poses and optionally also landmarks. The edges in this graph are measurements that put constraints between these variables. By associating probability distributions to the constraints, the graph can be interpreted as a Bayes network.

Under the assumption that measurements are corrupted by zero-mean Gaussian noise, the maximum likelihood solution of the joint probability distribution is found by solving a nonlinear ~~least-squares~~ least squares problem. Many iterative solutions to the SLAM problem have been presented, such as stochastic gradient descent [29, 61], relaxation [10], preconditioned conjugate gradient [44], and loopy belief propagation [64].

Faster convergence is provided by direct methods that are based on matrix factorization. Dellaert and Kaess [9] introduced the square root smoothing and mapping (SAM) algorithm ~~,~~using matrix factorization to solve the normal equations of the sparse ~~least-squares~~ least squares problem. Efficiency is achieved by relating the graphical model to a sparse matrix in combination with variable reordering for maintaining sparsity. Similar methods are used by [45, 46], and more efficient approximate solutions include [28].

The aforementioned incremental smoothing and mapping algorithm provides an efficient incremental solution [40]. In iSAM the matrix factorization is incrementally updated using Givens rotations, making the method better suited for ~~on-line~~ online operations. In addition they developed an efficient algorithm for recovering parts of the covariance matrix [37], which is useful for ~~on-line~~ online data association decisions.

Recently, further exploration of the connection between graphical models and linear algebra allowed a fully incremental formulation of iSAM. The Bayes tree data structure [38] can be considered as an intermediate representation between the Cholesky factor and a junction tree. While not obvious in the matrix formulation, the Bayes tree allows a fully incremental algorithm, with incremental variable

Fig. 8.1 Factor graph for the pose graph formulation of the SLAM problem. The ~~large circles~~ *large circles* are variable nodes, here the AUV states x_i. The ~~small solid circles~~ *small solid circles* are factor nodes: relative pose measurements u_i, absolute pose measurements ψ_i, a prior on the first pose p_0, and loop closure constraints c_j

~~re-ordering and fluid re-linearization~~reordering and fluid relinearization. The resulting sparse nonlinear ~~least-squares~~ least squares solver is called iSAM2 [39].

Using a nonlinear solver for the full SLAM problem overcomes the problems caused by linearization errors in filtering methods, and it is also the case that estimation of the full trajectory results in a sparse estimation problem [9]. It is not necessary to explicitly store the correlation between all the landmarks, making these methods very efficient. One downside is that the problem grows with time (or at least distance traveled) instead of the size of the environment, although the rate of growth is not significant for the applications discussed in this chapter.

8.2.2 Mathematical Summary

In this section, we will briefly present the mathematical formulation of the full SLAM problem as a nonlinear least squares optimization. The full SLAM problem can be described as a constantly growing factor graph. A factor graph is a bipartite graph consisting of variable nodes and factor nodes, connected by edges. The factor graph represents a factorization of a function $f(X)$ over some variables $X = \{x_i\}_{i=0}^{N}$:

$$f(X) = \prod_{k=1}^{K} f_k(X_k), \tag{8.1}$$

where X_k denotes the subset of variables involved in the kth factor. The factor nodes $F = \{f_k\}_{k=1}^{K}$ represent constraints involving one or more variables. Each edge connects one factor node with one variable node.

For our navigation setting, consider the simple factor graph example in Fig. 8.1, where the variable nodes $x_1 \ldots x_N$ represent the vehicle states sampled at discrete times, together forming the vehicle trajectory. Here, the factor nodes F are partitioned into multiple types that represent relative pose constraints u_i between

consecutive poses, absolute pose constraints ψ_i on individual poses, and loop closure constraints c_j on arbitrary pairs of poses form these measurements.

When assuming Gaussian measurement noise, we arrive at a *nonlinear ~~least-squares~~least squares* problem. Under the Gaussian assumption, a measurement z_k is predicted based on the current estimate X_k through a deterministic function h_k and with added zero-mean Gaussian measurement noise v_k with covariance Λ_k:

$$z_k = h_k(X_k) + v_k \qquad v_k \sim \mathcal{N}(0, \Lambda_k). \tag{8.2}$$

Hence, the factor f_k to encode the actual measurement z_k is defined as

$$f_k(X_k) \propto \exp\left(-\frac{1}{2}\|h_k(X_k) - z_k\|^2_{\Lambda_k}\right), \tag{8.3}$$

where $\|x\|^2_\Sigma := x^\top \Sigma^{-1} x$. To find the nonlinear ~~least-squares~~ least squares solution $\hat{X}$ we make use of the monotonicity of the logarithm function for converting the factorization into a sum of terms:

$$\hat{X} = \arg\max_X \prod_{k=1}^{K} f_k(X_k) \tag{8.4}$$

$$= \arg\min_X -\log \prod_{k=1}^{K} f_k(X_k) \tag{8.5}$$

$$= \arg\min_X \sum_{k=1}^{K} -\log f_k(X_k) \tag{8.6}$$

$$= \arg\min_X \sum_{i=k}^{K} \|h_k(X_k) - z_k\|^2_{\Lambda_k}. \tag{8.7}$$

Standard Gauss-Newton [27]~~based~~ -based solutions, such as Levenberg-Marquardt or Powell's dog leg, repeatedly linearize and solve this sparse nonlinear ~~least-squares~~ least squares problem. By stacking the linearized equations, a sparse matrix A is obtained whose block structure mirrors the structure of the factor graph

$$\delta\hat{X} = \arg\min_{\delta X} \|A\delta X - b\|^2 \tag{8.8}$$

The vector b contains the measurements and residuals; details are given in [9]. This linear system can be solved by matrix factorization and ~~forward- and backsubstitution~~forward and back substitution. After each iteration the current estimate is updated by $\hat{X} \leftarrow \hat{X} + \delta\hat{X}$. The new estimate is then used as new linearization point, and the process is iterated until convergence.

iSAM [39, 40] provides an incremental solution to Gauss-Newton style methods, in particular Powell's dog leg [66]. When new measurements are received,

this approach updates the existing matrix factorization rather than ~~re-calculating~~ recalculating the nonlinear least squares system anew each iteration. For a detailed account of this process, the reader is referred to the original papers.

8.2.3 Data Association

A fundamental problem in feature-based SLAM is the correct association of point measurements from different time steps to one another. Given a series of raw laser, camera, or sonar measurements, the challenge is to identify the observed features which originated from the same physical entity. Knowledge of this data association provides a set of valid measurement constraints. As explained previously, these constraints can be optimized efficiently~~, however~~; however, this data association problem must first be solved.

Data association in its most generalized form is a ~~well-studied~~ well-studied problem, for example [59]. Where the measurements are indistinct, noisy, or contradictory, there remains the possibility of association errors. A core weakness of current SLAM approaches is brittleness and ~~sub-optimality~~ suboptimality resulting from these errors becoming ~~'baked into'~~ "baked into" the optimization problem. Currently, the predominant approach is to avoid adding such associations if not absolutely confident in their correctness—instead assuming access to informative sensor data at a later time. That is the approach we are taking for ship hull inspection in Sect. 8.6, where navigation uncertainty of the ~~on-board~~ onboard sensors is low, allowing for many minutes of ~~open-loop~~ open-loop navigation without significant loss of accuracy.

Discarding uninformative sensor information unfortunately is not a luxury available in many AUV applications in which interesting features are often rare. While approaches which maintain multiple data association ~~hypothesizes~~ hypotheses for an extended time have been proposed, the exponential growth in the size of a hypothesis tree cannot be supported indefinitely. In Sect. 8.5 we present an application which tackles this problem in a typical marine environment for a ~~low-cost~~ low-cost AUV with significant navigation uncertainty. Data association decisions are taken just after a feature has left the field of view so as to have access to all available observations of a particular feature before making the critical association decision.

AQ: Please check if edit to sentence starting "While approaches which maintain..." is okay.

While a detailed discussion of the field of data association is outside the scope of this work, it remains a problem specific to each problem or application.

8.3 Navigation in Marine Environments

In the following sections we will motivate the use of the smoothing and mapping approach by way of three separate autonomous marine applications. In particular, we will demonstrate that the estimation problem at the heart of each application can be reduced to a set of navigation and perception constraints which can be optimally, incrementally, and efficiently solved using the iSAM algorithm.

First we will give a more general overview of SLAM in marine environments. The modern AUV contains proprioceptive sensors such as compasses, ~~fiber optic~~ fiber-optic gyroscopes (FOG), and Doppler velocity loggers (DVL) [83]. The sensor output of these ~~senors~~ sensors is fused together using navigation filters, such as the EKF, to produce a ~~high quality~~ high-quality estimate of the AUV position and uncertainty. This estimate is then used by the AUV to inform ~~on-board decision making~~ onboard decision-making logic and to adaptively complete complex survey and security missions. Kinsey et al. provides survey of state-of-the-art approaches to AUV navigation [43].

Acoustic ranging has been widely used to contribute to AUV navigation [85,86]. Long baseline (LBL) navigation was initially developed in the ~~1970's~~ 1970s [31,34] and is commonly used by industrial practitioners [52]. It requires the installation of stationary beacons at known locations surrounding the area of interest which measure round-trip acoustic time of flight before triangulating for ~~3D~~ 3-D position estimation. Operating areas are typically restricted to a few square kilometers.

~~Ultra short~~ Ultrashort baseline (USBL) navigation [49] is an alternative method which is typically used for tracking an underwater vehicle's position from a surface ship. Range is measured via time of flight to a single beacon, while bearing is estimated using an array of hydrophones on the surface vehicle transducer. Overall position accuracy is dependent on many factors, including the range of the vehicle from the surface ship, the motion of the surface ship, and acoustic propagation conditions.

In addition, many modern AUVs have multiple exteroceptive sensors. Side-scan sonar, initially developed by the US Navy, has been widely used for ship, ROV, and AUV survey since its invention in the 1950s. More recently, forward looking sonars, with the ability to accurately position a field of features in two dimensions, have also been deployed for a variety of applications such as 3-D reconstruction [32] and harbor security [13,41,50,65]. In scenarios in which water turbidity is not excessively high, cameras have been used to produce accurate maps of ~~ship-wrecks~~ shipwrecks and underwater historical structures, for example, the mapping of RMS Titanic [16] and of Iron Age shipwrecks [3].

These more recent applications have a common aspect; to maintain consistency of sensor measurements over the duration of an experiment, smoothing ~~on-line~~ online of an AUV's trajectory and the location of measured features ~~is~~ are necessary. We will now demonstrate how SLAM smoothing in a marine environment is applied in practice.

Fig. 8.2 Optimizing the entire set of vehicle poses and target observations facilitates explicit alignment of sonar mosaics and understanding of the motion of the AUV during the mission. This allows for reactive decision making in the water—as opposed to post-processing which is common currently. In this figure this optimization allows three different observations of a single target to be explicitly aligned

8.4 Smoothing: Cooperative Acoustic Navigation

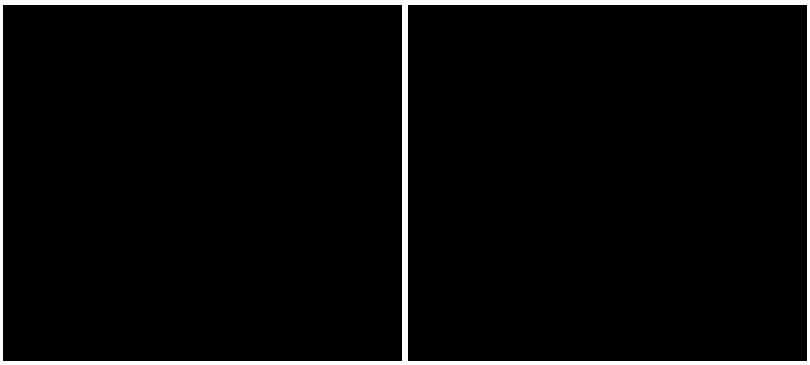
The first application we will consider is that of cooperative acoustic navigation. In this application ~~non-traditional~~ nontraditional sources of acoustic range measurements can be used to improve the navigation performance of a group of AUVs with aim of achieving bounded error or at the least reducing the frequency of GPS fix surfacings.

