Acoustic Camera Pose Refinement Using Differentiable Rendering

Chujie Wu[1], Yusheng Wang[1], Yooghoon Ji[2], Hiroshi Tsuchiya[3], Hajime Asama[1], and Atsushi Yamashita[4]

Abstract— **Acoustic cameras, also known as 2D forward looking sonars, show high reliability in underwater environments as they can produce high resolution images even if the illumination is limited. However, due to the unique imaging principle, it is hard to estimate ground-truth-level extrinsic parameters even in a known 3D scene. Usually, there are methods such as direct measurements by rulers to acquire a rough pose with centimeter-level error. It is necessary to refine the pose to millimeter-level error. In this work, we develop a novel differentiable acoustic camera simulator, which can be applied for estimating accurate 6 degrees of freedom pose of the acoustic cameras. We calculate the derivatives of synthetic acoustic images with respect to camera pose, and further integrated them into a gradient-based optimization pipeline to refine the pose. To mitigate the domain gap between real and synthetic images, an unpaired image translation method is used to transfer the real image to synthetic domain. Experiments prove the feasibility of the proposed method. It outperforms methods of previous research for higher efficiency and accuracy.**

I. INTRODUCTION

In recent years, automated underwater tasks have become increasingly important and extensive for purposes such as underwater construction, inspection, and resource exploration [1]. Underwater vehicles like autonomous underwater vehicles (AUVs) and remotely operated vehicles (ROVs) are widely used in these tasks as to improve the efficiency and protect human beings from the potential hazards in open water environment. To obtain underwater visual information, optical cameras and acoustic cameras are commonly mounted on underwater vehicles as perception sensors. However, the performance of optical cameras is extremely limited to weak illumination and water turbidity. Acoustic cameras, also known as the 2D forward looking sonars, show higher reliability in underwater scenarios [2]. They can continuously generate acoustic images while being resistant to low-light circumstances, making them important sensors for underwater tasks [3]–[5].

Finding the accurate 6 degree-of-freedom (6DoF) relative pose between cameras and scenes, which is also known as extrinsic parameter calibration, is a non-trivial problem for acoustic cameras. It can be used to tasks such as robot navigation and underwater monitoring. Furthermore, due to the rapid development of deep learning, there are pressing needs for datasets with accurate ground truth. A dataset with accurate pose ground truth is vital for tasks such as 3D reconstruction and localization [6]. Considering the range resolution of the state-of-the-art acoustic cameras (3 mm for ARIS EXPLORER 3000), the acceptable pose error should be at millimeter level. However, due to the unique imaging principle of acoustic cameras, it is hard to estimate the extrinsic parameters in a known 3D scene. Currently, the pose is usually measured manually with equipment such as rulers, or ultra-short baseline system (USBL), but these approaches are constrained by the inability to pinpoint the acoustic camera's origin, which can only offer a rough pose with centimeter-level error. Comparing synthetic acoustic images from the simulator and the real images is a possible method to estimate more accurate pose [7], [8]. Such methods usually follow a coarse-to-fine scheme, by globally estimating the pose with a rough-grained sampling of the large search space, and then locate the camera in a fine-grained local space. In [7], the aforementioned scheme is applied; however, such method has limitations in that only 3DoF poses can be estimated. In [8], the similar scheme is utilized, although 6DoF pose estimation is achieved, the search in local space is discrete, which requires fine-grained sampling for high accuracy results. This leads to synthetic image generation with a huge number, which may take more than 11 hours.

In this work, we propose a method for acoustic camera pose refinement by developing *the first differentiable acoustic camera simulator*. Differentiable rendering techniques allow the gradients of rendered images to be calculated with regard to scene parameters such as camera pose. It can be integrated in gradient-based optimization pipeline to refine the 6DoF pose for the acoustic cameras [9]. With GPU acceleration, it can generate synthetic images with faster speed and require much less synthetic images compared to the brute force search in [10]. For comparing the synthetic and the real images, one main concern is the domain gap. Previous methods use edge features to mitigate the domain gap [7], [10]. Inspired by [11], this work uses a generative adversarial network (GAN) to deal with the gap. The network is trained on unpaired datasets with contrastive learning [12]. Experiments prove the feasibility of the proposed method, it outperforms methods of previous works for a higher efficiency and accuracy.

