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

Deep Underwater Monocular Depth Estimation with Single-Beam Echosounder

Haowen Liu, Monika Roznere, and Alberto Quattrini Li

***Abstract*— Underwater depth estimation is essential for safe Autonomous Underwater Vehicles (AUV) navigation. While there has been recent advances in out-of-water monocular depth estimation, it is difficult to apply these methods to the underwater domain due to the lack of well-established datasets with labelled ground truths. In this paper, we propose a novel method for self-supervised underwater monocular depth estimation by leveraging a low-cost single-beam echosounder (SBES). We also present a synthetic dataset for underwater depth estimation to facilitate visual learning research in the underwater domain, available at `https://github.com/hdacnw/sbes-depth`. We evaluated our method on the proposed dataset with results outperforming previous methods and tested our method in a dataset we collected with an inexpensive AUV. We further investigated the use of SBES as an additional component in our self-supervised method for up-to-scale depth estimation providing insights on next research directions.**

I. Introduction

This paper presents an underwater depth estimation framework that is suited for Autonomous Underwater Vehicles (AUVs) with inexpensive sensor configurations, i.e., a monocular camera and a single-beam echosounder (SBES) – see Fig. 1.

Underwater perception is an important task for AUVs in navigation, localization, and survey [1]. Typically AUVs use sensors, such as multi-beam sonar [2], photon beam [3], or LiDAR [4], but they are generally expensive, difficult to setup, and do not provide dense depth estimates, limiting their adoption. Cameras are an inexpensive solution that are available on many AUVs [1], [5]. However, camera-based depth estimation is challenging in underwater compared to out-of-water; the underwater image formation is affected by light attenuation and backscattering due to suspended particulates and other environmental factors [5], [6].

Recent advances in deep learning has the potential to address such challenges [7]. Many of these deep monocular depth estimation methods have emerged for out-of-water domains [8]. They are generally trained on large well-established datasets such as KITTI [9] and NYU-depth [10]. However, there are no comparable datasets in the underwater domain to facilitate large-scale training for deep underwater depth estimation. While there are a few Generative Adversarial Network (GAN) based underwater depth estimation methods available (e.g., [11], [12]), they are still trained with out-of-water dataset augmented with synthetic underwater effects or style transfer. As such, there is a need for a dataset focused on underwater depth estimation task. Also, monocular depth estimation suffers from the problem of scale ambiguity [8], which hinders its usage in many robotics applications that require metric depth estimation. A potential direction is to integrate absolute measurements from other sensors as additional cues to monocular depth estimation.

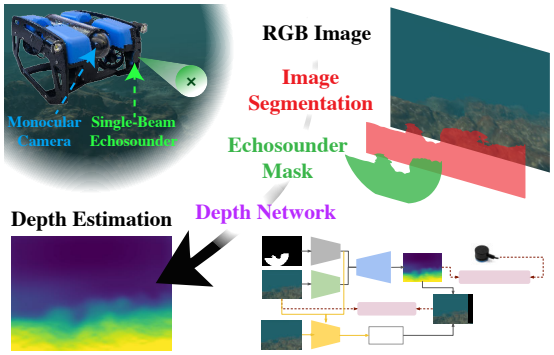


Fig. 1: Overview of the proposed deep underwater depth estimation framework using a monocular camera and a single-beam echosounder.

In this paper, we propose a novel self-supervised network for monocular depth estimation that takes in as input both monocular imagery and a single range measurement from a low-cost SBES. The main contributions of this paper are:

- The first deep learning method for underwater monocular depth estimation with SBES.
- Mask representation of enchosounder measurement to transform single distance reading to a denser representation for more effective fusion with monocular imagery.
- A large-scale synthetic dataset for underwater depth estimation with SBES data generated via simulation.
- A Pytorch implementation with tests in the proposed dataset and real-world data that highlight the feasibility of our approach for depth estimation.

This work pushes forward advancements in autonomy of low-cost underwater robots so that they become accessible to the broader community and support important tasks, such as environmental monitoring and ocean exploration.

II. Related Work

Situational awareness is important for any robot to achieve robust control, navigation, and planning. However, with a basic sensor configuration (e.g., monocular camera, SBES) on an inexpensive underwater robot, any perception based method will suffer from scale ambiguity and error induced by water medium complexities. In the following, we will discuss past works on how deep learning can help estimate image depth while handling image distortions, and how sensor fusion can improve the quality of depth estimation

The authors are with the Department of Computer Science, Dartmouth College, USA {`haowen.liu.gr, monika.roznere.gr, alberto.quattrini.li`} `@dartmouth.edu`

This work is in part supported by the Dartmouth Burke Research Initiation Award, NSF CNS-1919647, 2024541, 2144624.

frameworks. We will also discuss datasets required to train a deep learning framework well.

