Sonar-Aided Manipulation in Low-Visibility Conditions by Novice Users

Amy Phung*†, Gideon Billings§, and Richard Camilli†
*Massachusetts Institute of Technology, Cambridge, MA
†Woods Hole Oceanographic Institution, Woods Hole, MA
§Australian Centre for Robotics, The University of Sydney, Sydney NSW 2006, Australia

***Abstract*—Underwater intervention tasks in the deep ocean are typically completed with remotely operated vehicles (ROVs) equipped with robotic manipulator arms, and rely on optical-based perception of the scene to guide the manipulation tasks. However, the performance of optical sensors is highly degraded in turbid water conditions, which can arise from a variety of causes. This work investigates the use of an imaging sonar and doppler velocity log mounted on the wrist of a manipulator for close-range scene mapping to aid in underwater intervention. We integrate this sonar-based perception method with a shared autonomy framework to facilitate safe intervention tasks by users without ROV piloting experience in turbid environments, and conduct experiments to validate our system.**

***Index Terms*—Underwater manipulation, imaging sonar, doppler velocity log, eye-in-hand perception**

I. Introduction

PERFORMING underwater intervention tasks in low-visibility conditions remains an outstanding challenge for existing underwater manipulation systems. Typically, intervention tasks (e.g., scientific seafloor sampling, infrastructure maintenance) with work-class systems are completed by trained pilots who teleoperate remotely operated vehicles (ROVs) equipped with robotic manipulator arms. ROV pilots complete intensive training to operate these vehicles safely, and considerable ship-side infrastructure is necessary to support teleoperation [1]. These factors combined with high operational expense and the limited berthing available onboard the support ships restricts access to underwater intervention operations.

Several prior works aim to mitigate these barriers to access by integrating various degrees of robot autonomy to reduce the training and infrastructure requirements [2]–[5]. However, due to their reliance on optical cameras for perception, both teleoperated and autonomy-based ROV operations can be significantly hampered by turbid conditions [6]–[8]. Unfortunately, such conditions frequently occur due to a variety of natural and anthropogenic causes, including but not limited to wind and waves [9], marine traffic [10], and urban runoff [11]. Thus, operating in these environments requires a robust method of perception that remains functional in visually degraded conditions. For particularly sensitive applications, such as unexploded ordinance (UXO) remediation, operational robustness is especially critical since errors can pose a serious threat to human health and the environment.

Automated optical 3D reconstruction methods typically employed onboard robotic systems often rely on the detection of natural features in the environment, which leads to poor reconstruction quality in areas with turbid water or suboptimal lighting conditions [8]. Compared with optical imaging, the longer wavelength of acoustic sensors reduces their susceptibility to signal attenuation in turbid conditions. This attribute is potentially useful for extending manipulation capabilities to low-visibility environments.

ROVs are typically equipped with sonars for navigation and obstacle avoidance. Forward-looking imaging sonars typically have a wide vertical aperture which increases their spatial coverage and provides pilots with context of the ROV's surroundings. However, this wide aperture results in elevation angle ambiguity when using the data for mapping construct maps. Resolving this ambiguity to create 3D scene reconstructions remains an area of active research, with prior works being roughly grouped into feature based methods [12], model-based [13], [14], and volumetric techniques [15]. Feature based methods and generative models typically require prior knowledge or make assumptions about geometry (e.g., flat seafloor and monotonic surfaces [16]), vehicle motion (e.g., rectilinear transits using passively roll and pitch stable vehicles [14]), or material properties [13]. Volumetric techniques reconstruct the environment based on intersecting regions among a collection of sonar images [17], [18]. Unlike other mapping methods, volumetric methods do not impose constraints on environmental structures or vehicle motion. Rather, the quality of the reconstructions with these methods benefit from collecting data across a wide variety positions and orientations. Since robotic manipulators often operate across all six degrees of freedom with nonlinear velocities in irregular and potentially unknown environments, volumetric methods show promise in enabling intervention operations in low-visibility conditions.

In this work, we address the challenge of underwater intervention in turbid conditions by employing a volumetric sonar-based perception method during manipulation tasks, which has the benefit of remaining operational regardless of visibility and lighting conditions. Through our experiments, we evaluate the

This work was supported by the Strategic Environmental Research and Development Program Grant W912HQ24P0024. Amy Phung would like to acknowledge financial support from the National Science Foundation Graduate Research Fellowship (No. 2141064), from the National Aeronautics and Space Administration (NASA) through the FINESST program (No. 80NSSC23K1391), and the Link Foundation's Ocean Engineering & Instrumentation Fellowship Program

efficacy of mapping an unstructured workspace using wrist-mounted sonars. Additionally, we examine their utility in assisting novice users (i.e., non-ROV pilots) with intervention tasks in low-visibility conditions through integration with a shared autonomy framework.