The rest of the paper is organized as follows. In Section II, preliminaries about acoustic camera principles are intro-

This work was partly supported by JSPS KAKENHI Grant Number 21J12362.

[1]Chujie Wu, Yusheng Wang, Hajime Asama are with the Department of Precision Engineering, Graduate School of Engineering, The University of Tokyo, Tokyo 113-8654, Japan. {wuchujie,wang,asama}@robot.t.u-tokyo.ac.jp

[2]Yonghoon Ji is with the Graduate School of Advanced Science and Technology, Japan Advanced Institute of Science and Technology, Ishikawa 923-1211, Japan.

[3] Hiroshi Tsuchiya is with the Research Institute, Wakachiku Construction Company, Ltd. Chiba 299-0268, Japan.

[4]A. Yamashita is with the Department of Human and Engineered Environmental Studies, Graduate School of Frontier Sciences, The University of Tokyo, Japan. yamashita@robot.t.u-tokyo.ac.jp

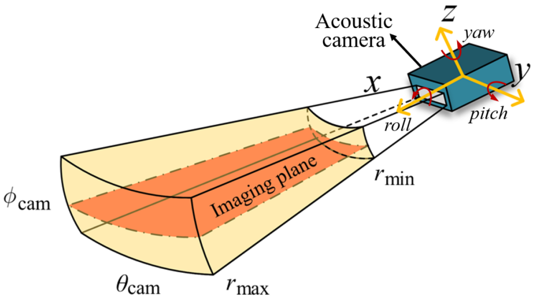


Fig. 1. Viewing scope of acoustic camera. The camera view is determined by the azimuth angle θ_{cam}, elevation angle ϕ_{cam}, and ranges $(r_{\min}, r_{\max})$.

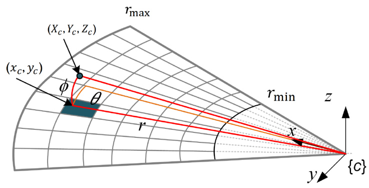


Fig. 2. Acoustic camera projection. A point (X_c, Y_c, Z_c) in 3D space is projected to (x_c, y_c) in 2D image plane.

duced. The proposed differentiable acoustic camera simulation system and camera pose refinement are explained in Section III and Section IV respectively. Section V presents the experiments and evaluations. Finally, in Section VI, conclusions and future work are summarized.

II. PRINCIPLES OF ACOUSTIC CAMERAS

An acoustic camera generates images by transmitting acoustic waves within a fan-shaped scope determined by fixed parameters of the camera including the azimuth angle θ_{cam}, elevation angle ϕ_{cam}, and its minimum and maximum ranges $(r_{\min}, r_{\max})$, as illustrated in Fig. 1. The acoustic waves are transmitted forwards from the camera and reflected backwards when hitting objects.

A 3D point in the acoustic camera coordinate system can be represented as (r, θ, ϕ) and can be transferred into Euclidean coordinates (X_c, Y_c, Z_c) based on Eq. (1):

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} r\cos\phi\cos\theta \\ r\cos\phi\sin\theta \\ r\sin\phi \end{bmatrix}. \tag{1}$$

The acoustic camera projection model is illustrated in Fig. 2. When projecting from 3D space to 2D image plane, each point (r, θ, ϕ) in the 3D area is mapped to (r, θ) in the 2D image. The point (x_c, y_c) in the Euclidean image coordinates can be represented as Eq. (2).

$$\begin{bmatrix} x_c \\ y_c \end{bmatrix} = \begin{bmatrix} r\cos\theta \\ r\sin\theta \end{bmatrix}. \tag{2}$$

In other words, the elevation information ϕ is missing during projection.

III. DIFFERENTIABLE SIMULATION SYSTEM

Collecting underwater acoustic image dataset is difficult because operating cameras in open water can be risky and of a high cost [13]. A promising solution to this challenge is developing an acoustic camera simulation system which allows controlling systems and evaluating algorithms without going to the real underwater environment. It helps reduce the cost and risk of in-field experiments.

In previous research, the acoustic image simulator is based on 3D rendering using a modeling software Blender [11], [14]. However, as the rendering process is non-differentiable, it cannot be integrated to other frameworks to solve the inverse rendering problem like pose estimation. Besides, it requires several seconds for processing one image, which is slow.

In our work, the first differentiable acoustic camera simulation system is proposed and used to simulate scenarios of underwater tasks and to generate synthetic acoustic images.