A. Underwater Depth Estimation

There has been an increasing number of studies on underwater depth estimation in recent years. Early works [13]–[15] used the underwater image formation model and the dark channel prior assumption to estimate the transmission map and backscattering constant for depth estimation. Li *et al.* [16] jointly estimated scene depth and restored the image from an underwater video sequence using stereo matching and fog information. Peng *et al.* [17] used both image blurriness and light absorption for estimating depth. For learning based methods, Gupta and Mitra [12] proposed a GAN based method for depth estimation of a single underwater image. Their method learns mapping functions between unpaired in-air RGB-D images and arbitrary underwater images to indirectly estimate depth images based on cycle-consistent learning. Similarly, Hambarde *et al.* [11] used GANs for coarse and fine-level depth map estimation, with synthetic underwater images constructed from in-air RGB-D images as input. Accurate up-to-scale depth estimation underwater is still an open problem.

B. Sensor-Camera Fusion

Monocular depth estimation using deep learning has primarily focused on in-land scenarios [18]–[21] than on underwater cases. Self-supervised monocular depth methods formulate depth estimation as an image reconstruction problem, where depth maps are used as an intermediate product that integrates into the image reconstruction loss [22].

It is well known that fusing data from other sensors can help increase the quality of depth estimates. Sensor configurations can be as simple as multiple cameras [23], [24], IMU [25]–[31], and sonar [32]. They can also be more complex and of higher cost, such as the setup by Richmond *et al.* [33] with multibeam sonar, fiber-optic gyroscope IMU, and doppler velocity log, or the SVIn2 [34] setup with profiling scanning sonar, IMU, stereo camera, and pressure sensor. Within the learning-based literature, to improve and recover dense depth maps from monocular images, prior work have proposed to merge sparse depth measurements from other sensors, e.g., LiDAR. For example, Zhao *et al.* [35] used a multi-scale co-attention-guided graph propagation network adaptive to the sparsity patterns of sparse LiDAR depth input to better associate the spatial context with observed depth values. Park *et al.* [36] employed a novel architecture that additionally re-injects the LiDAR signal at the refinement stage to mitigate signal degradation due to normalization layers after concatenating image and LiDAR features at the start. Feng *et al.* [37] proposed a novel pseudo dense LiDAR representation to overcome the sparsity issue of a 4-beam LiDAR for more effective fusion with monocular image features. There has also been work on fusion of RGB images with single-row scanning automotive radars [38] or binaural echoes [39] for improving depth estimation. Our method on fusing echosounder with monocular images for depth prediction takes inspiration from these LiDAR based methods; however, one of the main differences is that, while the LiDAR provides multiple signals, the SBES provides a single value, thus requiring strategies to handle this challenge.

C. Underwater Dataset

There are very few existing datasets in the underwater domain, mainly targeted for SLAM or for image enhancement, compared to general out-of-water computer vision tasks. Examples include AQUALOC [40] which contains seabed recordings at different water depths from Remotely Operated Vehicles equipped with a monocular monochromatic camera, IMU, and a pressure sensor; the Underwater Caves Sonar [41] dataset consisting of data from a mechanically scanned imaging sonar, depth sensor, and camera imagery; and SQUID [42] containing images taken under varying water properties and respective 3D structures of the scenes. However, due to the challenges of underwater data collection, most underwater datasets only include a small number of samples obtained under specific settings and lack corresponding ground truth measurements, rendering them less suitable for deep learning based methods. While methods such as WaterGAN [7] proposed synthesizing underwater images via introducing artificial underwater effects to in-air images, they often introduce unwanted artifacts and fail to represent real underwater landscapes.

III. Dataset

Given the scarcity of underwater datasets and the lack of datasets comprising of echosounder measurements and monocular images, we created a dataset for underwater depth estimation models. Our dataset contains monocular and stereo images, semantic segmentations, and echosounder readings of realistic simulated underwater scenes to facilitate vision-based learning in the underwater domain. Note that stereo images were collected for future analysis, as here we focus on a monocular configuration. We describe the data creation and collection pipeline in the following.

To construct underwater scenes in simulation, we first selected models of various underwater structures, such as shipwrecks, reefs, and caves built using professional photogrammetry software from 3D model collection site SketchFab[1]. As these models were constructed from real-world images, they provide accurate representation of structures found in real-world underwater environments. Then, each model is placed in a simulated underwater environment created using Unity[2] game engine. The simulated environments are modified from an existing underwater simulator framework which contains custom shaders that incorporates a light transmission model, simulating underwater optical effects, thus providing a good amount of realism. As individual texture colors of the collected models vary greatly according to the color of their surrounding waters during the photogrammetry data collection process, color of the simulated water in each scene

[1] https://sketchfab.com/
[2] https://unity.com/

TABLE I: Summary reporting the number of images of the proposed dataset and split in our experiments.

	Shipwreck(large)	Shipwreck(small)	Reef	Rock	Misc. Structure	Misc. wreck
Total	11450	6669	2758	1553	2971	2694
Train	10426	6019	2464	1366	2647	2377
Val	586	390	180	129	210	232
Test	438	260	114	58	114	85

is sampled to match the texture color of the assigned model to look realistic. Table I summarizes the images collected from different categories of structures, and a sample of the different underwater scenes simulated can be found in Fig. 2.

