2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

Predictive 3D Sonar Mapping of Underwater Environments via Object-specific Bayesian Inference

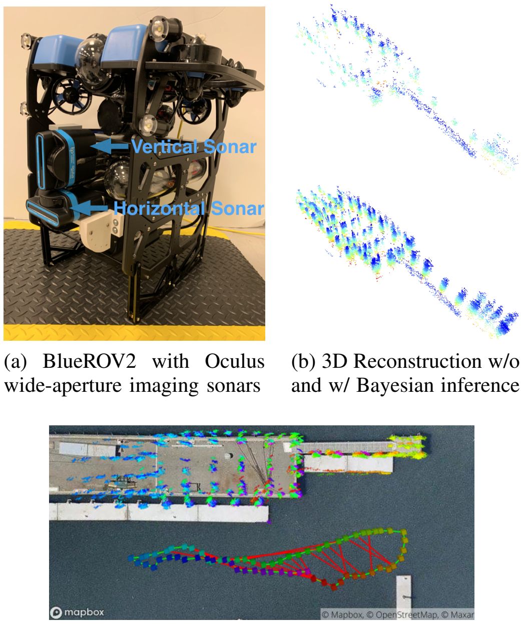
John McConnell and Brendan Englot

Abstract— **Recent work has achieved dense 3D reconstruction with wide-aperture imaging sonar using a stereo pair of orthogonally oriented sonars. This allows each sonar to observe a spatial dimension that the other is missing, without requiring any prior assumptions about scene geometry. However, this is achieved only in a small region with overlapping fields-of-view, leaving large regions of sonar image observations with an unknown elevation angle. Our work aims to achieve large-scale 3D reconstruction more efficiently using this sensor arrangement. We propose dividing the world into semantic classes to exploit the presence of repeating structures in the subsea environment. We use a Bayesian inference framework to build an understanding of each object class's geometry when 3D information is available from the orthogonal sonar fusion system, and when the elevation angle of our returns is unknown, our framework is used to infer unknown 3D structure. We quantitatively validate our method in a simulation and use data collected from a real outdoor littoral environment to demonstrate the efficacy of our framework in the field.**

I. Introduction

Autonomous underwater vehicles (AUVs) offer important capabilities that support subsea inspection, surveillance, and environmental monitoring. Due to variable water clarity and frequently poor ambient lighting, AUVs often rely on acoustic rather than optical sensors for perception in real-world operational settings; among the most capable and versatile acoustic sensors are multi-beam profiling sonars and wide aperture imaging sonars. Profiling sonars are highly accurate, but only provide a narrow vertical beam. This narrow beam limits the volume of water which can be imaged at each time step, making large-scale 3D mapping with these sensors a prolonged endeavor. Further, using these sensors may be prohibitively expensive, especially considering the high-grade inertial navigation systems (INS) often required to support 3D mapping. Wide aperture, multi-beam imaging sonar, in contrast, maximizes situational awareness by imaging a large volume of water at every time step. However, while these sensors image large 3D volumes, only the range and bearing of their returns are recorded; not the elevation angle.

Recent work [1]–[3] has investigated how to supply the missing information using a stereo pair of wide aperture multi-beam imaging sonars to *measure* the unknown elevation angle rather than estimate it. This methodology requires no prior assumptions about the geometry of the objects in view, unlike other algorithms for recovering elevation angle from a single sonar image. In the specific method implemented in our prior work [1], the need for overlapping views across orthogonally oriented sonars means that 3D volumetric sensing can only be performed across a 20°-by-20° field of view (the horizontal field of view of each sensor would normally be 130°). Using sensors with only 20° horizontal and vertical fields of view proves tedious when mapping a large-scale outdoor environment. The question then remains: how can accurate, large-scale 3D volumetric mapping be achieved efficiently by sonar-equipped AUVs in cluttered environments?



(a) BlueROV2 with Oculus wide-aperture imaging sonars

(b) 3D Reconstruction w/o and w/ Bayesian inference

(c) Satellite image overlay of 3D reconstruction

Fig. 1: **System Overview:** (a) shows our AUV hardware, (b) shows a sample reconstruction from SUNY Maritime's marina in the East River (raw 3D observations at top and with Bayesian inference at bottom) and (c) shows a top-down view of (b) with a satellite image [4] overlay, and the corresponding SLAM pose graph. Squares in (c) show AUV poses with color corresponding to time; green lines are sequential factors; red lines are loop closures. Note: The point cloud in (c) uses the same time mapping as the corresponding poses, while the point cloud in (b) has color mapped to the vertical axis.

In this paper, we consider the mapping of shallow water, littoral settings often characterized by repeating objects, such as pier pilings, throughout the environment. We will exploit this repeating structure in conjunction with the previously proposed orthogonal stereo sonar fusion system to accelerate 3D reconstruction of large scale littoral environments. The contributions of this paper are as follows:

- A framework to identify and exploit the observation of repeating objects in the environment, using past 3D measurements to infer unknown 3D structure when

J. McConnell and B. Englot are with the Department of Mechanical Engineering, Stevens Institute of Technology, Hoboken, NJ, 07030, USA {jmcconn1,benglot}@stevens.edu

978-1-7281-9077-8/21/$31.00 ©2021 IEEE

instances of the same class are partially re-observed.

