Bathymetric Reconstruction From Sidescan Sonar With Deep Neural Networks

Yiping Xie, Nils Bore, and John Folkesson, *Senior Member, IEEE*

***Abstract*—In this article, we propose a novel data-driven approach for high-resolution bathymetric reconstruction from sidescan. Sidescan sonar intensities as a function of range do contain some information about the slope of the seabed. However, that information must be inferred. In addition, the navigation system provides the estimated trajectory, and normally, the altitude along this trajectory is also available. From these, we obtain a very coarse seabed bathymetry as an input. This is then combined with the indirect but high-resolution seabed slope information from the sidescan to estimate the full bathymetry. This sparse depth could be acquired by single-beam echo sounder, Doppler velocity log, and other bottom tracking sensors or bottom tracking algorithm from sidescan itself. In our work, a fully convolutional network is used to estimate the depth contour and its aleatoric uncertainty from the sidescan images and sparse depth in an end-to-end fashion. The estimated depth is then used together with the range to calculate the point's three-dimensional location on the seafloor. A high-quality bathymetric map can be reconstructed after fusing the depth predictions and the corresponding confidence measures from the neural networks. We show the improvement of the bathymetric map gained by using sparse depths with sidescan over estimates with sidescan alone. We also show the benefit of confidence weighting when fusing multiple bathymetric estimates into a single map.**

***Index Terms*—Bathymetric mapping, data-driven, neural network, sidescan sonar (SSS).**

I. Introduction

SIDESCAN and multibeam echo sounder (MBES) are the commonly used sonars for surveying the seabed. Sidescan sonar (SSS) is used for obtaining detailed seabed images due to its high resolution and wide swath coverage, while MBES is used when constructing a bathymetric map due to its ability to directly measure the seafloor's 3-D geometry. The MBES are normally mounted on ships or large autonomous underwater vehicles (AUVs). Ones small enough for smaller AUVs will not have sufficient resolution in the across track direction. They are also relatively expensive compared to single array SSS. Since sidescans do not have any across track array, they can easily be mounted on small and more affordable AUVs. The disadvantage is that there is also no across-track angular resolution, and thus, the sidescan gives a 2-D projection of the 3-D seabed. An estimate of the depth coordinates would resolve the 3-D positions. The sidescan intensities do contain some information about the seafloor's material and elevation changes: harder materials tend to have higher return intensities; the nadir range in every ping gives a single altitude reading; and, more importantly, the intensity changes indicate the change of the incidence angle [1]. Even though the unknown bottom material and other disturbances make it difficult to estimate the depth from sidescan returns analytically, data-driven methods [2] have shown promising results in estimating depth contours from sidescan intensities. Here, we will add sparse height constraints from the altimeter readings along the trajectory to the neural networks estimated depth approach. This will significantly improve the accuracy of the method.

Over the last decades, deep learning has made a significant impact on the computer vision field. Among the various computer vision tasks, 3-D reconstruction from monocular camera images can be seen as an analogous task to bathymetry from sidescan. Early on, shape from shading techniques based on physical principles were used in 3-D scene reconstruction, but recently deep neural networks (DNNs) have become the state-of-the-art methods. Usually, DNNs estimate the depth from monocular images, and a pinhole model is used to reconstruct the 3-D point clouds [3].

Similar to how neural networks have been used for estimating depth from monocular camera images with sparse depth provided by low-resolution depth sensors such as LiDARs [4], we train convolutional neural networks (CNNs) with sparse depth provided from the altimeter to predict a dense depth image from sidescan images, as shown in Fig. 1.

In theory, the sidescan intensities contain information on the surface gradients. If we integrate the surface gradients, it will inevitably drift further as we get far from the starting point at the nadir. Our prior work [2] shows that the estimated errors are most significant as one moves further from the sensor. As a matter of fact, there are problems of treating sidescan images more or less as camera images in a CNN. A convolutional filter assumes that the interpretation is invariant to pixel position. For sidescan images, that is not exactly the case. There is a changing interpretation as one moves further away. The geometry shifts and the per column change rate of the incidence angle are less in the far than the close region.

Manuscript received 23 March 2021; revised 21 July 2022 and 28 October 2022; accepted 31 October 2022. Date of publication 23 December 2022; date of current version 14 April 2023. This work was supported in part by the Wallenberg AI, Autonomous Systems and Software Program funded by the Knut and Alice Wallenberg Foundation and in part by Stiftelsen för Strategisk Forskning through the Swedish Maritime Robotics Centre under Grant IRC15-0046. *(Corresponding author: Yiping Xie.)*

Associate Editor: M. Hayes.

The authors are with the Robotics, Perception and Learning Lab, Royal Institute of Technology, SE-100 44 Stockholm, Sweden (e-mail: yipingx@kth.se; nbore@kth.se; johnf@kth.se).

Digital Object Identifier 10.1109/JOE.2022.3220330

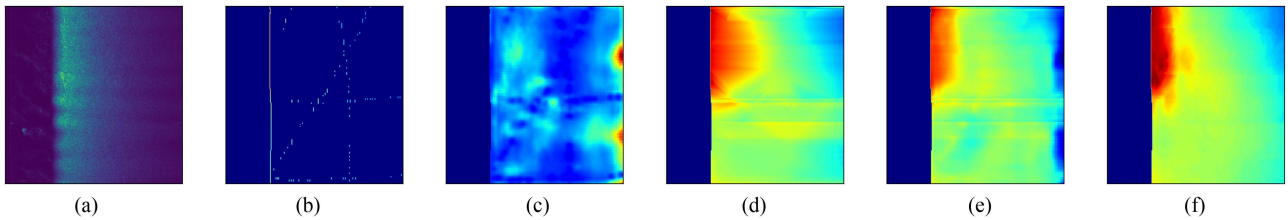


Fig. 1. Example of training image pairs. We divide the downsampled sidescan waterfall images into smaller windows in our network training step and associate the depth to each pixel given bathymetry from the multibeam survey to form the ground truth. (a) Sidescan intensities input window with size 256×256. (b) Sparse depth input window. (c) Uncertainty estimation output windows. (d) Interpolated depth window from the sparse depth data. (e) Predicted depth output window. (f) Ground truth depth window.

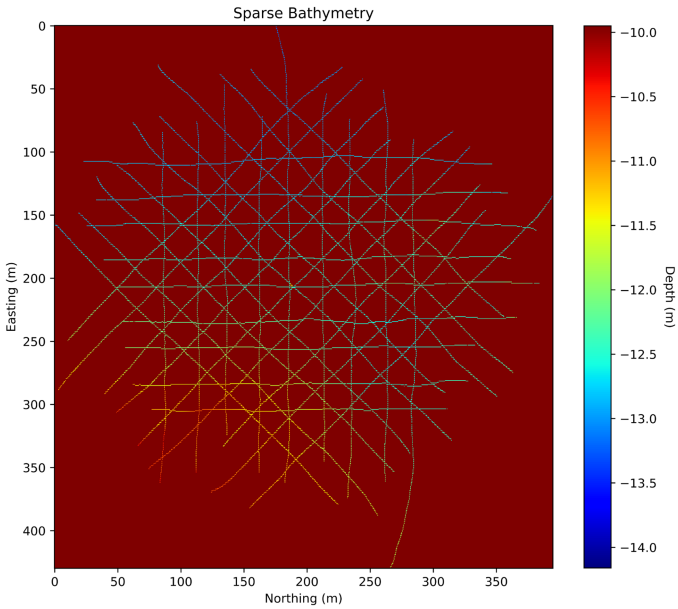


Fig. 2. Example of sparse seabed bathymetry from data set 2. Brown represents no data and the colored tracks are the lines of sparse data.

To address the increasing errors further from the sidescan sensor, we utilize the sparse depth (see Fig. 2) and an estimate of uncertainty to reconstruct a better bathymetry. We use the sparse depth provided by navigating and altitude measurement, as a constraint to the neural network, to reduce the drift errors. We use the uncertainty estimation to form a probabilistic model to fuse different estimations from different lines of the survey.

Our contributions are as follows.