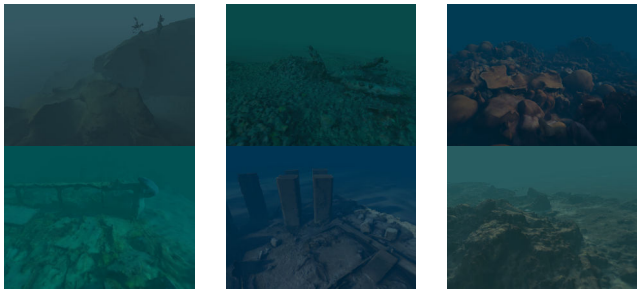


Fig. 2: Examples of underwater scenes constructed in simulation

A. Data Collection

We simulated an AUV, performing data collection in the underwater environments in Unity. The simulated AUV is equipped with an SBES and two RGB-D cameras in a stereo configuration with a baseline of $0.5\,\mathrm{m}$. The echosounder has a range of 0.5 to 10 meters, and camera parameters such as resolution and focal length are kept the same as the actual camera used on a relatively inexpensive underwater robot, BlueRobotics BlueROV2[3]. Both the echosounder and cameras are placed at a pitch of 30-degree towards the sea floor to reduce the area of background water captured. For each scene, the AUV traveled to waypoints randomly generated within a bounding box around each scene. The path is calculated via three-dimensional A*. Each path was constrained to have length greater than a pre-defined distance and the process stopped only when a target number of waypoints had been reached. Vertical motion of the AUV was limited within a range of 20 degrees to avoid large and sudden changes in pitch which rarely occurs in the real-world. During navigation, the cameras and the echosounder measurements were recorded at a fixed rate of 10 frames per second, together with the AUV's pose. To represent different possible real-world underwater environments better, underwater visibility – the gradient and conditions in visibility over distance – was randomized [43]. Effects such as camera grain and vignette were also included randomly. Fig. 3 shows an instance of the data gathered. Note that semantic segmentation of the model and background region was also provided to facilitate potential segmentation tasks.

IV. Approach

In this section, the proposed methodology is described in detail and the main components are outlined – an overview is shown in Fig. 5.

[3]https://bluerobotics.com/store/rov/bluerov2/

Fig. 3: Dataset instance with RBG image, depth map, and semantic label.

A. Echosounder Fusion

While our method uses both monocular imagery and SBES measurements as input for dense depth estimation, it is difficult to fuse an echosounder measurement naively with image features as it is extremely sparse compared to the number of image pixels (1 vs >100k). Hence, we first attempt to address the sparsity issue via transforming the sparse measurement into a denser representation so that echosounder information can be more effectively fused with monocular image features.

To measure the distance to an object, an SBES emits an acoustic pulse - approximately in the shape of a cone and listens to the reflected pulses, using the time of flight for distance calculation. According to the echosounder and camera model described in our previous work [44], as shown in Fig. 4a, the base of the sound cone can be approximated as a circle with the center ${}_C\mathbf{c}_i$ in the camera reference frame as:

$$ {}_C\mathbf{c}_i = {}_C\mathbf{t}_E + m_i \cdot \tilde{\mathbf{v}} \tag{1} $$

where m_i is the echosounder distance reading, ${}_C\mathbf{t}_E$ is the echosounder's location, and $\tilde{\mathbf{v}}$ is the direction unit vector with respect to the camera reference frame. Given cone angle a, the radius of the circular region is calculated as:

$$ r_i = m_i \cdot \tan(a) \tag{2} $$

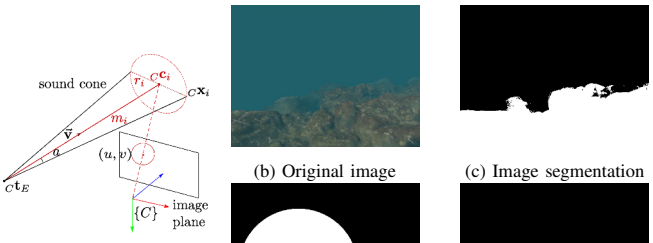
and with camera intrinsics K, assuming the pinhole camera model, we can identify the pixel coordinate (u, v) that corresponds to the center of the same circular region, projected onto the image plane, as:

$$ \begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = K \cdot {}_C\mathbf{c}_i \tag{3} $$

where

$$ u = \frac{u'}{w'} \qquad v = \frac{v'}{w'} \tag{4} $$

Although we now know that the echosounder measurement must correspond to the nearest point within the projected circle, it still covers a relatively large area on the image in our particular setting. Hence, we further reduced the candidate region by performing image segmentation to exclude background regions where the echosounder measurement is unlikely to pick up from. After preliminary experiments, we found a clustering algorithm based on Gaussian mixture model (GMM) [45] to perform well with underwater imagery. Finally, the remaining area was assigned the distance measurement returned by the echosounder while the rest are assigned 0, indicating an unknown value. This generated a mask of size $H \times W$, where H and W represent the height and width of the monocular image, respectively. The



(a) Echosounder-camera model modified from our previous work [44].

