2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Neural Implicit Surface Reconstruction using Imaging Sonar

Mohamad Qadri, Michael Kaess, and Ioannis Gkioulekas

Abstract— **We present a technique for dense 3D reconstruction of objects using an imaging sonar, also known as forward-looking sonar (FLS). Compared to previous methods that model the scene geometry as point clouds or volumetric grids, we represent the geometry as a neural implicit function. Additionally, given such a representation, we use a differentiable volumetric renderer that models the propagation of acoustic waves to synthesize imaging sonar measurements. We perform experiments on real and synthetic datasets and show that our algorithm reconstructs high-fidelity surface geometry from multi-view FLS images at much higher quality than was possible with previous techniques and without suffering from their associated memory overhead.**

I. Introduction

Imaging or forward-looking sonar (FLS) is an extensively used sensor modality by Autonomous Underwater Vehicles (AUV). The key motivation for using FLS sensors is their ability to provide long-range measurements, unlike optical cameras whose range is severely limited in turbid water—a common situation in the field. Their versatility has resulted in their incorporation as a core sensor modality in applications including robotic path planning [1, 2], localization [3]–[7], and the automation of tasks potentially dangerous or mundane for humans such as underwater inspection [8] and mapping [9]–[12].

FLS outputs 2D image measurements of 3D structures by emitting acoustic pulses and measuring the energy intensity of the reflected waves. While the sonar resolves azimuth and range, the elevation angle is ambiguous, and an object at a specific range and azimuth can be located anywhere along the elevation arc. Hence, the task of 3D reconstruction using FLS measurements can be equivalently framed as the task of resolving the elevation ambiguity from the image readings.

Existing algorithms for 3D reconstruction from FLS measurements can be grouped into geometry-based, physics-based and, more recently, learning-based methods. However, most existing approaches either place restrictions on the robotic/sensor setup (elevation aperture, motion of the vehicle, etc.); rely on volumetric grids that are prohibitively expensive for large scenes or scenes with fine-grained geometry; or, specific to learning approaches, require the use of large labeled training sets that are difficult to collect in underwater environments.

To address these shortcomings, we approach the problem of underwater FLS-based 3D reconstruction through the lens of differentiable rendering and leverage the representational power of neural networks to encode the imaged object as an implicit surface. Our overall reconstruction approach comprises the following components:

- A differentiable volumetric renderer that models the propagation of acoustic spherical wavefronts.
- A representation of 3D surfaces as zero-level sets of neural implicit functions.
- A regularized rendering loss for 3D reconstruction using imaging sonars.

To the best of our knowledge, this work is the first to introduce a physics-based volumetric renderer suitable for dense 3D acoustic reconstruction using wide-aperture imaging sonars. We evaluate our approach against different unsupervised methods on simulated and real-world datasets, and show that it outperforms previous state of the art. We have made our code and different datasets publicly available[2].

II. Related Work

A. 3D Reconstruction Using Imaging Sonar

Different methods have been introduced to produce both sparse [5, 10, 11, 13]–[15] and dense 3D reconstructions using FLS. The focus of this work is on dense object-level 3D reconstruction. A number of algorithms enforced assumptions or constraints on the physical system to obtain reliable 3D models. Teixeira et al. [16] successfully reconstructed a 3D map of a ship hull by leveraging probabilistic volumetric techniques to create submaps which are later aligned using Iterative Closest Point (ICP). However, the sonar aperture was set to 1° and all detected points were assumed to lie on the zero-elevation plane which leads to reconstruction errors and prohibits extending the method to larger apertures.

A line of work [17]–[20] uses two complementary sensors (imaging and profiling sonars) and performs sensor fusion to disambiguate the elevation angle. In our work, we restrict our setup to a single imaging sonar. Westman et al. [21] proposed a method to reconstruct specific points on surfaces (aka. Fermat Paths). However, it places constraints on the vehicle's motion as it needs a view ray perpendicular to the surface at each surface point and hence, requires a large number of images collected from specific views.

Another set of methods uses generative models to obtain dense 3D reconstructions. Aykin et al. [22], [23] attempt to estimate the elevation angle of each pixel by leveraging information from both object edges and shadows which restricts the object to be on the seafloor. Westman et al. [24] further extended the idea to do away with the seafloor assumption but still required estimates of object edges. Negahdaripour et al. [25] proposed an optimization-based algorithm to refine an initial 3D reconstruction obtained using space carving by encouraging consistency between the actual sonar images and the images produced by the generative model. However, generative methods generally rely on assumptions of the

[1]M. Qadri, M. Kaess, and I. Gkioulekas are with The Robotics Institute, Carnegie Mellon University, USA. {mqadri, kaess, igkioule}@cs.cmu.edu

This work was supported by the Office of Naval Research grant N00014-21-1-2482 and National Science Foundation awards 1730147 and 1900849. Ioannis Gkioulekas was supported by a Sloan Research Fellowship.

[2]Code available at https://github.com/rpl-cmu/neusis

surface reflectivity proprieties and on 3D estimates of object edges which makes them impractical in real scenarios.

Various volumetric methods have also been proposed. Wang et al. [26] introduced an inverse sonar sensor model to update the occupancy in a voxel grid and later extended it to handle errors in pose estimates by aligning local submaps using graph optimization [27]. Although these methods, as probabilistic frameworks, can be more robust compared to space carving techniques [9, 22], they consider each voxel independently and ignore inherent surface constraints. Guerneve et al. [28] frame the problem of 3D volumetric reconstruction as a blind deconvolution with a spatially varying kernel which can be reformulated as a constrained linear least squares objective. However, the method makes a linear approximation to the vertical aperture and places restrictions on the motion of the sonar limiting its practical application. Westman et al. [29] noted the equivalence of 3D sonar reconstruction to the problem of Non-Line-of-Sight (NLOS) imaging. It introduced a regularized least square objective and solved it using the alternating direction method of multipliers (ADMM). All aforementioned volumetric methods, however, share similar limitations since extracting high-fidelity surfaces from volumetric grids is difficult. These approaches can also be computationally expensive for larger scenes or a fine discretization of volumes.