II. Methods

We adopt an approach that integrates acoustic perception methods with a shared autonomy framework to enable reliable underwater intervention in low-visibility conditions. This approach leverages the complementary strengths of robotic systems and human operators: the robot automates the low-level control, motion planning, and spatial reasoning of the data, while human operators contribute their expertise in identifying objects, discerning erroneous data, and making informed decisions. In this section, we introduce an acoustic eye-in-hand workspace mapping approach, present a real-time volumetric mapping method using multibeam data, and detail their integration with a shared autonomy framework to render the data in real-time for the end user.

A. Acoustic Eye-In-Hand Workspace Mapping

To identify objects and obstacles in the workspace during intervention, we integrate a wrist-mounted multibeam imaging sonar, doppler velocity log (DVL), and optical camera on a robotic manipulator arm. The "eye-in-hand" approach of mounting sensors on the end-effector of a robotic manipulator is often adopted with optical cameras since the manipulator's dexterity allows for flexible viewing angles, which can be useful for structure from motion (SFM)-based mapping approaches [19]. We leverage the manipulator's ability to acquire multiple viewing angles to resolve the imaging sonar's elevation angle ambiguity, which is discussed in Section II-B. Although the focus of our work is on sonar-aided perception, we include an optical camera to aid in close-range perception (<30cm), particularly when the distance between the gripper and the environment is within the blanking distance of the sonars.

The DVL and camera are mounted facing downward towards the gripper, while the multibeam sonar faces forward, away from the base of the arm. Representative data examples from using this sensor configuration are illustrated in Figure 2. This configuration optimally positions the multibeam sonar to perceive the arm's workspace and allows for the construction of an occupancy grid of the area in front of the manipulator with minimal movement (Figure 3). Although DVLs are typically employed for estimating vehicle velocity and dead-reckoning based navigation, in this method it is used primarily for assessing occupancy volumes and workspace geometry directly beneath the gripper. The DVL complements the multibeam sonar by providing vertical range data using a narrow beamwidth, which helps to counteract the vertical "stretching" distortion caused by the multibeam's wide vertical aperture.

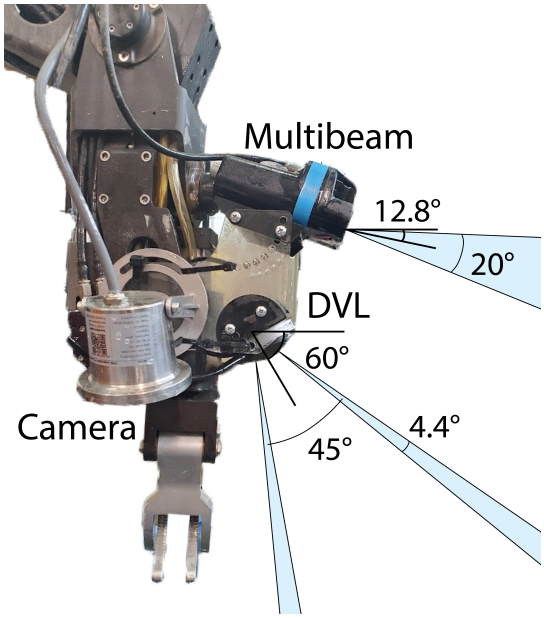

Fig. 1. Wrist-mounted multibeam sonar, DVL, and camera, with annotated fields of view. With the manipulator's gripper pointed downwards, the multibeam and DVL sonars are mounted 12.8 and 60 degrees from horizontal, respectively.

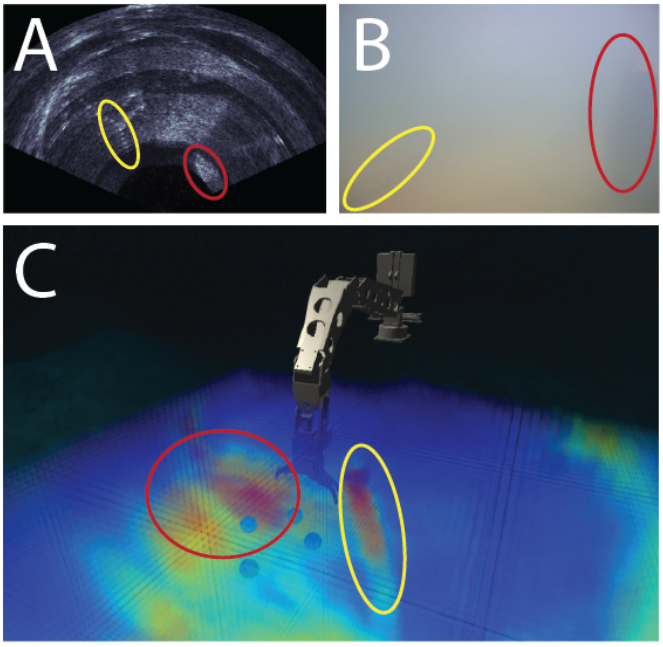

Fig. 2. Representative images from the (A) multibeam sonar and (B) optical camera in a turbid test tank. (C) The 3D occupancy map generated from the multibeam data, with DVL data projected on top (green). While a chain (yellow) and rock (red) are visible in the sonar data, they are difficult to identify in the camera data due to the turbidity level.