1) We show that a novel framework to reconstruct bathymetry with high resolution and high quality with sidescan and sparse depth improves the accuracy over only using either of those two inputs.
2) We show that an aleatoric uncertainty estimation as a confidence measure for the depth estimation can improve the bathymetric map formed by combining estimates from overlapping survey lines.

A. Related Work

Woock and Frey [5] summarize the challenges of extracting depth information from SSS, which requires knowledge of sediment characteristics, surface and volume scattering properties, sound absorption and dispersion, water currents, variations in sound speed and the sonar transducer beam pattern. Assumptions must be made to simplify the methods, such as isospeed sound velocity profile (SVP).

Many attempts to reconstruct a shape from sidescan are based on "Lambert's cosine law." Li and Pai's work [6] is inspired by shape-from-shading methods with camera images [7], determining a Lambertian sonar model to obtain the approximation of the surface normals. However, the diffuse reflections assumptions' work much better for light than for sound. Coiras et al. [8] model the intensity as a function of bathymetry, the reflectance and the incident energy. The authors model the bathymetry and reflectance by splines and the incident energy by polynomials to reduce the dimensionality and apply standard gradient descent to the square error in modeled intensity versus measured. In [8], quality and quantity validations are done on a pipe of known diameter. Jones and Traykovski [9] collect data with a rotary SSS, which is mounted on underwater frames and rotated 360° to get a circular image. The authors exploit the shadows in sidescan images to estimate the elevation of bedform in shallow water and validate their methods on wave-orbital ripples and megaripples comparing with the multibeam data. Usually sidescan data are collected with overlapping swaths and this overlap can be used to infer depth from the images. Burguera and Oliver [10] exploit a physics-based SSS model to correct the raw data, including beam corrections and motion estimation, leading to a probabilistic framework to build a high-resolution bathymetric map from sidescan data. Johnson and Hebert [11] initialize the estimated bathymetry with the sparse direct measurements and form the bathymetry estimation from a full survey line as a global optimization. Bore and Folkesson [12] also perform a global optimization to estimate the bathymetry, but they use the sparse direct bathymetric measurements as constraints and a neural network to represent the estimated bathymetry instead of a grid or a mesh. Also, Bore and Folkesson [12] estimate the bathymetry from many survey lines of SSS so that the final bathymetric model is self-consistent. The authors in [11] and [12] remove the SSS data corresponding to shadows since the reflection model cannot model the shadows. Cuschieri and Hebert [13], on the other hand, identify shadows in the SSS data and use trigonometry to calculate the height of the objects that cause the shadows. Subsequently, the authors integrate the individual geometries to a full seafloor map. Zhao et al. [14] also integrate the sparse bathymetry into the reconstruction, utilizing

a bottom tracking algorithm as an altimeter to obtain an initial seabed topography, which is then used as a constraint for the reconstruction model based on Lambertian law. The evaluation is done compared to the bathymetry constructed by a single-beam bathymetric system.

Deep learning approaches have been used in sidescan images for other tasks in recent years, such as object classification [15], [16], object detection [17] and semantic segmentation [18], [19]. Dzieciuch et al. [15] show that a simple CNN can be used for mine detection in SSS imagery and achieve comparable accuracy as human operators. Huo et al. [16] show that with deep transfer learning, a CNN could achieve high accuracy on the multiclass classification task on sidescan images. The authors also propose a semisynthetic data generation method to handle the imbalanced training data, which is a most common case in the real sidescan data sets. Einsidler et al. [17] show that deep transfer learning could also be used for underwater object detection. The authors adapt the state-of-the-art object detection algorithm, YOLO (You Only Look Once) [20], to sidescan images and achieve reasonable accuracy in anomaly detection after some fine-tuning on the real sidescan data set. Rahnemoonfar and Dobbs [18] propose a novel CNN architecture and illustrate its performance on pothhole semantic segmentation of sidescan images. Wu et al. [19] propose ECNet to perform semantic segmentation on sidescan with much fewer parameters and much faster speed, making it possible to be applied to real-time tasks on embedded platforms.

Our previous work [2] shows promising results for the task of depth estimation from sidescan images with deep learning techniques. Inspired by deep learning methods to estimate depth from single camera images [21], in [2], we propose a method to extract 3-D information from 2-D sonar images with DNNs. In this work, based on our prior one, we further exploit the sparse depth as a constraint for the DNNs and propose a framework of building a complete bathymetric map from sidescan. The use of the sparse depth, namely, a deep regression network taking the sidescan and sparse depth data as input, is inspired by Ma and Karaman [4]. In [4], the authors demonstrate their proposed framework outperforms the other depth fusion techniques on the task of depth completion from camera images and the available sparse depth. The uncertainty estimation in this work is closely related to [22] and [23]. In [22], the authors propose a simple framework to quantify predictive uncertainty in neural networks, which is easy to adapt to most of the deep learning approaches. They show that maximizing likelihood is a proper scoring rule, which measures the quality of predictive uncertainty [24]. In [23], the authors demonstrate that the framework proposed by Lakshminarayanan et al. [22] can also achieve reasonable results on pixel-level applications, such as depth estimation. And simply by filtering out the few extreme outliers with high uncertainty, one can improve the overall performance of the 3-D reconstruction.

The major difference between our work and the others to reconstruct bathymetry from sidescan is that we use a data-driven approach, whereas the prior works are model based. Our motivation is that some effects are not plausible to model yet the sidescan images do contain some information about them.

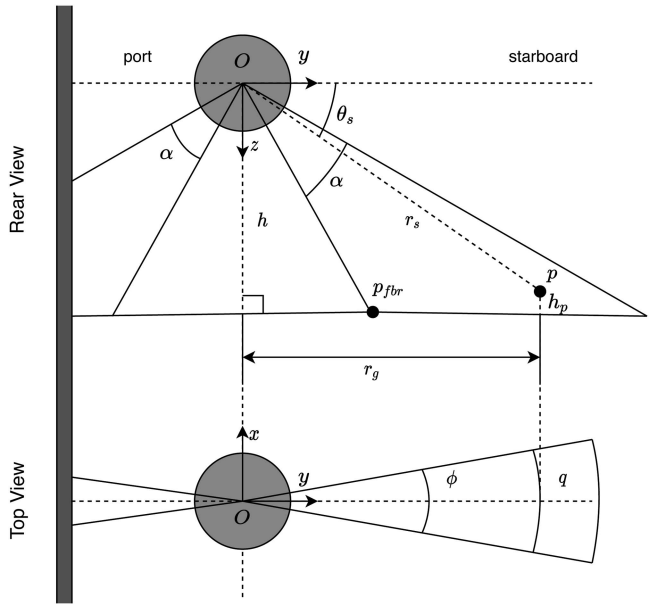


Fig. 3. SSS formation.

For example, an expert can tell if a sidescan image appears to be hard or soft bottom, rocks appear often as part of a larger geological formation and so on. It is not practical, however, to have experts estimate all the sediment characteristics in every sidescan images and model the surface scattering properties accordingly. Thus, we exploit data-driven methods to leverage deep learning's advantages of learning from patterns in the data distributions to compensate for those unmodeled effects.

B. Summary of the Proposed Method

Reconstructing the bathymetry from SSS is difficult. Many properties that are hard to model have large impacts on estimating depth contours from sidescan. With a data-driven approach, some of these can be partially compensated, but naturally there will be errors. Besides the unmodeled properties of the seabed and water column, the main source of errors is the navigation error between lines of the survey that provide the sparse depth information.

In this article, we develop a method that reconstructs the bathymetry relying on SSS, vehicle position and the altimeter. Such a data-driven method could, in principle, work with data produced by most standard sidescan surveys. We utilize the sparse depth to reduce the errors and propose a framework to estimate the depth and uncertainty at the same time and a probabilistic model to reconstruct the bathymetry.