More recently, learning-based methods were proposed to resolve the elevation ambiguity. DeBortoli et al. [30] proposed a self-supervised training procedure to fine-tune a Convolutional Neural Network (CNN) trained on simulated data with ground truth elevation information. Wang et al. [31] use deep networks to transfer the acoustic view to a pseudo frontal view which was shown to help with estimating the elevation angle. However, these methods are limited to simple geometries or require collecting a larger dataset of real elevation data. Arnold et al. [32] propose training a CNN to predict the signed distance and direction to the nearest surface for each cell in a 3D grid. However, the method requires ground truth Truncated Signed Distance Field (TSDF) information which can be difficult to obtain.

In this work, we propose a physics-based renderer which uses raw FLS images and known sonar pose estimates to represent objects as zero-level sets of neural networks. It does not require hand-labeled data for training nor does it place restrictions on the setup or environment (voxel size, need for reflectance information, etc.)

B. Neural Implicit Representation

NeRF [33] introduced a volume rendering method to learn a density field aimed at novel view synthesis. It samples points along optical rays and predicts an output color which is then checked against that of the ground truth pixel. IDR [34] introduced a surface rendering technique that contrary to the volume rendering technique of NeRF, only considers a single point intersection on a surface. Hence, it fails to properly capture areas of abrupt changes in the scene. NeuS [35] leveraged the volume rendering technique of NeRF to perform 3D surface reconstructions and showed impressive results against state-of-the-art neural scene representation methods for scenes with severe occlusions and complex structures. NeTF [36] applied ideas from NeRF to the problem of NLOS imaging which was shown in [29] to have close similarity to FLS 3D reconstruction. All these methods focus on 3D reconstruction using optical sensors, either intensity or time-of-flight based. Our focus is on learning surfaces from acoustic sonar images.

III. Approach

A. Image Formation Model

An FLS 2D image $\mathcal{I}$ comprises pixels corresponding to discretized range and azimuth (r_i, θ_i) bins. Each pixel value is proportional to the sum of acoustic energy from all reflecting points $\{\mathbf{P}_i = (r_i, \theta_i, \phi_i); \phi_{\min} \leq \phi_i \leq \phi_{\max}\}$, ϕ_i being the elevation angle (Fig. 1c). However, the elevation angle ϕ_i is lost since each column θ_i of an FLS image is the projection onto the $z = 0$ plane of a circular sector π_i constrained to the sonar vertical aperture $(\phi_{\min}, \phi_{\max})$ and containing the z axis (Fig. 1b).

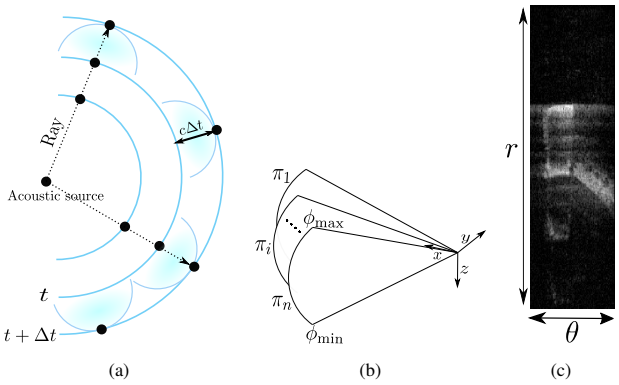


Fig. 1: (a) Sound propagates as spherical wavefronts. An acoustic ray is defined as the ray starting at the acoustic source and terminating at the wavefront (figure inspired by the *Discovery of Sound in the Sea* project [37]) (b) Each image column θ_i is the projection of the circular arc π_i onto the plane $z = 0$. (c) Example of a sonar image. Each pixel at (r, θ) corresponds to the intensity reading of all points along the elevation arc.

We now present our rendering equation. Imaging sonars are active sensors that emit a pulse of sound and measure the strength of the reflected acoustic energy. Let E_e be the emitted acoustic energy by the sonar. Now, consider a unique infinitesimal reflecting patch $\mathcal{P}_i$ "illuminated" by the acoustic wave and located on the arc $\mathcal{A}(\phi) \in \pi_i$ which passes through $(r_i, \theta_i, 0)$ (Fig. 2). The reflected acoustic energy at $\mathcal{P}_i$ and received by the sonar can be approximated as:

$$E_r(r_i, \theta_i, \phi_i) = \int_{r_i-\epsilon}^{r_i+\epsilon} \frac{E_e}{r^2} \underbrace{e^{-\int_0^{r_i} \sigma(r', \theta_i, \phi_i) \mathrm{d}r'}}_{T} \sigma(r, \theta_i, \phi_i) r \mathrm{d}r \tag{1}$$

where 2ϵ is the patch thickness, σ is the particle density at $\mathcal{P}_i$, and the factor $\frac{1}{r^2}$ accounts for spherical spreading on both the transmit and receive paths. T is the transmittance, corresponding to exponential attenuation of a wave due to particle absorption—equivalently, the probability that the acoustic wave travels between two points unoccluded. We note that, when the sonar emitter and receiver are collocated, this probability is identical during the transmit (sonar to $\mathcal{P}_i$) and return ($\mathcal{P}_i$ to sonar) paths; thus, transmittance is

accounted for only once for both paths. This is analogous to the effect of coherent backscattering in optical wave propagation with collocated emitter and receiver [38].

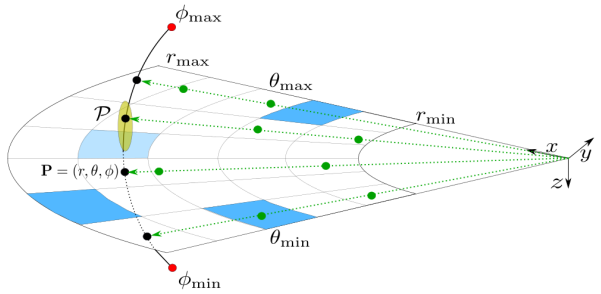


Fig. 2: 1) All points $\mathbf{P} = (r, \theta, \phi)$ on the arc are projected onto the $z = 0$ elevation plane. 2) An example of an infinitesimally small patch on the arc $\mathcal{P}$ is shown in yellow. 3) Illustrating our sampling scheme: sampled pixels are colored in blue. Sampled points on the arc are shown in black. For each point on the arc, we construct the acoustic ray (green arrow) and sample points on each ray (green points).