B. Multibeam Occupancy Mapping

Our method, which we call the min-max filtering approach, builds upon prior work using volumetric reconstruction methods which evaluate the intersecting regions among a collection of sonar images [17], [18]. To maximize its utility in a

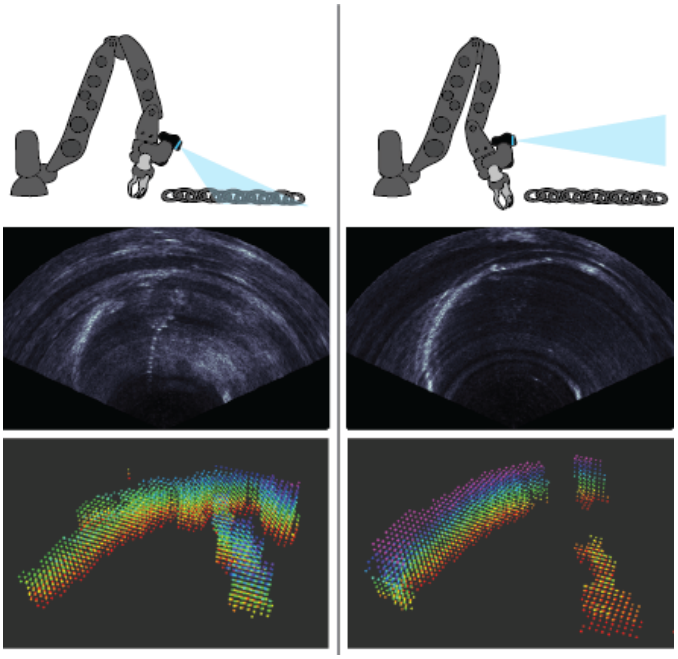

Fig. 3. Pitching the multibeam sonar quickly resolves elevation ambiguity while requiring minimal manipulator movement. (left) After the first image, the chain appears vertically stretched in the voxel grid. (right) After moving the arm slightly to adjust the sonar pitch angle, the new data resolves the ambiguity and corrects the chain's height in the voxel grid.

workspace manipulation context, we prioritize its design to be real-time capable and require minimal assumptions about the workspace geometry or sensor motion. This application also requires that the generated voxel grid maintains a low false-negative rate. While a moderate false-positive rate may cause the manipulator's accessible areas to appear smaller than their actual size, a false-negative can cause catastrophic collisions with the workspace if the manipulator incorrectly classifies a voxel as being unoccupied. The min-max filtering process is described in detail below:

Initialization: Construct Voxel Grid We first initialize a voxel grid, which we store as a 3D meshgrid. The meshgrid is box-shaped, with the dimensions of the grid fully containing the manipulator workspace. The spacing between the points in the grid are set to a fixed voxel size. We refer to the set of points in this meshgrid as set M, and additionally define the following notation:

$\mathbf{q} = (i, j, k)$: grid voxel at index (i,j,k)

$\mathbf{p} = (x, y, z)$: arbitrary cartesian point in world frame

$\bar{\mathbf{p}} = (x, y, z)$: meshgrid-aligned point in world frame

$= \mathbf{q} * \mathbf{s} - \bar{\mathbf{o}}$

where $\mathbf{s}$ is the voxel size, and $\bar{\mathbf{o}}$ is the voxel grid origin

We refer to individual points in the voxel grid with the notation $M({}^W\mathbf{q}^V)$, where ${}^W\mathbf{q}^V$ refers to the index (i,j,k) of voxel V, expressed in the world frame W. Equivalently, we can refer to the same points with the notation $M({}^W\bar{\mathbf{p}}^V)$, where ${}^W\bar{\mathbf{p}}^V$ refers to the cartesian position of voxel V in the world frame.

With a fixed-base manipulator, the size of the reachable workspace is known in advance, and thus the memory for our voxel grid can be pre-allocated. We use a 3D meshgrid to prioritize the read and write speeds of our voxel grid, which contributes to our method's real-time performance.

Initialization: Construct Voxel Template Next, we construct a "voxel template" to discretize the area imaged by the multi-beam sonar. We start by initializing a second 3D meshgrid with the following dimensions to fully contain the sonar's field-of-view:

$$
\begin{aligned}
&x : r \\
&y : r * \cos(\pi/2) - \text{horizontal FOV} \\
&z : r * \sin(\pi/2) - \text{vertical FOV} \\
&\text{where } r = \text{sonar range}
\end{aligned}
$$