- A simulation study demonstrating the accuracy of the proposed Bayesian inference framework in predicting the 3D structure of partially observed objects.
- Real-world experiments that show the efficacy of this framework in mapping large-scale littoral environments, as well as its compatibility with sonar-based simultaneous localization and mapping (SLAM) – an illustrative overview is provided in Figure 1.

First, we will discuss related work and precisely define the problem we aim to solve. Next, we will present the developed algorithm in detail. Finally, we will provide experimental results that confirm our algorithm's utility, both in simulation and using real data from the East River in the Bronx, NY.

II. Related Work

A. 3D Reconstruction with Wide Aperture Sonar

Wide aperture multi-beam imaging sonar is a relatively low cost option for underwater perception which provides a large field of view. However, it can only measure two of three dimensions in the spherical coordinate frame: range and bearing, leaving the third, elevation angle, unknown. This limitation is addressed in [5], where 3D reconstruction is achieved by estimating each sonar contact's unknown elevation angle. This approach is built upon by [6], which accurately maps the surfaces of concrete pier pilings. However, both studies assume that elevation angle monotonically increases or decreases with range, limiting their application to environments where this assumption holds. Further, both methods require the sonar to be positioned at a downward grazing angle; in shallow, littoral environments, this may create difficulty when trying to distinguish between the structures of interest and returns from the seafloor.

B. Space Carving

Another set of 3D reconstruction methods used to address these issues is space carving. Typically, low-intensity background pixels are used to remove sections of a voxel grid, using multiple views of an object [7]–[9]; with [10] being a recent innovation. These methods prove effective in their respective validations but fail to address two fundamental requirements for large scale mapping: the large memory required for such a voxel grid needed to map a large scale outdoor environment, and the accurate online pose estimates required to support this approach. Underwater SLAM is often solved incrementally, with pose estimates fluctuating as the solution evolves. This makes space carving a challenge to implement and a research area unto itself.

C. Deep Learning

Recently, deep learning was applied to this problem in Elevate-net [11] by making use of a convolutional neural network (CNN) to estimate the elevation angle at each pixel in a wide aperture sonar's acoustic image. While an exciting line of inquiry, Elevate-net requires pre-training on synthetic data generated with CAD models.

D. Using a Stereo Pair of Sonars for 3D Reconstruction

Using a stereo pair of imaging sonars to address each sonar's missing elevation angle requires no prior assumptions about object geometry, while providing fully defined 3D points at every time step [1]–[3]. Moreover, because the resulting points are generated independently, one pair of images at a time, this approach is easily integrated with a variety of SLAM solutions. In the method proposed in our prior work [1], the use of orthogonally oriented sonars as depicted in Fig. 1 enables dense mapping, but constrains 3D volumetric sensing to a 20°-by-20° field of view (when the horizontal aperture of each sensor would normally be 130°), making large scale 3D mapping a time consuming process.

E. Inference Aided 3D Reconstruction and Mapping

There is a large body of work that applies probabilistic inference to enhance 3D mapping and reconstruction. Most closely related are the methods that use inference to improve point cloud or voxel mapping, which often use insights from large data sets or object models provided a priori, which is not always available in an underwater robotics setting.

The use of a variational auto-encoder to infer the 3D distribution of an object given a single view from an RGB camera is explored in [12], [13]. As mentioned above, this work requires a large data set to pre-train the network weights. [14] explores the use of synthetic data from a generative adversarial network to avoid this requirement. [14] uses a single view voxel grid as the input for 3D reconstruction; the focus is 3D-to-3D densification, rather than the 2D-to-3D inference considered in our work. A similar concept to our work is explored in [15], where mapping is performed at an object level. Here, object models are produced using high-quality depth camera scans from a controlled setting. These scans are used to improve a 6-DoF SLAM solution. Similar to [12], [13], the applications of this approach in underwater robotics are limited due to the need for object models a priori. A notable use of inference in an underwater setting is [16], where CAD models of objects of interest are provided a priori, and are used to improve the map output.

A related set of methods use probabilistic inference to enhance occupancy grid maps [17]–[20]. While these methods are excellent at improving occupancy map coverage, they solve mapping under sparse inputs by performing gap-filling and semantic inference on a data structure of the same dimensionality as those sparse inputs. In contrast, we focus on 2D-to-3D inference to enhance the dimensionality of inputs that are not directly observed in 3D, whose inference is conditioned on prior 3D observations of the same class.