(b) Original image

(c) Image segmentation

(d) Projected sound cone

(e) Echosounder mask

Fig. 4: Conversion of a single echosounder measurement into a mask representation. From the raw echosunder reading, we projected the sound cone onto the image plane, took its intersection with foreground regions identified using image segmentation, and assigned the measurement to pixels within the area.

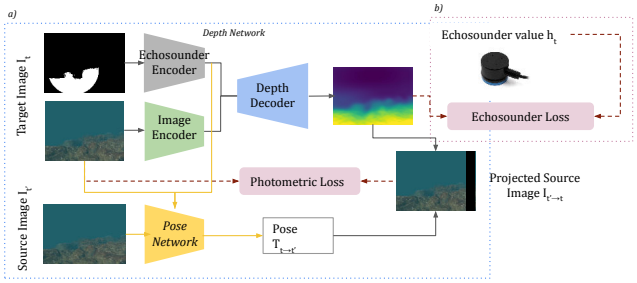


Fig. 5: Overview of our framework. a) - Photometric loss only. b) Ablation with direct supervision from echosounder measurements.

conversion process is shown in Fig. 4 (b)-(d). Compared to naive concatenation of image and echosounder measurement, this mask provides a denser representation that can be better encoded and fused with the input image features for learning.

B. Networks

Our depth estimation network takes in the target monocular image I_t and the corresponding echosounder mask M_t as input to estimate dense depth D_t. To work with the echosounder mask input, we extended Monodepth2 [18], a state-of-the-art self-supervised monocular depth estimation network to include an additional encoder for encoding the echosounder mask input, as shown in Fig. 5 a). Separate encoders, based on ResNet-18 [46], were used for the monocular image and echosounder mask to encode features from the two different modalities independently. Intermediate fusion was used to integrate the multi-modal features progressively, as the decoder network takes in multi-scale deep features of both modalities and concatenates them together to estimate depth at multiple scales.

A separate pose network was used to estimate camera ego-motion between successive image pairs. The pose network takes in a target-source monocular image pair and the echosounder mask M_t as input and outputs the camera rotation and translation between the image pair. Both the depth and pose networks were trained simultaneously.

C. Loss Functions

Self-supervised depth estimation can be expressed as a view-synthesis problem, where we try to predict a target image I_t given the viewpoint of adjacent images $I_{t'}$. For successful synthesis of the target image from source images, we need to accurately estimate both depth D_t and the relative pose between the target-source images. Hence, similar to [18], [19], photometric reprojection loss L_p was used to estimate the pixel-level similarity between the target image I_t and the synthesized target image $I_{t'\to t}$:

$$L_p = \sum_{t'} pe(I_t, I_{t'\to t}) \tag{5}$$

and

$$I_{t'\to t} = I_t\Big\langle \text{proj}(D_t, T_{t\to t'}, K)\Big\rangle \tag{6}$$

where relative pose of source view $I_{t'}$ with respect to the target image I_t is expressed as $T_{t\to t'}$, and proj() is the projected coordinates of depth D_t in $I_{t'}$ using camera intrinsics K. $\langle\rangle$ is the sampling operator. *pe*, the photometric reconstruction error consisting of the L1 distance in pixel space and the SSIM loss [47], is calculated by:

$$pe(I_a, I_b) = \frac{\alpha}{2}(1 - \text{SSIM}(I_a, I_b) + (1-a)\|I_a - I_b\|_1) \tag{7}$$

where α is a hyper-parameter to weigh contributions of the two terms.

We also include the edge-aware smoothness L_s [48] to encourage smooth depth estimations locally while preserving sharp boundaries:

$$L_s = |\partial_x d_t^*| e^{-|\partial_x I_t|} + |\partial_y d_t^*| e^{-|\partial_y I_t|} \tag{8}$$

where $d_t^* = d_t/\bar{d}$ is the mean-normalized inverse depth.

Similarly to [18], we further applied a binary per-pixel mask μ to the photometric reprojection loss L_p to filter out static frames and texture-less regions, which can be dominant in underwater scenes:

$$\mu = [\min_{t'} pe(I_t, I_{t'\to t}) < \min_{t'} pe(I_t, I_{t'})] \tag{9}$$

Finally, the overall loss L is the linear combination of the photometric reprojection loss and smoothness loss:

$$L = \mu L_p + L_s \tag{10}$$

V. Experiments

We evaluated our method quantitatively and qualitatively on our proposed dataset as well as on real-world images. Results are presented in the following section.

A. Evaluation on Proposed Dataset

There are 25299 images and 1069 images used for training and validation, respectively. Our method was implemented in Pytorch and experiments were done on a workstation with a Nvidia GeForce RTX 3090 GPU. Images were scaled down to a size of 160×224, using batch size of 24 and an initial learning rate of $1e-4$ for training.