II. Method

A. Sidescan Sonar Formation

Fig. 3 illustrates the top view and the rear view of an SSS with its sensor origin O at altitude h. Let p be a point in the ensonified region on the bathymetric surface $\mathcal{M} \subset \mathbb{R}^3$ with point altitude h_p, whose polar coordinates can be expressed in its slant range r_s and its grazing angle θ_s. The grazing angle θ_s can be calculated

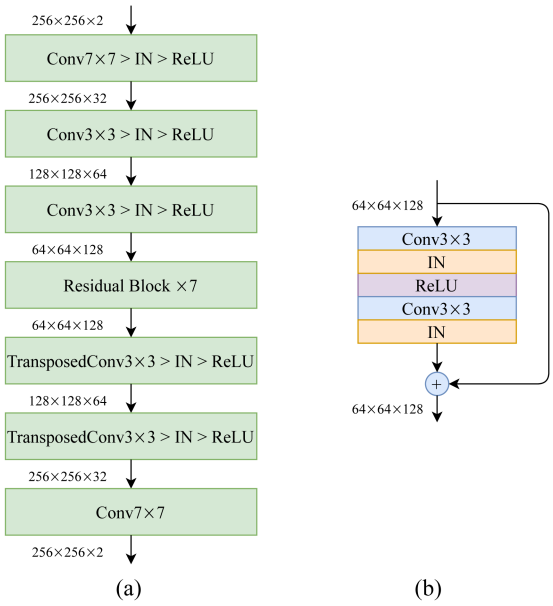


Fig. 4. CNN architecture. (a) Network architecture (ResNet architecture). (b) Resblock (residual block).

as follows if we ignore ray bending effects and h_p is known:

$$\theta_s = \arcsin\left(\frac{h - h_p}{r_s}\right). \tag{1}$$

The ground range r_g is the projection of vector $\vec{op}$ over the Y-axis. The vertical beamwidth α, sometimes referred to as sensor opening in the YZ plane, is usually 40°–60°, and the horizontal beamwidth ϕ, sometimes referred to as sensor opening in the XY plane, is usually around 0.1° [25]. Due to the horizontal beamwidth ϕ, the exact point position of p in the XY plane is ambiguous over the arc q; however, the assumption is usually that this fact can be neglected since ϕ is very small.

B. Sparse Depth Association

The idea is to use the set of points directly below the AUV along with its altitude and pose reading to generate a set of sparse depths, x, y, z. Then, for a waterfall image from any line in our survey, we can compute the range from the sonar at each ping to points from the sparse depth set that fall within its range and beam angle. We then create a second sparse depth image where the pixels correspond to the sidescan waterfall image but the values are now depths relative to the depth of the sonar [see Fig. 5(b)].

C. Uncertainty Estimation

Predicting depth from sidescan images can be seen as a pixel-level regression problem that can be addressed using neural networks. We preprocess the data so that the network is trying to estimate the point altitude h_p. We set the final layers of the neural network to output two values: mean $\mu(h_p)$ and variance $\sigma^2(h_p)$. We do a variational fit of the point altitude to a Laplacian distribution with the predicted mean $\mu(h_p)$ and

TABLE I
DATA SETS' DETAILS

	Dataset 1			Dataset 2
	Training	Validation	Testing	Testing
Survey lines	45	6	6	36
Image pairs	4352	640	592	1994
Max Depth	25.07 m			13.66 m
Min Depth	9.03 m			10.45 m
Sonar type	Edgetech 4200MP			Edgetech 4200MP
Sonar range	∼50 m			∼50 m
Sonar frequency	850 kHz			850 kHz

variance[1] $\sigma^2(h_p)$. The loss using negative log-likelihood (NLL) will be

$$-\log p_\theta(h_{p,gt}) = \frac{\|h_{p,gt} - \mu_\theta(h_p)\|}{\sigma_\theta^2(h_p)} + \log \sigma_\theta^2(h_p) + \log 2 \tag{2}$$

where θ represents the weights that parameterize the neural network and $h_{p,gt}$ denotes the ground truth.

As a comparison, the mean absolute error (MAE) loss can be seen as a special case of minimizing the above loss with a constant variance $\sigma_\theta^2 = 1$.[2] We model the likelihood to follow Laplacian distribution instead of Gaussian because we find $L1$ loss is more suitable than $L2$ loss for depth regression, as observed in [23]. Therefore, the NLL averaged over each pixel in sidescan images is considered as an aleatoric loss function for training the neural network [23].

During the test phase, we can use

$$c_p = \frac{1}{|\sigma_\theta^2(h_p)|}$$

as a confidence measure of the depth estimates, where $\sigma_\theta^2(h_p)$ is the uncertainty output of the network. By fusing all confidence estimates together, we are able to create a confidence map $\mathcal{U} \subset \mathbb{R}^3$ for the corresponding reconstructed bathymetry $\hat{\mathcal{M}} \subset \mathbb{R}^3$.

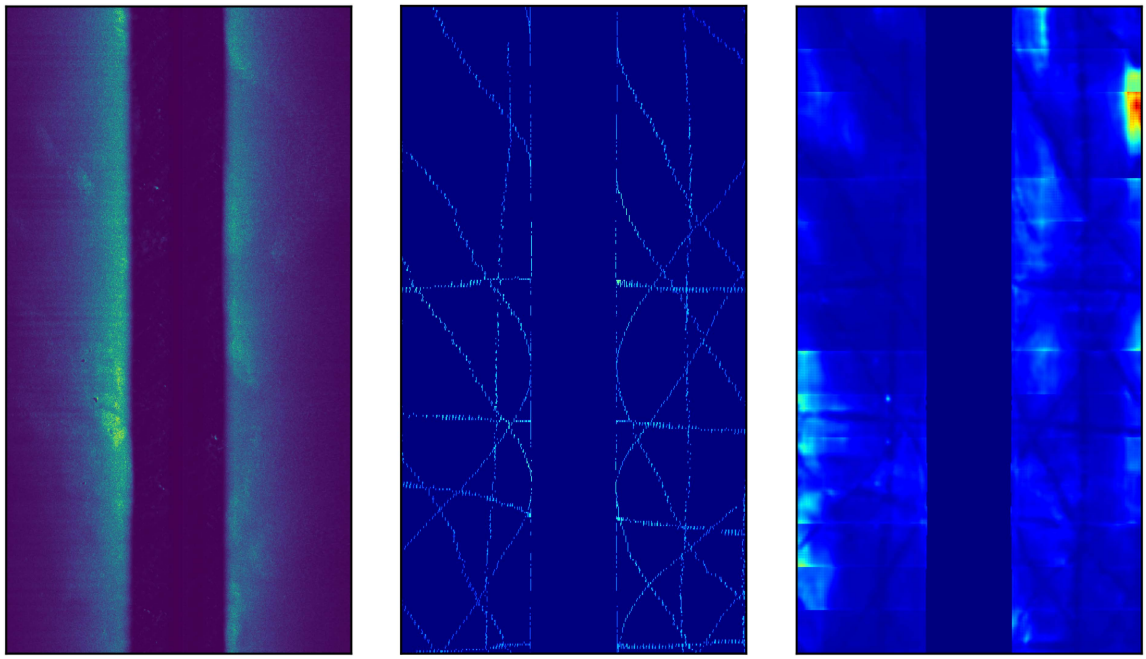
D. Bathymetry Reconstruction Model

For a sidescan waterfall image, let $I^{k,i}$ denote the returned intensity corresponding to ping number k and echo travel time interval i, $r_s^{k,i}$ denote the corresponding slant range and $r_g^{k,i}$ denote the ground range. The slant range can be deduced from the sound speed c^k and the two-way travel time $t^{k,i}$ between the SSS and the point at seafloor $\mathbf{p}^{k,i}$ as follows:

$$r_s^{k,i} = \frac{c^k \cdot t^{k,i}}{2}. \tag{3}$$

[1]Similar to [22], we enforce the positivity constraint on $\sigma^2(h_p)$ by passing it through the softplus function $\log(1 + \exp(\cdot))$ and add a minimum constant, e.g., 10^{-6}, for numerical stability.

[2]The value of 1 is arbitrary but other choices would only scale the loss and change the constant part and, thus, have no effect on the optimization.