A. Overview

The proposed simulation system consists of three parts, including 3D scene assembling, perspective camera view rendering and differentiable acoustic camera view generation, as illustrated in Fig. 3. Generally, the acoustic camera simulator takes scene parameters including camera, geometry and materials as inputs, and produces synthetic acoustic images via differentiable rendering. Materials and the geometry are fixed information defined by the underwater environment, while the camera pose is initialized by a rough guess. Ground truth image is a real image taken from the target camera pose. We define an objective function to measure the difference between synthetic images and ground truth images. Since the image generation process is differentiable, we can calculate the gradients of image with respect to the input camera pose. The gradients are further used in backward pose optimization to update pose parameters until loss being converged.

B. 3D Scene Assembling

In 3D scene assembling, scene parameters including camera, geometry and materials are defined. We construct 3D scene objects and underwater seafloor plane using Blender. An example of constructed scene is shown in Fig. 4. Each object is represented in the format of triangle meshes and is assumed to follow the Lambertian reflection rules.

To simulate the acoustic camera, we use the perspective camera model with the same aperture angle, and set a point light at the camera position with the attenuation of the ray strength based on the inverse square law. The sensing range is defined by the specifications of the acoustic camera.

C. Perspective Camera View Rendering

Rendering can be defined as a function that takes a 3D scene as an input and outputs a 2D image. Differentiable rendering makes the derivatives of this function be calculated with respect to different scene parameters.

In perspective camera view rendering, the task is as the following. Given a 3D scene with a continuous parameter

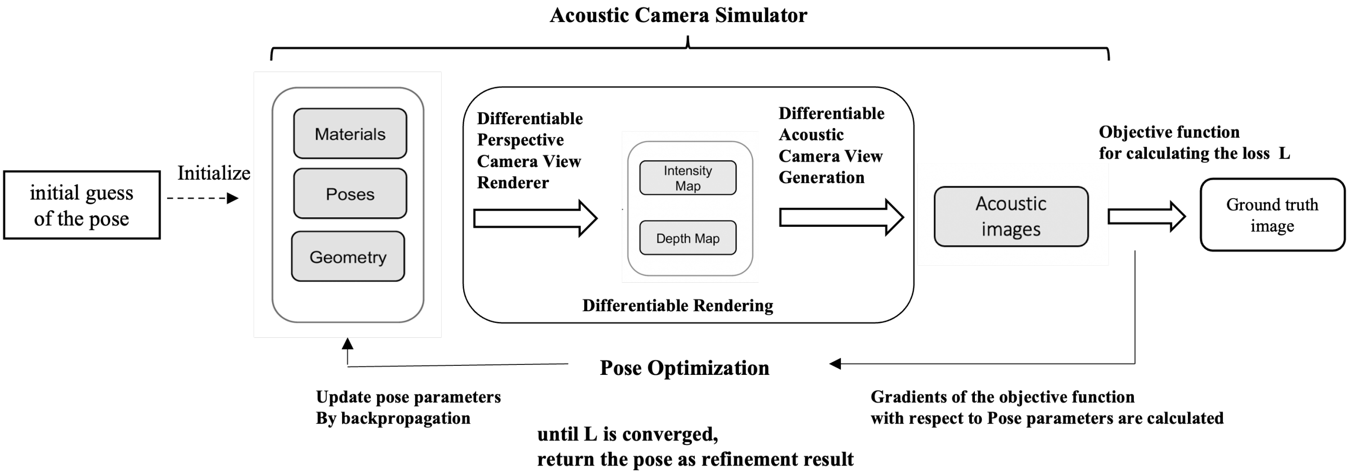

Acoustic Camera Simulator
initial guess of the pose
Initialize
Materials
Poses
Geometry
Differentiable Perspective Camera View Renderer
Intensity Map
Depth Map
Differentiable Acoustic Camera View Generation
Differentiable Rendering
Acoustic images
Objective function for calculating the loss L
Ground truth image
Pose Optimization
Update pose parameters By backpropagation
Gradients of the objective function with respect to Pose parameters are calculated
until L is converged, return the pose as refinement result

Fig. 3. Overview of the proposed differentiable acoustic camera simulation system.