Within the context of the data association discussion in Sect. 8.2.3, this application is much simpler in that the acoustic range measurements are paired with the location of the surface beacon originating them—by design. This avoids data association entirely.

Following on from traditional LBL navigation, the moving long baseline (MLBL) concept proposed two mobile autonomous surface vehicles (ASVs) aiding an AUV using acoustic modem ranging. This was proposed by Vaganay et al. [76] and extended by Bahr et al. [1, 2]. This concept envisaged the ASVs transmitting acoustic modem messages containing their GPS positions paired with a modem-estimated range to the AUV which could then uniquely fix its position while maintaining full mobility—which is not afforded by typical LBL positioning.

More recent research has focused on utilizing only a single surface vehicle to support an AUV using a recursive state estimator such as the extended Kalman filter [19] or the distributed extended information filter (DEIF) [82].

For many robotic applications, however, estimating the vehicle's entire trajectory as well as the location of any observed features is important (~~for example~~ e.g., in survey missions). As mentioned previously, the EKF has been shown to provide an inconsistent SLAM solution due to information lost during the linearization step [36]. Furthermore, our previous work, [22], demonstrated (off-line) the superior performance of NLS methods in the acoustic ranging problem domain versus both an EKF and a particle filtering implementation—although requiring growing computational resources. For these reasons we present here an application in which iSAM is used for full pose trajectory estimation using acoustic range data.



a b

Fig. 8.3 ~~The vehicles used in our experiments:~~ **(a)** As the ~~Hydroid Remus 100~~ AUV ~~was supported by~~ travels through the ~~MIT Scout ASV or by~~ water, the ~~research vessel—the *Steel Slinger*~~side-scan sonar images laterally with objects on the ocean floor giving strong returns. **(b)** A top-down projection of the side-scan sonar for a 120 m of vehicle motion *(left to right)*. The lateral scale is 30 m in each direction which yields a 1:1 aspect ratio. Note that in this case targets 1 and 2 have been observed twice each after a turn

Additionally we demonstrate that mapping of bottom targets (identified in side-scan sonar imagery) can be integrated within the same optimization framework. The effect of this fusion is demonstrated in Fig. 8.2. This figure demonstrates the alignment of ~~sidescan~~ side-scan sonar mosaics from three separate observations of the same feature. Without optimizing the entire global set of constraints, the resultant data reprojection would be inconsistent.

As an extension, we demonstrate the ability to combine relative constraints across successive missions, enabling multi-session AUV navigation and mapping, in which data collected in previous missions is seamlessly integrated ~~on-line~~ online with data from the current mission ~~on-board~~ on board the AUV.

8.4.1 Problem Formulation

The full vehicle state is defined in three Cartesian and three rotation dimensions, $[x, y, z, \phi, \theta, \psi]$. Absolute measurements of the depth z, roll ϕ, and pitch θ ~~,~~are measured using a water pressure sensor and inertial sensors. This leaves three dimensions of the vehicle to be estimated in the horizontal plane: x, y, ψ.

The heading is instrumented directly using a compass, and this information is integrated with inertial velocity measurements to propagate estimates of the x and y

position[1] ~~.~~.[1] This integration is carried out at a high frequency (~ 10 Hz) compared to the exteroceptive range and sonar measurements (~ 1 z).

The motion of the vehicle at time step i is described by a Gaussian process model as follows:

$$u_i = h_u(x_{i-1}, x_i) + w_i \quad w_i \sim N(0, \Sigma_i) \tag{8.9}$$

where x_i represents the 3-D vehicle state (as distinct from the dimension x above). Note that while the heading is directly estimated using a compass, we use this estimate only as a prior within the smoothing framework. In this way the smoothed result will produce a more consistent combined solution.

8.4.1.1 Acoustic Ranging

Instead of either LBL or USBL, our work aims to utilize acoustic modems, such as the WHOI Micro-Modem [25], which are already installed on the majority of AUVs for command and control. The most accurate inter-vehicle ranging is through one-way ~~travel-time~~ travel-time ranging with precisely synchronized clocks, for example, using the design by Eustice [17], which also allows for broadcast ranging to any number of vehicles in the vicinity of the transmitting vehicle. An alternative is ~~round-trip~~ round-trip ranging, which, while resulting in more complexity during operation and higher variance, requires no modification of existing vehicles.

Regardless of the ranging method, the range measurement $r_{j,3D}$, a 2-D estimate of the position of the transmitting beacon, $g_j = [x_{gj}, y_{gj}]$, and associated covariances will be made known to the AUV at intervals on the order of 10–120 seconds. Having transformed the range to a 2-D range over ground r_j (using the directly instrumented depth), a measurement model can be defined as follows:

$$r_j = h_r(x_j, b_j) + \mu_j \quad \mu_j \sim N(0, \Xi_j) \tag{8.10}$$

where x_j represents the position of AUV state at that time. GPS measurements of the beacon position are assumed to be distributed via a normal distribution represented by Φ_j.

Comparing the ~~on-board~~ onboard position estimates of the AUV and the ASV in the experiments in Sect. 8.4.2, ~~round-trip~~ round-trip ranging is estimated to have a variance of approximately 7 m, compared with a variance of 3 m for one-way ranging reported in [22]. An additional issue is that with the ranging measurement occurring as much as 10 s before the position and range are transmitted to the AUV, an acausal update of the vehicle position estimate is required.

---

[1] ~~In our case this integration is carried out on a separate proprietary vehicle control computer and the result is passed to the payload computer.~~

[1] In our case this integration is carried out on a separate proprietary vehicle control computer, and the result is passed to the payload computer.

The operational framework used by Webster *et al.* [81, 82] is quite similar to ours. Their approach is based on a decentralized estimation algorithm that jointly estimates both the AUV position and that of a supporting research vessel using a distributed extended information filter. Incremental updates of the surface vehicle's position are integrated into the AUV-based portion of the filter via a simple and compact addition which, it is assumed, can be packaged within a single modem data packet.

This precise approach hypothesizes the use of a surface vehicle equipped with a high accuracy ~~gyro-compass~~ gyrocompass and a survey-grade GPS (order of 0.5 m accuracy). Furthermore, as described in [81], the approach can be vulnerable to packet loss, resulting in missing incremental updates which would cause the navigation algorithm to fail. While ~~re-broadcasting~~ rebroadcasting strategies to correct for such a failure could be envisaged, it is likely that significant (scarce) bandwidth would be sacrificed, making multi-vehicle operations difficult.

Our approach instead aims to provide independent surface measurements to the AUV in a manner that is robust to inevitable acoustic modem packet loss. The goal is a flexible and scalable approach that fully exploits the one-way ~~travel time~~ travel-time ranging data that the acoustic modems enable. The solution should be applicable to situations in which only low-cost GPS sensors are available on the ASVs or gateway buoys ~~,~~ to provide maximum flexibility.

8.4.1.2 Side-Scan Sonar

To demonstrate the compatibility of this approach with typical side-scan sonar surveys, the algorithm was extended to support relative observations from the sonar in a SLAM framework.

Side-scan sonar is a common sonar sensor often used for ocean sea-floor mapping. As the name suggests, the sonar transducer device scans laterally when towed behind a ship or flown attached to an AUV through the water column. A series of acoustic pings are transmitted, and the amplitude and timing of the returns combined with speed of sound in water ~~is~~ are used to determine the existence of features located perpendicular to the direction of motion.

~~**(a)** As the AUV travels through the water the side-scan sonar images laterally with objects on the ocean floor giving strong returns. **(b)** A top down projection of the side-scan sonar for a 120m of vehicle motion (left to right). The lateral scale is 30m in each direction which yields a 1:1 aspect ratio. Note that in this case Targets 1 and 2 have been observed twice each after a turn~~

By the motion of the transducer through the water column, two-dimensional images can be produced which survey the ocean floor and features on it. See Fig. 8.3 for an example side-scan sonar image. These images, while seemingly indicative of what exists on the ocean floor, contain no localization information to register them with either a relative or global position. Also it is often difficult to repeatedly detect and recognize features of interest~~,~~ ; for example, Fig. 8.3 illustrates two observations each of two different targets of interest. Target 1 (a metallic icosahedron) appears

AQ: "Figures 8.3, 8.4 and 8.5" have been renumbered to appear in sequence. Please check.

differently in its two observations. Also ~~,~~targets are typically not identified using the returned echoes from the target itself ~~,~~but by the shadow cast by the target [12].

For these reasons we must be careful in choosing side-scan sonar features for loop closure. Appearance-based matching techniques, such as FABMAP [8], would most likely encounter difficulties with acoustic imagery. Metric-based feature matching requires access to accurate, fully optimized position and uncertainty estimates of the new target relative to all previously observed candidate features. For these reasons, we will use iSAM to optimize the position and uncertainty of the entire vehicle trajectory, the sonar target positions, as well as all the beacon range estimates mentioned in Sect. 8.4.1.1.

The geometry of the side-scan sonar target positioning is illustrated in Fig. 8.3. Distance from the side-scan sonar to a feature corresponds to the *slant range*, $d_{m,3D}$, while the distance of the AUV off the ocean floor (*altitude*, a_m) can be instrumented. We will assume the ocean floor to be locally flat which allows the slant range to be converted into the horizontal range, resulting in the following relative position measurement:

$$d_{m,2D} = \sqrt{d_{m,3D}^2 - a_m^2} \tag{8.11}$$

$$\rho_m = \pm\pi/2 \tag{8.12}$$

where Ψ_m is the relative bearing to the target defined from the front of the vehicle ~~anti-clockwise~~anticlockwise. These two measurements paired together give a relative position constraint, $z_m = [d_{m,2D}, \rho_m]$ for an observation of target s_m. This target can either be a new, previously unseen target or a ~~re-observation~~ reobservation of an older target. In the experiments in Sect. 8.4.2 this data association is done manually, while in future work we will aim to do this automatically as in [70]. The resultant measurement model will be as follows:

$$z_m = h_z(x_m, s_m) + v_m \quad v_m \sim N(0, \Lambda_m) \tag{8.13}$$

where x_m is the pose of the AUV at that time. In effect, repeated observations of the same sonar target correspond to ~~loop-closures~~loop closures. Such repeated observations of the same location allow uncertainty to be bounded for the navigation between the observations.

8.4.1.3 Integration into the SAM Framework

Using the set of J acoustic ranges, M ~~sidescan~~ side-scan sonar constraints as well as the N incremental inertial navigation constraints, the optimization problem is formulated as follows:

Fig. 8.4 Factor graph formulation of the measurement system showing vehicle states x_i, surface beacons b_j, and sonar targets s_k. Also illustrated are the respective constraints: range r_j in the case of the surface beacons and range and relative bearing z_m in the case of sonar targets. Ranges are paired with surface beacon measurements, while multiple observations of a particular sonar target is in effect a loop closure. The initial pose is constrained using an initial prior p_0 using the GPS position estimate when the AUV dived

$$\begin{aligned}\hat{X} = \arg\min_{X} & \sum_{i=1}^{N} \|h_u(x_{i-1}, x_i) - \hat{u}_i\|^2_{\Sigma_i} \\ & + \sum_{j=1}^{J} \left\|b_j - \hat{g}_j\right\|^2_{\Phi_j} + \sum_{j=1}^{J} \left\|h_r(x_j, b_j) - \hat{r}_j\right\|^2_{\Xi_j} \\ & + \sum_{m=1}^{M} \|h_z(x_m, s_m) - \hat{z}_m\|^2_{\Lambda_m} \end{aligned} \tag{8.14}$$

In summary, x_j represents the vehicle pose when measuring the range r_j to beacon b_j, x_m is the pose when observing sonar target s_m at relative position z_m, and finally u_i is the control input between poses x_{i-1} and x_i. Unlike the simple derivation outlined in Sect. 8.2.2, the beacon and target positions require explicit insertion into the problem factor graph.

The corresponding factor graph is illustrated by Fig. 8.4.