(a) (b) (c)

Fig. 5. Example of part of one sidescan waterfall image, the associated sparse depth and the estimated uncertainty from a survey line in data set 2. Rainbow colormap is used to show the uncertainty. We can observe that the uncertainty is low (dark blue) at pixels where sparse depth is available. Also note that the uncertainty does not increase with distance from the nadir as would be the case without sparse depths. (a) SSS waterfall image. (b) Sparse depth waterfall image. (c) Uncertainty waterfall image.

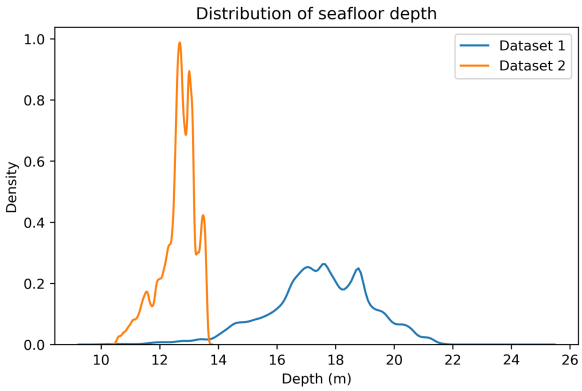


Fig. 6. Distributions of the depth of the seafloor in two data sets. The altitude distribution would be the same offset by the nearly constant depth of the sonar.

Note here we assume an isospeed SVP, which will introduce additional errors that could be eliminated if the SVP were known. The rotation $\mathcal{R}^k \in SO(3)$ and position $\mathbf{s}^k$ of the sonar is given by the navigation, and the point altitude $h_p^{k,i}$ can be estimated from our neural network. If we assume that the arc parameterized by $r_s^{k,i}$ has only one intersection with the seafloor surface, the point position $\mathbf{p}^{k,i} \in \mathbb{R}^3$ can be calculated by simply solving the following equation:

$$\begin{aligned} r_s^{k,i} &= \|\mathbf{s}^k - \mathbf{p}^{k,i}\|_2 \\ &= \sqrt{(r_g^{k,i})^2 + (h^k - h_p^{k,i})^2}. \end{aligned} \tag{4}$$

We can now fuse all estimates $\mathbf{p}^{k,i} = (p_x^{k,i}, p_y^{k,i}, p_z^{k,i})$ from the neural network from every survey line. We add them in a probabilistic fusion model to form a bathymetric mesh $\hat{\mathcal{M}}$ using the confidence estimates $c_p^{k,i}$. So a fused depth for point p_z on the reconstructed bathymetry grid $\hat{\mathcal{M}}$ is

$$\hat{p}_z = \frac{\sum_{p\in\mathcal{P}} p_z^{k,i} c_p^{k,i}}{\sum_{p\in\mathcal{P}} c_p^{k,i}} \tag{5}$$

where $\mathcal{P} \subset \mathbb{R}^3$ is the set of points that fall within the grid cell. The fused confidence map $\mathcal{U}$ can be obtained by averaging $c_p^{k,i}$ over $\mathcal{P}$

$$\hat{c} = \frac{\sum_{p\in\mathcal{P}} c_p^{k,i}}{|\mathcal{P}|}. \tag{6}$$

E. Sidescan Draping and Data Set Generation

1) Sidescan Geographic Referencing: To generate ground truth for the training and validation data sets, we need to associate sidescan intensities $I^{k,i}$ to its georeferenced coordinates $\mathbf{p}^{k,i}$ on a bathymetric mesh $\mathcal{M} \subset \mathbb{R}^3$, which is also referred to as *sidescan draping* [26]. To do so, the MBES is used to form such mesh, and the SVP is needed to determine the sound speed of the water layer. Also, the sensor position $\mathbf{s}^k \in \mathbb{R}^3$ and the rotation matrix $\mathcal{R}^k \in SO(3)$ of the sidescan must be known. Using this, we find the intersection of each SSS arc with the mesh.

2) Data Set Generation: The two data sets (see Table I) we used in this article were both collected with MMT Ping, a survey vessel equipped with a hull-mounted sidescan Edgetech 4200MP and RTK GPS to ensure high accuracy positioning. For both data sets, we have the high-resolution multibeam bathymetry collected with Reson 7125, treated as the ground truth. For every survey line, we divide each side of the waterfall

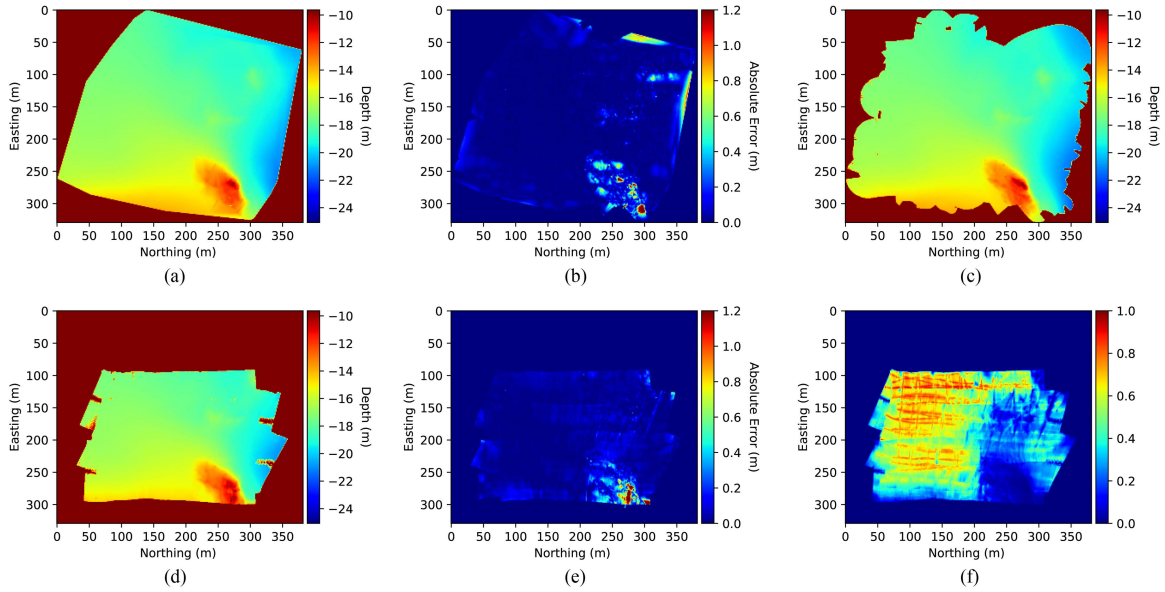


Fig. 7. Bathymetry on data set 1, produced by the altimeter, multibeam and sidescan respectively. (a) Bathymetry from linear interpolation of 57 lines of altimeter readings. (b) Absolute error map between (a) and the ground truth. (c) Ground truth bathymetry produced with multibeam data. (d) Bathymetry from six sidescan survey lines. (e) Absolute error map between (d) and (c). (f) Normalized confidence map for the bathymetry produced from sidescan, color red indicating high-confidence low uncertainty.