III. Problem Description

In this work, we consider 3D reconstruction using a pair of orthogonal imaging sonars with an overlapping field of view. A robot visits a series of poses x_t, with transformations $\mathbf{T} \in \mathbb{R}^{4\times4}$. Each pose has associated observations z_t, with two components: horizontal sonar observations z^h and vertical sonar observations z^v. Each set of observations is defined as an intensity image in spherical coordinates with range

$R \in \mathbb{R}_+$, bearing $\theta \in \Theta$, and elevation $\phi \in \Phi$, with $\Theta, \Phi \subseteq [-\pi, \pi)$, and an associated intensity value $\gamma \in \mathbb{R}_+$. These measurements can be converted to Cartesian space:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = R \begin{pmatrix} \cos\phi\cos\theta \\ \cos\phi\sin\theta \\ \sin\phi \end{pmatrix}. \tag{1}$$

Each recorded measurement z^h and z^v is characterized by an omitted degree-of-freedom (DoF), and in the robot frame, due to the orthogonality of the two sonars, these DoFs differ:

$$z^h = (R^h, \theta, \gamma^h)^\top, \qquad z^v = (R^v, \phi, \gamma^v)^\top. \tag{2}$$

As in [1], we associate measurements across concurrent, orthogonal images to fully define the measurements in 3D, yielding Equation (3):

$$z^{Fused} = \left(\frac{R^h + R^v}{2}, \theta^{(h)}, \phi^{(v)}\right)^\top. \tag{3}$$

However, this association can only take place within the small region of overlap between the sonars' fields of view, equivalent in size (in both bearing and elevation) to the sonar's vertical beamwidth. This leaves large portions of each image with a missing DoF and, therefore, undefined in 3D. In this work, we wish to perform a 3D reconstruction by mapping the observations into fixed frame $\mathcal{I}$ as in Eq. (4).

$$\mathcal{M} = \{\hat{z}^{(\mathcal{I})} | \hat{z}^{(\mathcal{I})} = \mathbf{T}\hat{z}^{Fused} \ \forall \ \hat{z} \in \hat{z}\} \tag{4}$$

The information missing from a portion of all images introduces ambiguity into the application of Eqs. (1) and (4). The question to be answered in this paper becomes: How can we leverage the fully defined measurements from Eq. (3) to infer the unknown 3D structure of the rest of the imagery, producing a more comprehensive 3D reconstruction?

IV. Proposed Algorithm

We propose to exploit the observation of repeating objects throughout the environment, using the 3D observations occasionally obtained for specific objects, to infer the 3D structure when those same objects are re-observed without a known elevation angle. First, we identify objects in the sonar image and provide a semantic class label. Next, we develop a Bayesian inference model for each object class to later query this model for points in the horizontal sonar image corresponding to that same class. Note that we only apply this inference procedure to the horizontal sonar image. Due to the shallow depths at which we operate our vehicle, the vertical image contains fewer meaningful observations of subsea structures and is not subjected to Bayesian inference.

A. SLAM Solution

In this work, we utilize a pose SLAM formulation to estimate our robot's pose history through time. We restrict our formulation to 3-DoF estimation in the plane to provide an efficient and robust SLAM pipeline that prioritizes the DoFs of greatest uncertainty (surge, sway, and yaw).

Here we formulate SLAM on a factor graph, which is solved with the aid of GTSAM [21] and iSAM2 [22]. At each keyframe, features are extracted from our robot's horizontal sonar using the method described below in Sec. IV.B, and the unknown angle ϕ is assigned as zero. The constraints relating adjacent keyframes are estimated using iterative closest point (ICP) [23], with global initialization via consensus set maximization [24] - we denote this step sequential scan matching (SSM). Loop closures are added by applying the same scan matching process to non-consecutive keyframes, applying ICP to frames within a designated radius of the current keyframe - we denote this step non-sequential scan matching (NSSM). To reject loop closure outliers, we use pairwise consistent measurement set maximization [25]. Our factor graph is given by

$$\mathbf{f}(\mathbf{\Theta}) = \mathbf{f}^0(\mathbf{\Theta}_0) \prod_i \mathbf{f}_i^{\mathrm{O}}(\mathbf{\Theta}_i) \prod_j \mathbf{f}_j^{\mathrm{SSM}}(\mathbf{\Theta}_j) \prod_q \mathbf{f}_q^{\mathrm{NSSM}}(\mathbf{\Theta}_q).$$

This planar SLAM solution is used to provide estimates of surge, sway and yaw for registering our algorithm outputs to the global frame; the remaining degrees of freedom come from our vehicle's pressure sensor and inertial measurement unit (IMU). Note that we will only analyze the sonar images corresponding to SLAM keyframes to develop our 3D map. We do not analyze images between keyframes, due to the high drift rate of our low-cost dead reckoning system.

B. Sonar Fusion System

At each time step, the robot receives a set of observations constituting a pair of sonar images, as described in Eq. (2). To fully define these in 3D as denoted in Eq. (3), we need to extract and associate features across the images.