8.4.2 Experiments

A series of experiments were carried out in St. Andrews Bay in Panama City, Florida to demonstrate this proposed approach. A Hydroid REMUS 100 AUV carried out four different missions while collecting side-scan sonar data (using a Marine ~~Sonics~~ Sonic transducer) as well as range and GPS position information transmitted from either the Scout ASV (Fig. 8.5) or a ~~deck-box~~ deck box on the 10 m support vessel. In each case, a low-cost Garmin 18x GPS ~~Sensor~~ sensor was used to provide GPS position estimates.

a

b

Fig. 8.5 The vehicles used in our experiments: the Hydroid REMUS 100 AUV was supported by the MIT Scout ASV or by the research vessel—the *Steel Slinger*

The Kearfott T16 INS, connected to the REMUS front-seat computer, fused its FOG measurements with those of a Teledyne RDI DVL, an accelerometer and a GPS sensor to produce excellent navigation performance. For example after a 40 min mission the AUV surfaced with a 2 m GPS ~~correction - drift~~ correction—drift of the order of 0.1 % of the distance traveled.

The AUV did not have the ability to carry out one-way ranging, and as a result, two-way ranging was used instead. The navigation estimate was made available to a *backseat* computer which ran an implementation of the algorithm in Sect. 8.2.2 (less the sonar portion).

Given the variance of two-way ranging ($\sim$7 m) and the accuracy of the vehicle INS, it would be ambitious to expect to demonstrate significant improvement using cooperative ranging-assisted navigation in this case. For this reason these missions primarily present an opportunity to validate and demonstrate the system with combined sensor input and multiple mission operation.

For simplicity, we will primarily focus on the longest mission—Mission 3 in Fig. 8.7—before discussing the extension to successive missions in Sect. 8.4.2.3. The missions are numbered chronologically.

8.4.2.1 Single Mission

During Mission 3, the AUV navigation data was combined with the acoustic range/position pairs and optimized ~~on-line on-board~~ online on board the AUV using iSAM to produce a real-time estimate of its position and uncertainty. After the experiments, sonar targets were manually extracted from the Marine ~~Sonics~~ Sonic data file and used in combination with the other navigation data to produce the combined optimization illustrated in Fig. 8.6. (The two remaining applications of this chapter describe ~~on-line~~ online algorithms for sonar processing.)

An overview of the mission is presented in Fig. 8.7 as well as quantitative results from the optimization where 3σ uncertainty was determined using $3\sqrt{\sigma_x^2 + \sigma_y^2}$.

Starting at (400, 250), the vehicle carried out a set of four *re-identification* (RID) patterns. These overlapping patterns are designed to provide multiple opportunities to observe objects on the ocean floor using the side-scan sonar. Typically this mission is carried out after having first coarsely surveyed the entire ocean floor. In this case two artificial targets were placed at the center of patterns 2 and 3 and were detected between 15–24 min (6 times) and 27–36 min (7 times), respectively. The surface beacon, in this case the support vessel on anchor at (400, 250), transmitted round-trip ranges to the AUV on a ~~20 second~~ 20-second cycle.

8.4.2.2 Analysis

A quantitative analysis of the approach is presented in Fig. 8.7. The typical case (black) of using only dead reckoning for navigation results in ~~ever increasing~~ ever-increasing uncertainty. The second approach (blue) utilizes target re-identifications in the sonar data but not acoustic range measurements. This temporarily halts the growth of uncertainty, but monotonic growth continues in their absence.

Acoustic ranging by comparison (red) can achieve bounded error navigation—in this case with a 3σ-bound of about 2 m. As the AUV's mission encircled the support vessel, sufficient observability was achieved to properly estimate the AUV's state—which results in the changing alignment of the uncertainty function. However performance deteriorates when the relative positions of the vehicles do not vary significantly (such as during patterns 3–4; 40–53 min).

Finally, the best performance is observed when the sonar and acoustic ranging data are fully fused. Interestingly, the two modalities complement each other: during *re-identification* patterns 2 and 3, sonar target observations bound the uncertainty while the AUV does not move relative to the support vessel. Later the vehicle transits between patterns—allowing for the range observability to improve.

In summary, the combination of the ~~on-board, sonar~~onboard, sonar, and ranging sensor measurements allows for ~~on-line~~ online navigation to be both globally bounded and locally drift-free.

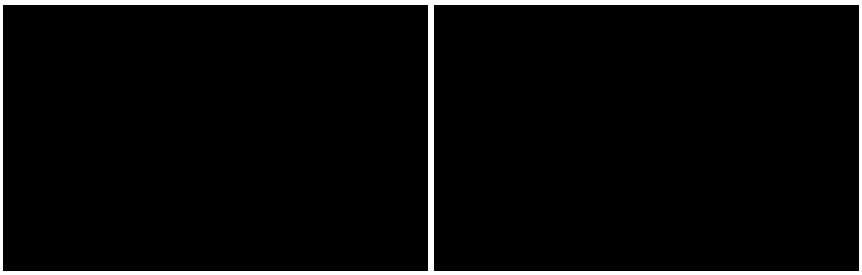


Fig. 8.6 An overview of the optimized trajectory estimates of the AUV (~~(~~*blue*~~)~~) and the surface vehicle (~~(~~*red*~~)~~), as well as the estimated position of three sonar targets (~~(~~*magenta*~~)~~) for two of the missions. The mutually observed feature in the ~~South-East~~ southeast allows for the joint optimization of the two missions. This corresponds to ~~Target~~ target 3 in Fig. 8.7. The ~~red lines~~ *red lines* indicate the relative vehicle positions during ranging, while the ~~ellipses~~ *ellipses* indicate position uncertainty

8.4.2.3 Multiple Missions

In this section we will describe how the algorithm has been extended to combine the maps produced by multiple successive AUV missions within a single optimization framework. As mentioned in previous sections, it is advantageous to provide a robot with as much prior information of its environment before it begins its mission, which it can then improve on as it navigates.

Space considerations do not permit a full analysis of this feature, but briefly: during Missions 1 and 2, surface information was transmitted from an autonomous surface vehicle, MIT's Scout kayak (shown in Fig. 8.5), which moved around the AUV so as to improve the observability of the AUV, as previously demonstrated in [22]. In Mission 4, as in Mission 3, the support vessel was instead used—although in this case, the support vessel moved from a location due east of the AUV to another location due west of the AUV, as illustrated in Fig. 8.6. This demonstrates that a basic maneuver by the support vessel is sufficient to ensure mission observability. The mission started at (350, 200).

Figure 8.7 illustrates the ~~inter-mission~~ intermission connectivity. This demonstrates that the two targets were observed numerous times during the missions, which allows us to combine the navigation across all of the missions into a single fully optimized estimate of the entire operation area.

While such an approach could possibly be carried out for several vehicles operating simultaneously, sharing minimal versions of their respective maps [21], it is unclear if the acoustic bandwidth available would be sufficient to share sonar target observation thumbnails to verify loop closure.

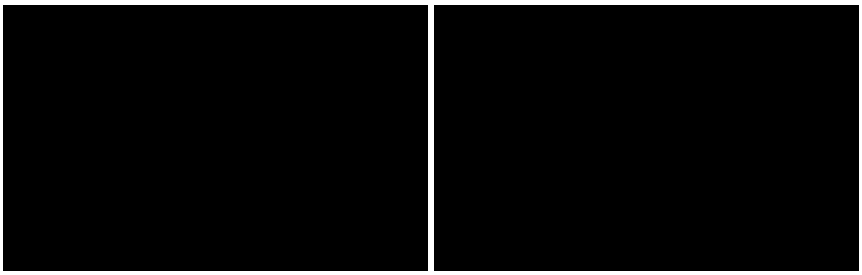


Fig. 8.7 ~~**(a):**~~ (a) Navigation uncertainty for Mission 3 for four different algorithm configurations. Acoustic ranging alone can bound error growth—subject to observability (~~(~~*red*~~);~~), while the full sonar and acoustic fusion produces the solution with minimum uncertainty (~~(~~*magenta*~~)~~). ~~**(b):**~~ (**b**) During the four (consecutive) missions, range measurements (represented by the ~~red lines~~*red lines*) were frequently received from the ASV (Mission 1 and 2) or the research vessel (Mission 3 and 4). Occasionally targets were detected in the side-scan sonar data. Repeated observations of the same target (illustrated in magenta) allow for a SLAM loop closure and for ~~inter-loop~~ interloop uncertainty to be bounded

8.4.3 Discussion

In this section we presented a method for the fusion of ~~on-board~~ onboard proprioceptive navigation and relative sonar observations with acoustic ranges transmitted from an autonomous surface vehicle. It allows for operation for many hours in ~~real-time~~ real time for missions of the type described above. Factors resulting in a reduction in performance of this approach are as follows: (1) infrequent ranging, (2) ranging from the same relative direction, and (3) sonar targets not being present or being infrequently observed. We estimate that the bounded error for a non-FOG enabled AUV with several percent drift would be of the order of 3–5 m (depending on the relative geometry and frequency of the one-way travel-time range measurements).

The specific acoustic ranging problem defined above is one problem in a wider class of problems each of which is defined by the connectivity of the fleet of vehicles and the direction of information flow (which result in inter-vehicle correlations being created). A recent overview of the various ~~sub-problems~~ subproblems is presented in [78].

8.5 Localization Using a Prior Map

In this second application we consider the challenge of using a prior map (generated using techniques described above) as part of a greater mission to neutralize mines in very shallow water—a task that has traditionally been carried out by human divers. The potential for casualties associated with this method of mine countermeasures

Fig. 8.8 iRobot Ranger—a ~~low cost~~ low-cost single-man portable AUV

(MCM) motivates the use of unmanned systems to replace human divers. While tethered robotic vehicle could be remotely controlled to perform MCM, a solution using untethered AUVs offers numerous advantages.

When mission requirements dictate that vehicle cost must be extremely low, the navigation problem for target reacquisition is quite challenging. The crux of the problem is to achieve good navigation performance despite the use of sensors with very low cost.

Resultantly the application unfolds within the context of a multiple-step effort, involving a variety of vehicles and technologies. The mission assumes a target field of moored and bottom mines along a shoreline. In this scenario, a remote environmental measuring unit (REMUS) AUV [77] (Fig. 8.5 a) performs a survey of the operating area, scouting the operating area, and collecting data using its ~~side scan~~ side-scan sonar. The REMUS data are used to create an a priori map of the underwater environment via processing software developed by SeeByte, Ltd. This a priori map consists of the locations of any strong sonar features in the target field.

~~Typical prior map generated using a REMUS 100 equipped with a marine-sonic side-scan sonar. A series of features were extracted by trained human operates from the side-scan sonar imagery to produce an a priorimap for the target reacquisition mission. The distance between the features is approximately 20m. Fig. courtesy of SeeByte, Ltd.~~

This map and the location of the feature of interest (FOI) acts as input to a second *low-cost* ~~re-localization~~ relocalization vehicle. In the mission scenario we aim to release this vehicle at a distance of 100 to 1,500 m from the center of the prior map and have it swim at the surface close to the feature field before diving to the seabed. Upon ~~re-entering~~ reentering this feature field, the vehicle will extract features from its sonar and use these features to build a map of the features.

Having ~~re-observed~~ reobserved a sufficient number of features, the AUV will localize relative to the a priori map and attach itself to the FOI. If successful, the AUV will self-detonate or place an explosive charge. Because of this the vehicle is not intended to be recoverable. For these reasons a low-cost vehicle design requirement has had significant impact on the SLAM algorithms mentioned here.

Overview of Vehicles Used

The vehicle used in development has been the iRobot Ranger AUV [67]. This vehicle was equipped with a depth sensor, altimeter, GPS receiver, a ~~3D~~ 3-D compass, an acoustic pinger, and a Blueview Proviewer 900 kHz forward looking sonar. The vehicle's design was intended to be low cost and light weight. As indicated by Fig. 8.8, it is single-man portable and deployable.

The design of the vehicle incorporates a propeller which is entirely ~~servo-ed~~servoed. This allows the vehicle to be highly maneuverable with a very tight turning radius of 0.5 m (compared with 10 m for the REMUS 100). This is of particular importance for the target homing at the end of the mission. The cruising speed of the AUV is quite low at about 0.6 m/s—comparable with typical surface currents. Thus, the dead-reckoning error due to the current can be quite significant. Given the small diameter of the vehicle, a processor smaller than the typical PC104 generation with limited capability ~~,~~ was used. This resulted in severe processing restrictions which are mentioned in subsequent sections.