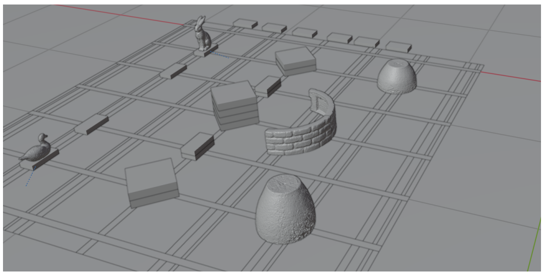


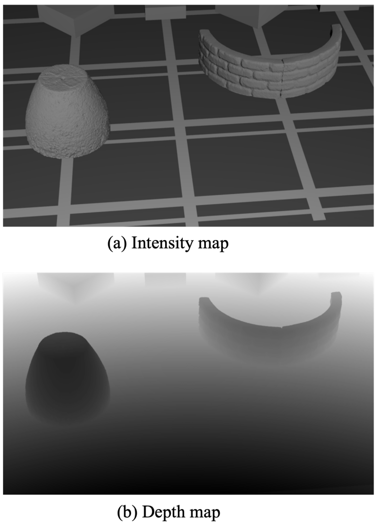
Fig. 4. 3D scene objects constructed by Blender.

set Φ constructed in 3D Scene Assembly, we generate an intensity map and a depth map as output. In the intensity map, each pixel represents the backscattered intensity value, while in the depth map, each pixel represents the range from camera to objects. Both intensity maps and depth maps are $\theta - \phi$ images that each pixel can be denoted as (θ, ϕ) in acoustic camera coordinates. An example of perspective camera view rendering output is shown in Fig. 5.

We implement this part using a differentiable renderer framework named Redner [15]. It offers a comprehensive solution to compute derivatives of scalar functions over a rendered image with respect to arbitrary scene parameters without approximation. The backscattered intensity map is simulated in an optical camera way as we simplify the sound propagation process as multiple ray casting within the scope. Monte Carlo sampling [16] is used to estimate both the integral and the gradient of the integral. The core strategy for computing the gradient integral is to split it into smooth and discontinuous regions [17]. For smooth regions, traditional area sampling with automatic differentiation is employed, while for discontinuous regions, a novel edge sampling method is used to capture the changes at boundaries.

D. Differentiable Acoustic Camera View Generation

In differentiable acoustic camera view generation part, acoustic image is generated from intensity map and depth map, following a differentiable pipeline. As both intensity map and depth map are $\theta - \phi$ images, we firstly convert them into pixel coordinates by transforming and mapping along the origin. We denote pixel p_i in intensity map as (θ_i, ϕ_i). The pixel coordinate transformation is performed as:



(a) Intensity map

(b) Depth map

Fig. 5. Perspective camera view rendering outputs. Both intensity map and depth map are expanded by (θ, ϕ).

$$\theta_i^{'} = \frac{\theta_i}{\theta_r} \times \theta_{\text{ap}} - \frac{\theta_{\text{ap}}}{2}, \tag{3}$$

$$\phi_i^{'} = \frac{\phi_i}{\phi_r} \times \phi_{\text{ap}} - \frac{\phi_{\text{ap}}}{2}, \tag{4}$$

where θ_{ap} and ϕ_{ap} refer to aperture angles in azimuth direction and elevation direction, and (θ_r, ϕ_r) is the image resolution. The coordinate of p_i is updated as $(\theta_i^{'}, \phi_i^{'})$.

According to equation (1), we can further integrate 2D information in both depth map and intensity map into a set of 3D points in Euclidean coordinates. We denote the set as $\mathbf{P} = \{\langle \mathbf{x}_i, y_i \rangle\}_{i=1}^{N}$, which includes N tuples of a point position $\mathbf{x}_i = (x_{i1}, x_{i2}, x_{i3})$ and a intensity value y_i of pixel

p_i. The point position is calculated as follows, where r_i is the corresponding pixel value in the depth map.

$$\begin{bmatrix} x_{i1} \\ x_{i2} \\ x_{i3} \end{bmatrix} = \begin{bmatrix} r_i \cos\phi_i^{'} \cos\theta_i^{'} \\ r_i \cos\phi_i^{'} \sin\theta_i^{'} \\ r_i \sin\phi_i^{'} \end{bmatrix}. \tag{5}$$