Here we use a similar approach to that of our prior work [1], dividing the image-feature matching problem into multiple, smaller, subproblems. Recall that range R is discrete within a sonar image, and since associated features should be at the same range, we use the range to define these subproblems. All features at the same range in each sonar image are gathered and processed using intensity-based association. The cost function used here is defined as follows:

$$\mathcal{L}(z_i^h, z_j^v) = ||\nu^h - \nu^v||, \tag{5}$$

where ν^h and ν^v are square patches of the sonar image around the given feature. These patches have a fixed size of 5x5 pixels in this work. Note that ν^v is rotated 90 degrees to account for the orthogonality of the images. Moreover, before this comparison is made, the images' intensity values are normalized at every time step. The cost function in Eq. (5) is used to find the solution that minimizes the sum of costs between features for each subproblem.

To estimate our confidence in these matches, we compare the two best solutions for each feature association [26]:

$$C = \frac{\mathcal{L}(z_i^h, z_j^v)_{min2} - \mathcal{L}(z_i^h, z_j^v)_{min}}{\Sigma_{0,0}^{i,j} \mathcal{L}(z_i^h, z_j^v)}. \tag{6}$$

While we compute subproblem cost totals in order to find sets of associated features in Eq. (5), confidence is evaluated on a feature-wise basis, comparing the costs for each individual feature association made in the given solution. This gives us a simple metric with which to cull uncertain associations.

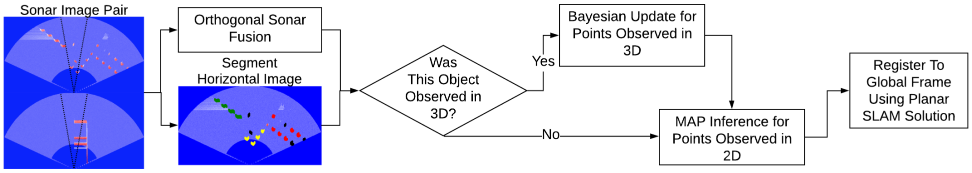

Sonar Image Pair
Orthogonal Sonar Fusion
Segment Horizontal Image
Was This Object Observed in 3D?
Yes
No
Bayesian Update for Points Observed in 3D
MAP Inference for Points Observed in 2D
Register To Global Frame Using Planar SLAM Solution

Fig. 2: **System block diagram.** A pair of orthogonal sonar images is provided as input (black lines bound the region of overlap between the two sonar fields-of-view). The images are processed according to Section IV.B. The horizontal image is segmented as in Section IV.C (colors denote different object classes - seawall in green, rectangular pilings in yellow, cylindrical pilings in red). The resulting 3D points enrich each object's model (Eq. (7)), while MAP inference is applied to 2D points (Eqs. (8), (9)). We then use the planar SLAM solution to register the resulting point cloud. The synthetic sonar images shown here are sampled from the virtual environment depicted in Fig. 3.

C. Image Feature Extraction and Object Classification

To extract features from sonar imagery, we use the method employed in [1], used in various radar and sonar applications. The technique, smallest-of cell averaging constant false alarm rate (SOCA-CFAR) detection [27], takes local area averages around the pixel in question and produces a noise estimate. If the signal is greater than a designated threshold, the pixel is identified as an image-feature.

Next, objects must be identified; for this, we utilize DBSCAN [28], since the number of clusters in an image is not known a priori. The results from this step are image-features clustered into objects with unknown class labels. Note that even with a feature detector as robust as SOCA-CFAR, noise is still present. Accordingly, only clusters with n or more image-features are passed on to the next step.

Lastly, we must provide a class label for each object. In this work, we use a simple neural network to perform semantic labeling of object instances. Specifically, we use a CNN that accepts 40x40 pixel sonar image patches in grayscale, with two convolutional layers. Inputs are generated by fitting a bounding box around each object identified in a sonar image and resizing the bounding box into 40x40 pixels, while preserving the object's aspect ratio. We utilize Monte-Carlo dropout in this CNN to reject outliers and uncertain classifications by making m predictions for each object. In this way, we can assess the network's confidence in its predictions, as shown in [29]. Uncertain predictions are simply provided with a label "unknown class." An example of this pipeline in action is shown in Fig. 2.

To train this CNN, a small hand-annotated data set of representative sonar imagery is used, which is not included in the sequences used for validation in this work. To generate sufficient training samples to properly train the model, data augmentation is used. We augment our data by applying Gaussian noise, random flips and random rotations.

D. Bayesian Inference for Objects Observed in 3D

Each detected object in the horizontal sonar image is now represented by a cluster of features with a class label. These features have a range, bearing, and unknown elevation angle. At this step, the dual sonar fusion system provides an elevation angle for a subset of these points, which lie inside the small region with overlapping fields of view. These are the points we concern ourselves with in this subsection.

We assume objects of the same class will have similar geometries, as is typical in the humanmade littoral environments, populated with piers, used to validate this algorithm. Semantic classes are defined with this goal in mind, so that objects with similar geometries are grouped together.