The vehicle specifically did not have a DVL, used for precise velocity estimation due to cost regions. It would be remiss for us not to mention that the current range of FLS devices are comparable in price to a typical DVL~~, however~~; however, a significant proportion of this price represents the overhead cost of research and development. The manufacturer expects that mass production can reduce cost by an order of magnitude. Nonetheless the utility of the capabilities outlined herein go far beyond this particular application.

While the Hydroid REMUS 100 was primarily used as a survey vehicle (as discussed in Sect. 8.5.2), it was also used in several experiments demonstrated in Sect. 8.5.5.

Marine Vehicle Proprioception

At ~~high frequency~~ high-frequency depth estimates, altimeter altitudes, GPS fixes and compass estimates of roll, pitch and heading are fused with actuation values (orientation of the ~~servo-ed~~ servoed propeller and the estimated propeller RPM) using a typical EKF prediction filter to produce an estimate of the vehicle position and uncertainty at each time. In benign current-free conditions, with careful tuning and excellent compass calibration, this procedure produced a dead-reckoning estimate with about 1 % error per distance traveled.

However as we transitioned to more challenging current-prone conditions in later stages of the project (as discussed in Sec. 8.4.2), a current estimation model was developed so as to reject the vehicle's drift in this situation. (Because of the nature of this project, it is not possible to use the aforementioned DVL-enabled vehicle's estimate of the current profile~~).~~.) This module is designed to be run immediately prior to the mission as the vehicle approaches the target field.

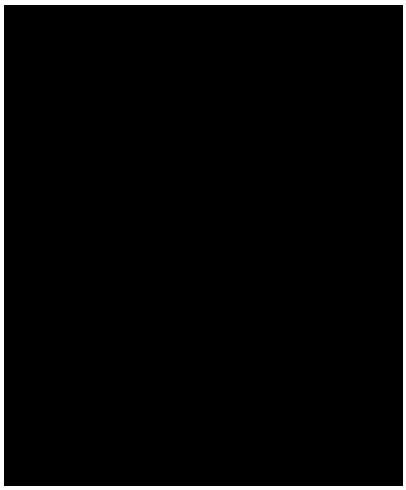


Fig. 8.9 The sonar image generation process for a single sonar beam. Each beam return is an vector of intensities of the returned sonar signal with objects of high density resulting in high returns and shadows resulting in lower intensities

This simplistic model performed reasonably well in smaller currents (below 0.3 m/s) and allowed the AUV to enter the field. After this, success was primarily due to the sonar-based SLAM algorithm (outlined in Sect. 8.5.2). In this current regime, we were able to enter the field approximately 85 % of the time using this model, and we estimate the error as about 5 % per distance traveled.

8.5.1 Forward ~~Look~~ Looking Sonar Processing

The sonar is our most important sensor allowing the AUV to perceive its environment. During the project a series of Blueview Proviewer FLS sonars were used. In this section we will give an overview of the sensor technology before presenting our sonar processing algorithms in Sect. 8.5.1.1.

The Proviewer FLS operates using Blazed Array technology [71]. Typically the sonar consisted to two transducer heads (horizontal and vertical) each with a field of view of 45°, although 90° and 135° units were later used.

An outgoing ensonifying signal (colloquially known as a ~~'ping'~~"ping") reflects off of objects of incidence (in particular metal and rock), and the phase, amplitude, and delay of the returned signals are processed to produce a pattern as indicated in Fig. 8.9 (by the manufacturer ~~Blueview~~BlueView). This return is evaluated for each array element with ~~one degree~~ one-degree resolution in the plane of the head, and the output is then fused together via digital signal processing to produce the image in Fig. 8.10.

AQ: "Figures 8.9–8.14" have been renumbered to appear in sequence. Please check.

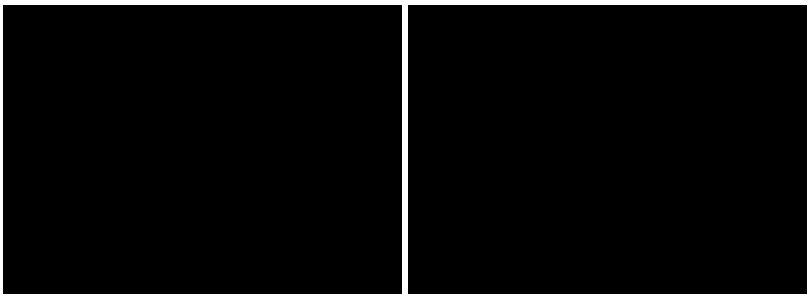


Fig. 8.10 Typical underwater camera and sonar images. The clear water and ~~well lit~~ well-lit scenario represents some of the best possible optical conditions~~,~~; nonetheless, visibility is only a few meters. This 90° Blazed Array sonar horizontal image indicates 3 features (one at 5 m in front; one at 20 m and 5° to the left and one at 35 m and 40° to the left)—which is more than typical

The outgoing sonar signal also has a significant lobe width, $\phi \sim 20°$, which means that there is significant ambiguity as to the location of the returning object in the axis off of the return. This distribution was used to estimate the elevation of detections using only the horizontal image.

8.5.1.1 Sonar Feature Detection

In this section we will outline our algorithms which extract point features from regions of high contrast. Forward looking sonar has previously been used for obstacle detection and path planning as in [63]~~,~~; in this application the feature extraction is focused on conservative estimation of all detected objects given the very noisy output of the FLS systems. Finally, [7] carried out multi-target tracking of multiple features from a FLS using a PHD filter.

Most normal objects are visible at 20 m, while very brightly reflective objects are detectable to 40 m. Adaptability to bottom reflective brightness was achieved by the ~~on-line~~ online estimation of an average background image immediately after the vehicle leveled out at its cruising depth. Estimating this background noise image was essential for us to achieve excellent performance in both sandy and muddy bottom types. Having done this, the features are detected based on gradients of the sonar image in each of four different sonar regions. The specific details of our feature detector and a quantitative analysis of its performance are available in [20, 23].

In terms of processing, the feature detector uses negligible processing power. The formation of the input image (using the Blueview SDK)—the input to this process—requires a substantial 180 ms, per image. The feature detector requires about 18 ms while the remaining CPU power is used to fuse the measurements, to make high level mission decisions and to control the AUV.

Fig. 8.11 As the robot explores, a pose (~~(~~*black*~~)~~) is added to the graph at each iteration, while feature detections (~~(~~*red*~~)~~) are also added to produce a dense trajectory. This dense trajectory is very large, so we periodically marginalize portions of the trajectory and the feature observations into composite measurements (~~(~~*green*~~)~~) at a much lower rate

8.5.2 Marine Mapping and Localization

As in the case of cooperative acoustic navigation, this application results in a series of constraints which can be optimized to best inform the AUV of its location relative to the map.

The complexity of the system of equations is tied to the sparseness of the A matrix which is itself dependent on the fill-in caused by loops in the graph structure. We explicitly avoid carrying out loop closures in this filter so as to maintain sparsity. All of this ensures that the matrices remain sparse and computation complexity predictable. Decomposition will not grow in complexity at each iteration, while the computational cost of back substitution will grow, but it is linear.

So as to avoid computational growth due to an ~~ever increasing~~ ever-increasing graph size and also to produce an input to the next estimation stage, we periodically rationalize the oldest measurements from this graph to form a *composite measurement*. To do this we marginalize out all the poses that have occurred during the previous (approximately) 10 s period to produce a single node for the relative motion for that period as well as nodes for fully detected features and the associated covariances. This approach is very similar in concept to ~~key-frames~~ key frames in vision SLAM and is illustrated in Fig. 8.11.

We time this marginalization step to occur after a feature has left the sonar field of view as this allows us to optimally estimate its relative location given all available information. This composite measurement is then added to a lower-frequency ~~higher level~~ higher-level graph. This ~~low–frequency~~ low-frequency graph is used as input to the prior ~~map-matching~~ map matching algorithm in Sect. 8.5.3. Meanwhile the ~~high frequency~~ high-frequency graph begins to grow again by the insertion of newer constraints into A_i.

An alternative approach would be to maintain the dense trajectory of the robot pose at all times. This is the approach taken by iSAM [40]~~, however~~; however, given the size of the resultant graph, we are not certain that such an approach would have been able to yield a computationally constant solution required for our low-powered embedded CPU.

Additionally and unlike most land-based systems, the underwater domain is characterized by extended periods where the seabed is featureless for long distances and the resultant composite measurement is simply the relative trajectory of the distance traveled.

8.5.2.1 Feature Tracking

While the section above explains how the graph of the trajectory and sonar observations is optimized and efficiently solved, we have not discussed the way in which sonar targets are proposed.

The sonar detector passes point extractions to a target nursery which maintains a vector of all recent detections. The nursery feature projects the detections into a local coordinate frame using the recent vehicle ~~dead-reckoning~~ dead reckoning and uses a probabilistic distance threshold to associate them with one another. Should a sufficiently large number of detections be clustered together (approximately 7–8 but dependent on the spread and intensity of detections), it is inferred that a consistent physical feature is present.

At this stage this nursery feature is added to the square root smoother. All of the relative AUV-to-point constraints for that feature are then optimized which results in improved estimation of the feature and the AUV trajectory. Subsequent point detections, inserted directly into the pose graph, result in an improved estimate via further square root smoothing. This approach also estimates the height/altitude of the sonar target using the sonar intensities measured at each iteration.

Finally it should be noted that the input to this feature tracker are point features characterized only by their location and covariance (due to the poor resolution of the sensor). This makes it difficult to robustly infer SLAM loop closures on the graph structure.

8.5.3 Global Estimation and Map Matching

Given this ~~high level~~ high-level graph of the robot trajectory and observed feature locations, it still remains for the autonomous system to make a critical judgment of where it is relative to the *a priori* map and to decide if this relative match is certain enough to be declared convincingly. To do this we maintain a set of match hypotheses in parallel. We compare them probabilistically so as to quantify the quality of the map match.

This comparison is implemented using a bank of estimators—each tracking a different match hypothesis in parallel. The relative likelihood of one match hypothesis over another is computed using positive information (of prior features detected by the sonar) as well as negative information (of prior features that were expected but undetected by the sonar), and in this way matching can be done in

a probabilistically rigorous manner. Simply put: *if one expects to detect features predicted to lie along the trajectory of a robot and these features were not seen, then the trajectory must be less likely.*

The inclusion of this extra information is motivated by the regular rows of feature in the field and the inability of positive information metrics to estimate the relative position of the AUV along these lines. The incorporation of negative information in this way is to, our knowledge, a novel contribution and was by motivated information not captured by algorithms such as joint compatibility branch and bound (JCBB) algorithm [59].

8.5.3.1 Negative and Positive Scoring

In SLAM, multi-hypothesis comparison can typically be reduced to a scoring algorithm of the relative probabilities of candidate solutions. Here we propose an algorithm for multi-hypothesis scoring which uses both positive as well as negative information which we name the negative and positive scale (NAPS). An early version of this concept was introduced in [23]. More details are provided in [20].

At time t, we define NAPS for hypothesis i as the ratio of the probability of its map matching hypothesis, $h_{i,t}$, compared to a null hypothesis, h_{null}, when both are conditioned on the measurements $z_{1:t}$

$$NAPS(h_{i,t}) = \ln\left(\frac{p(h_{i,t}|z_{1:t})}{p(h_{null,t}|z_{1:t})}\right) \tag{8.15}$$

We define a hypothesis as the combination of an estimate of the graph structure of the SLAM problem x_h (the vehicle trajectory and all detected features) as well as all data association matches of these features to map features in the prior map. The null hypothesis is a special version of this hypothesis in which no data associations exist and in which it is proposed that each detected features is a new feature independent of the map. We use it as normalization for maps of growing size.

Dropping reference to i for simplicity and using Bayes' rule gives

$$NAPS(h_t) = \ln\left(\frac{p(z_t|h_t)p(h_t)}{p(z_t|h_{null})p(h_{null})}\right) \tag{8.16}$$

We split $p(z_t|h)$ into two terms representing both negative and positive information

$$p(z_t|h) = \eta\, p(z_{pos}|h)p(z_{neg}|h) \tag{8.17}$$

Positive information is then defined, in the same way as for JCBB, as the likelihood of the measurements given the hypothesis

$$\begin{aligned} p(z_{t,pos}|h) &= \eta_{z,pos}\mathrm{e}^{-\frac{1}{2}(x_h - z_t)^{\mathrm{T}}\Sigma^{-1}(x_h - z_t)} \\ &= \eta_{z,pos}\mathrm{e}^{-D_h} \end{aligned}$$

where Σ represents the covariance, $\eta_{z,pos}$ is a normalization constant, and D_h is the Mahalanobis distance.