The spacing between the points is set to the fixed voxel size from the prior step. We refer to individual points in the voxel template T with the notation $T({}^S\mathbf{p}^V)$, where ${}^S\mathbf{p}^V$ refers to the cartesian position of voxel V relative to the sonar's frame S. The set of points contained in T only contains the meshgrid points within the sonar's FOV. Formally,

$$
\begin{aligned}
T := &\left\{ {}^S\mathbf{p}^V \;\middle|\; {}^S\mathbf{p}^V \text{ satisfies the following constraints} \right\} \\
&\text{a) } \|{}^S\mathbf{p}^V\| < \text{sonar range} \\
&\text{b) } \left| \arctan\left(\frac{{}^S\mathbf{p}^V_y}{{}^S\mathbf{p}^V_x} \right) \right| < \frac{\text{horizontal FOV}}{2} \\
&\text{c) } \left| \arctan\left(\frac{{}^S\mathbf{p}^V_z}{{}^S\mathbf{p}^V_x} \right) \right| < \frac{\text{vertical FOV}}{2}
\end{aligned}
$$

To illustrate through an example, the set of points contained within the voxel template T for a voxel size of 0.05 meters is depicted in Figure 4.

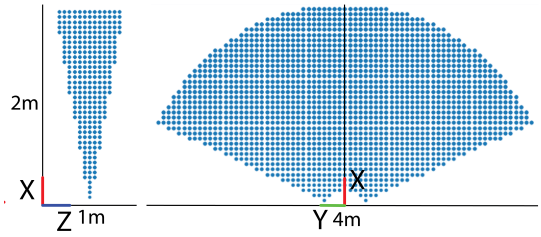

Fig. 4. (left) X-Z and (right) X-Y view of cartesian points included in the voxel template. Points are plotted in the multibeam sonar's reference frame

This voxel template is used to reduce the computational resources required to project the sonar data into the global frame. Rather than projecting each individual pixel in the image, we use the template to project the minimum number of points to sufficiently represent the data, based on the voxel size.

Step 1: Project Template into World Frame For each new multibeam image, we use the template to project the sonar's field of view into the world frame. For each voxel ${}^S\mathbf{p}^V$ in the template T, we compute its position in the world frame W

$$^{W}\mathbf{p}^{V} = {}^{W}\mathbf{X}^{S}\,{}^{S}\mathbf{p}^{V}$$

where ${}^{W}\mathbf{X}^{S}$ denotes the transformation between the world (W) and the sonar (S). ${}^{W}\mathbf{X}^{S}$ is estimated using the manipulator arm's joint angle sensors and its forward kinematics model.

Step 2: Compute Nearest Grid Voxel Indices Given the set of template voxel points in the world frame ${}^{W}\mathbf{p}^{V}$, we compute the nearest grid voxel index to each point using the following equation:

$$^{W}\mathbf{q}^{V} = \text{round}\left(\frac{^{W}\mathbf{p}^{V} + \bar{\mathbf{o}}}{\mathbf{s}}\right)$$

We disregard invalid values of ${}^{W}\mathbf{q}^{V}$ (i.e., where indices are less than 0 or greater than the size of the map). Invalid indices indicate that the corresponding template voxel falls outside of our voxel grid (and thus our manipulator workspace). Using this indexing approach enables us to identify the intersecting region between the sonar data and the voxel grid without projecting each individual pixel or iterating thorugh the entire grid.

Step 3: Convert Grid Voxel Indices to World Coordinates Our set of ${}^{W}\mathbf{q}^{V}$ values identifies the voxels in our voxel grid that intersect with the sonar data. The next step is to compute the cartesian position of each of these voxels in the world frame, which can be written as follows:

$$^{W}\bar{\mathbf{p}}^{V} = {}^{W}\mathbf{q}^{V} * \mathbf{s} - \bar{\mathbf{o}}$$

Step 4: Transform Voxel Positions to Sonar Frame To convert from voxels in the world frame to the sonar frame, we multiply each voxel coordinate ${}^{W}\bar{\mathbf{p}}^{V}$ by the inverse of the transformation matrix between the world and sonar frames (${}^{W}\mathbf{X}^{S}$ is given in step 1)

$$^{S}\bar{\mathbf{p}}^{V} = \left[{}^{W}\mathbf{X}^{S}\right]^{-1}\,{}^{W}\bar{\mathbf{p}}^{V}$$

Step 5: Compute Max Intensity of Projected Voxel For each value of ${}^{S}\bar{\mathbf{p}}^{V}$, we record the max intensity value contained within the voxel based on its projection on the sonar image (Figure 5).

$$I\left({}^{S}\bar{\mathbf{p}}^{V}\right) = \text{max intensity in projected voxel}$$

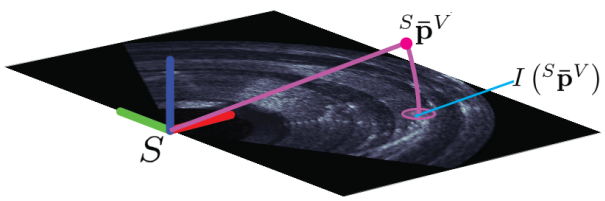

Fig. 5. In Step 5, the voxel associated with each value of ${}^{S}\bar{\mathbf{p}}^{V}$ is projected onto the cartesian multibeam image. The max intensity of the projected voxel in the new data is compared with the voxel's previous intensity value.