Acoustic images are formed by azimuth angle θ and range r coordinates. Points in set **P** forms a 3D cube represented by $[r, \theta, \phi]$. We re-scale the coordinates by the camera aperture parameters and acoustic image resolutions in three directions. To allow for the gradient flow in 3D-to-2D projection, we represent each point $\mathbf{x}_i$ by a smooth Gaussian density function $f_i(\cdot)$ and the occupancy function of the point cloud is a clipped sum of the individual per-point functions [18]:

$$o(\mathbf{x}) = \text{clip}(\sum_{i=1}^{N} f_i(\mathbf{x}), [0, 1]), \tag{6}$$

$$f_i(\mathbf{x}) = c_i \text{exp}\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}_i)^T \Sigma_i^{-1}(\mathbf{x} - \mathbf{x}_i)\right), \tag{7}$$

where c_i and Σ_i are size parameters, and $\mathbf{x}$ is the set of points $\mathbf{x}_i$. The resulting function can be discretized to a grid of resolution. We perform integration along elevation angle to get synthetic acoustic images.

IV. Camera Pose Refinement

The pose refinement task is regarded as, given a target real acoustic image $\mathbf{I}_1$ taken from a known 3D scene, and an initial guess of 6DoF camera pose $\mathbf{w}$, we aim at finding the accurate pose of camera from which the target image is taken.

The derivatives of synthetic acoustic images can be obtained by differentiable rendering. Therefore, we adopt an optimization method using gradient descent algorithm [19] on the loss between target image and synthetic acoustic image to refine the translation and rotation of the pose. Acoustic camera 6DoF pose $\mathbf{w}$ is denoted as follows.

$$\mathbf{w} = \left[x, y, z, \varphi_x, \varphi_y, \varphi_z\right], \tag{8}$$

where x, y, x refer to the translation in world coordinate and $\varphi_x, \varphi_y, \varphi_z$ refer to the rotation in Euler angles.

The prior knowledge of the camera pose is used as initial guess $\hat{\mathbf{w}}$ before pose refinement. It can be obtained by manually measuring the approximate pose of the camera using a ruler. Another approach is creating a synthetic dataset consisting images of a large variety of viewpoints and searching for the closest one to the target image within this dataset [8]. The corresponding camera pose can be used as a guess. Since the initial guess is not sufficiently precise, it usually differs from the ground truth pose by approximately 5 cm per scale.

The goal of the pose refinement is to determine a more accurate pose of the camera. Starting from the initial guess $\hat{\mathbf{w}}$, the differentiable simulator R outputs the rendered acoustic image $\mathbf{I}_2$ taking $\mathbf{w}$ as scene parameter. However, there is a large domain gap between the real acoustic image $\mathbf{I}_1$ and the synthetic image $\mathbf{I}_2$, for example, the heavy noise in real acoustic images can not be represented in simulation. It is hard to directly compare and utilize the difference between real acoustic images and synthetic images. In this case, we train a constrative learning model named Contrastive Unpaired Translation (CUT) [12] on pairs of real acoustic images and synthetic images, and use it to transfer real acoustic images to be like synthetic images. We denote the converted image as $\mathbf{I}_1'$.

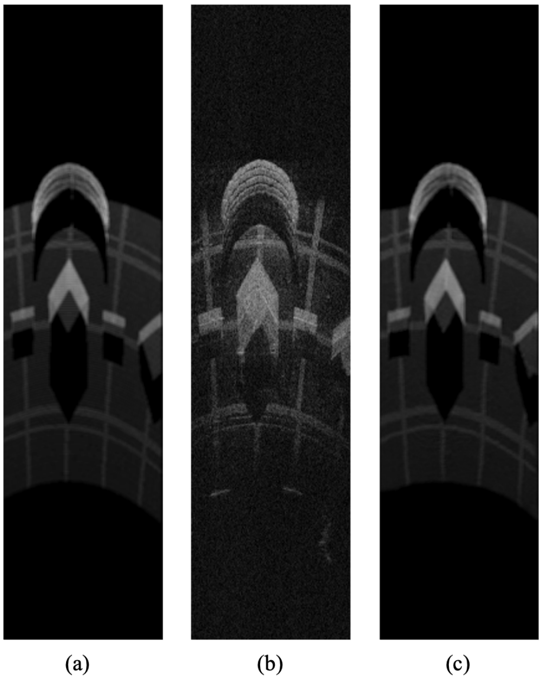


Fig. 6. Test result of Constrastive Unpaired Translation (CUT) model. (a) Synthetic acoustic image generated by our simulation system, (b) Real acoustic image, (c) Fake synthetic acoustic image, converted from (b) using CUT.