The term $p(h)$ represents a prior probability of a particular map hypothesis being created by the robot which we propose is directly related to the number of features N_f matched to the prior map is given by

$$p(h) = \eta_x e^{\lambda N_f} \tag{8.18}$$

where η_x is a normalization constant, λ is a free parameter, and N_f is an integer between zero and the total number of features in the prior map. While this formulation does not take into account aspects such as a target's measured visibility or other such specific terms, it does give us a measure of the confidence of a map match.

Combining these terms and canceling where possible gives give the following expressions for NAPS and as well as more common positive-only scoring (POS) metrics:

$$NAPS_t(h) = -D_h + \lambda N_f + C_{h,neg} \tag{8.19}$$

$$POS_t(h) = -D_h + \lambda N_f \tag{8.20}$$

This specifically indicates the contribution of negative information, $C_{h,neg}$, that we believe is neglected in typical multi-hypothesis scoring algorithms. POS algorithms implicitly assume $C_{h,neg} = 0$ and don't do not account for it in scoring the hypotheses. Most approaches assume very high very high λ: essentially selecting the hypotheses that match the most total features and then ordering those by Mahalanobis distance—as in the case of JCBB. A overview of such algorithms is presented in [58, 62].

8.5.4 Evaluating Negative Information

We define negative information as :

$$\begin{aligned} C_{h,neg} &= \ln\left(\frac{p(z_{t,neg}|h)}{p(z_{t,neg}|h_{null})}\right) \\ &= \ln(p(z_{t,neg}|h)) - \ln(p(z_{t,neg}|h_{null})) \end{aligned} \tag{8.21}$$

As each hypothesis NAPS score will eventually be compared to one another, the second term need not be calculated.

For a particular hypothesis, consider an entire vehicle trajectory and the sonar footprint that it traced out (such as in Fig. 8.12). Also consider a prior map feature which is located within this footprint but was not detected. We wish to measure the

number of times that this feature ought to have been detected, given that trajectory. NI is formed as the product of the probability of each undetected feature given the hypothesized vehicle trajectory

$$\begin{aligned} p(z_{t,neg}|h) &= p(z_{t,neg,f_1} \cap \ldots \cap z_{t,neg,f_{n_u}}|h) \\ &= \prod_{f \in N_u} p(z_{t,neg,f}|h) \\ &= \prod_{f \in N_u} \left(1 - p(z_{t,pos,f}|h)\right) \end{aligned} \tag{8.22}$$

where s_t is whole area sensed during measurement z_t~~, thus:~~ ; thus,

$$p(z_{t,pos,f}|h) = \int_{p(f) \cap p(s_t)} v_f p(f) p(s_t) \mathrm{d}A \tag{8.23}$$

where v_f is the visibility of feature f and $p(f)$ is the prior probability of that feature.

In words, the probability of not detecting each ~~conditionally-independent~~ conditionally independent feature is the product of one minus the probability of detecting each feature, integrated across the intersection of the PDF of each feature and the PDF of the scanned sensor area. This formulation is subject to the following assumptions: (1) the sensor occlusion model is well-defined and accurate, (2) all features are static, (3) feature detections are independent, and (4) feature visibility can be accurately modeled. This calculation, often intractable due to complicated integration limits, theoretically defines the probability of a set of negative measurements $z_{t,neg}$ given sensed area s_t.

More information about its precise evaluation is presented in [20]. The result of the metric is a positive value which scores a particular hypothesis more likely when its observations do not contradict the prior map.

In particular, combining negative information with the other (positive-only) metrics in Eq. 8.19 allowed us to disambiguate similar locations along a row of otherwise indistinguishable features, as indicated in Fig. 8.12.

While the AUV operated in the field this metric is evaluated for each hypothesis. The vehicle controls itself off of the most likely hypothesis: giving heading, speed, and depth commands to the low level vehicle controller so as to travel to a set of preprogrammed ~~way-points~~ waypoints in the field. When the metric for a particular hypothesis exceeds a threshold, it is decided that the AUV is matched to the prior map and switches to a final target capture mode.

When it approaches this location, the FOI should be observed in the sonar imagery. The mission controller then transitions to directly controlling using the sonar detections using a PID—which we call sonar servoing. It opens a pair of tines with a tip separation of approximately 1m and drives onto the mooring line of the FOI.

Fig. 8.12 Illustration of the effect of negative and positive scoring (NAPS). Consider the AUV trajectory from A to B with the sonar sensor footprint enclosed in ~~green~~*green*. If the AUV observes the ~~red~~ *red* feature, how do we match its trajectory to the prior map (~~purple squares~~*purple squares*)? Using JCBB, the observed feature will be matched equally well to either prior feature. However, using negative information, NAPS indicates that the match in the ~~lower figure~~ *lower figure* is more likely. The ~~upper figure~~ *upper figure* is less likely because we would have expected to have observed both features—but only observed one

8.5.5 Field Experiments

The system has undergone extensive testing and evolution over a number of years. Starting in November, 2006, we have conducted approximately 14 sea trials, each lasting 2 to 3 weeks. Our experiments began in fairly benign environments, using highly reflective moored objected as features, and progressed to more challenging conditions such as natural sea bottom targets and strong currents. After each trial we have refined and improved the system. In the following we summarize the progress of the development of the algorithms and the vehicle platform (Fig. 8.13).

AQ: Please check if inserted citation for Fig. 8.13 is correct.

Typically the ingress point/direction to the field was varied for each mission, while the choice of feature of interest was taken at random just before placing the AUV in the water. After reaching the field, the vehicle typically traveled along the rows of features indicated in Fig. 8.14. This was so as to keep the number of map match hypotheses low (to about 4–5). The typical mission duration was 15–25 min, although the mission planner could be programmed to repeat the mission if the AUV failed to find the feature field. A typical water depth was 15 m.

Detailed comparison of mission parameters is difficult as the effect of the vehicle's control decisions is that different paths and observations follow. For this reason, this section focuses on the progression of our core map matching algorithm.

St. Andrews Bay, Florida, June 2007: The NAPS and joint compatibility branch and bound (JCBB) criteria were alternately used over 18 trials on a field of strongly reflective moored targets. The JCBB implementation uses a threshold on the Mahalanobis distance for multiple pair matching and chooses the most compatible pairs. The results of this live test and selected other tests are summarized in Table 8.1. The difference in the frequencies is 1.41 standard deviations which

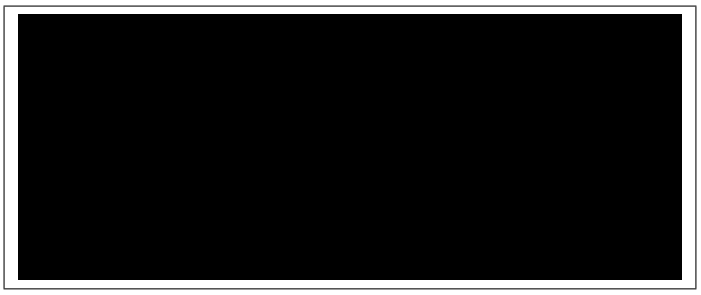


Fig. 8.13 A top-down overview of a successful mission using the accurate ~~REMUS 100~~ *REMUS 100* vehicle. The vehicle approached from the ~~north-west~~ northwest and extracted feature points (~~purple dots~~*purple dots*). Using these points and the prior map (~~(~~*blue squares*~~)~~), the SLAM map (~~black squares~~*black squares*) and the vehicle trajectory estimate (~~(~~*magenta line*~~)was~~) were formed. Having matched against the map, the vehicle homed to the feature of interest. The abrupt position changes are the result of the square root smoother. The scale of the grid is 10 m. It is important to note that the DVL-INS-enabled AUV ~~would have failed~~ *would have failed* to reacquire the FOI without using sonar as the map itself was only accurate to 5 m (~~(~~*blue line*~~)~~)

Fig. 8.14 Typical prior map generated using a REMUS 100 equipped with a Marine Sonic side-scan sonar. A series of features were extracted by trained human operates from the side-scan sonar imagery to produce an a priori map for the target reacquisition mission. The distance between the features is approximately 20 m (Fig. courtesy of SeeByte, Ltd.)

gives a 91 % significance. We believe this demonstrates that the NAPS outperforms the simpler JCBB matching criteria in our application.

Narragansett Bay, Rhode Island, June 2008: Using the data from June 2007, significant improvements to our sonar processing algorithms allowed for improved detection of man-made and natural bottom features. This includes the addition of an adaptive noise floor model discussed in Sect. 8.5.1.1 and a reimplementation in integer logic for increased efficiency. The field for these tests consisted of various man-made and naturally occurring objects on the sea bottom as well as moored targets. The bay had a significant tidal current comparable to the 0.5 m/s velocity of the vehicle, which gave us substantial dead-reckoning errors.

Of the nine runs, we attached to the target once and had two mechanical failures. In both cases the tine mechanism broke upon hitting the mine mooring line. Thus the overall success rate of the sonar navigation system was 33 %. After these tests the current model mentioned in Sect. 8.5 was developed.

Gulf of Mexico, near Panama City, Florida, June 2009: The entire system was tested on a field of 12 bottom objects and 3 moored objects over a ~~two week~~ two-week period. These experiments tested an improved model for current estimation along with minor adjustments to the feature modeling. The current during this period was estimated as typically being 0.2 m/s using GPS surfaces. We had 17 successful target attachments in 26 runs.

Gulf of Mexico, July 2010: An additional set of experiments were carried out. In this circumstance we observed much higher currents which changed significantly. These currents varied from day to day but were estimated to be greater than the vehicle velocity (greater than 0.5 m/s) on certain days meaning that the vehicle could not make any headway against the current when it found itself down field from the features.

Presented in Table 8.1 are two different results for this experiment. One result gives the overall percent success when including all of the 42 runs carried out: 31 %. Filtering the runs to the 18 runs in which the AUV was able to enter the field (as defined by at least a single feature detection in the sonar) produced a success percentage of 72 % which we believe is more in fitting with the performance of the SLAM algorithm and comparable to the previous year's results. Nonetheless, this demonstrates the limitation of this particular vehicle platform as well as current estimation without a direct sensor.

8.5.6 Discussion

This section described target reacquisition system for small low-cost AUVs, based on forward looking sonar-based SLAM aided by a prior map. Our results indicate that when the AUV correctly matches to the prior feature map, it is regularly able to ~~re-visit~~ revisit a designated feature of interest.

The main failure mode of the algorithm is failing to enter the feature field, due to disturbances that exceed the vehicle's control authority. For small to moderate ocean currents, we developed an ~~on-line~~ online current estimation procedure which allows the vehicle to avoid being driven off course during the initial vehicle dive. Room exists to improve this estimation procedure by estimation of the known current features mentioned in Sect. 8.5. Unsurprisingly in currents of more than 50–70 % of the vehicle's velocity, successful performance was limited. This presented an obvious engineering limitation for this technology.

While more research is necessary to understand the many variables that can effect the system performance, such as the density and complexity of environmental

Table 8.1 Selected results in different conditions, with and without use of NAPS

Match ~~Criteria~~ criteria	No. of ~~Runs~~ runs	Successes	Frequency (%)	$\sqrt{s_n^2/n}$ (%)
Bright targets - June 2007				
NAPS	9	6	~~67 %~~ 67	~~17 %~~ 17
JCBB	9	3	~~33 %~~33	~~17 %~~ 17
NAPS and JCBB			~~33 %~~ 33	~~24 %~~ 24
NAPS ~~Multi-hypothesis~~ and multi-hypothesis	18	14	~~78 %~~78	~~10 %~~10
Normal targets - June 2008				
NAPS ~~Multi-hypothesis~~ multi-hypothesis	9	3	~~33 %~~ 33	~~17 %~~ 17
Normal targets, low currents - June 2009				
NAPS ~~Multi-hypothesis~~multi-hypothesis	26	17	~~65 %~~ 65	~~10 %~~ 10
Normal targets, high currents - June 2010				
NAPS ~~Multi-hypothesis~~multi-hypothesis	42	13	~~31 %~~ 31	~~7 %~~ 7
As above, having reached the field - June 2010				
NAPS ~~Multi-hypothesis~~multi-hypothesis	18	13	~~72 %~~ 72	~~11 %~~ 11

features, the project has shown the viability of the viability of the FBN concept for feature reacquisition with low-cost vehicles.