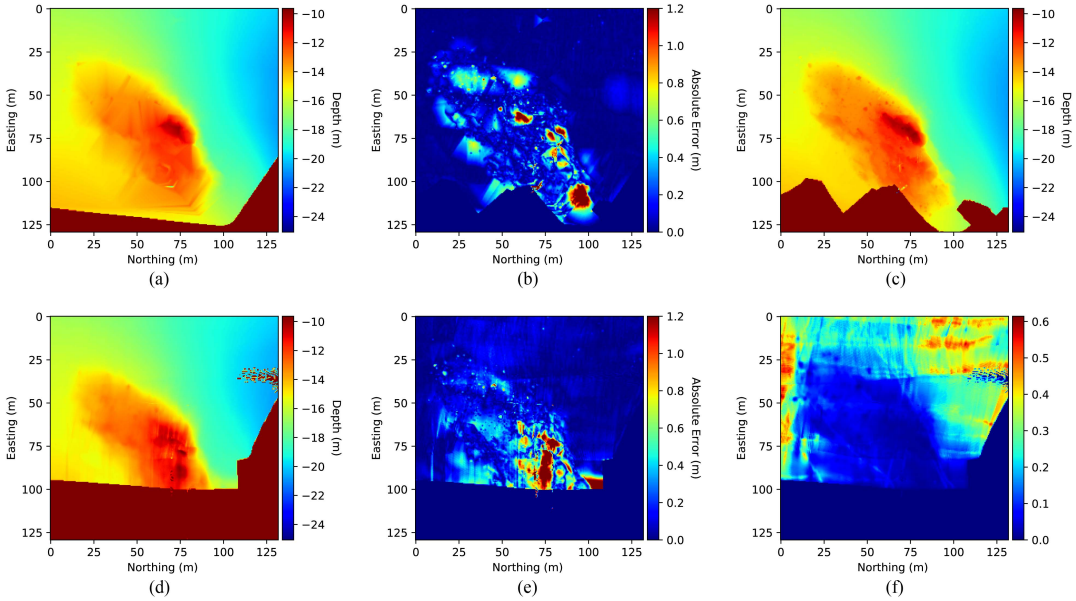


Fig. 8. Zoomed-in section of Fig. 7 where the area of interest contains a hill with multiple boulders. (a) Bathymetry from linear interpolation. (b) Error map—linear interpolation. (c) Bathymetry from MBES. (d) Bathymetry from SSS. (e) Error map—SSS. (f) Normalized confidence map—SSS.

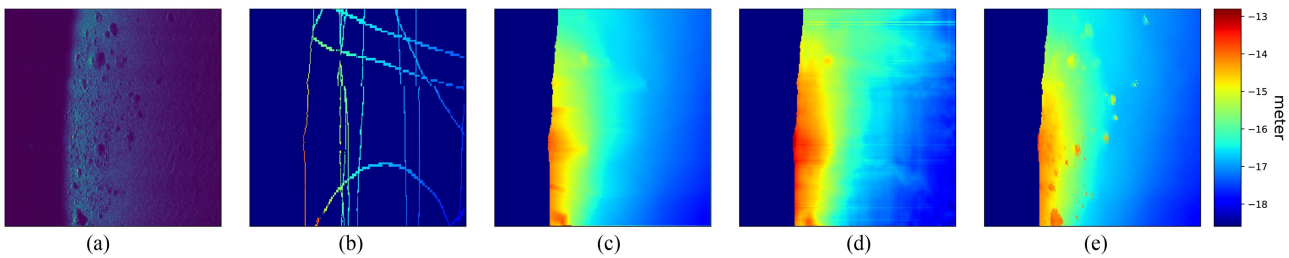


Fig. 9. Another example of image pairs from data set 1, where multiple rocks are observable from sidescan. (a) Sidescan intensities input window. (b) Sparse depth input window. (c) Interpolated depth window. (d) Predicted depth output window. (e) Ground truth depth window.

image into smaller windows with height $H = 256$ and width $W = 256$ downsampled from $\sim$6000 bins. The selection of H and W is chosen to fit the CNNs and, at the same time, ensure the sidescan's across-track resolution higher than the bathymetry resolution.

F. CNN Model

The model takes the sidescan intensities window [see Fig. 1(a)] and sparse depth window [see Fig. 1(b)] directly concatenated together as the input and output the estimated depth and uncertainty. The loss function is the NLL averaged over each valid pixel in the window

$$\mathcal{L} = \frac{1}{|D_{k,i}|} \sum_{d \in D_{k,i}} \left(\frac{\|d_{gt} - \mu_\theta(d)\|}{\sigma_\theta^2(d)} + \log \sigma_\theta^2(d) \right) \quad (7)$$

where $\sigma_\theta^2(d)$ is ensured to be positive and $D_{k,i}$ is the set of all valid depth points by masking out the nadir area and missing data.

The neural network architecture is a fully convolutional network (FCN) based on our prior work [2] with some minor modifications to adapt the sparse depth and uncertainty estimation, shown in Fig. 4(a). For the normalization, we choose instance normalization (IN), and for the activation functions, we use rectified linear unit (ReLU). The downsampling layers consist of three convolutional modules in the form of convolution-IN-ReLU. These are followed by the seven residual blocks, as shown in Fig. 4(b). The residual blocks consist of several layers with no change in the image dimension, the output of which is summed with the input and fed to the next residual block. By feeding the input directly to the output, one gets a direct link across all the blocks that facilitate propagation of the gradient. The residual blocks are followed by two upsampling layers with two transposed convolution layers and the convolution operation in the end.

III. Experiments

The method is evaluated on two data sets from different areas. Data set 1 is divided into training, validation and test sets, while data set 2 is only used for testing. The details are given in Table I. Using *sidescan draping* described in Section II-E, we can associate the waterfall images to corresponding depth images and the sparse depth available from the altimeter reading along the trajectory (see Fig. 5). For each side of the waterfall image, we divide it into smaller windows with height $H = 256$ and width $W = 256$ downsampled from $\sim$6000 bins. The square images with size 256×256 make it easier to adapt the architecture of the mainstream neural networks for computer vision. To generate more training data, we augment the data by allowing the windows to overlap by 75% and flipping the windows in the along-track direction to simulate the sonar to move exactly the opposite direction.

The network is trained on the training set with 4352 windows from data set 1 with different hyper-parameters. The validation set is used to select the three best models, whose results will be used for ensemble in the testing phase later to make better estimation of the predictive uncertainty from the neural network. The six lines from validation set, data set 1 and the six lines from test set, data set 1 are evenly distributed across the whole area but orthogonal to each other. The purpose of this is to test the generalization of the network when the sidescan images are from 90° angles. The whole data set 2 from totally another place is also used as the test set to test the generalization of the network when coming to different environments.

To evaluate the methods, we compare the bathymetric map generated from the network and the one from the MBES pings. The bathymetric map is generated by solving the reverse problem of *sidescan draping* with the methodology described in Section II-E. Due to sidescan's wide swath coverage and high resolution, for most of the points on the seafloor, there are usually many estimates. During the bathymetry fusion, we first discard the extreme outliers with uncertainty larger than a certain threshold, and then use the confidence measurement $c_p^{k,i}$ as weights for the corresponding depth estimates, as (5) in Section II-D. The threshold is selected empirically by increasing the threshold to be just large enough for the final bathymetric map to have enough coverage.

To compare the result, we use the same resolution (0.5 m) as the bathymetric map from the MBES data, where in theory the resolution is not integral to the method, meaning one could choose much higher resolution to build a super-resolution bathymetric map based on sidescan data. The potential challenge is the lack of super-resolution bathymetry to evaluate and the GPU power to train the network.

IV. Results

A. Reconstruction Results

Fig. 6 shows the seafloor depth distribution of two data sets. We can notice that data set 1 covers a large range of 9–21 m, whereas data set 2 mainly concentrates on the range of 10–14 m.

1) Data set 1: Fig. 7 shows the reconstructed bathymetry with six sidescan survey lines on the test data from data set 1, with an MAE 0.059 m. Looking at Fig. 7(a), we observe that the reconstructed bathymetry reproduces the seafloor topography on a large scale. It also highlights one advantage of sidescan over multibeam wider swath coverage. Only six survey lines can cover around 60% of the surveyed area. The MBES from these six lines would only cover about 35% of the area, indicating that the proposed method could in theory significantly improve the survey efficiency. Fig. 7(c) shows the corresponding confidence map where clearly the bottom right of the map has low confidence, high uncertainty. In Fig. 7(d), we see that the areas with the highest errors have high predicted uncertainty. In the zoomed-in Fig. 8, we can observe that the low-confidence area contains a huge hill with many boulders where the network's prediction performs worst. The network manages to recover the contour of the hill and some boulders but the details are less accurate. One possible explanation is that this is an area with many topography variations, and a small error in depth prediction will cause a relatively large error in its position in the map based on the trigonometry calculation described in (4).