Now consider a surface composed of many such patches. The received energy by the sonar is simply the sum of the reflected energy by all patches $\{\mathcal{P}_i\} \in \mathcal{A}(\phi)$ which approximate the surface. Hence, we arrive at the following image formation model:

$$I(r_i, \theta_i) = \int_{\phi_{\min}}^{\phi_{\max}} \int_{r_i-\epsilon}^{r_i+\epsilon} \frac{E_e}{r^2} e^{-\int_0^{r_i} \sigma(r', \theta_i, \phi) \mathrm{d}r'} \sigma(r, \theta_i, \phi) r \mathrm{d}r \mathrm{d}\phi$$
$$= \int_{\phi_{\min}}^{\phi_{\max}} \int_{r_i-\epsilon}^{r_i+\epsilon} \frac{E_e}{r} T(r, \theta_i, \phi) \sigma(r, \theta_i, \phi) \mathrm{d}r \mathrm{d}\phi. \quad (2)$$

Note that although sound propagation through liquids is fundamentally different from that of light (longitudinal vs. transverse waves), different geometric acoustic modeling techniques still borrowed heavily from graphics and ray optics [39]. These methods fundamentally rely on Huygen's principle of sound travel through mediums which approximates the spherical wavefront as many energy-carrying particles travelling at the speed of sound. Hence, analogous to the concept of a light ray, we view an acoustic ray as the ray starting at the sonar acoustic center and ending at $\mathcal{P}_i$ (Fig. 1).

B. Rendering Procedure

Similarly to Yariv et al. [34] and Wang et al. [35], we represent the surface using two multi-layer perceptrons (MLPs): a neural implicit surface, $\mathbf{N}$, which maps a spatial coordinate $\mathbf{x} = (r, \theta, \phi)$ to its signed distance; and a neural renderer, $\mathbf{M}$, which outputs the outgoing radiance at $\mathbf{x}$. Once the surface $\mathcal{S}$ is learned, we can extract it as the zero level set of $\mathbf{N}$:

$$\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^3 : \mathbf{N}(\mathbf{x}) = 0\}. \quad (3)$$

To train our network using the rendering loss (Eq. 2), we leverage the following equation from Wang et al. [35] to estimate the value of the density $\sigma(\mathbf{x})$ from the SDF:

$$\sigma(\mathbf{x}) = \max\left(\frac{\frac{-\mathrm{d}\Phi_s(\mathbf{N}(\mathbf{x}))}{\mathrm{d}\mathbf{x}}}{\Phi_s(\mathbf{N}(\mathbf{x}))}, 0\right) \quad (4)$$

where $\Phi_s(\tau) \equiv (1+e^{-s\tau})^{-1}$ is the sigmoid function used as a smooth approximator of the occupancy indicator function $\mathcal{O}(\mathbf{x}) \equiv \mathbf{1}[\mathbf{N}(\mathbf{x}) \geq 0]$.

C. Sampling Procedure

Existing work that targets optical cameras leverages ray optics where sampling points along a ray originating at some pixel is sufficient to approximate the rendering loss. On the contrary, our rendering loss in Eq. 2 requires producing point samples along the arc at $p_i = (r_i, \theta_i)$ as well as samples along each acoustic ray. To obtain a balanced dataset of zero and non-zero intensity samples when processing an image, we sample $\mathbf{N}_{\mathcal{P}^1}$ random image pixels as well as $\mathbf{N}_{\mathcal{P}^2}$ pixels with an intensity $I(r_i, \theta_i)$ greater than a threshold. Let $\mathcal{P}$ be the set of sampled pixels.

For each pixel $p_i \in \mathcal{P}$, we use stratified sampling [40] to obtain samples along the arc. We discretize the elevation range $[-\phi_{\min}, \phi_{\max}]$ into $\mathbf{N}_{\mathcal{A}}$ equally spaced angles. Hence, the difference between two consecutive angles is $\Delta\phi = \frac{\phi_{\max}-\phi_{\min}}{\mathbf{N}_{\mathcal{A}}}$. We perturb these angles by adding $\mathbf{N}_{\mathcal{A}}$ randomly generated noise values $\sim \mathrm{Uniform}(0, 1)\Delta\phi$ to obtain a set of points $\mathbf{A}_p = \{\mathbf{P}_p = (r_i, \theta_i, \phi_{\mathbf{P}_p})\}$ on the arc.

For each sampled point $\mathbf{P}_p$, we first construct the acoustic ray $\mathcal{R}_{\mathbf{P}_p}$ which starts at the acoustic center of the sonar and terminates at $\mathbf{P}_p$ and then sample $\mathbf{N}_{\mathcal{R}} - 1$ points along each ray. Specifically, we first sample $\mathbf{N}_{\mathcal{R}} - 1$ range values r' such that $r' < r$ and $r' = i\epsilon_r$ for some $i > 0$ (ϵ_r being the sonar range resolution). We obtain the set of points $\mathbf{R}_{\mathbf{P}_p} = \{\mathbf{p} = (r', \theta, \phi_{\mathbf{P}_p})\}$. The points $\mathbf{R}_{\mathbf{P}_p} \cup \mathbf{A}_p$ constitute a set of $\mathbf{N}_{\mathcal{R}}$ points along the ray ($\mathbf{N}_{\mathcal{R}} - 1$ points along the ray + exactly 1 point on the arc). Finally, we perturb the range value of all points by adding uniformly distributed noise $\sim \mathrm{Uniform}(0, 1)\epsilon_r$ (Fig. 3).

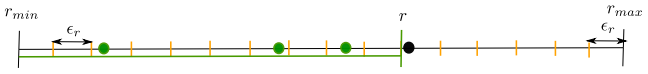


Fig. 3: Sampling along radius r. We first sample range bins and then sample one point in each bin (green points). This is the set $\mathbf{R}_{\mathbf{P}_p}$. The black point is the perturbed point on the arc $\mathbf{P}_p$.