We use a Bayesian inference framework to estimate the conditional distribution P$(z^h_Z|z^h_R, z^h_\theta)$ for each object class incrementally and online. Note that in this process, we estimate Cartesian z_Z and not elevation angle. z_Z is a more accurate indicator of the absolute, rather than relative, height at which an object is observed, since in this work we consider scenarios in which our robot maps the environment at fixed depth, employing planar SLAM. An object's distribution is updated for every measured 3D point per Bayes rule:

$$P(z^h_Z|z^h_R, z^h_\theta) = \frac{P(z^h_R, z^h_\theta|z^h_Z)P(z^h_Z)}{P(z^h_R, z^h_\theta)}. \tag{7}$$

Elevation angles measured by the dual sonar fusion system are treated as measurements of z^h_Z at the given range and bearing, corrupted with zero-mean Gaussian noise, $\mathcal{N}(\mu, \sigma^2)$, forming the measurement likelihood $P(z^h_R, z^h_\theta|z^h_Z)$. The prior, $P(z^h_Z)$, is simply the existing distribution corresponding to the z^h_R and z^h_θ of the newly observed 3D point. We note that these distributions are maintained throughout the whole time-history of the robot's mission, so they incorporate observations from the current frame along with observations from all previous frames. If an update has never been performed, an initial uniform distribution is used.

At times we may view an object class at different distances and orientations; for this reason, we register each object to a *reference coordinate frame* before we apply Bayes rule in Eq. (7). The first time we see an object, we note its minimum range and median bearing as the reference frame's origin. These object points are then maintained as a "reference point cloud" to register the object class's future instances to this coordinate frame. When an object is detected and its distribution P$(z^h_Z|z^h_R, z^h_\theta)$ is updated, the object is first registered to the given object class's reference coordinate frame using ICP. The transformed points are evaluated via Eq. (7) and are added to the points used to register future object sightings to the reference frame. Our object-specific distributions P$(z^h_Z|z^h_R, z^h_\theta)$ will next allow us to predict the height of the sonar returns observed only in 2D.

E. Predicting 3D Structure via MAP Estimation

At each time step, after the process detailed in Section IV.D is completed for the objects measured in 3D, we again consider all objects with class labels comprised of a minimum number of features. We now use the posterior distribution of each object's geometry, $P(z_Z^h|z_R^h, z_\theta^h)$, to predict the height of all 2D points lacking this information.

Suppose an object belongs to a class with a posterior updated by at least one application of Eq. (7). In that case, we proceed, first with registration to the object class's reference frame as described above, without adding new points to the reference point cloud. Due to the sonar's ambiguity, it may be that there is more than one true z_Z^h for a given range and bearing. For this reason, we break maximum a posteriori (MAP) estimation into two steps, as shown in Eqs. (8), (9).

$$z_Z^h = argmaxP(z_Z^h|z_R^h, z_\theta^h), z_Z^h \leq 0 \tag{8}$$

$$z_Z^h = argmaxP(z_Z^h|z_R^h, z_\theta^h), z_Z^h > 0 \tag{9}$$

If one or both maxima correspond to confidence exceeding a designated threshold, those values are adopted for inclusion in the robot's map. Eq. (1) is solved to provide an output in local Cartesian coordinates, $[X, Y, Z]^T$. This process is completed for all objects in view – the result is a horizontal sonar image with more observations fully defined in 3D, rather than just the few observations inside the region of dual-sonar overlap. The observations are converted to a point cloud and registered to the global map frame per Eq. (4).

V. Experiments and Results

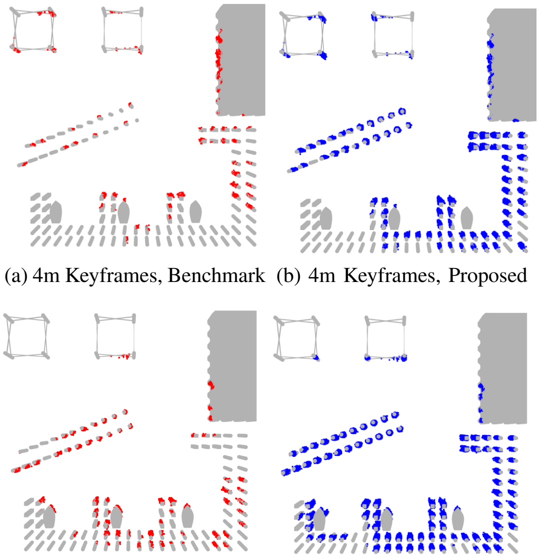
A. Hardware Overview

In order to perform real experiments and derive a simulation environment for this work, we use our customized BlueROV2 heavy-configuration robot, shown in Fig. 1a. This vehicle is equipped with an on-board Pixhawk, Raspberry Pi and NVIDIA Jetson Nano for control and computation. We use a Rowe SeaPilot Doppler velocity log (DVL), VectorNav VN100 inertial measurement unit (IMU) with integrated Kalman filter, and a Bar30 pressure sensor. For perceptual sensors we use a pair of wide aperture multi-beam imaging sonars, a Blueprint subsea Oculus M750d and M1200d. We use the M750d as our horizontal sonar and the M1200d as the vertical sonar. Note that the entirety of this work takes place with these sonars and their simulated versions operating at a range of 30 meters, with 5cm range resolution.