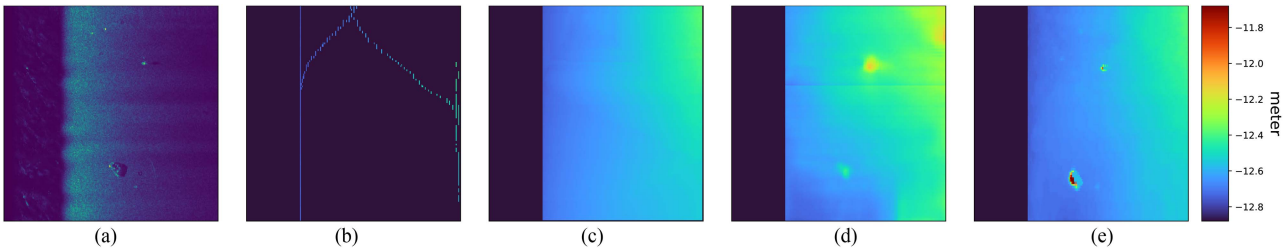


Fig. 10. Example of image pairs from a relatively flat area in data set 2, where the density of sparse depth is reduced. (a) Sidescan intensities input window. (b) Sparse depth input window. (c) Interpolated depth window. (d) Predicted depth output window. (e) Ground truth depth window. Here, we can note that at the bottom of (a) and (e), there is a rock whose shape cannot be reconstructed only through interpolation as in (c) but can be reconstructed with a NN as in (d). However, the height of reconstructed rock is about 20 cm, whereas from the "ground truth" from MBES data, it should be around 1 m high.

One interesting finding is that, in Fig. 9, the shape of some rocks appeared in SSS is correctly reconstructed by the network, as seen in Fig. 9(c). The network accurately infers the elevation rising in front of the shadows, which indicates the necessity and importance of sidescan data. However, not all rocks in Fig. 9(a) and (d) are shown as prominent in the prediction. This issue could possibly be addressed in the future work by adding constraints on the gradients of the sidescan intensities and increasing the sidescan's across-track resolution.

2) Data Set 2 and Generalization: Fig. 11 shows the comparison between the reconstructed bathymetry and the ground truth from data set 2 and the corresponding confidence map. From Fig. 11(a) and (b), we can again observe that the reconstructed bathymetry captures most of the seafloor topography. The bottom of Fig. 11(a) shows a distinguishing sidescan's characteristic that there is no measurement in the nadir area, hence no estimates about the terrain. Fig. 11(c) presents the confidence map of the prediction, where we can clearly see that the confidence is high along the sonar's trajectory (see Fig. 2) while relatively low near to the boundary of the surveyed area. The reasons that the periphery has high uncertainty are the uncertainty of the depth estimation is naturally high as one moves away from the sonar [see Fig. 1(c)] and there is no longer sparse depth available to reduce the drifting errors. Another observation is that in the middle of the surveyed area, there are two places with sudden low confidence, where they are two boulders. If we zoom in there, as seen in from Fig. 12(a), we can see that the two boulders are not as sharply shown as in Fig. 12(b).

Data set 2 covers a relatively flat area with a different depth distribution in the seafloor. The reconstructed bathymetry has a 0.043 m absolute error, indicating a good generalization ability on the unseen data of a different natural environment. This indicates that one could train a network using SSS and MBES. Thereafter, use it on many AUVs equipped only with the same SSS. However, since we do not apply any geometric correction to the SSS images, if we want to use the trained CNN on the SSS data from another sensor setup, some techniques, such as domain adaption, have to be applied to address this.

B. Effects of Sparse Depth

Another interesting observation is that the quality and quantity of the sparse depth are critical for our proposed method to reconstruct a good bathymetry. The provided sparse depth acts as a boundary constraint in the optimization, so if the quality of the input sparse depth is low, i.e., the measurements being corrupted, the depth estimation will have large errors. As seen in Fig. 13, when the provided sparse depth is inaccurate, the shape of the predicted depth contour is more or less right but with an offset due to the errors in sparse depth. Several reasons could cause inaccurate sparse depth measurement, e.g., errors from the altimeter sensor, affecting the quality of reconstructed bathymetry. Not only the quality of the sparse depth but also the quantity affect the prediction accuracy. In Tables II and III, we compare the MAE and the standard deviation of the errors on the bathymetric map generated by the neural network with the baseline bathymetric model obtained through interpolation with different numbers of survey lines to provide sparse depth as constraints. As we can see in the tables, when all of the survey lines are utilized, the prediction accuracy is the highest, and as the quantity of provided sparse depth decreases, the prediction accuracy decreases.

In practice, for example, data set 2, one could certainly use less than 36 survey lines for the sidescan to cover the whole area. We mentioned in Tables II and III that the reconstructed error is still relatively low even with 30% sparse depth provided, indicating 70% efficiency improvement. So one could carefully plan the sidescan survey to cover a much larger area within one mission to construct a high-quality bathymetric map with the proposed method. Note that when 100% sparse depth is used, the baseline method, linear interpolation through the altimeter readings, can achieve slightly better results (MAE and STD) than the proposed method. However, when fewer lines of the altimeter readings are used, the proposed method outperforms the baseline significantly on data set 1, where the terrain has more topography variations. Even when the terrain is rather simple such as data set 2, the proposed method still shows its advantages over linear interpolation of altimeter readings when there are interesting features between the altimeter readings, as shown in Fig. 10.

C. Effects of Uncertainty Estimation

Uncertainty estimation is useful when fusing the estimated bathymetry from each sidescan line. We use data set 1 to illustrate that uncertainty could improve the quality of the reconstructed bathymetry. Assuming we lack uncertainty estimation and fuse the bathymetry estimation simply averaging