Note that the points $\mathbf{R}_{\mathbf{P}_p} \cup \mathbf{A}_p$ are expressed in spherical coordinates in the local sonar coordinate frame and hence need to be re-expressed in a global reference frame common to all sonar poses. We first transform the points to Cartesian coordinates: $x = r\cos(\theta)\cos(\phi), y = r\sin(\theta)\cos(\phi), z = r\sin(\phi)$, and then transform to world frame by multiplying with the sonar to world transform $T_W^{\text{sonar}} = \begin{bmatrix} R_W^{\text{sonar}} & t_W^{\text{sonar}} \\ \mathbf{0}^T & 1 \end{bmatrix}$. The resulting set of points expressed in world frame $\mathbf{R}_{\mathbf{P}_p}^W \cup \mathbf{A}_p^W$ are used as inputs to the SDF neural network $\mathbf{N}$. Finally, the direction of each ray is defined by the unit vector

$$D(\mathbf{P}_p) = \frac{T_W^{\text{sonar}}\mathbf{P}_p - t_W^{\text{sonar}}}{|T_W^{\text{sonar}}\mathbf{P}_p - t_W^{\text{sonar}}|} \quad (5)$$

Let $\mathbf{X}$ be the set of all sampled points across all pixels, arcs and rays. This is the input batch to the neural network.

D. Discretized Image Formation Model

The discrete counterpart of the image formation model in Eq. 2 is:

$$\hat{I}(r, \theta) = \sum_{\mathbf{P}_p \in \mathcal{A}_p} \frac{1}{r_{\mathbf{P}_p}} T[\mathbf{P}_p] \alpha[\mathbf{P}_p] \mathbf{M}(\mathbf{P}_p), \quad (6)$$

where: $\mathcal{A}_p$ is the arc located at (r, θ), $r_{\mathbf{P}_p}$ is the range of the perturbed point $\mathbf{P}_p$ on the arc, $\mathbf{M}(\mathbf{P}_p)$ is the predicted intensity at $\mathbf{P}_p$ by the neural renderer,

$$\alpha[\mathbf{p}_i] = 1 - \exp\left(-\int_{\mathbf{p}_i}^{\mathbf{p}_{i+1}} \sigma(p)\mathrm{d}p\right) \tag{7}$$

is the discrete opacity [35] at a point $\mathbf{p}_i$ ($\mathbf{p}_i$ and $\mathbf{p}_{i+1}$ being consecutive samples along the ray) which was further shown to equal:

$$\alpha[\mathbf{p}_i] = \max\left(\frac{\mathbf{\Phi}_s(\mathbf{N}(\mathbf{p}_i)) - \mathbf{\Phi}_s(\mathbf{N}(\mathbf{p}_{i+1}))}{\mathbf{\Phi}_s(\mathbf{N}(\mathbf{p}_i))}, 0\right). \tag{8}$$

Finally,

$$T[\mathbf{P}_p] = \prod_{\mathbf{p}^1 \in \mathbf{R}_{\mathbf{P}_p}} (1 - \alpha[\mathbf{p}^1]) \tag{9}$$

is the discrete transmittance value at $\mathbf{P}_p$ (the endpoint of the ray). This is the product of one minus the opacity values α of all points on the acoustic ray excluding the α at $\mathbf{P}_p$.

E. Training Loss

Our loss function is constituted of three terms: the intensity loss in addition to eikonal and ℓ_1 regularization terms. The intensity loss

$$\mathcal{L}_{\text{int}} \equiv \frac{1}{\mathbf{N}_{\mathcal{P}}^1 + \mathbf{N}_{\mathcal{P}}^2} \sum_{p \in \mathcal{P}} ||\hat{I}(p) - I(p)||_1, \tag{10}$$

encourages the predicted intensity to match the intensity of the raw input sonar images. The eikonal loss [41]

$$\mathcal{L}_{\text{eik}} \equiv \frac{1}{\mathbf{N}_{\mathcal{R}}\mathbf{N}_{\mathcal{A}}(\mathbf{N}_{\mathcal{P}}^1 + \mathbf{N}_{\mathcal{P}}^2)} \sum_{\mathbf{x} \in \mathbf{X}} (||\nabla \mathbf{N}(\mathbf{x})||_2 - 1)^2, \tag{11}$$

is an implicit geometric regularization term used to regularize the SDF encouraging the network to produce smooth reconstructions. Finally, we draw inspiration from the NLOS volumetric albedo literature [29, 42], and add the ℓ_1 loss term

$$\mathcal{L}_{\text{reg}} \equiv \frac{1}{\mathbf{N}_{\mathcal{R}}\mathbf{N}_{\mathcal{A}}(\mathbf{N}_{\mathcal{P}}^1 + \mathbf{N}_{\mathcal{P}}^2)} \sum_{\mathbf{x} \in \mathbf{X}} ||\alpha[\mathbf{x}]||_1, \tag{12}$$

to help produce favorable 3D reconstructions when we use sonar images from a limited set of view directions. Hence, our final training loss term is:

$$\mathcal{L} = \mathcal{L}_{\text{int}} + \lambda_{\text{eik}}\mathcal{L}_{\text{eik}} + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}}. \tag{13}$$

IV. Network Architecture

We model $\mathbf{N}$ and $\mathbf{M}$ as two MLPs each with 4 hidden layers of size 64 (Fig 4). We additionally apply positional encoding to the input spatial coordinates and use weight normalization similar to IDR. While existing works that use optical cameras typically rely on larger networks to successfully learn high-frequency color and texture information, we found the proposed architecture to have sufficient capacity to learn different shapes from FLS images. Decreasing the size of the network was especially important to handle GPU memory overhead during training caused by the added sampling dimension (arcs).

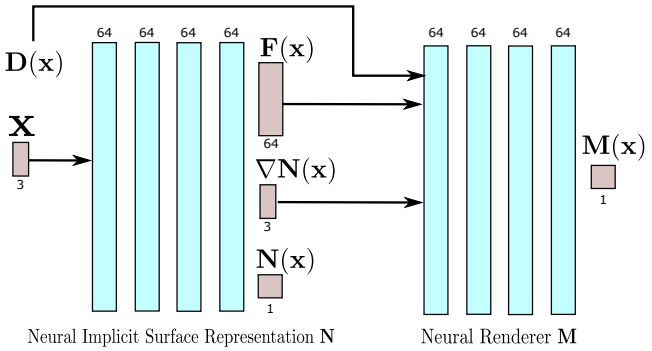


Fig. 4: Our neural network architecture. The neural implicit representation $\mathbf{N}$ takes 3D spatial coordinates $\mathbf{x}$ as input and outputs their signed distance to the surface as well as a learned feature vector $\mathbf{F}(\mathbf{x})$. We use PyTorch's autodiff [43] to compute $\nabla\mathbf{N}(\mathbf{x})$, the gradient of the signed distance at $\mathbf{x}$.