Table II shows the quantitative results on the proposed test set in comparison to existing methods, all trained on our

TABLE II: Quantitative evaluation of our method against other monocular depth estimation methods. 'Superv' column describes training supervision of each method. '*S*' - supervised, '*M*' - self-supervised, '*GA*' - GAN. All '*M*' and '*GA*' methods are scaled using echosounder measurement for evaluation for fair comparison. Pixels with ground truth depth $> 20\,\mathrm{m}$ are masked out.

Method	Superv.	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
BTS [49]	S	0.2348	0.658	2.188	0.226	0.642	0.941	0.986
SfMLearner [19]	M	0.471	3.321	3.651	0.429	0.452	0.727	0.860
SC-Depth [50]	M	0.394	2.272	2.896	0.335	0.492	0.842	0.926
PackNet-Sfm [51]	M	0.319	1.302	2.437	0.291	0.574	0.871	0.945
Monodepth2 [18]	M	**0.309**	1.257	2.391	0.280	**0.587**	0.887	0.952
UW-Net [12]	GA	0.939	8.459	5.980	0.625	0.246	0.484	0.683
Ours	M	**0.309**	**1.230**	**2.356**	**0.277**	0.582	**0.897**	**0.957**

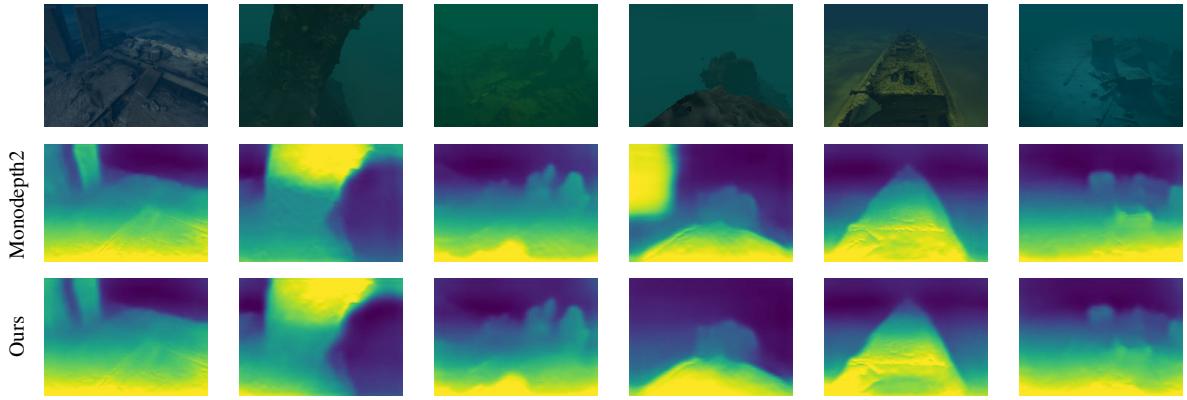


Fig. 6: A sample of qualitative comparison of Monodepth2 [18] and ours.

dataset from scratch. The existing methods include recent self-surpervised and GAN-based methods. We included also a supervised method as a lower bound of performance, even if it would be difficult to obtain accurate ground truth dataset in the underwater domain. For all unsupervised methods that only predict depth up to an unknown scale factor, the resulting depths were scaled according to the ratio of echosounder distance measurement to the nearest depth within the projected echosounder sound cone to have a fair comparison. This strategy was used as it follows the measurement returned by a real echosounder used on our underwater robot, where the strongest signal return would determine the corresponding distance measurement, while dense ground truth depth used for median scaling cannot be obtained in this setting. From Table II, as expected, we can see that BTS [49], with ground truth supervision, performed better, than other self-supervised method with ours being relatively close. Our method achieved positive results, outperforming (or ties) all other self-supervised and GAN-based methods, in almost all evaluation metrics. These results demonstrate that our self-supervised approach can still work reasonably well and is preferable than a supervised one, when a large dataset with accurate ground truth is not available.

Fig. 6 also shows some examples of depths maps produced for qualitative evaluation. Consistent with our quantitative results, our method produced the best depth maps overall. Compared to depth maps generated using Monodepth2 [18], our method had less incorrectly predicted background regions, see, e.g., top left corner of image in the fourth column. The echosounder mask encoding provided additional information that helped distinguish foreground and background regions, which were prone to error due to the lack of texture for photometric self-supervision. Also, compared to our method, Monodepth2 [18] tended to predict objects to be nearer than in reality, as seen in image 1 (second pillar from top left) and 5 (depression at near end of the boat). However, we notice that depth maps produced by our method were sometimes less sharp compared to [18], e.g., image 6, which may lead to loss of details. This may be due to the fact that the echosounder mask un-conservatively assigns the echosounder measurement to all regions within the mask, overlooking slight variances in depth.