8.6 Loop Closure: Ship Hull Inspection

The third application we consider is autonomous ship hull inspection. Hull inspections of large ships are frequently performed for safety and security purposes. It is not feasible to put the ships into ~~dry-dock~~ dry dock every time an inspection is required. Currently, this inspection is primarily carried out by divers. This is a ~~time-consuming~~ time-consuming and dangerous task for the divers. To address these risks, Bluefin Robotics and MIT built a ship hull inspection vehicle (see Fig. 8.15) ~~,~~ called the hovering autonomous underwater vehicle (HAUV) [75]. The HAUV is equipped with a DVL to measure velocity relative to a surface, an IMU with ring laser gyro for attitude measurements and a ~~dual frequency~~ dual-frequency identification sonar (DIDSON) [4] for imaging the structures being inspected.

For autonomous ship hull inspection, it is crucially important, but difficult, to accurately track the vehicle position during mission execution. Accurate position

Fig. 8.15 Top view of the Bluefin-MIT hovering autonomous underwater vehicle (HAUV). The vehicle is equipped with a Doppler velocity log (DVL), an imaging sonar, an optical camera, and a light strobe. The sonar and DVL can be actuated independently

information is essential for ensuring full coverage of the area being inspected. The ship hull inspection task further requires reporting the location of potential targets, so they can later be identified and removed. It is difficult, however, to obtain the global position estimate underwater from an external source. GPS is only available at the surface, so acoustic beacons would need to be deployed. Employing only rate gyros and odometry, over time sensor errors accumulate, and the position estimate will drift.

Using ~~time of flight~~ time-of-flight measurements with acoustic beacons has been commonly used in underwater navigation [84, 85, 87] to obtain a global position estimate~~,~~; it has also proved successful in various applications like underwater archaeology [53] and ship hull inspection [30]. Here, we want to avoid the need for external infrastructure ~~,~~and instead are interested in achieving drift-free navigation by using the ~~on-board~~ onboard imaging sonar. In particular, registering current data with previously observed sonar frames provides the necessary constraints to eliminate long-term drift.

Augmenting vehicle localization using sonars has been undertaken in a number of prior works. Walter et al. [80] used manually extracted landmarks ~~,~~and later automatic feature detection [79] with the ESEIF to produce a map of the environment. An automatic feature detector and a landmark formulation using an EKF filter ~~was~~ were used in [13]. Sekkati et al. used extracted corner features from DIDSON frames to estimate vehicle motion over several frames [68]. In related work Negahdaripour et al. combined the DIDSON with an optical camera for 3-D target reconstruction using opti-acoustic stereo [57]. Eustice *et al.* [16] used constraints from overlapping camera frames within a SLAM information filter to estimate vehicle pose. A full 360-degree sonar scanner has been used in partially structured underwater environments [65] for localization ~~,~~by tracking line features in the environment using an EKF for the estimation process. Mallios et al. recently showed promising results in [51] using an mechanical scanning sonar and scan matching in an EKF framework.

Here, we use the pose graph formulation from Sect. 8.2.2 to combine onboard navigation information with sonar registration based on automated dense feature extraction [35]. We focus on imaged areas that are roughly flat, such as the open areas of ship hulls and the ~~sea floor~~seafloor. Our system allows for drift-free navigation without depending on any external infrastructure.

8.6.1 Drift-Free Navigation ~~using~~ Using Imaging Sonar

The goal of this application is to correct drift in the vehicle state estimate over time using the imaging sonar—we begin by defining the quantities to be estimated. The vehicle pose consists of position and attitude. The vehicle position in ~~3D~~ 3-D is specified by Cartesian coordinates x, y, z with respect to some arbitrary reference frame, such as the starting point of the mission or a GPS frame acquired before diving. The attitude of the vehicle is specified by the standard Euler angles ϕ, θ, ψ or roll, pitch, and heading, respectively.

Without the imaging sonar information, only three out of the six degrees of freedom can be estimated without long-term drift. The ring laser gyro is used for heading estimation by integrating the measured angular rotations. A magnetic compass is not a viable option in close vicinity to a ship hull. The DVL provides velocities that are used for dead reckoning. In addition to the relative measurements, absolute measurements of depth from pressure and roll and pitch from the IMU are available. These absolute measurements are integrated into the estimation of the pose graph shown in Fig. 8.1. Nonetheless, no global information is available to limit long-term drift in the heading and x, y position.

Adding loop closure constraints from imaging sonar into the optimization problem eliminates long-term drift in the remaining three dimensions. The loop closure constraints are obtained by registering current sonar images to previously observed ones. Next, we describe the imaging sonar geometry, followed by our approaches to feature extraction and sonar registration.

8.6.1.1 Imaging Sonar Geometry

Following the formulation in [56, 57, 68], we define the geometry of the imaging sonar and derive a model that describes how the image is formed. To generate an image, the sonar emits narrow-beam sound wave and then listens to the returns, sampling the acoustic energy returned from different directions. The sonar provides time of flight and intensity for each azimuth angle. Combining the returns from all the elements provides an image of the reflective surfaces in front of the sonar. We use an imaging sonar with vertical beam width of 28°, covering 96 beams over a 29-degree horizontal field of view. Note that for a given point in the image, it can lie anywhere on an arc at a fixed range, spanning the vertical beam width.

Mathematically, the imaging process can be described as follows. We define the coordinate system for the sonar as shown in Fig. 8.16a. Let us consider a point $\mathbf{p} = [x\ y\ z]^\top$ in the sensor coordinate frame, and let $\mathbf{s} = [r\ \theta\ \phi]^\top$ be the same point in spherical coordinates, where r is the range, θ is the azimuth, and ϕ is the elevation of the point. We can relate the spherical and Cartesian coordinates with the following equations:

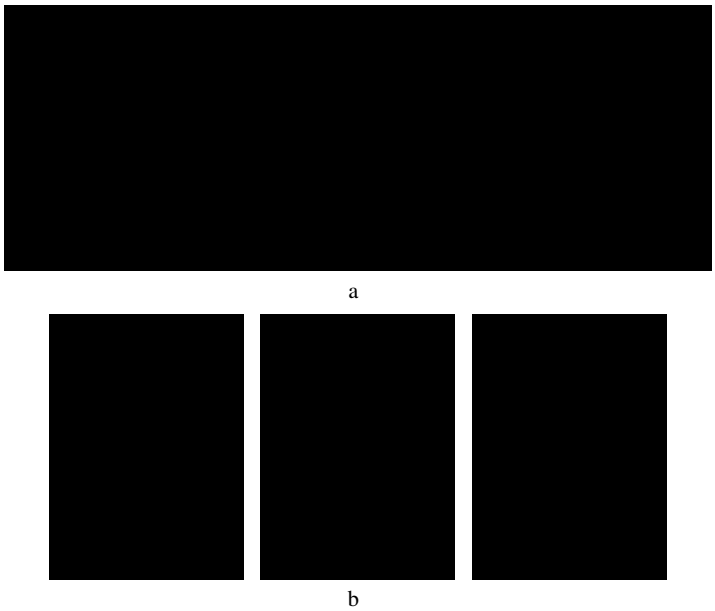


Fig. 8.16 **(a)** DIDSON imaging sonar geometry. **(b)** Sample images showing a clean hull ((~~*left*~~)) and several views of intakes and other structures

$$\mathbf{p} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r\cos\phi\cos\theta \\ r\cos\phi\sin\theta \\ r\sin\phi \end{bmatrix} \tag{8.24}$$

$$\mathbf{s} = \begin{bmatrix} r \\ \theta \\ \phi \end{bmatrix} = \begin{bmatrix} \sqrt{x^2+y^2+z^2} \\ \arctan 2(y,x) \\ \arctan 2\left(z, \sqrt{x^2+y^2}\right) \end{bmatrix} \tag{8.25}$$

The sonar does not provide azimuth ϕ, so we measure point $\mathbf{p}$ is $I(\mathbf{p}) = [r\ \theta]^\top$, and the Cartesian projection of this point is

$$\hat{I}(\mathbf{p}) = \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} r\cos\theta \\ r\sin\theta \end{bmatrix} \tag{8.26}$$

For a small vertical beam width, this can be viewed as an approximation to an orthographic projection.

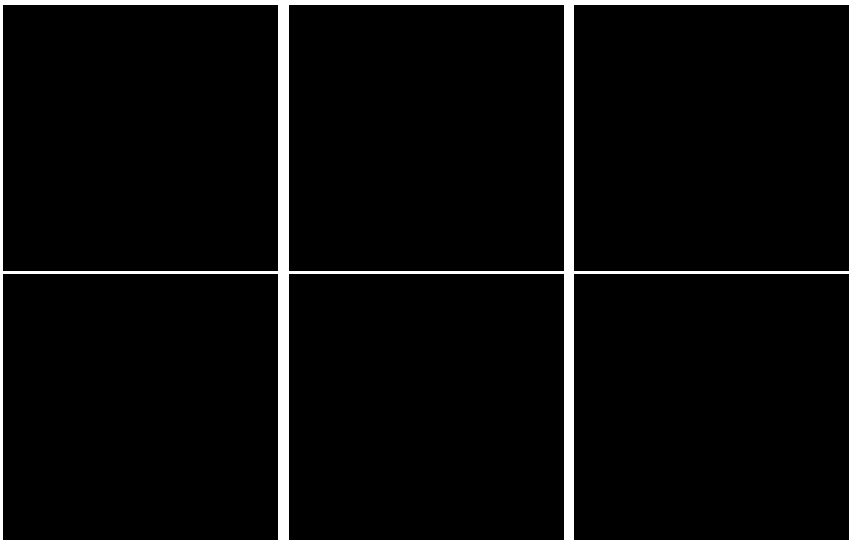


Fig. 8.17 Intermediate steps of the feature extraction process. The extracted features are shown in red ((~~a~~)) Initial sonar image ((~~b~~)) Smoothed ((~~c~~)) Gradient ((~~d~~)) Threshold ((~~e~~)) Clustering ((~~f~~)) Extracted ~~Features~~features

8.6.1.2 Feature Extraction

The imaging sonar returns intensity measurements at a number of ranges along each azimuth beam. The example sonar image in Fig. 8.17a shows some features on a flat surface. A strong return followed by a shadow likely indicates an object standing above the imaged surface, while a shadow on its own indicates a hole or a depression in the surface. Variations in the returned signal are also caused by changes in material properties, the strength of the transmitted signal, receiver sensitivity, distance to target, and the grazing angle, among other factors.

Stable features are extracted from sharp transitions in image intensity that mark the boundary between a strong return and a shadow. The main steps of the algorithm are as follows:

1. Smooth the image.
2. Calculate gradient.
3. Threshold a top fraction as features.
4. Cluster points and discard small clusters.

First, the image is smoothed using a median filter, significantly reducing noise, while still preserving edges, as shown in Fig. 8.17b. Next, the gradient is calculated by computing the difference between the local value and the mean of the n_p previous values along the beam (Fig. 8.17c). The number of previous values n_p used to calculate the mean around the current values affect the type of objects that are

detected. Then points with gradient exceeding a given threshold are marked as candidate features (Fig. 8.17d). The threshold is adaptively chosen, such that a fixed fraction of the features is retained. Note that strong positive gradients are ignored because these correspond to the ends of shadows and are not as stable as negative gradients, which are closer to the sensor. Next, spurious features are eliminated by clustering the points and eliminating small clusters (Fig. 8.17e). The remaining extracted features are shown in Fig. 8.17f, typically containing on the order of one thousand points.

Assuming a locally flat surface, the Cartesian error associated with a successful registration arises mostly from the vertical beam width. The inclination angle between sensor plan and imaged surface is typically around twelve degrees, with a perpendicular distance between one and two meters. For a vertical beam width of 28°, the error can therefore reach 15 cm ~~,~~but is typically much smaller.