Fig. 6 illustrates a test example of CUT. The model is trained on a small number of paired images while generating realistic results. The fake synthetic acoustic image $\mathbf{I}_1'$ transferred from real acoustic image $\mathbf{I}_1$ is very close to the ground truth synthetic acoustic image $\mathbf{I}_2$.

Since we eliminate the domain gap by transferring the target image $\mathbf{I}_1$ to $\mathbf{I}_1'$, we can use loss $\mathcal{L}$ to measure the difference between target image $\mathbf{I}_1'$ and the generated acoustic image $\mathbf{I}_2$. We aim at finding a pose $\mathbf{w}$ such that the loss $\mathcal{L}$ is minimized.

$$\mathbf{w} = \operatorname{argmin} L(\mathbf{w}). \tag{9}$$

We iteratively refine the guess using gradients and back propagation.

$$\mathbf{w}_{i+1} = \mathbf{w}_i - \alpha \nabla L(\mathbf{w}), \tag{10}$$

where α is the learning rate determines how far we move along the negative gradient direction. We use L1 loss (Mean Absolute Error, MAE) $\mathcal{L}_1$ to measure the difference, which is the sum of the all the absolute differences between the true pixel values in target image $\mathbf{I}_1'$ and the pixel values in predicted image $\mathbf{I}_2$.

$$\mathcal{L}_1 = \sum_{i=1}^{n} |\mathbf{I}_1 - \mathbf{I}_2|, \tag{11}$$

.

Algorithm 1 shows the overall framework of pose refinement, which is based on a neural network structure performing forward and backward optimization. Pose is initialized as $\hat{\mathbf{w}}$ for we assume we have a prior knowledge of the rough camera pose. $\mathbf{I}_2$ is the rendering output acoustic image from the differentiable simulator R, which takes $\mathbf{w}$ as scene parameters.

Algorithm 1 Acoustic Camera Pose Refinement

```
procedure POSE REFINEMENT($\hat{\mathbf{w}}, \mathbf{I}_1'$)
    Initialize the camera pose $\mathbf{w} = \hat{\mathbf{w}}$
    while not converge do
        Render from $\mathbf{w}$ in simulator $R$
        Get image $\mathbf{I}_2 = R(\mathbf{w})$
        Calculate $\mathcal{L}(\mathbf{I}_1', \mathbf{I}_2)$
        Calculate gradients of $L$ with respect to $\mathbf{w}$
        Backpropagate and update the camera pose $\mathbf{w}$
    end while
    the loss $L$ is converged, return pose $\mathbf{w}$ as output
end procedure
```

To measure the quality of acoustic images after pose refinement, we introduce two metrics including Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM) for comparing the target image and the final reconstructed image [20]. PSNR is an expression for the ratio of signal power to noise power:

$$PSNR = 10\log_{10}(\frac{m^2}{\mathcal{L}_2}), \tag{12}$$

where m is the maximum value of a pixel in image which in this case equals 255. A higher PSNR score suggests a higher image quality. SSIM is used to measure the similarity between images based on three comparison measurements including luminance, contrast, and structure. SSIM index is calculated as:

$$SSIM = \frac{(2\mu_1\mu_2 + C_1)(2\sigma_{12} + C_2)}{(\mu_1^2 + \mu_2^2 + C_1)(\sigma_1^2 + \sigma_2^2 + C_2)}, \tag{13}$$

where μ_1, μ_2 are the average and σ_1, σ_2 are the variance of $\mathbf{I}_1, \mathbf{I}_2$. Besides, C_1, C_2 are the variables to stabilize the division with weak denominator and σ_{12} is the covariance.

V. Experiments

We utilize the real dataset in [6]. The 3D models of the objects are generated in land by structure from motion. Then, the objects are placed in a water tank with designed configuration. All objects on the scene are represented in triangular meshes format for applying rendering. Regarding the underwater seafloor, there are grids formed by strips of specific material. Acoustic camera 6DoF pose is represented as translation x, y, z in meter scale and rotation $\varphi_x, \varphi_y, \varphi_z$ as angle in radius. The output images of perspective camera view rendering are in resolution $(720, 1200)$, while the synthetic acoustic images are in resolution $(512, 128)$. We set the aperture angles in elevation direction and azimuth direction of acoustic cameras to be 18 and 30 respectively in degree and the resolution in range direction to be 0.006 m for a standard sensing scope based on the specification of ARIS EXPLORER 3000.