In order to manage the BlueROV's sensors, SLAM system, and companion 3D mapping system, we use the Robot Operating System [30], both for operating the vehicle and for playback of its data. The proposed 3D mapping algorithm is applied to real-time playback of our data using a computer equipped with an NVIDIA Titan RTX GPU and Intel i9 CPU. Due to the keyframe spacing employed within our SLAM framework, our mapping process (which operates only on keyframes) sees a sonar frame-rate that does not exceed 0.4Hz. Note that all experiments take place at a fixed depth.

B. Simulation Study

In this section, we utilize Gazebo [31] with the UUV simulator [32] to quantitatively validate our method. The UUV simulator provides an implementation of [33] to simulate wide aperture multi-beam sonar. This simulation environment allows us to perform a quantitative study impossible with field data, where perfect ground truth information is available. We design a humanmade, littoral environment that resembles many marinas and waterfronts. This environment includes two differently shaped pilings (cylindrical and rectangular), boat hulls, corrugated seawalls, and trusses.



(a) 4m Keyframes, Benchmark (b) 4m Keyframes, Proposed

(c) 2m Keyframes, Benchmark (d) 2m Keyframes, Proposed

Fig. 3: **Qualitative Simulation Results.** The top row and bottom row compare results from 4m keyframe spacing and 2m keyframe spacing, respectively. Our proposed method's results are displayed in the right column in blue, and benchmark results (without Bayesian inference) are displayed in the left column in red.

This work proposes an inference method that divides the world into semantic classes by estimating each object class's geometry. These estimates are leveraged to perform 2D-to-3D inference over wide aperture multibeam sonar data. However, not all objects can be treated this way. For example, a wall detected in a sonar image will be difficult to accurately register to a reference coordinate frame, since it may not be fully captured within a single image. Moreover, a pier piling near the edge of a sonar image cannot be distinguished from the edge of a wall that extends beyond the image. Conversely, objects that fit entirely inside the sonar image can be estimated with some confidence, as the registration process described above is readily applicable. It is for this reason that we confine our inference framework in simulation to two classes, cylindrical and rectangular pier pilings. The remaining classes of boat hull, truss and wall, though present in our Gazebo model, are not considered in our predictive mapping framework. A 200-image per class hand-annotated dataset is used to train the CNN.

Recall that we use planar pose SLAM to provide transformations to the global map frame. To ensure accurate mapping, only imagery from SLAM keyframes is introduced as input to our framework. We compare two configurations of varying density in this study, a sparse keyframe spacing of 4 meters and a denser keyframe spacing of 2 meters. Note that in this simulation study, while we utilize the same downsampling as a real SLAM solution, we provide the SLAM pipeline with the robot's true pose in order to isolate errors

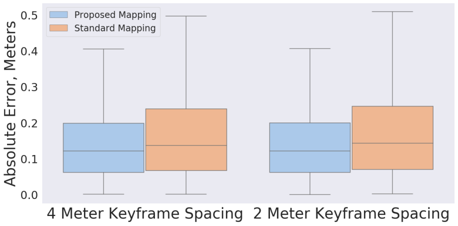


Fig. 4: **Simulation Error.** On the left we show the distribution of absolute errors for a keyframe spacing of 4m. On the right we show the distribution of absolute errors for a keyframe spacing of 2m. Errors of our proposed method are displayed in blue; the gold shows errors of simply registering dual sonar fusion outputs. Outliers not shown comprise between 1.7-2.0 percent of each dataset.

to only the 3D reconstruction system. As a benchmark for comparison we use a simple method, registering the dual sonar fusion system's 3D output over each keyframe, without classification or Bayesian inference. While naive, this method represents the state of the art in 3D reconstruction with wide aperture multibeam sonar [1], while minimizing assumptions with regard to the appearance of the environment.

For every trial, our robot navigates to each of 20 randomly sampled goals, sequencing them using nearest neighbor; starting at a random location. Collision avoidance is achieved using a 2D, in-plane roadmap for navigation purposes only, which is in no way used in our 3D mapping algorithm. We use A* in conjunction with the roadmap to generate trajectories from one goal to the next. Ten trials are run for each SLAM configuration. Fig. 4 shows that our predictive 3D mapping method has comparable error values to the benchmark 3D mapping method. Moreover, per Table I, the proposed method provides an order of magnitude greater coverage for each SLAM configuration. Most critically though, our method with half as many keyframes (spaced 4m apart) has an order of magnitude better coverage than the benchmark with twice as many keyframes (2m). Qualitative results are shown in Fig. 3, which clearly illustrates the improvement in point cloud density over repeating objects, in this case the two different types of pier pilings, without impacting the reconstruction of the trusses, boats, or seawall.