V. Evaluation

As our comparison metric, we use the mean and root mean square (RMS) Hausdorff distance defined as:

$$d_H(\mathcal{M}_1, \mathcal{M}_2) = \max\Big(\max_{\mathbf{p} \in \mathcal{M}_1} \min_{\mathbf{q} \in \mathcal{M}_2} ||p - q||_2, \max_{\mathbf{q} \in \mathcal{M}_2} \min_{\mathbf{p} \in \mathcal{M}_1} ||p - q||_2\Big) \tag{14}$$

$\mathcal{M}_1$ and $\mathcal{M}_2$ being respectively the ground truth (GT) and reconstructed meshes. We evaluate our method against back-projection (BP) and volumetric albedo (VA) [29], two state-of-the-art optimization-based methods for unsupervised object-centric 3D reconstruction using imaging sonar [1]. BP is similar to the occupancy grid mapping method (OGM) as it uses the inverse sensor model to update the voxel occupancy while, however, ignoring the correlation between grid cells. We note that both VA and BP generate a density field $\mathbf{F}(\sigma)$. Hence, for each possible density σ (i.e., $\sigma \in [0, 1]$), we extract a surface using marching cubes and report the metrics based on the best σ value. The mesh quality generated by VA also depends on the regularization weight terms which we empirically tuned for each object. With our approach, extracting the zero-level set of $\mathbf{N}$ directly generates a high-quality mesh. However, for the purpose of metric generation, we also try different level-sets near zero: $\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^3 : \mathbf{N}(\mathbf{x}) = s \mid s \in [-0.1, 0.1]\}$. We run our experiments on a system with an NVIDIA RTX 3090 GPU, an Intel Core i9-10900K, and 32GB of RAM. Our network training time until convergence is ~ 6 hours.

A. Simulation

We use HoloOcean [44, 45], an underwater simulator to collect datasets of different objects of various shapes and sizes. We use the simulator's default noise parameters; namely a multiplicative noise $w^{\text{sm}} \sim \mathcal{N}(0, 0.15)$ and additive noise $w^{\text{sa}} \sim \mathcal{R}(0.2)$ (where $\mathcal{R}$ is the Rayleigh distribution and parameters are in units of normalized pixel intensity in the range $[0, 1]$). We also enable the simulation of multipath effects. The maximum range of the sonar was set to 8m. Before feeding the raw data to the three algorithms, we perform minimal filtering of speckle noise in the images by

[1] We use the implementation of Westman et al. [29].

zeroing out pixels whose intensities are less than a threshold. After generating the meshes, we align them to the GT using ICP and report in Table I the mean and RMS Hausdorff distance to the GT for different objects and two sonar vertical apertures (14° and 28°). Figure 5 shows example 3D reconstructions obtained using each algorithm. We see

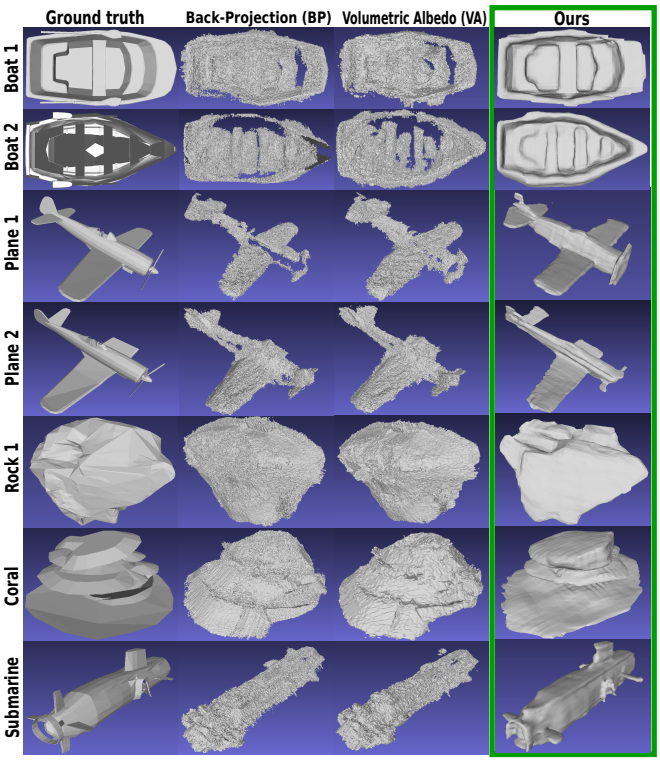


Fig. 5: 3D reconstructions generated by each method using simulated data from HoloOcean with a 14° elevation angle. Qualitatively, our method outputs more faithful 3D reconstructions compared to VA and BP.

that our method produces more accurate reconstructions compared to the baselines in terms of 3D reconstruction accuracy and mesh coverage. The neural network implicit regularization combined with the eikonal loss favors learning smooth surfaces while avoiding bad local minimas when the input images potentially do not contain enough information to completely constrain and resolve the elevation of every 3D point in space. For large objects (specified by an asterisk in the table), we decreased the grid voxel resolution of the baseline methods by one-half (increased the voxel size from the default value of 0.025m to 0.05m) to prevent the system from running out of memory (OOM): the VA and BP baselines do not leverage stochastic updates and hence, need to construct the optimization objective by processing all images in one go. This leads to memory overhead for larger objects, objects that require a fine discretization of the volume, or in the presence of a large number of non-zero pixel intensities [2]. In contrast, we train our renderer on a different subset of images in every iteration and use stochastic updates (the Adam optimizer) to optimize the function which significantly reduces memory requirements.

[2]A re-implementation of the baselines which solves the optimization problem using stochastic updates can help dealing with OOM errors.