TABLE I

Camera Pose Before and After Refinement

	x [m]	y [m]	z [m]	φ_x [rad]	φ_y [rad]	φ_z [rad]
Target	0.1635	-1.5768	2.0047	0.900	-0.027	1.503
Initial	0.1500	-1.6000	1.9900	0.890	-0.010	1.510
Refined	**0.1643**	**-1.5775**	**2.0037**	**0.901**	**-0.024**	**1.500**

TABLE II

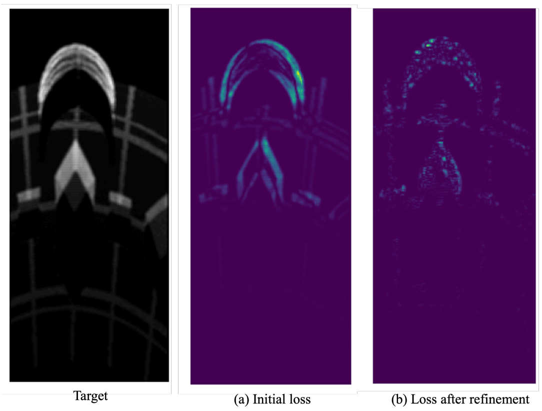
Evaluation of Refinement Results (measured in average).

	Initial	Refined
Converge Time	–	17 min 36 sec
PSNR (dB)	26.86	42.25
SSIM	0.9135	0.9940

Datasets of two domains are constructed for training the CUT model. For real acoustic image dataset, we perform image generation experiments in a water tank within a 1.8 m × 1.8 m × 0.4 m space and set the camera poses to be a variety of orientations at different positions. The poses of camera are recorded and used as inputs to simulation system to generate corresponding synthetic acoustic images. Synthetic images have an average of 0.33 second generation time and each image is cropped and resized to be $(128, 512)$ in resolution. We randomly select 295 pairs of images to form the training set and select 98 pairs of image to form the test set. The CUT model is trained of 400 epochs on Intel(R) Core(TM) i9-11900K CPU and a NVIDIA GeForce RTX(TM) 3090 GPU. The output test results have 0.993 similarity score in average to the ground truth synthetic image.

For pose refinement, the target image is selected from CUT test result images. It was measured by ruler to get an initial guess of camera pose. We implemented the gradient descent algorithm using Pytorch framework and trained the model on NVIDIA Tesla T4 GPU. In optimization, we use six independent Adam optimizers for 6DoF respectively, and set six MultiStepLR schedulers that decay the learning rate with a multiplicative factor when a certain milestone is reached. Specifically, our learning rate for translation x, y is set to 0.0002 while learning rate for translation z and rotations is set to 0.0001.

Table I shows camera poses before and after refinement and Table II shows the quantitative evaluation for refinement results. The refined camera poses show high accuracy compared to ground truth with an average position error being less than 0.001 m. Both PSNR and SSIM score reveal the improvement of image quality after refinement. Besides, regarding the refinement time, it is significantly more efficient than methods in [8] based on fine-grained sampling. It is also


Target

(a) Initial loss

(b) Loss after refinement

Fig. 7. L1 Loss before and after refinement. It is calculated between the target image and synthetic image.

intuitive from the comparison in Fig. 7 that loss between ground truth and reconstructed image has been reduced greatly after the pose refinement. The experiment results prove the effectiveness of our proposed acoustic camera pose refinement approach using differentiable rendering.

VI. CONCLUSIONS

In this paper, a method for acoustic camera pose refinement is proposed. We developed a novel acoustic camera simulator using differentiable rendering. With the help of differentiable rendering, we are able to compute the gradients of the synthetic acoustic image with respect to camera pose, and use them to optimize the six degree-of-freedom pose for the acoustic cameras. Considering there is a domain gap between real acoustic image and synthetic acoustic image, we trained an image-to-image translation network CUT to transfer real acoustic images to be synthetic images. And use the transferred image as target for optimization of pose. The experiments prove the feasibility of this method. It outperforms the methods proposed in [8] for a higher efficiency.

In the future, efforts should be made in further improving the accuracy and precision when the initial guess error becomes larger and solving the inconsistency between real and synthetic images in terms of image illuminance. We also believe that the differentiable acoustic simulator has a broader perspective, which can be applied to various tasks such as inverse rendering.