B. Supervision with Echosounder Measurement

In addition to indirectly providing information on ground truth measurements through echosounder mask encoding, we also explored the use of direct supervision using the echosounder measurement itself (Fig. 5 b). Using the self-supervised losses described in the previous section, we were only able to estimate depth up to an unknown scale factor. Hence, to estimate metric depth, we included an additional L1 loss L_b using the single echosounder distance measurement h_t as the supervisory signal:

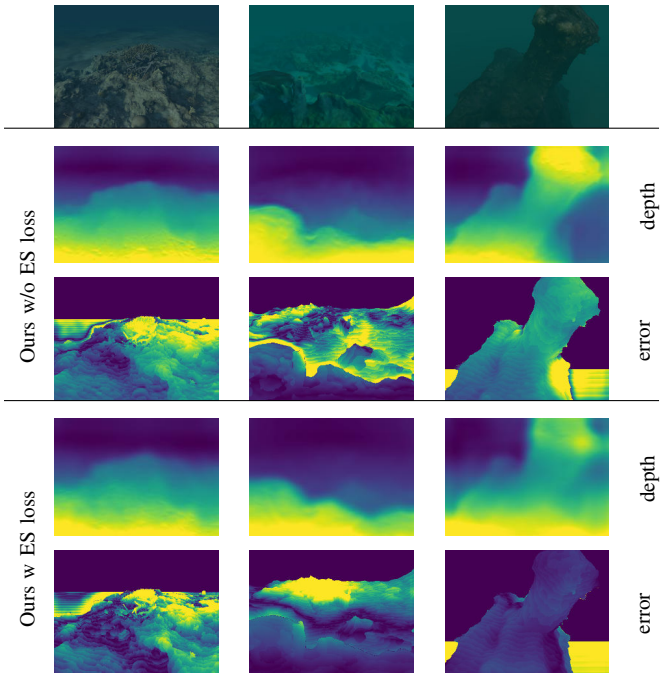


Fig. 7: Comparison of depth maps and absolute error maps from our method with and w/o echosounder loss. Depth maps: lighter color represents nearer; error maps: lighter color represents bigger error. Pixels with ground truth depth > 20 are masked out for error maps.

$$L_b = |\frac{1}{k} \sum_{p \in P_k} D_t(p) - h_t| \tag{11}$$

where P_k consists of the smallest k depths of regions within the echosounder mask. Average of the smallest k depths instead of the smallest depths was used to protect against outliers.

The total loss now becomes:

$$L = L_p + L_s + L_b \tag{12}$$

where each component is weighed using hyper-parameters accordingly.

Qualitative results of depth and absolute error maps produced using models trained with the new loss, compared to only using the echosounder mask are shown in Fig. 7. We can see that, in general, the quality of the depth maps produced by our method with echosounder loss is a little less sharp in comparison. However, in terms of absolute error maps, our method with echosounder loss performed better (2.258 vs 2.356). This suggests that by using the echosounder value as a supervisory signal does indeed allow our method to learn about the overall world scale to some extent, hence allowing it to outperform its counterpart without echosounder loss. Nevertheless, with only a single echosounder measurement with uncertain position, it proved to be difficult for our model to learn jointly the *absolute* scale and *relative* depth relationships between objects in the scene.

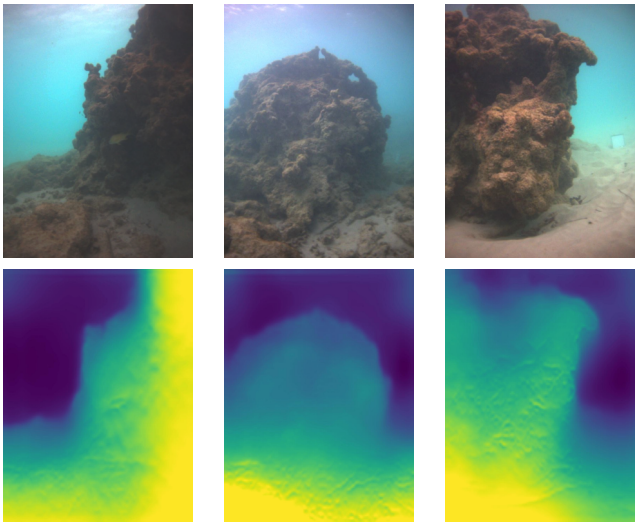


Fig. 8: Qualitative evaluation on a dataset we collected from a BlueROV2.

C. Evaluation on Real-World Images

Next, we carried out some qualitative evaluation on real-world data to demonstrate the transferability of our method trained using the proposed dataset. Real-world data collection were performed in the Caribbean Sea with a BlueROV2 installed with the Sony IMX273 camera and Ping echosounder. The camera has a resolution of 1.6 MP, a horizontal and vertical FOV of 96° and 72°. The echosounder has a beam width a of 30° and a maximum range set to $4\,\mathrm{m}$. Actual images and their predicted depth maps are shown in Fig. 8. From the figure, we can see that despite the images being different from our dataset used for training, our model was still able to generalize relatively well to real-world images without fine-tuning. We also measured the inference time of a single image on a .8 GHz Intel i7 laptop with Nvidia Geforce 1660Ti, which is about 0.07 seconds per inference, suggesting feasibility for real-time application.