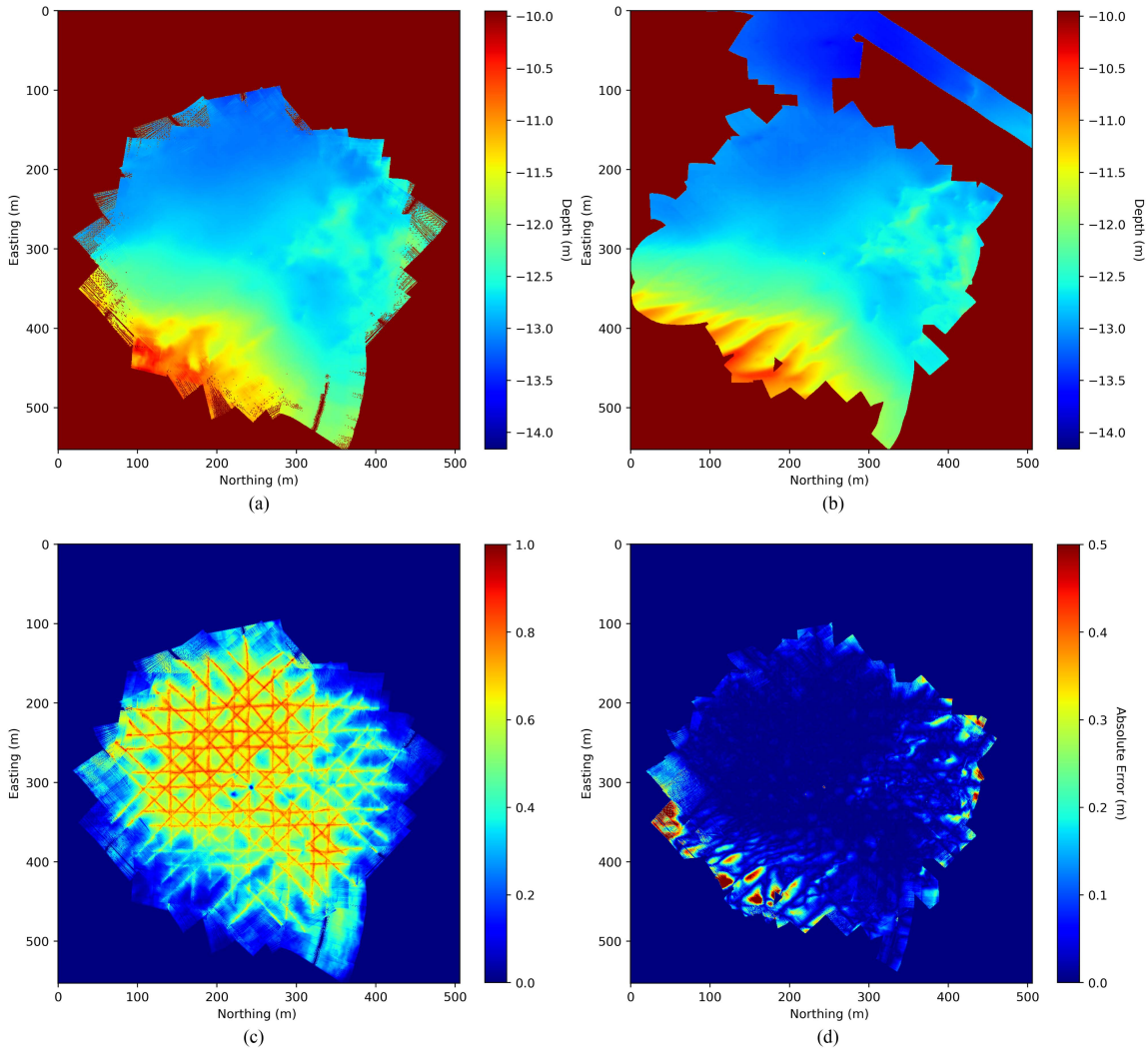


Fig. 11. Bathymetry on data set 2, produced by sidescan and multibeam. (a) Bathymetry from 36 sidescan survey lines, covering about 0.16 km^2 area. (b) Ground truth bathymetry produced with multibeam data. (c) Normalized confidence map for the bathymetry produced from sidescan. (d) Absolute error map between (a) and (b).

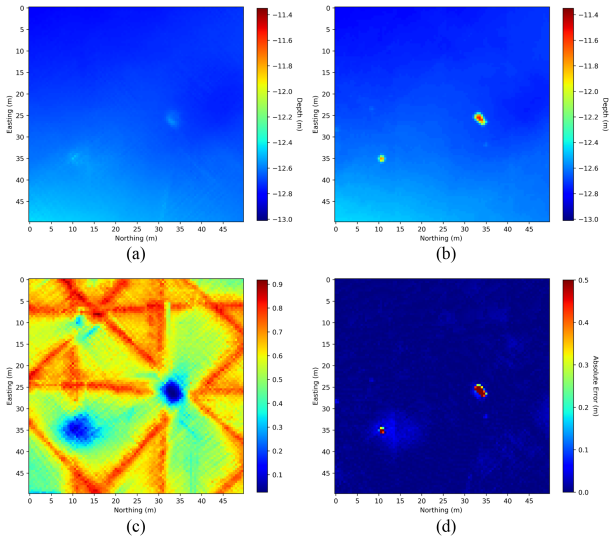


Fig. 12. Zoomed-in section of Fig. 11 where the area of interest contains two boulders. (a) Bathymetry from SSS. (b) Bathymetry from MBES. (c) Normalized confidence map. (d) Error map.

each estimate, we can generate a bathymetry with absolute error 0.071 m, while using uncertainty as described in (5), we can achieve 0.059 m error. Besides that, the fusion without using uncertainty performs much worse in the areas that are supposed to be highly uncertain. Fig. 14 shows the same place as Fig. 8 but without using the uncertainty estimation. We can clearly observe from Fig. 14(a) that the bathymetric map is much worse when uncertainty is not used.

D. Bathymetry With Higher Resolution

When reconstructing the bathymetry from SSS, we use the grid size 0.5 m because our bathymetric map from MBES (data set 1) has 0.5 m resolution, which is used to generate the training data. Nevertheless, for data set 2, we do have a bathymetric map from MBES with 0.25 m resolution, which can be used as ground truth to compare the bathymetry from sidescan with a grid size of 0.25 m, as shown in Fig. 15. To generate such map, we use the same outputs from the neural network but only use a smaller grid size when constructing the bathymetry from predicted depth windows. We can clearly observe the effects of ship turning on

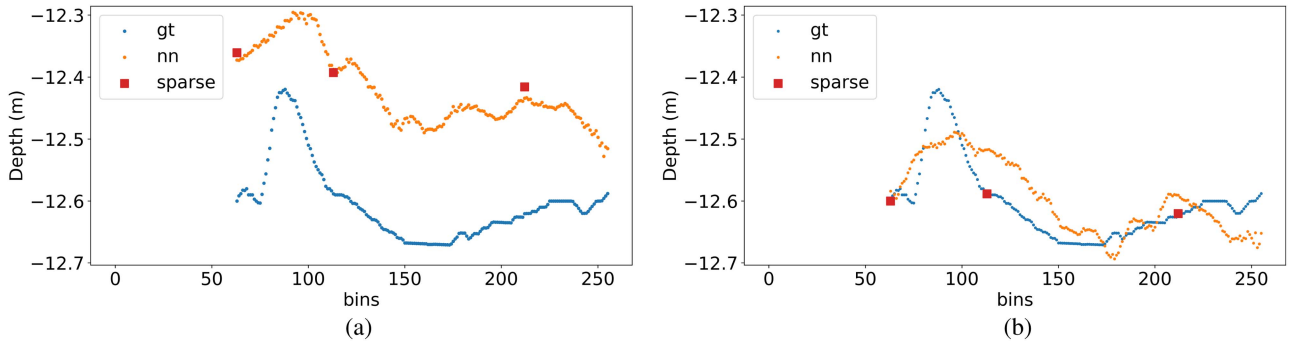


Fig. 13. Effects of sparse depth quality: Here, we plot one row of the network's input and output; one ping of ground truth depth in blue, one ping of the depth prediction in orange and the provided sparse depth as input in red. (a) Sparse depth provided is corrupted, thus not aligned with the ground truth depth, leading the prediction of the network off by a lot. (b) Sparse depth provided is accurate, leading the prediction aligned much better with the ground truth.