Step 6: Update Voxel Grid Finally, we update the voxel grid based on the value of $I\left({}^{S}\bar{\mathbf{p}}^{V}\right)$. If the existing value of $M(i,j,k)$ is uninitialized or less than the new intensity value, the voxel grid is updated using the new value. Formally,

$$M(i,j,k) = \max\left(M(i,j,k), I\left({}^{S}\bar{\mathbf{p}}^{V}\right)\right)$$

C. Shared Autonomy Integration

We integrate the manipulator joint angle feedback, camera data, sonar image, projected DVL data, and the computed voxel grid with the shared autonomy framework SHARC [2]. The SHARC-VR interface (Figure 6) provides the end-user a virtual reality (VR)-based interface for viewing the data in context of the manipulator arm, and automates the low-level motion planning and control to safely enable operations by non-pilot users. In addition to rendering the projected DVL and multibeam data in context of the manipulator arm, we provide the user with the option to view the raw 2D camera and sonar images directly.

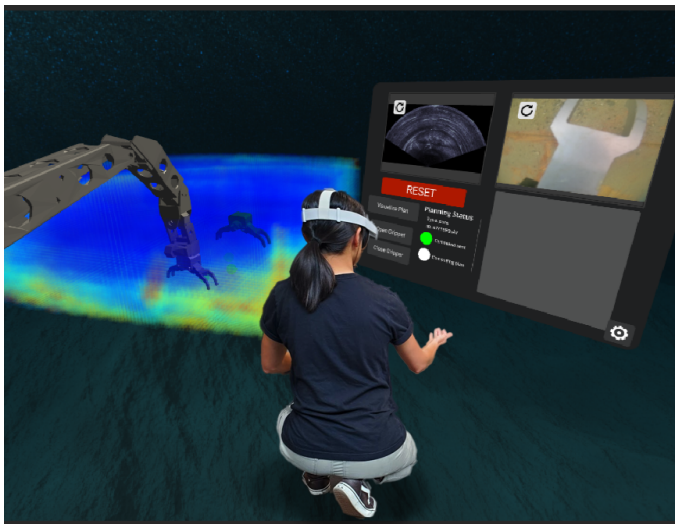

Fig. 6. Mixed-reality view of the SHARC-VR with integrated data feeds from the multibeam sonar, DVL, and optical camera. Hand gestures are used to interact with the interface and control the manipulator.

III. Experimental Setup

To validate our approach, we conduct experiments with an Oculus M1200d multibeam imaging sonar (Blueye; Trondheim, Norway) operating at 1.2 MHz and a DVL A50 (Water-Linked; Trondheim, Norway) operating at 1 MHz with a 10 Hz update rate for all sensors (Figure 1). The Oculus sonar was configured with a gain setting of 40% and a range of 2 m. These sensors are mounted on the wrist of a fixed-base seven-degree-of-freedom hydraulic manipulator arm (Kraft TeleRobotics; Overland Park, KS) and are used to image various objects in a sandbox submerged in a turbid 0.6 m deep test tank (Figure 7). All of the sensors communicated via a wired Ethernet connection to a desktop computer running ROS1. A separate laptop connected to the same wired network ran the SHARC-VR interface (developed in Unity), and displayed the interface in an Oculus Quest 2 headset (Meta; Menlo Park, CA).