8.6.1.3 Registration

We align two overlapping sonar images by registration of the extracted features using the normal distribution transform (NDT) algorithm [5]. The NDT algorithm assigns a scan ~~'s feature points~~ feature point to cells of a regular grid spanning the covered area. For each cell we calculate the mean and variance of its assigned points. This is done for four overlapping grids, where each grid is shifted by half a cell width along each axis. Using multiple shifted grids alleviates the effect of discontinuities resulting from the discretization of space. Two of the benefits using the NDT are that it provides a compact representation of the scan ~~,~~and no exact correspondences between points are needed for alignment. This is useful here, because the movement of the HAUV causes variations in the returns from surfaces, causing some points to drop in and out of the extracted feature set.

The NDT of a scan serves as our model for registration. Given a new scan, a score is calculated for each point by evaluating the Gaussian of the NDT cell that receives the point. This provides a measure of the likelihood that a given point is observed based on the model. We define a cost function as the sum of the negative scores of all the points in the current view. Minimizing the cost with respect to the ~~the~~ change in x, y position and heading ψ of the sonar provides the transformation between the scans. Because the main goal of the registration method is to close loops after drift has accumulated, we do not use the current estimate of the vehicle location to initialize the search. Instead, we repeat optimization from several initial values in an attempt to find the global minimum. To avoid incorrect matches, acceptance is based on a conservative threshold of a normalized score ~~,~~and also requires a minimum number of points to be matched. A successful registration is added to the pose graph as a ~~loop closing~~ loop-closing constraint.

Fig. 8.18 HAUV trajectory along a ship hull with sonar footprint visualized against a ~~3D~~ 3-D ship model. The current sonar image with detected features is shown on the ~~right~~*right*

Fig. 8.19 SS Curtiss in San Diego, with the submerged HAUV visible in the foreground as the ~~yellow~~ *yellow* object in the water

Fig. 8.20 Estimated vehicle trajectory along the ship hull of the USCGC Seneca in ~~Boston – detected~~ Boston—detected loop closures are shown in ~~pink~~*pink*. The vehicle is shown partially under the ship hull, with the sonar viewing cone in ~~blue~~ *blue* indicating the part of the hull visible at this instance

8.6.2 Experiments and Results

Initial experiments were performed in a testing tank ~~,~~ to verify that the system runs ~~on-line~~ online and can stay localized over an extended period of time. Further tests were performed inspecting a patch of the bottom of the Charles river near the MIT Sailing Pavilion ~~,~~ and on a small vessel with flat bottom in Boston harbor. While initial experiments focused on flat surfaces, the open areas of larger ship hulls are only locally flat, requiring a full six degrees of freedom state estimation. A model

view of a ship from an actual experiment is shown in Fig. 8.18, with the trajectory in cyan and the sonar viewing cone in blue. Note the ~~large scale~~ large-scale difference between the vehicle and the ships this system is targeted for.

The HAUV is tethered during experiments ~~, but uses on-board~~ but uses onboard power. The ~~on-board~~ onboard battery allows operation for several hours, while the vehicle moves at 0.25 meters per second. An ~~on-board~~ onboard computer controls the vehicle. Our software runs on a laptop on shore, connected by a fiber tether to the vehicle. The laptop serves for development purposes as well as for visualization during missions. Without the tether, the trajectory as well as selected data can be sent by acoustic communication to the shore. The DVL is locked to the hull to allow operation at a constant distance from the hull.

Recently we demonstrated ~~on-line~~ online ship hull inspection on various large ships. One experiment was performed in early 2011 at the ~~U.S.~~ US Naval Station in San Diego on the 183 ~~m long~~ m-long SS Curtiss, a roll-on/roll-off container ship shown in Fig. 8.19. Another experiment was performed in ~~mid 2011~~ mid-2011 in Boston, on a 82 m medium endurance ~~U.S.~~ US Coast Guard cutter, the USCGC Seneca. One trajectory segment is shown in Fig. 8.20, with ~~loop closing~~ loop-closing constraints obtained from sonar registration. The ~~track lines~~ track lines are spaced approximately four meters apart, providing redundant coverage of the hull with sonar for the inspection mission.

We demonstrated the accuracy of our SLAM-derived state estimate by revisiting ~~way-points~~ waypoints under closed-loop control. During inspection, the operator selected interesting ~~way-points~~ waypoints along the hull using our real-time visualization. The ~~way-points~~ waypoints were selected based on human-recognizable features, which were saved for later comparison. Later in the mission, the vehicle was commanded back to various ~~way-points~~ waypoints, and the current and recorded images were shown for comparison. This served as verification of the consistency and accuracy of our SLAM system. The uncorrected coordinates estimated by the vehicle significantly differed after longer operation, showing the effectiveness of our navigation system in eliminating long-term drift.

A fuller and more complete overview of this particular project can be found in [33], which also presents more detailed experimental results and demonstrations. In addition to the sonar-based measurements discussed in this section, the wider project also incorporated information extracted from the optical camera illustrated in Fig. 8.15. Combining sonar and visual constants within the same SLAM estimation problem allows for sensor redundancy as well, taking advantage of complementary information. The specific application of visual SLAM to this problem is described in [42].

8.7 Conclusions

We have outlined three very different applications of simultaneous localization and mapping in the marine environment. SLAM is increasingly mature and is contribut-

ing to ever more complex marine problems which move to closer interaction with the underwater environment. In particular we have demonstrated that pose graph optimization methods for SLAM can operate onboard modern AUVs in ~~real-time, enabling closed loop~~ real time, enabling closed-loop autonomous operation for many missions of interest.

Despite the substantial progress in SLAM for AUVs over the past decade, there are numerous important topics for future research. Challenges include ~~:~~ (1) robust long-term operation incorporating recovery from failures and detection of environmental changes, (2) cooperative mapping by multiple AUVs using undersea acoustic modems, and (3) integration of SLAM with motion planning and task control to problem close-range subsea inspection and intervention tasks.

Acknowledgments The work described in this chapter was partially supported by the Office of Naval Research under grants N00014-05-10244, N00014-11-10119, N00014-07-11102, N00014-06-10043, N00014-10-10936, and N00014-12-10020, and by the MIT Sea Grant College Program under research grant 2007-R/RCM-20. We wish to acknowledge the contributions of Michael Benjamin, Joseph Curcio, Georgios Papadopoulos, and Andrew Patrikalakis at MIT. Each project had a number of commercial and academic partners which we also wish to acknowledge including Ryan Eustice and Ayoung Kim (University of Michigan)~~,~~; Doug Horner and Tad Mazek (Naval Postgraduate School)~~,~~; Ed Matson and Bryan Schulz (iRobot)~~,~~; Jerome Vaganay, Kim Shurn, and Mike Elkins (Bluefin Robotics)~~,~~; Scott Reed, Alastair Cormack, and Jose Vasquez (~~Seebyte)~~SeeByte); and Dan Kucik and Paul Carroll (Naval Surface Warfare Center-Panama City).

References

1. Bahr A ~~,~~(2009) Cooperative Localization for Autonomous Underwater Vehicles. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA.
2. Bahr A, Leonard J, Fallon M (2009) Cooperative localization for autonomous underwater vehicles. Intl. J. Robotic. Res. 28(6):714–728
3. Ballard RD, Stager LE, Master D, Yoerger D, Mindell D, Whitcomb, LL, Singh H, Piechota D (2002) Iron Age shipwrecks in deep water off Ashkelon, Israel. Am ~~. J . Archaeol.~~J Archaeol 106(2):151–168
4. Belcher E, Hanot W, Burch J (2002) Dual-frequency identification sonar (DIDSON). In: Underwater Technology, 2002. Proceedings of the 2002 International Symposium on, p 187–192
5. Biber P, Strasser W (2003) The normal distributions transform: a new approach to laser scan matching. In: IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS), vol 3, pp 2743–2748
6. Bosse M, Newman P, Leonard J, Teller S (2004) Simultaneous localization and map building in large-scale cyclic environments using the Atlas framework. Intl ~~. J . of Robotic . Res.~~J Robotic Res 23(12):1113–1139
7. Clark DE, Bell J (2005) Bayesian multiple target tracking in forward scan sonar images using the PHD filter. IEE Radar, Sonar and Navigation ~~,~~152:327–334
8. Cummins M, Newman P (2007) Probabilistic appearance based navigation and loop closing. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), pp 2042–2048
9. Dellaert F, Kaess M (2006) Square Root SAM: Simultaneous localization and mapping via square root information smoothing. Intl ~~. J . Robotic . Res.~~J Robotic Res 25(12):1181–1203

10. Duckett T, Marsland S, Shapiro J (2002) Fast, on-line learning of globally consistent maps. Auton ~~. Robot.~~Robot 12(3):287–300
11. Durrant-Whyte H, Bailey T (2006) Simultaneous localisation and mapping (SLAM): Part I. Robot ~~. Automat . Mag.~~Automat Mag ~~,~~13(2):99 –110
12. Edgerton HE (1986) Sonar ~~Images~~images. ~~Englewood Cliffs, NJ:~~ Prentice-Hall, ~~New york~~Englewood Cliffs
13. Englot B, Johannsson H, Hover F (2009) Perception, stability analysis, and motion planning for autonomous ship hull inspection. In: Proceedings of the ~~International Symposium on Unmanned Untethered Submersible Technology~~ international symposium on unmanned untethered submersible technology (UUST)
14. Eustice R (2005) Large-Area Visually Augmented Navigation for Autonomous Underwater Vehicles. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA.
15. Eustice R, Singh H, Leonard J (2006). Exactly sparse delayed-state filters for view-based SLAM. IEEE Trans. Robot., 22(6):1100–1114
16. Eustice R, Singh H, Leonard J, Walter M, Ballard R (2005) Visually navigating the RMS Titanic with SLAM information filters. In:Robotics: Science and Systems (RSS), pp 3281–3288
17. Eustice RM, Whitcomb LL, Singh H, Grund M (2007) Experimental results in synchronous-clock one-way-travel-time acoustic navigation for autonomous underwater vehicles. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), pp 4257–4264, Rome, Italy ~~.~~
18. Fallon M, Kaess M, Johannsson H, Leonard J (2011) Efficient AUV navigation fusing acoustic ranging and side-scan sonar. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), Shanghai, China ~~.~~
19. Fallon M, Papadopoulos G, Leonard J (2009) Cooperative AUV navigation using a single surface craft. In: Field and Service Robotics
20. Fallon MF, Folkesson J, McClelland H, Leonard JJ (2012). Relocating underwater features autonomously using sonar-based SLAM. IEEE J. Ocean Engineering. To Appear ~~.~~
21. Fallon MF, Papadopoulos G, Leonard, JJ (~~2010a~~2010) A measurement distribution framework for cooperative navigation using multiple AUVs. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), pp 4803–4808
22. Fallon MF, Papadopoulos G, Leonard JJ, ~~and~~ Patrikalakis NM (~~2010b~~2010) Cooperative AUV navigation using a single maneuvering surface craft. Intl ~~. J . Robot . Res.~~J Robot Res 29(12):1461–1474
23. Folkesson J, Leonard J (2009) Autonomy through SLAM for an underwater robot. In: Proc. of the Intl. Symp. of Robotics Research (ISRR) ~~, pp 55-70~~
24. Folkesson J, Leonard J (2011) Autonomy through SLAM for an underwater robot. Robot ~~. Res.~~Res 70:55–70
25. Freitag L, Grund M, Singh S, Partan J, Koski P, Ball K (2005) The WHOI micro-modem: An acoustic communications and navigation system for multiple platforms. In: Proceedings of the IEEE/MTS OCEANS Conference and Exhibition, vol 1, pp 1086–1092
26. Geyer E, Creamer P, D'Appolito J, Gains R (1987). Characteristics and capabilities of navigation systems for unmanned untethered submersibles. In: Proc. Int. Symp. on Unmanned Untethered Submersible Technology, pp 320–347
27. Golub G, Loan CV (1996) Matrix ~~Computations~~computations, 3rd edn. Johns Hopkins University Press, Baltimore ~~, MD, 3rd edn.~~
28. Grisetti G, Kümmerle R, Stachniss C, Frese U, Hertzberg C (2010) Hierarchical optimization on manifolds for online 2D and 3D mapping. In :IEEE Intl. Conf. on Robotics and Automation (ICRA), Anchorage, Alaska
29. Grisetti G, Stachniss C, Grzonka S, Burgard W (2007) A tree parameterization for efficiently computing maximum likelihood maps using gradient descent. In: Robotics: Science and Systems (RSS).
30. Harris S, Slate E (1999) Lamp Ray: Ship hull assessment for value, safety and readiness. In: OCEANS '99 MTS/IEEE. Riding the Crest into the 21st Century, vol ~~.~~1, pp 493 –500

AQ: Please update Ref. "[20]".