		BP		VA		Ours	
		RMS	Mean	RMS	Mean	RMS	Mean
Boat 1	14°	0.092	0.068	0.100	0.073	**0.055**	**0.042**
(3.8 × 1.7 × 0.84)	28°	0.196	0.149	0.136	0.101	**0.063**	**0.046**
Boat 2	14°	0.121	0.090	0.084	0.064	**0.076**	**0.062**
(5.7 × 2.3 × 1.2)	28°	0.101	0.071	0.111	0.081	**0.083**	**0.068**
Plane 1	14°	0.204*	0.138*	0.191*	0.147*	**0.160**	**0.096**
(13.5 × 11.5 × 3.6)	28°	0.256*	0.206*	0.236*	0.165*	**0.167**	**0.098**
Plane 2	14°	0.204*	0.167*	0.181*	0.139*	**0.122**	**0.082**
(9.1 × 12.6 × 3.0)	28°	0.333*	0.251*	0.313*	0.224*	**0.166**	**0.116**
Rock 1	14°	0.194	0.153	0.187	0.132	**0.109**	**0.081**
(5.7 × 3.5 × 2.8)	28°	0.202	0.159	0.202	0.159	**0.139**	**0.098**
Rock 2	14°	0.083	0.065	0.079	0.060	**0.071**	**0.056**
(2.2 × 2.2 × 2.0)	28°	0.084	0.065	0.082	0.063	**0.072**	**0.058**
Rock 3	14°	0.149	0.093	0.149	0.098	**0.102**	**0.082**
(3.2 × 3.7 × 2.8)	28°	0.192	0.152	0.166	0.114	**0.148**	**0.103**
Coral	14°	0.241*	0.192*	0.241*	0.176*	**0.134**	**0.106**
(4.4 × 5.6 × 3.3)	28°	0.289*	0.232*	0.285*	0.218*	**0.212**	**0.166**
Concrete column	14°	0.125	0.097	0.128	0.099	**0.084**	**0.055**
(1.9 × 1.2 × 4.3)	28°	0.149	0.113	0.150	0.115	**0.094**	**0.060**
Submarine	14°	0.187*	0.122*	0.204*	0.144*	**0.173**	**0.101**
(5.1 × 16.7 × 4.7)	28°	0.229*	0.176*	0.237*	0.181*	**0.149**	**0.102**

TABLE I: Size (W × L × H), root mean square (RMS) and mean Hausdorff distance errors (all in meters) for different simulated objects. For certain objects (*), we increased the voxel size from 0.025m to 0.05m to prevent OOM errors with the baseline methods.

We analyze our method's performance as a function of the training set size. We vary the size of the training set while maximizing object coverage for the submarine object (this is the largest object which also contains varied geometric details. The maximum training set size is ∼ 1300 images). We report in Fig. 6 the associated performance metrics. We note that the reconstruction quality improves significantly with ∼ 200 images and plateaus after ∼ 600 images. This potentially suggests the existence of an optimal set of sonar collection points which minimize the number of data samples and maximize reconstruction quality. We leave the investigation of such active policy for future work.

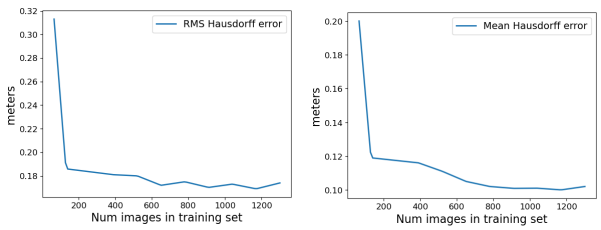
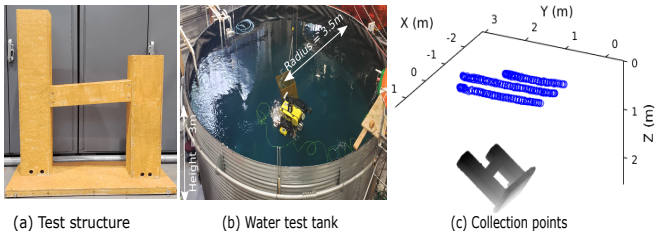


Fig. 6: Reconstruction quality as a function of the training set size for the simulated submarine dataset. The maximum training set contains ∼ 1300 images collected with a 14° vertical aperture.

B. Water Tank Experiments



(a) Test structure (b) Water test tank (c) Collection points

Fig. 7: (a) Test structure. (b) Test tank (height=3m and radius=3.5m). (c) Sample positions where a sonar image was taken. Over 900 images were collected for 1° and over 400 images for 14° and 28°.

We evaluate our method on real-world datasets of a test

structure (Fig. 7a) submerged in a test tank (Fig. 7b) using a SoundMerics DIDSON imaging sonar mounted on a Bluefin Hovering Autonomous Underwater Vehicle (HAUV). The sonar can achieve three different elevation apertures $(1°, 14°, 28°)$. We test our method on three different datasets, one for each feasible aperture. The vehicle uses a high-end IMU and a Doppler Velocity Log (DVL) to provide accurate vehicle pose information (i.e., minimal drift for the duration of data capture).

Fig. 9 shows the RMS and mean Hausdorff distance error of the three methods. The quality of the mesh generated by VA and BP depends on the selected marching cubes threshold σ. Hence, we plot the metrics generated using different σs and report the best value. With our method, we can extract the zero-level set of $\mathbf{N}$ directly alleviating the need for a post-processing step for surface generation. Since the structure is submerged and lying at the bottom of the test tank (and hence, no sonar image captures the backside of the object - Fig. 7c), we limit the matching distance of the Hausdorff metric to $0.15\text{m}, 0.2\text{m},$ and 0.25m for the $1°, 14°,$ and $28°$ apertures respectively. We see that our method generates higher quality reconstructions especially when using larger apertures: With $14°$, our method achieves an (RMS, Mean)=$(0.058\text{m}, 0.040\text{m})$ while BP and VA are respectively at $(0.077\text{m}, 0.063\text{m})$ and $(0.069\text{m}, 0.052\text{m})$. Similarly for a $28°$ aperture, our method achieves a lower (RMS, Mean) = $(0.072\text{m}, 0.055\text{m})$ compared to BP $(0.104\text{m}, 0.079\text{m})$ and VA $(0.091\text{m}, 0.070\text{m})$.

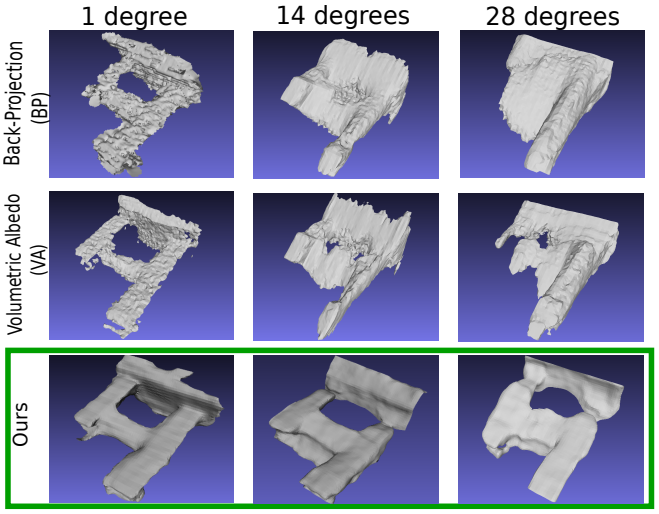


Fig. 8: The output 3D reconstructions for each method and each elevation aperture. Our method is able to capture the main components of the structures while VA and BP struggle for large vertical apertures.