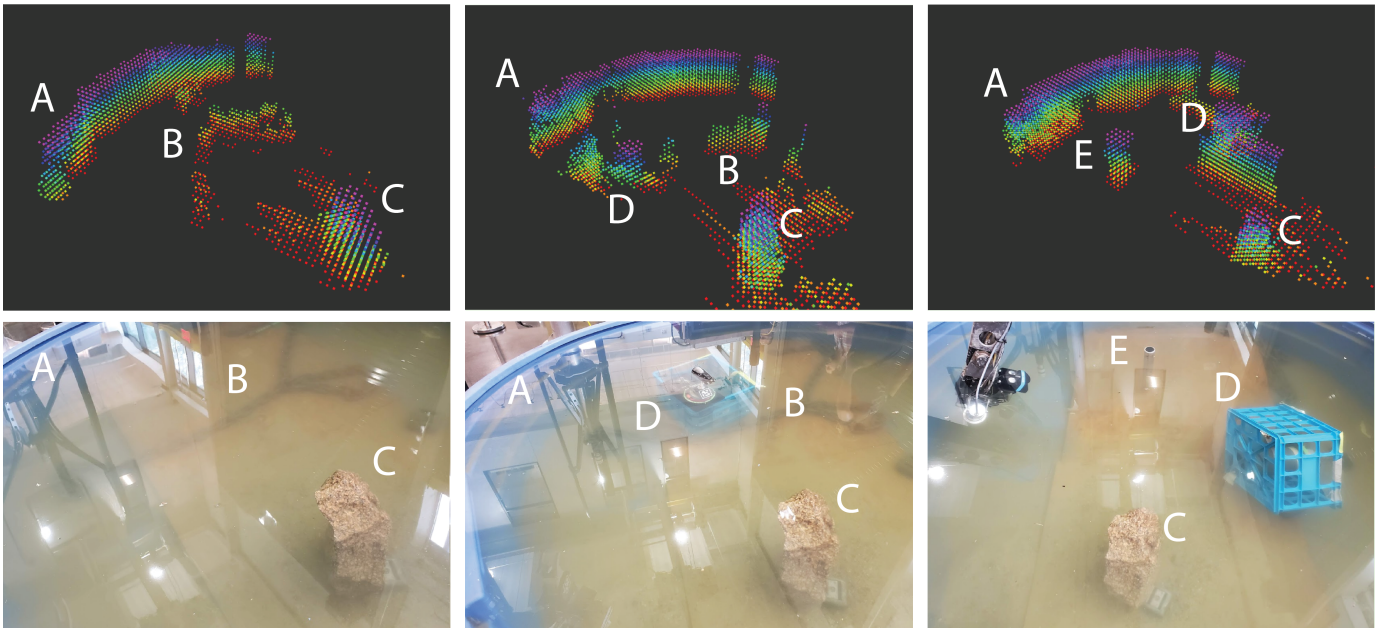

Fig. 7. (Top) 3D reconstruction results from our min-max filtering approach with the multibeam data. Although unneccesssary when viewing in 3D, in this figure we omit data points below a certain intensity value, crop the data to exclude the bottom of the tank and regions outside of it, and color the points based on the Z-axis value to aid in interpretation. (A) The tank walls, (B) chain, (C) rock, (D) tool basket, and (E) metal pipe in the reconstruction closely match their actual locations in the tank (Bottom).

IV. Results

Results using the min-max filtering method for multibeam data recorded using the experimental setup described in Section III are presented in Figure 7 and Table IV. Testing used a voxel grid size of 4x3x1 meters, with a 0.03 meter voxel size (∼450,000 voxels).[1] The voxel grid was constructed by initially rotating the sonar pitch with the arm in place (before constructing the grid, we assume that only the area immediately around the arm is likely to be free of obstacles). Once a preliminary grid was constructed, we identify regions with low-intensity voxels in the grid as safe regions within the workspace where the arm can then be moved to improve data coverage.

TABLE I
Min-Max Filtering Performance Metrics

Voxel size (m)	Time per Image (s)	Update Rate (Hz)
0.06	0.07	14.3[1]
0.05	0.12	8.3
0.04	0.25	4
0.03	0.52	1.9
0.02	1.6	0.6
0.01	12	0.08

[1] The maximum update rate for the Oculus sonar is 10 Hz

[1]In this implementation, the primary performance bottlenecks are attributable to 3D rendering and data conversion processes. With the workspace gridded at ∼50,000 voxels, the renderer started to to lag, and data conversion requirements decreased the voxel grid update rate to 2 Hz. We circumvented these performance issues by only sending a subset of the full voxel grid to the SHARC-VR interface.

Although the figure illustrates thresholded and cropped data with points colored according to their Z-axis value for ease of interpretation in 2D, when rendering the data in 3D we display all of the data and color the points based on the recorded intensity value. As evident by Figure 7, The tank walls, chain, rock, tool basket, and metal pipe are recognizable in the 3D reconstruction.

Although the objects appear vertically stretched in voxel grid after the first few sonar images due to the multibeam's wide vertical aperture, their height in the final reconstruction accurately reflects their true size. For example, in Figure 7 the additional viewing angles generated by moving the manipulator across a variety of pitch angles and locations helps to resolve the chain's elevation angle ambiguity from individual images.

It is also worth noting that in the full voxel grid, multipath artifacts appear beyond the edge of the tank as a result of beams reflecting off the bottom of the tank and echoing off of the water surface. During our testing, these artifacts did not affect our ability to conduct manipulation tasks since they generally appeared well beyond the reachable area of the manipulator. Artifacts that did appear within the manipulator's workspace were removed by the min-max filter as data was recorded from additional view angles.

The raw camera and sonar images, projected DVL data, and voxel grid were rendered in the SHARC-VR interface in real-time, which enabled non-pilot users to retrieve objects from the tank despite the turbid conditions. While the projected data proved to be useful for understanding the workspace geometry relative to the arm, it was easier for users to identify objects using the sonar and camera images directly. The voxel grid

was particularly helpful for navigating the arm around the workspace while avoiding obstacles. Once the manipulator was close to the object (within ∼10 cm), the user could switch over to using the camera and DVL data for alignment. Although it was difficult to see objects in the camera image at a distance due to the turbidity (Figure 2B), objects were generally visible at shorter ranges (Figure 8-4). The process used to retrieve objects from the tank using the interface is illustrated in Figure 8.