VI. Conclusion and Future Steps

We presented the first deep underwater monocular depth estimation approach with single-beam echosounder input, as well as a new large-scale dataset for underwater vision tasks. We also validated our approach with evaluations in both the proposed dataset and real-world underwater images.

Our proposed method provides potential in correcting the depth estimates with an inexpensive sensor configuration. The experiments provided some insights on next steps. We will investigate replacing the GMM algorithm for echosounder mask generation with a learnable network to capture closer objects in finer detail. We will also explore a denser integration of echosounder-camera data via including all the signal strengths and via a temporal network to improve the depth estimation which currently is limited due to the single supervisory signal from the echosounder. We also plan to expand our current dataset with more path variations, e.g., circular paths and dynamic objects such as fish, to enhance the realism of our dataset and cover scenarios that are encountered in the real world.

REFERENCES

[1] Y. R. Petillot, G. Antonelli, G. Casalino, and F. Ferreira, "Underwater robots: From remotely operated vehicles to intervention-autonomous underwater vehicles," *IEEE Robotics & Automation Magazine*, vol. 26, no. 2, pp. 94–101, 2019.

[2] H. Cho, B. Kim, and S.-C. Yu, "Auv-based underwater 3-d point cloud generation using acoustic lens-based multibeam sonar," *IEEE Journal of Oceanic Engineering*, vol. 43, no. 4, pp. 856–872, 2018.

[3] A. Maccarone, A. McCarthy, X. Ren, R. E. Warburton, A. M. Wallace, J. Moffat, Y. Petillot, and G. S. Buller, "Underwater depth imaging using time-correlated single-photon counting," *Optics express*, vol. 23, no. 26, pp. 33 911–33 926, 2015.

[4] A. Filisetti, A. Marouchos, A. Martini, T. Martin, and S. Collings, "Developments and applications of underwater lidar systems in support of marine science," in *OCEANS 2018 MTS/IEEE Charleston*. IEEE, 2018, pp. 1–10.

[5] B. Joshi, S. Rahman, M. Kalaitzakis, B. Cain, J. Johnson, M. Xanthidis, N. Karapetyan, A. Hernandez, A. Quattrini Li, N. Vitzilaios, and I. Rekleitis, "Experimental Comparison of Open Source Visual-Inertial-Based State Estimation Algorithms in the Underwater Domain," in *Proc. IROS*, 2019.

[6] M. Roznere and A. Quattrini Li, "Real-time model-based image color correction for underwater robots," in *Proc. IROS*, 2019.

[7] J. Li, K. A. Skinner, R. Eustice, and M. Johnson-Roberson, "Watergan: Unsupervised generative network to enable real-time color correction of monocular underwater images," *IEEE Robotics and Automation Letters (RA-L)*, 2017.

[8] M. Poggi, F. Tosi, K. Batsos, P. Mordohai, and S. Mattoccia, "On the synergies between machine learning and binocular stereo for depth estimation from images: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5314–5334, 2021.

[9] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[10] P. K. Nathan Silberman, Derek Hoiem and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *ECCV*, 2012.

[11] P. Hambarde, S. Murala, and A. Dhall, "Uw-gan: Single-image depth estimation and image enhancement for underwater images," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021.

[12] H. Gupta and K. Mitra, "Unsupervised single image underwater depth estimation," in *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2019, pp. 624–628.

[13] P. L. Drews, E. R. Nascimento, S. S. Botelho, and M. F. M. Campos, "Underwater depth estimation and image restoration based on single images," *IEEE computer graphics and applications*, vol. 36, no. 2, pp. 24–35, 2016.

[14] A. Galdran, D. Pardo, A. Picón, and A. Alvarez-Gila, "Automatic red-channel underwater image restoration," *Journal of Visual Communication and Image Representation*, vol. 26, pp. 132–145, 2015.

[15] C. O. Ancuti, C. Ancuti, C. De Vleeschouwer, L. Neumann, and R. Garcia, "Color transfer for underwater dehazing and depth estimation," in *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 695–699.

[16] Z. Li, P. Tan, R. T. Tan, D. Zou, S. Zhiying Zhou, and L.-F. Cheong, "Simultaneous video defogging and stereo reconstruction," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4988–4997.

[17] Y.-T. Peng and P. C. Cosman, "Underwater image restoration based on image blurriness and light absorption," *IEEE transactions on image processing*, vol. 26, no. 4, pp. 1579–1594, 2017.

[18] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth prediction," in *The International Conference on Computer Vision (ICCV)*, October 2019.

[19] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1851–1858.

[20] Z. Yin and J. Shi, "Geonet: Unsupervised learning of dense depth, optical flow and camera pose," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1983–1992.

[21] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3828–3838.

[22] X. Dong, M. A. Garratt, S. G. Anavatti, and H. A. Abbass, "Towards real-time monocular depth estimation for robotics: A survey," *arXiv preprint arXiv:2111.08600*, 2021.

[23] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, 2017.

[24] P. Corke, C. Detweiler, M. Dunbabin, M. Hamilton, D. Rus, and I. Vasilescu, "Experiments with underwater robot localization and tracking," in *Proc. ICRA*, 2007.