Fig. 8 shows the resulting meshes for each method. While all three methods perform well with a $1°$ aperture, the difference in reconstruction quality becomes visually more apparent as the aperture angle increases. With a $14°$ aperture, we begin to lose the main feature of the object with VA and BP: the hole, short piling and crossbar are not easily discernible. When the aperture is increased to $28°$, both baseline methods perform poorly: the hole, crossbar, and short piling are lost. On the other hand, our proposed method successfully captures the major components of the structure for all three different apertures (a base, two vertical pilings, and a crossbar).

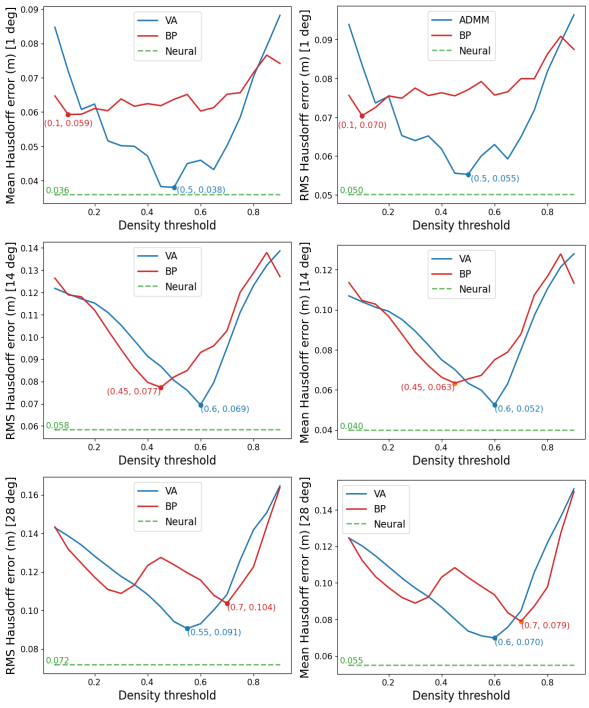


Fig. 9: Plots showing the root mean square (RMS) and mean Hausdorff distance in meters for all three methods on the real datasets $(1°, 14°,$ and $28°$ elevation apertures). To easily compare against the baselines, we add the constant green dashed line to report our method's metrics. Note however that our algorithm does not depend on the σ values in the x axis.

VI. Conclusion and Future Work

We proposed an approach for reconstructing 3D objects from imaging sonar which represents imaged surfaces as zero-level sets of neural networks. We performed experiments on simulated and real datasets with different elevation apertures and showed that our method outperforms current state-of-the-art techniques for unsupervised 3D reconstruction using FLS in terms of reconstruction accuracy. While existing volumetric methods can suffer from memory overhead as well as require a separate step to extract meshes from volumetric grids (a process often difficult and prone to error), our method allows for easy surface extraction from implicit representations and uses stochastic updates to lessen the computational requirements.

Our algorithm has some limitations, all of which create opportunities for future work. First, we currently focus on single-object reconstruction but plan to expand our method to large-scale reconstruction of marine environments at the scale of harbors by taking inspiration from techniques such as Block-Nerf [46]. Second, our method is currently mostly suited for offline 3D reconstructions but using techniques such as Instant-NGP [47] and Relu-Fields [48] can bring it to real-time performance needed for robotic navigation applications. Finally, our experiments now use only sonar but underwater robots are typically equipped with other sensors such as optical cameras. Hence, another direction from future work is to fuse multi-modal sensor inputs (acoustic and optical) where, for example, optical cameras are used to obtain high resolution models of specific interest areas in the scene while a sonar, with longer range, is used elsewhere.

REFERENCES

[1] B.-J. Ho, P. Sodhi, P. Teixeira, M. Hsiao, T. Kusnur, and M. Kaess, "Virtual occupancy grid map for submap-based pose graph slam and planning in 3d environments," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 2175–2182.

[2] P. Sodhi, B.-J. Ho, and M. Kaess, "Online and consistent occupancy grid mapping for planning in unknown environments," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 7879–7886.

[3] H. Johannsson, M. Kaess, B. Englot, F. Hover, and J. Leonard, "Imaging sonar-aided navigation for autonomous underwater harbor surveillance," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 4396–4403.

[4] S. Negahdaripour, "On 3-d motion estimation from feature tracks in 2-d fs sonar video," *IEEE Transactions on Robotics*, vol. 29, no. 4, pp. 1016–1030, 2013.

[5] E. Westman, A. Hinduja, and M. Kaess, "Feature-based slam for imaging sonar with under-constrained landmarks," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 3629–3636.

[6] Y. Yang and G. Huang, "Acoustic-inertial underwater navigation." in *ICRA*, 2017, pp. 4927–4933.

[7] S. Arnold and L. Medagoda, "Robust model-aided inertial localization for autonomous underwater vehicles," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 4889–4896.

[8] J. Albiez, S. Joyeux, C. Gaudig, J. Hilljegerdes, S. Kroffke, C. Schoo, S. Arnold, G. Mimoso, P. Alcantara, R. Saback *et al.*, "Flatfish-a compact subsea-resident inspection auv," in *OCEANS 2015-MTS/IEEE Washington*. IEEE, 2015, pp. 1–8.

[9] 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, 2016.

[10] T. A. Huang and M. Kaess, "Incremental data association for acoustic structure from motion," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 1334–1341.

[11] 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, 2018.

[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)*. IEEE, 2019, pp. 3471–3477.

[13] N. T. Mai, H. Woo, Y. Ji, Y. Tamura, A. Yamashita, and H. Asama, "3-d reconstruction of underwater object based on extended kalman filter by using acoustic camera images," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 1043–1049, 2017.

[14] T. A. Huang and M. Kaess, "Towards acoustic structure from motion for imaging sonar," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 758–765.