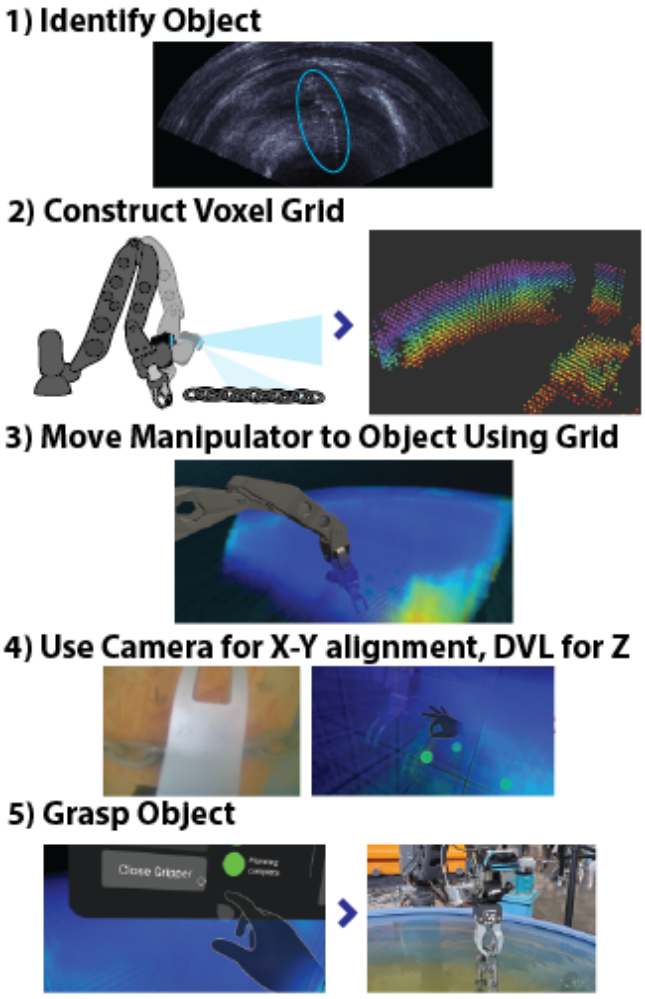

Fig. 8. Illustrated workflow of an object retrieval task in turbid water using the available data in the SHARC-VR interface.

V. Discussion

For each new multibeam image, the min-max filtering approach identifies the maximum intensity value contained within a voxel in the new data, and updates the voxel grid with the minimum between the new and the existing data in the map. By recording the maximum intensity value over a region and using a "voxel template" to represent the data rather than processing and reprojecting individual pixels in the image, our method minimizes computational load and can process multibeam data in real time. Despite this approximation, this approach maintains a sufficient degree of robustness by minimizing false-negatives in the generated map. Objects such as thin metal pipes can show up as very small but strong high-intensity returns across a limited number of voxels, and thus recording the maximum intensity value contained within a voxel prevents the method from over-estimating the free space in the voxel grid. While recording the lower intensity value between new voxel data and the current voxel grid helps to resolve the elevation ambiguity in static workspaces, this method is inadequate for characterizing dynamic obstacles in the work environment. Rather than simply recording the lowest intensity value, future work on quantifying uncertainty based on the new data can help to address this limitation.

Although the min-max filtering approach generally resolved multi-path artifacts within the manipulator's reachable workspace with sufficient data, multi-path artifacts persisted beyond the edges of the workspace. While this had minimal effect on conducting manipulation tasks in a confined tank since it can be assumed that returns beyond the tank's walls were invalid, it could make object identification more difficult in open-water settings. However, most of the multi-path artifacts were caused by sonar beams reflecting off extremely close proximity of the tank walls and water surface, and thus these artifacts may appear less pronounced in open-water settings, which will be explored in future work.

The current implementation of the min-max filtering approach focuses on generating a real-time capable method that produces robust maps with a low false-negative rate, which are essential properties of a mapping method in a sonar-aided manipulation context. However, it currently ignores issues of navigation drift and scalability, both of which need to be addressed in future work for the method to be used for free-floating manipulation.

As discussed in Section IV, the min-max filtering approach and additional data from a DVL and camera enabled the completion of object retrieval tasks in turbid conditions using the SHARC-VR interface. Minimal manipulator movements were required to pitch the multibeam sonar to resolve the elevation angle ambiguity in the sonar data and construct a voxel grid sufficient for manipulation. Meanwhile, the camera and DVL data proved to be useful for small adjustments to align the gripper with objects in the tank.

VI. Conclusion

In this paper, we present a new sonar-based structure from motion method for underwater scene reconstruction and introduce the min-max filtering method for constructing a 3D occupancy map. This process addresses the need for robust perception during intervention operations in low-visibility underwater environments. By utilizing sonar-based perception, we extend the range of operating conditions for underwater manipulation beyond the limitations posed by optical imagery. Through experiments and integration with a shared autonomy framework, we demonstrate the ability to create a 3D reconstruction of a manipulator workspace in real time and complete object retrieval tasks in turbid conditions. By augmenting perception capabilities in low-visibility conditions, this work

takes a step towards enabling safe intervention missions in areas currently intractable due to turbidity, and improving operational robustness in existing systems.