31. Heckman DB, Abbott RC (1973) An acoustic navigation technique. In: IEEE Oceans '73, pp 591–595
32. Horner DP, McChesney N, Masek T, and Kragelund SP (2009) 3D reconstruction with an AUV mounted forward looking sonar. In: Proc. Int. Symp. on Unmanned Untethered Submersible Technology, pp 1464–1470
33. Hover FS, Eustice RM, Kim A, Englot B, Johannsson H, Kaess M, Leonard JJ (2012) Advanced perception, navigation and planning for autonomous in-water ship hull inspection. Intl . J . Robotic . Res.J Robotic Res
34. Hunt M, Marquet W, Moller D, Peal K, Smith W, and Spindel R (1974) An acoustic navigation system. Technical Report WHOI-74-6, Woods Hole Oceanographic Institution.
35. Johannsson H, Kaess M, Englot B, Hover F, and Leonard J (2010) Imaging sonar-aided navigation for autonomous underwater harbor surveillance. In: IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS), Taipei, Taiwan
36. Julier S, Uhlmann J (2001) A counter example to the theory of simultaneous localization and map building. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), vol 4, pp 4238–4243
37. Kaess, M Dellaert F (2009) Covariance recovery from a square root information matrix for data association. J . Robotic . Auton . Syst.Robotic Auton Syst 57(12):1198–1210
38. Kaess M, Ila V, Roberts R, Dellaert F (2010) The Bayes tree: An algorithmic foundation for probabilistic robot mapping. In: Intl. Workshop on the Algorithmic Foundations of Robotics, WAFR, Singapore .
39. Kaess M, Johannsson H, Roberts R, Ila V, Leonard JJ, and Dellaert, F (2012) iSAM2: Incremental smoothing and mapping using the Bayes tree. Int . J . Robotic . Res.J Robotic Res 31:217–236
40. Kaess M, Ranganathan A, Dellaert F (2008) iSAM: Incremental smoothing and mapping. IEEE Trans . Robotic.Robotic 24(6):1365–1378
41. Kim A, Eustice R (2009) Pose-graph visual SLAM with geometric model selection for autonomous underwater ship hull inspection. In: IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS) , pp 549-593
42. Kim A, Eustice RM (2011) Combined visually and geometrically informative link hypothesis for pose-graph visual SLAM using bag-of-words. In: IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS), San Francisco, CA, pp 1647–1654
43. Kinsey JC, Eustice RM, Whitcomb LL (2006) A survey of underwater vehicle navigation: Recent advances and new challenges. In: IFAC Conference of Manoeuvering and Control of Marine Craft, Lisbon, Portugal. Invited paper .
44. Konolige K (2004) Large-scale map-making. In: AAAI'04: Proceedings of the 19th national conference on Artificial intelligence, pp 457–463. AAAI Press/The MIT Press
45. Konolige K, Grisetti G, Kummerle R, Burgard W, Limketkai B, Vincent R (2010) Efficient sparse pose adjustment for 2D mapping. In: IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS), Taipei, Taiwan, pp 22–29
46. Kümmerle R, Grisetti G, Strasdat H, Konolige K, Burgard W (2011) g2o: A general framework for graph optimization. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), Shanghai, China
47. Leonard JJ, Bennett AA, Smith CM, Feder HJS (1998) Autonomous underwater vehicle navigation. Technical Report Marine Robotics Laboratory Technical Memorandum 98–1, MIT
48. Lu F, Milios E (1997) Globally consistent range scan alignment for environmental mapping. Auton . Robot.Robot 4:333–349
49. Mandt M, Gade K, Jalving, B (2001) Integrating DGPS-USBL positioning measurements with inertial navigation in the HUGIN 3000 AUV. In:Saint Petersburg International Conference on Integrated Navigation Systems, Russia
50. Mahon I, Williams S, Pizarro O, Johnson-Roberson M (2008) Efficient view-based SLAM using visual loop closures. IEEE Trans . Robotic.Robotic 24(5):1002–1014
51. Mallios A, Ridao P, Hernandez E, Ribas D, Maurelli F, Petillot , Y (2009) Pose-based SLAM with probabilistic scan matching algorithm using a mechanical scanned imaging sonar. In: OCEANS 2009-EUROPE, 2009. OCEANS '09., pp 1–6

AQ: Please provide the volume and page range for the reference [33]

AQ: Please check the author names for the reference [49]

52. Milne PH (1983) Underwater ~~Acoustic Positioning Systems~~acoustic positioning systems. E. F. N. Spon, London
53. Mindell D (2007) Precision navigation and remote sensing for underwater archaeology. Remote sensing in archaeology, p 499
54. Montemerlo M, Thrun S, Roller D, Wegbreit B (2003) FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In: Intl. Joint Conf. on Artificial Intelligence, Morgan Kaufmann~~Publishers Inc~~, pp 1151–1156
55. Moutarlier P, Chatila R (1990) An experimental system for incremental environment modeling by an autonomous mobile robot. In: Intl. Sym. on Experimental Robotics (ISER).
56. Negahdaripour S, Firoozfam P, Sabzmeydani P (2005) On processing and registration of forward-scan acoustic video imagery. In: Computer and Robot Vision, 2005. Proceedings. The 2nd Canadian Conference on, pp 452–459
57. Negahdaripour S, Sekkati H, Pirsiavash H (2009) Opti-acoustic stereo imaging: On system calibration and 3-D target reconstruction. IEEE transactions on image processing: a publication of the IEEE Signal Processing Society ~~.~~
58. Neira J, Tardos J, Castellanos J (2003) Linear time vehicle relocation in SLAM. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), pp 427–433
59. Neira J, Tardos JD (2001) Data association in stochastic mapping using the joint compatibility test. IEEE Trans ~~. Robotic . Autom.~~Robotic Autom 17(6):890–897
60. Nuske S, Roberts J, Prasser D, Wyeth G (2010) Experiments in visual localisation around underwater structures. In: Howard A, Iagnemma K, Kelly A (eds), Field and ~~Service Robotics~~service robotics~~, vol 62 of~~ . Springer ~~Tracts in Advanced Robotics~~tracts in advanced robotics, vol 62. Springer, Berlin~~/ Heidelberg~~, pp 295–304
61. Olson E, Leonard J, Teller S (2006) Fast iterative alignment of pose graphs with poor initial estimates. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), pp 2262–2269
62. Paz L, Guivant J, Tardós J, Neira J (2007) Data association in *O(n)* for divide and conquer SLAM. In: Robotics: Science and Systems, RSS, Atlanta, GA, USA ~~.~~
63. Petillot Y, Ruiz IT, Lane DM (2001) Underwater vehicle obstacle avoidance and path planning using a multi-beam forward looking sonar. J ~~. Ocean . Eng.~~Ocean Eng 26:240–251
64. Ranganathan A, Kaess M, Dellaert F (2007) Loopy SAM. In: Intl. Joint Conf. on Artificial Intelligence, Hyderabad, India, pp 2191–2196
65. Ribas D, Ridao P, Neira J, Tardós, J (2006) SLAM using an imaging sonar for partially structured underwater environments. In: IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)
66. Rosen D, Kaess M, Leonard J (2012) An incremental trust-region method for robust online sparse least-squares estimation. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), St. Paul, MN, pp 1262–1269
67. Schulz B, Hughes R, Matson R, Moody R, Hobson B (2005) The development of a free-swimming UUV for mine neutralization. In: Proceedings of the IEEE/MTS OCEANS Conference and Exhibition, IEEE, pp 1443–1447
68. Sekkati H, Negahdaripour S (2007) 3-D motion estimation for positioning from 2-D acoustic video imagery. Lecture Notes in Computer Science, 4478:80
69. Smith R, Self M, Cheeseman P (1990) Estimating uncertain spatial relationships in robotics. In: Autonomous Robot Vehicles~~, SpringerVerlag~~. Springer, pp 167–193
70. Tena Ruiz I, de Raucourt S, Petillot Y, Lane D (2004) Concurrent mapping and localization using sidescan sonar. J ~~. Ocean . Eng.~~Ocean Eng 29(2):442–456
71. Thompson R, Zehner W (1999) Frequency-steered acoustic beam forming system and process. US Patent 5,923,617
72. Thrun S, Beetz M, Bennewitz M, Burgard W, Cremers A, Dellaert F, Fox D, Hahnel D, Rosenberg C, Roy N, Schulte J, Schulz D (2000) Probabilistic algorithms and the interactive museum tour-guide robot minerva. Intl ~~. J . of Robotic . Res.~~J Robotic Res 19(11):972–999
73. Thrun S, Burgard W, Fox D (2005) Probabilistic ~~Robotics~~robotics. The MIT press, Cambridge ~~, MA~~

74. Thrun S, Liu Y (2004) Simultaneous localization and mapping with sparse extended information filters. Intl ~~. J . Robotic . Res.~~J Robotic Res, 23(7)
75. Vaganay J, Elkins M, Esposito D, O'Halloran W, Hover F, Kokko M ~~.~~(2007) Ship hull inspection with the HAUV: US Navy and NATO demonstrations results. In: Proceedings of the IEEE/MTS OCEANS Conference and Exhibition, vol 1, pp 761–766
76. Vaganay J, Leonard J, Curcio J, Willcox J (2004) Experimental validation of the moving long base line navigation concept. In: Autonomous Underwater Vehicles, 2004 IEEE/OES, pp 59–65
77. von Alt C, Allen B, Austin T, Stokey R (1994) Remote environmental monitoring units. In: AUV 94
78. Walls JM, Eustice RM (2011) Experimental comparison of synchronous-clock cooperative acoustic navigation algorithms. In: Proceedings of the IEEE/MTS OCEANS Conference and Exhibition, Kona, HI, USA. Accepted, To Appear ~~.~~
79. Walter M (2008) Sparse Bayesian information filters for localization and mapping. PhD thesis, Massachusetts Institute of Technology.
80. Walter M, Hover F, Leonard J (2008) SLAM for ship hull inspection using exactly sparse extended information filters. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), pp 1463–1470
81. Webster SE (2010) Decentralized single-beacon acoustic navigation: combined communication and navigation for underwater vehicles. PhD thesis, Johns Hopkins University, Baltimore, MD, USA.
82. Webster SE, Whitcomb LL, Eustice RM (2010) Preliminary results in decentralized estimation for single-beacon acoustic underwater navigation. In: Robotics: Science and Systems (RSS), Zaragoza, Spain.
83. Whitcomb L, Yoerger D, Singh H (1999a) Advances in doppler-based navigation of underwater robotic vehicles. In: IEEE Intl. Conf. on Robotics and Automation (ICRA), vol 1, pp 399–406
84. Whitcomb L, Yoerger D, Singh H (1999b) Doppler/LBL based navigation of underwater vehicles. In: Proceedings of the International Symposium on Unmanned Untethered Submersible Technology (UUST)
85. Whitcomb L, Yoerger D, Singh H, Mindell D(1998) Towards precision robotic maneuvering, survey, and manipulation in unstructured undersea environments. In: Proc. of the Intl. Symp. of Robotics Research (ISRR), vol 8, Springer, London, pp 45–54
86. Yoerger D, Jakuba M, Bradley A, Bingham B (2007) Techniques for deep sea near bottom survey using an autonomous underwater vehicle. Intl ~~. J . of Robotic . Res.~~J Robotic Res 416–429
87. Yoerger D, Mindell D (1992) Precise navigation and control of an ROV at 2200 meters depth. In: Proceedings of Intervention/ROV, vol 92

AQ: Please provide page range for the reference [74]