[15] E. Westman and M. Kaess, "Degeneracy-aware imaging sonar simultaneous localization and mapping," *IEEE Journal of Oceanic Engineering*, vol. 45, no. 4, pp. 1280–1294, 2019.

[16] P. V. Teixeira, M. Kaess, F. S. Hover, and J. J. Leonard, "Underwater inspection using sonar-based volumetric submaps," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4288–4295.

[17] H. Joe, J. Kim, and S.-C. Yu, "Probabilistic 3d reconstruction using two sonar devices," *Sensors*, vol. 22, no. 6, p. 2094, 2022.

[18] H. Joe, H. Cho, M. Sung, J. Kim, and S.-c. Yu, "Sensor fusion of two sonar devices for underwater 3d mapping with an auv," *Autonomous Robots*, vol. 45, no. 4, pp. 543–560, 2021.

[19] J. McConnell, J. D. Martin, and B. Englot, "Fusing concurrent orthogonal wide-aperture sonar images for dense underwater 3d reconstruction," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 1653–1660.

[20] J. McConnell and B. Englot, "Predictive 3d sonar mapping of underwater environments via object-specific bayesian inference," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6761–6767.

[21] E. Westman, I. Gkioulekas, and M. Kaess, "A theory of fermat paths for 3d imaging sonar reconstruction," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5082–5088.

[22] M. D. Aykin and S. Negahdaripour, "On 3-d target reconstruction from multiple 2-d forward-scan sonar views," in *OCEANS 2015-Genova*. IEEE, 2015, pp. 1–10.

[23] M. D. Aykin and S. S. Negahdaripour, "Modeling 2-d lens-based forward-scan sonar imagery for targets with diffuse reflectance," *IEEE journal of oceanic engineering*, vol. 41, no. 3, pp. 569–582, 2016.

[24] E. Westman and M. Kaess, "Wide aperture imaging sonar reconstruction using generative models," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 8067–8074.

[25] S. Negahdaripour, V. M. Milenkovic, N. Salarieh, and M. Mirzargar, "Refining 3-d object models constructed from multiple fs sonar images by space carving," in *OCEANS 2017-Anchorage*. IEEE, 2017, pp. 1–9.

[26] Y. Wang, Y. Ji, H. Woo, Y. Tamura, A. Yamashita, and A. Hajime, "3d occupancy mapping framework based on acoustic camera in underwater environment," *IFAC-PapersOnLine*, vol. 51, no. 22, pp. 324–330, 2018.

[27] Y. Wang, Y. Ji, H. Woo, Y. Tamura, A. Yamashita, and H. Asama, "Three-dimensional underwater environment reconstruction with graph optimization using acoustic camera," in *2019 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, 2019, pp. 28–33.

[28] T. Guerneve and Y. Petillot, "Underwater 3d reconstruction using blueview imaging sonar," in *OCEANS 2015-Genova*. IEEE, 2015, pp. 1–7.

[29] E. Westman, I. Gkioulekas, and M. Kaess, "A volumetric albedo framework for 3d imaging sonar reconstruction," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9645–9651.

[30] 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)*. IEEE, 2019, pp. 8040–8047.

[31] Y. Wang, Y. Ji, D. Liu, H. Tsuchiya, A. Yamashita, and H. Asama, "Elevation angle estimation in 2d acoustic images using pseudo front view," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1535–1542, 2021.

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

[33] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[34] L. Yariv, Y. Kasten, D. Moran, M. Galun, M. Atzmon, B. Ronen, and Y. Lipman, "Multiview neural surface reconstruction by disentangling geometry and appearance," *Advances in Neural Information Processing Systems*, vol. 33, pp. 2492–2502, 2020.

[35] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang, "Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction," *Advances in Neural Information Processing Systems*, vol. 34, pp. 27 171–27 183, 2021.

[36] S. Shen, Z. Wang, P. Liu, Z. Pan, R. Li, T. Gao, S. Li, and J. Yu, "Non-line-of-sight imaging via neural transient fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 7, pp. 2257–2268, 2021.

[37] K. J. Vigness-Raposa, G. Scowcroft, C. Knowlton, and H. Morin, "Discovery of sound in the sea: An on-line resource." *The Journal of the Acoustical Society of America*, vol. 127, no. 3, pp. 1894–1894, 2010.

[38] M. I. Mishchenko, L. D. Travis, and A. A. Lacis, *Multiple scattering of light by particles: radiative transfer and coherent backscattering*. Cambridge University Press, 2006.

[39] S. Siltanen, T. Lokki, S. Kiminki, and L. Savioja, "The room acoustic rendering equation," *The Journal of the Acoustical Society of America*, vol. 122, no. 3, pp. 1624–1635, 2007.

[40] M. Kettunen, E. D'Eon, J. Pantaleoni, and J. Novák, "An unbiased ray-marching transmittance estimator," *ACM Trans. Graph.*, vol. 40, no. 4, jul 2021. [Online]. Available: https://doi.org/10.1145/3450626.3459937

[41] A. Gropp, L. Yariv, N. Haim, M. Atzmon, and Y. Lipman, "Implicit geometric regularization for learning shapes," *arXiv preprint arXiv:2002.10099*, 2020.

[42] F. Heide, L. Xiao, W. Heidrich, and M. B. Hullin, "Diffuse mirrors: 3d reconstruction from diffuse indirect illumination using inexpensive time-of-flight sensors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3222–3229.

[43] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[44] E. Potokar, S. Ashford, M. Kaess, and J. G. Mangelson, "Holoocean: An underwater robotics simulator," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 3040–3046.

[45] E. Potokar, K. Lay, K. Normal, D. Benham, T. Neilsen, M. Kaess, and J. Mangelson, "HoloOcean: Realistic sonar simulation," in *Proc. IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems, IROS*, Kyoto, Japan, Oct. 2022, to appear.

[46] M. Tancik, V. Casser, X. Yan, S. Pradhan, B. Mildenhall, P. P. Srinivasan, J. T. Barron, and H. Kretzschmar, "Block-nerf: Scalable large scene neural view synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 8248–8258.

[47] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *arXiv preprint arXiv:2201.05989*, 2022.

[48] A. Karnewar, T. Ritschel, O. Wang, and N. Mitra, "Relu fields: The little non-linearity that could," in *ACM SIGGRAPH 2022 Conference Proceedings*, 2022, pp. 1–9.