REFERENCES

[1] J. Li, M. Kaess, R. M. Eustice, and M. Johnson-Roberson, “Pose-graph slam using forward-looking sonar,” *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2330–2337, 2018.

[2] E. Belcher, W. Hanot, and J. Burch, “Dual-frequency identification sonar (didson),” in *Proceedings of the 2002 IEEE International Symposium on Underwater Technology (UT2002)*, Apr. 2002, pp. 187–192.

[3] N. Hurtos, D. Ribas, X. Cufi, Y. Petillot, and J. Salvi, “Fourier-based registration for robust forward-looking sonar mosaicing in low-visibility underwater environments,” *Journal of Field Robotics*, vol. 32, no. 1, pp. 123–151, 2015.

[4] L. Baumgartner, N. Reynoldson, L. Cameron, and J. Stanger, “Assessment of a dual-frequency identification sonar (didson) for application in fish migration studies,” *NSW Department of Primary Industries-Fisheries Final Report Series*, no. 84, pp. 1–33, 2006.

[5] Y. Wang, Y. Ji, H. Woo, *et al.*, “Acoustic camera-based pose graph slam for dense 3-d mapping in underwater environments,” *IEEE Journal of Oceanic Engineering*, vol. 46, no. 3, pp. 829–847, 2021.

[6] Y. Wang, Y. Ji, D. Liu, H. Tsuchiya, H. Asama, and A. Yamashita, “Learning pseudo front depth for 2d forward-looking sonar-based multi-view stereo,” in *Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2022)*, 2022.

[7] J. Park and J. Kim, “Robust underwater localization using acoustic image alignment for autonomous intervention systems,” *IEEE Access*, vol. 10, pp. 58 447–58 457, 2022.

[8] Y. Wang, Y. Ji, D. Liu, H. Tsuchiya, A. Yamashita, and H. Asama, “Simulator-aided edge-based acoustic camera pose estimation,” in *OCEANS 2022 - Chennai*, Feb. 2022, pp. 1–4.

[9] A. Grabner, Y. Wang, P. Zhang, *et al.*, “Geometric correspondence fields: Learned differentiable rendering for 3d pose refinement in the wild,” in *European Conference on Computer Vision*, Springer, 2020, pp. 102–119.

[10] Y. Wang, Y. Ji, D. Liu, *et al.*, “Acmarker: Acoustic camera-based fiducial marker system in underwater environment,” *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5018–5025, 2020.

[11] D. Liu, Y. Wang, Y. Ji, H. Tsuchiya, A. Yamashita, and H. Asama, “Simulator-aided edge-based acoustic camera pose estimation,” *Advanced Robotics*, vol. 35, no. 3-4, pp. 242–254, 2021.

[12] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu, “Contrastive learning for unpaired image-to-image translation,” in *European Conference on Computer Vision*, 2020.

[13] Y. Ji, S. Kwak, A. Yamashita, and H. Asama, “Acoustic camera-based 3d measurement of underwater objects through automated extraction and association of feature points,” in *Proceedings of 2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI2016)*, Sep. 2016, pp. 224–230.

[14] D. Liu, Y. Wang, Y. Ji, H. Tsuchiya, A. Yamashita, and H. Asama, “Cyclegan-based realistic image dataset generation for forward-looking sonar,” *Advanced Robotics*, vol. 35, no. 3-4, pp. 242–254, 2021.

[15] T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen, “Differentiable monte carlo ray tracing through edge sampling,” *ACM Transactions on Graphics*, vol. 37, no. 6, pp. 1–11, 2018.

[16] A. Shapiro, “Monte carlo sampling methods,” *Handbooks in operations research and management science*, vol. 10, pp. 353–425, 2003.

[17] T.-M. Li, “Differentiable visual computing,” 2019. [Online]. Available: `http://arxiv.org/abs/1904.12228`.

[18] E. Insafutdinov and A. Dosovitskiy, “Unsupervised learning of shape and pose with differentiable point clouds,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[19] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proceedings of the 3rd International Conference on Learning Representations (ICLR2015)*, Dec. 2014.

[20] A. Hore and D. Ziou, “Image quality metrics: Psnr vs. ssim,” in *2010 20th international conference on pattern recognition*, IEEE, 2010, pp. 2366–2369.