REFERENCES

[1] E. Simetti, "Autonomous underwater intervention," *Current Robotics Reports*, vol. 1, pp. 117–122, 2020.

[2] A. Phung, G. Billings, A. F. Daniele, M. R. Walter, and R. Camilli, "Enhancing scientific exploration of the deep sea through shared autonomy in remote manipulation," *Science Robotics*, vol. 8, no. 81, p. eadi5227, 2023.

[3] P. Di Lillo, E. Simetti, F. Wanderlingh, G. Casalino, and G. Antonelli, "Underwater intervention with remote supervision via satellite communication: Developed control architecture and experimental results within the dexrov project," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 1, pp. 108–123, 2020.

[4] J. Evans, P. Redmond, C. Plakas, K. Hamilton, and D. Lane, "Autonomous docking for intervention-auvs using sonar and video-based real-time 3d pose estimation," in *Oceans 2003. Celebrating the Past... Teaming Toward the Future (IEEE Cat. No. 03CH37492)*, vol. 4, pp. 2201–2210, IEEE, 2003.

[5] P. Cieslak, P. Ridao, and M. Giergiel, "Autonomous underwater panel operation by girona500 uvms: A practical approach to autonomous underwater manipulation," in *2015 IEEE International conference on robotics and automation (ICRA)*, pp. 529–536, IEEE, 2015.

[6] S. Sivčev, M. Rossi, J. Coleman, E. Omerdić, G. Dooly, and D. Toal, "Collision detection for underwater rov manipulator systems," *Sensors*, vol. 18, no. 4, p. 1117, 2018.

[7] S. Aldhaheri, G. De Masi, È. Pairet, and P. Ardón, "Underwater robot manipulation: advances, challenges and prospective ventures," in *OCEANS 2022-Chennai*, pp. 1–7, IEEE, 2022.

[8] G. Billings, M. R. Walter, O. Pizarro, M. Johnson-Roberson, and R. Camilli, "Towards automated sample collection and return in extreme underwater environments," *Field Robotics*, vol. 2, 2022.

[9] K. E. Fabricius, G. De'ath, C. Humphrey, I. Zagorskis, and B. Schaffelke, "Intra-annual variation in turbidity in response to terrestrial runoff on near-shore coral reefs of the great barrier reef," *Estuarine, Coastal and Shelf Science*, vol. 116, pp. 57–65, 2013.

[10] I. A. Callejas, C. M. Lee, D. R. Mishra, S. L. Felgate, C. Evans, A. Carrias, A. Rosado, R. Griffin, E. A. Cherrington, M. Ayad, *et al.*, "Effect of covid-19 anthropause on water clarity in the belize coastal lagoon," *Frontiers in Marine Science*, vol. 8, p. 648522, 2021.

[11] A. R. Orpin, P. V. Ridd, S. Thomas, K. R. Anthony, P. Marshall, and J. Oliver, "Natural turbidity variability and weather forecasts in risk management of anthropogenic sediment discharge near sensitive environments," *Marine Pollution Bulletin*, vol. 49, no. 7-8, pp. 602–612, 2004.

[12] J. Wang, T. Shan, and B. Englot, "Underwater terrain reconstruction from forward-looking sonar imagery," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 3471–3477, IEEE, 2019.

[13] R. DeBortoli, F. Li, and G. A. Hollinger, "Elevatenet: A convolutional neural network for estimating the missing dimension in 2d underwater sonar images," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8040–8047, IEEE, 2019.

[14] R. Ferretti, M. Bibuli, G. Bruzzone, A. Odetti, S. Aracri, C. Motta, M. Caccia, M. Rovere, A. Mercorella, F. Madricardo, *et al.*, "Acoustic seafloor mapping using non-standard asv: Technical challenges and innovative solutions," in *OCEANS 2023-Limerick*, pp. 1–6, IEEE, 2023.

[15] S. Arnold and B. Wehbe, "Spatial acoustic projection for 3d imaging sonar reconstruction," in *2022 International Conference on Robotics and Automation (ICRA)*, pp. 3054–3060, IEEE, 2022.

[16] M. D. Aykin and S. Negahdaripour, "Forward-look 2-d sonar image formation and 3-d reconstruction," in *2013 OCEANS-San Diego*, pp. 1–10, IEEE, 2013.

[17] 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.

[18] M. D. Aykin and S. Negahdaripour, "On feature matching and image registration for two-dimensional forward-scan sonar imaging," *Journal of Field Robotics*, vol. 30, no. 4, pp. 602–623, 2013.

[19] X. Zhi and S. Schwertfeger, "Simultaneous hand-eye calibration and reconstruction," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1470–1477, IEEE, 2017.