Algorithm	Mean Voxel Count	Std. Deviation
Standard Mapping, 4 Meter	2938.6	984.18
Semantic Mapping, 4 Meter	**60091.4**	**20038.116**
Standard Mapping, 2 Meter	3549.4	1144.72
Semantic Mapping, 2 Meter	**68422.2**	**25518.37**

TABLE I: **Simulation coverage results**, where point clouds are voxelized using a 10cm grid cell resolution.

We use two metrics to quantify performance: absolute error and voxel count. Absolute error is calculated by finding the shortest distance between a given point in the final point cloud and the CAD model of the environment (Fig. 4). Voxel count (Table I) is calculated by taking the final point cloud and voxelizing it, iterating over all the points and placing them in their respective discrete bins. If a voxel contains one or more points, it is counted; otherwise, it is not. This study uses voxel count to quantify coverage, while not using redundant information contained in the point cloud.

	Voxel Count	
Algorithm	2m Keyframe	4m Keyframe
Standard Mapping, Pier	4147	2040
Semantic Mapping, Pier	**38143**	**20766**
Standard Mapping, Waterfront	5891	2819
Semantic Mapping, Waterfront	**32131**	**13135**

TABLE II: **Field coverage results**, where point clouds are voxelized using a 10cm grid cell resolution. "Pier" (shown in Fig. 1) and "Waterfront" refer to two large structures in the SUNY Maritime marina, detailed in our video attachment.

C. Field Results

We next apply our method to real sonar data, using two datasets collected in SUNY Maritime College's marina on the East River in Bronx, NY. This environment represents the canonical humanmade littoral environment populated by piers, seawalls, and steel floating docks. Further, the East River poses serious environmental challenges such as low visibility, drastic tidal changes, and currents up to 2 knots.

To generate data, our BlueROV is manually piloted along the perimeter of structures in the marina while the vehicle heading is held near constant, with the vehicle strafing to either port or starboard. In the 300-image, hand-annotated dataset used to train our system, we use two classes: cylindrical pier piling and wall. Due to the aforementioned uncertainty associated with edge features, we only apply our inference method to the cylindrical pier piling class.

Since ground truth for mapping is not available, we analyze voxel count to measure coverage and assess the resulting point clouds. We produce separate maps for two structures in different areas of the marina, which we term "Pier" and "Waterfront". Once again, our method provides a dramatic increase in coverage, mapping many areas the benchmark leaves blank. Coverage results are shown in Table II, and detailed maps are provided in our video attachment. An overview of the Pier dataset is provided in Fig. 1. Roughly 3000 object classifications occur in this dataset. While we only apply our inference method to one class, because this class is constantly repeated throughout the environment, map coverage is greatly improved. Most critically, these datasets show the utility of our method on real-world sonar data gathered in a complex littoral environment.

VI. Conclusions

In this paper we have proposed using semantic classes to aid 2D-to-3D Bayesian inference over wide aperture multibeam sonar data. We have shown through experimental validation that exploiting the repeated observation of common structures in a littoral environment can permit highly accurate predictive mapping, without the need for CAD models a priori. In the absence of these structures, this method can still be employed, but with reduced coverage.

Acknowledgments

This research was supported by a grant from Schlumberger Technology Corporation - we thank Arnaud Croux for guidance and support. We also thank Robert Crafa for facilitating access to the SUNY Maritime College marina.

REFERENCES

[1] J. McConnell, J.D. Martin, and B. Englot “Fusing Concurrent Orthogonal Wide-aperture Sonar Images for Dense Underwater 3D Reconstruction,” *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1653-1660, 2020.

[2] S. Negahdaripour, “Analyzing Epipolar Geometry of 2-D Forward-Scan Sonar Stereo for Matching and 3-D Reconstruction,” *Proceedings of the IEEE/MTS OCEANS Conference*, 2018.

[3] S. Negahdaripour, “Application of Forward-Scan Sonar Stereo for 3-D Scene Reconstruction,” *IEEE Journal of Oceanic Engineering*, vol. 45, no. 2, pp. 547-562, 2020.

[4] “Mapbox,” *https://www.mapbox.com/*

[5] M.D. Aykin and S. Negahdaripour, “Forward-look 2-D sonar image formation and 3-D reconstruction,” *Proceedings of the IEEE/MTS OCEANS Conference*, 2013.

[6] E. Westman and M. Kaess, “Wide Aperture Imaging Sonar Reconstruction using Generative Models,” *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 8067-8074, 2019.

[7] M.D. Aykin and S. Negahdaripour, “On 3-D Target Reconstruction from Multiple 2-D Forward-Scan Sonar Views,” *Proceedings of the IEEE/MTS OCEANS Conference*, 2015.