[25] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual–inertial odometry using nonlinear optimization," *Int. J. Robot. Res.*, vol. 34, no. 3, pp. 314–334, 2015.

[26] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, 2018.

[27] J. Salvi, Y. Petillo, S. Thomas, and J. Aulinas, "Visual SLAM for underwater vehicles using video velocity log and natural landmarks," in *MTS/IEEE OCEANS*, 2008, pp. 1–6.

[28] C. Beall, F. Dellaert, I. Mahon, and S. B. Williams, "Bundle adjustment in large-scale 3d reconstructions based on underwater robotic surveys," in *Proc. OCEANS*, 2011, pp. 1–6.

[29] F. Shkurti, I. Rekleitis, M. Scaccia, and G. Dudek, "State estimation of an underwater robot using visual and inertial information," in *Proc. IROS*, 2011, pp. 5054–5060.

[30] G. Loianno, C. Brunner, G. McGrath, and V. Kumar, "Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and imu," *IEEE J. Robot. Autom.*, vol. 2, no. 2, pp. 404–411, 2016.

[31] Y. Zhang, J. Tan, Z. Zeng, W. Liang, and Y. Xia, "Monocular camera and imu integration for indoor position estimation," in *EMBS*, 2014.

[32] J. Folkesson, J. Leonard, J. Leederkerken, and R. Williams, "Feature tracking for underwater navigation using sonar," in *Proc. IROS*. IEEE, 2007, pp. 3678–3684.

[33] K. Richmond, C. Flesher, L. Lindzey, N. Tanner, and W. C. Stone, "SUNFISH®: A human-portable exploration AUV for complex 3D environments," in *MTS/IEEE OCEANS Charleston*, 2018, pp. 1–9.

[34] S. Rahman, A. Quattrini Li, and I. Rekleitis, "SVIn2: An Underwater SLAM System using Sonar, Visual, Inertial, and Depth Sensor," in *Proc. IROS*, 2019, pp. 1861–1868.

[35] S. Zhao, M. Gong, H. Fu, and D. Tao, "Adaptive context-aware multi-modal network for depth completion," *IEEE Transactions on Image Processing*, vol. 30, pp. 5264–5276, 2021.

[36] J. Park, K. Joo, Z. Hu, C.-K. Liu, and I. So Kweon, "Non-local spatial propagation network for depth completion," in *European Conference on Computer Vision*. Springer, 2020, pp. 120–136.

[37] Z. Feng, L. Jing, P. Yin, Y. Tian, and B. Li, "Advancing self-supervised monocular depth learning with sparse lidar," *arXiv preprint arXiv:2109.09628*, 2021.

[38] Y. Long, D. Morris, X. Liu, M. Castro, P. Chakravarty, and P. Narayanan, "Radar-camera pixel depth association for depth completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 507–12 516.

[39] K. K. Parida, S. Srivastava, and G. Sharma, "Beyond image to depth: Improving depth prediction using echoes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8268–8277.

[40] M. Ferrera, V. Creuze, J. Moras, and P. Trouvé-Peloux, "Aqualoc: An underwater dataset for visual–inertial–pressure localization," *The International Journal of Robotics Research*, vol. 38, no. 14, pp. 1549–1559, 2019.

[41] A. Mallios, E. Vidal, R. Campos, and M. Carreras, "Underwater caves sonar data set," *The International Journal of Robotics Research*, vol. 36, no. 12, pp. 1247–1251, 2017.

[42] D. Berman, D. Levy, S. Avidan, and T. Treibitz, "Underwater single image color restoration using haze-lines and a new quantitative dataset," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 8, pp. 2822–2837, 2020.

[43] P. Yang, H. Liu, M. Roznere, and A. Q. Li, "Monocular camera and single-beam sonar-based underwater collision-free navigation with domain randomization," *arXiv preprint arXiv:2212.04373*, 2022.

[44] M. Roznere and A. Quattrini Li, "Underwater monocular depth estimation using single-beam echo sounder," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[45] H. Permuter, J. Francos, and I. H. Jermyn, "Gaussian mixture models of texture and colour for image database retrieval," in *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, vol. 3. IEEE, 2003, pp. III–569.

[46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[47] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[48] C. Wang, J. M. Buenaposada, R. Zhu, and S. Lucey, "Learning depth from monocular videos using direct methods," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2022–2030.

[49] J. H. Lee, M.-K. Han, D. W. Ko, and I. H. Suh, "From big to small: Multi-scale local planar guidance for monocular depth estimation," *arXiv preprint arXiv:1907.10326*, 2019.

[50] J.-W. Bian, H. Zhan, N. Wang, Z. Li, L. Zhang, C. Shen, M.-M. Cheng, and I. Reid, "Unsupervised scale-consistent depth learning from video," *International Journal of Computer Vision (IJCV)*, 2021.

[51] V. Guizilini, R. Ambrus, S. Pillai, A. Raventos, and A. Gaidon, "3d packing for self-supervised monocular depth estimation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.