[8] M.D. Aykin and S. Negahdaripour, ”Three-Dimensional Target Reconstruction From Multiple 2-D Forward-Scan Sonar Views by Space Carving,” *IEEE Journal of Oceanic Engineering*, vol. 42, no. 3, pp. 574-589, 2017.

[9] T. Guerneve, K. Subr, and Y. Petillot, “Three-dimensional Reconstruction of Underwater Objects using Wide-aperture Imaging SONAR,” *Journal of Field Robotics*, vol. 35, no. 6, pp. 890-905, 2018.

[10] E. Westman, I. Gkioulekas and M. Kaess, “A Theory of Fermat Paths for 3D Imaging Sonar Reconstruction,” *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5082-5088, 2020.

[11] R. DeBortoli, F. Li, and G.A. Hollinger, “ElevateNet: A Convolutional Neural Network for Estimating the Missing Dimension in 2D Underwater Sonar Images,” *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 8040-8047, 2019.

[12] H.W. Yu and B.H. Lee, “A Variational Feature Encoding Method of 3D Object for Probabilistic Semantic SLAM,” *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3605-3612, 2018.

[13] H.W. Yu, J.Y. Moon and B.H. Lee, “A Variational Observation Model of 3D Object for Probabilistic Semantic SLAM,” *Proceedings of the International Conference on Robotics and Automation*, pp. 5866-5872, 2019.

[14] B. Yang, S. Rosa, A. Markham, N. Trigoni and H. Wen, “Dense 3D Object Reconstruction from a Single Depth View,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 12, pp. 2820-2834, 2019.

[15] R.F. Salas-Moreno, R.A. Newcombe, H. Strasdat, P.H.J. Kelly, and A.J. Davison, “SLAM++: Simultaneous Localisation and Mapping at the Level of Objects,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1352-1359, 2013.

[16] T. Guerneve, K. Subr and Y. Petillot, “Underwater 3D structures as semantic landmarks in SONAR mapping,” *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 614-619, 2017.

[17] J. Wang and B. Englot, “Fast, Accurate Gaussian Process Occupancy Maps via Test-data Octrees and Nested Bayesian Fusion,” *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1003-1010, 2016.

[18] K. Doherty, J. Wang, and B. Englot, “Learning-aided 3-D Occupancy Mapping with Bayesian Generalized Kernel Inference,” *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 953-966, 2019.

[19] C. O’Meadhra, W. Tabib, and N. Michael, “Variable Resolution Occupancy Mapping Using Gaussian Mixture Models,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 2015-2022, 2019.

[20] L. Gan, R. Zhang, J.W. Grizzle, R.M. Eustice, and M. Ghaffari, “Bayesian Spatial Kernel Smoothing for Scalable Dense Semantic Mapping,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 790-797, 2020.

[21] F. Dellaert, “Factor Graphs and GTSAM: A Hands-on Introduction,” Georgia Institute of Technology, Technical Report No. GT-RIM-CPR-2012-002, 2012.

[22] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert, “iSAM2: Incremental Smoothing and Mapping using the Bayes Tree,” *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216-235, 2012.

[23] P.J. Besl and N.D. McKay, “A Method for Registration of 3-D Shapes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*. vol. 14, no. 2, pp. 239–256, 1992.

[24] M. Fischler and R. Bolles, “Random Sample Consensus: A Paradigm for Model Fitting with Application to Image Analysis and Automated Cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381-395, 1981.

[25] J.G. Mangelson, D. Dominic, R. M. Eustice and R. Vasudevan, “Pairwise Consistent Measurement Set Maximization for Robust Multi-Robot Map Merging,” *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2916-2923, 2018.

[26] X. Hu and P. Mordohai, ”Evaluation of Stereo Confidence Indoors and Outdoors,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1466-1473, 2010.

[27] M. Richards, *Fundamentals of Radar Signal Processing*, McGraw Hill, 2005.

[28] M. Ester, H. Kriegel, J. Sander, and X. Xu, “A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise,” *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pp. 226-231, 1996.

[29] A. Loquercio, M. Segu and D. Scaramuzza, “A General Framework for Uncertainty Estimation in Deep Learning,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3153-3160, 2020.

[30] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Ng, “ROS: An Open-source Robot Operating System” *IEEE ICRA Workshop on Open Source Software*, 2009.

[31] N. Koenig and A. Howard, “Design and Use Paradigms for Gazebo, an Open-source Multi-robot Simulator,” *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, pp. 2149-2154, 2004.

[32] M.M.M. Manhães, S.A. Scherer, M. Voss, L.R. Douat and T. Rauschenbach, “UUV Simulator: A Gazebo-based Package for Underwater Intervention and Multi-robot Simulation,” *Proceedings of the IEEE/MTS OCEANS Conference*, 2016.

[33] R. Cerqueira, T. Trocoli, G. Neves, S. Joyeux, J. Albiez, and L. Oliveira, “A Novel GPU-based Sonar Simulator for Real-time Applications,” *Computers and Graphics*, vol. 68, pp. 66-76, 2017.