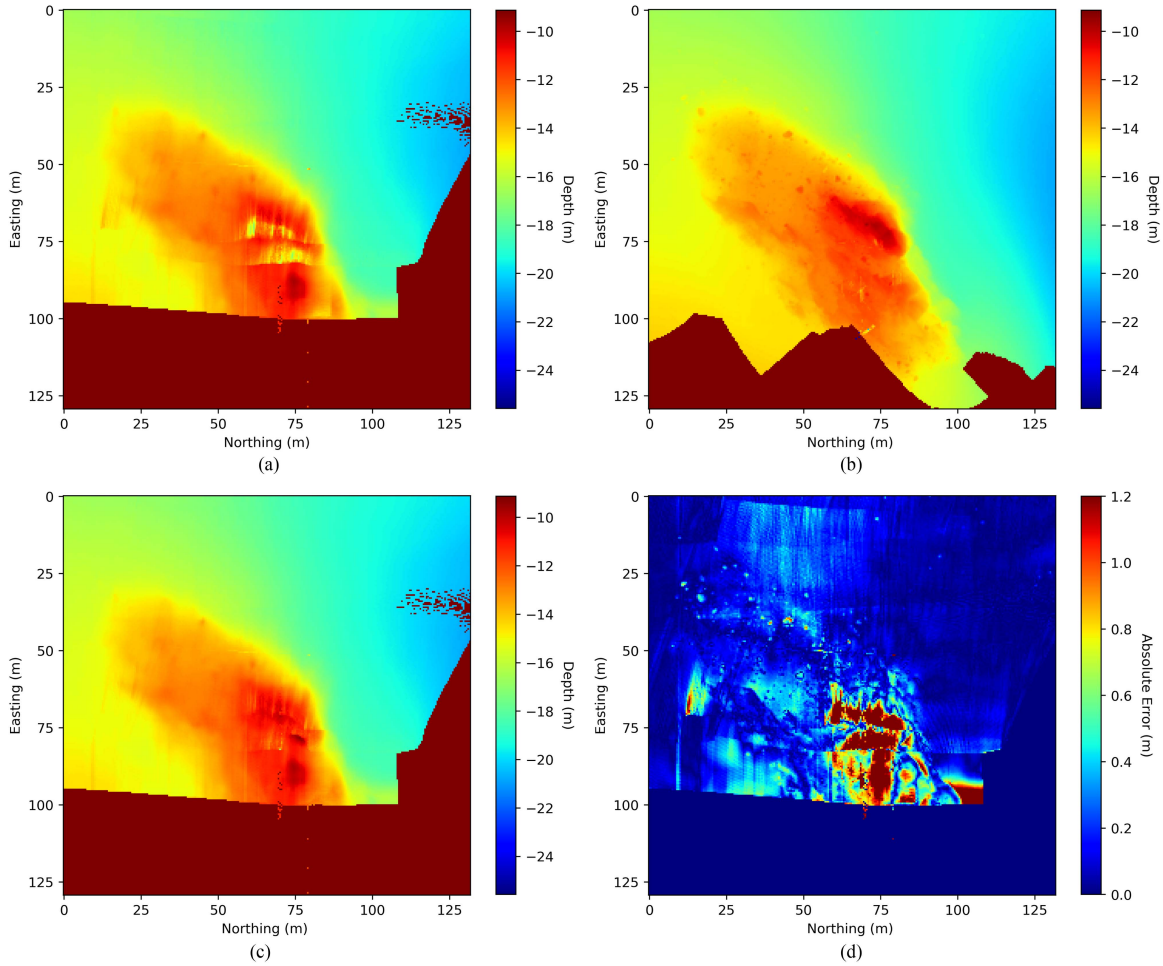


Fig. 14. Zoomed-in section of bathymetry for data set 1, without using the uncertainty estimation in (a). The same Fig. 8(a), bathymetry reconstructed with uncertainty in (c). (a) Bathymetry from SSS without uncertainty. (b) Bathymetry from MBES. (c) Bathymetry from SSS with uncertainty. (d) Error map between (a) and (b).

sidescan swaths from the bottom right of Fig. 15, where the portions in the inside corners overlap while the portions in the outside corners have incomplete coverage for this fine scale grid. We can also note that the coverage is low in Fig. 15 at the perimeter partially due to sidescan's narrow horizontal beamwidth. There exist research studies [27] investigating how to improve the coverage without increasing the error for wide and narrow aperture sonars. The absolute error compared to the ground truth is 0.042 m, very close to the error with 0.5-m grid size in Table III.

TABLE II
EFFECTS OF SPARSE DEPTH QUANTITY—DATA SET 1

	Ours		Linear Interpolation	
	Testing MAE (m)	STD (±m)	Testing MAE (m)	STD (±m)
100% sparse depth	0.059	0.146	0.049	0.139
50% sparse depth	0.074	0.193	0.314	0.615
30% sparse depth	0.082	0.250	0.430	0.866
no sparse depth	0.734	1.837	-	-

TABLE III
EFFECTS OF SPARSE DEPTH QUANTITY—DATA SET 2

	Ours		Linear Interpolation	
	Testing MAE (m)	STD (±m)	Testing MAE (m)	STD (±m)
100% sparse depth	0.043	0.093	0.039	0.084
50% sparse depth	0.053	0.107	0.046	0.094
30% sparse depth	0.060	0.116	0.053	0.098
no sparse depth	0.200	0.370	-	-

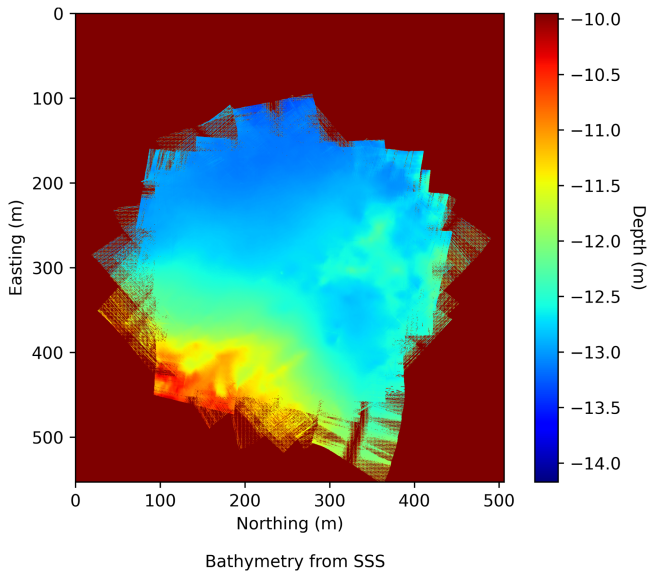


Fig. 15. Bathymetry reconstructed from sidescan with 0.25-m grid size for data set 2.

V. CONCLUSION

In this article, a novel approach to reconstruct high-resolution bathymetry from sidescan data using a neural network is presented. The neural network is trained in an end-to-end fashion to predict the depth and uncertainty from sidescan intensities and sparse depth. The predicted depth and uncertainty, modeled as following a Laplacian distribution, are fused to construct the bathymetry. In the qualitative and quantitative analysis, we showed that the generated bathymetry has high quality and low errors below the decimeter level. We also showed the important role both the sparse depth and the confidence estimate plays on the accuracy of the fused map. In this article, we rely on accurate navigation positioning, the absence of which will limit the reconstruction results, depending on how well the navigation errors can be reduced.

The current network architecture generates independent depth windows from sidescan data between a fixed time period without incorporating the sequential nature of the sidescan pings. In future work, we would like to better address this by using a recurrent FCN model conditioned on the previous pings.

Another interesting direction is to use sidescan measurements with a higher across-track resolution to fully utilize the advantages of the sidescan and generate a bathymetry with a higher resolution than the one generated from the multibeam. One challenge here will be the lack of ground truth to analyze the performance quantitatively. Another challenge is that with a higher across-track resolution, the larger the width of images will be. One may use super-resolution CNN model [28] to address such challenge, or one may treat the sidescan ping by ping so that higher resolution would not be too computational heavy.

ACKNOWLEDGMENT

Our data set was acquired in collaboration with Marin Mätteknik Gothenburg.

REFERENCES

[1] J. Folkesson, H. Chang, and N. Bore, "Lambert's cosine law and sidescan sonar modeling," in *Proc. IEEE/OES Auton. Underwater Veh. Symp.*, 2020, pp. 1–6.

[2] Y. Xie, N. Bore, and J. Folkesson, "Inferring depth contours from sidescan sonar using convolutional neural nets," *IET Radar, Sonar Navigation*, vol. 14, no. 2, pp. 328–334, 2019.

[3] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *Proc. Int. Conf. 3D Vis.*, 2016, pp. 239–248.

[4] F. Ma and S. Karaman, "Sparse-to-Dense: Depth prediction from sparse depth samples and a single image," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 4796–4803, doi: 10.1109/ICRA.2018.8460184.

[5] P. Woock and C. Frey, "Deep-sea AUV navigation using side-scan sonar images and SLAM," in *Proc. IEEE OCEANS Conf.*, 2010, pp. 1–8, doi: 10.1109/OCEANSSYD.2010.5603528.

[6] R. Li and S. Pai, "Improvement of bathymetric data bases by shape from shading technique using side-scan sonar images," in *Proc. IEEE OCEANS Conf.*, Honolulu, HI, USA, 1991, vol. 1, pp. 320–324, doi: 10.1109/OCEANS.1991.613950.

[7] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah, "Shape-from-shading: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 8, pp. 690–706, Aug. 1999, doi: 10.1109/34.784284.

[8] E. Coiras, Y. Petillot, and D. M. Lane, "Multiresolution 3-D reconstruction from side-scan sonar images," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 382–390, Feb. 2007, doi: 10.1109/TIP.2006.888337.

[9] K. R. Jones and P. Traykovski, "A method to quantify bedform height and asymmetry from a low-mounted sidescan sonar," *J. Atmos. Ocean. Technol.*, vol. 35, no. 4, pp. 893–910, 2018.

[10] A. Burguera and G. Oliver, "High-resolution underwater mapping using side-scan sonar," *PLoS One*, vol. 11, no. 1, pp. 1–41, 2016, doi: 10.1371/journal.pone.0146396.

[11] A. E. Johnson and M. Hebert, "Seafloor map generation for autonomous underwater vehicle navigation," *Auton. Robots*, vol. 3, no. 2, pp. 145–168, 1996.

[12] N. Bore and J. Folkesson, "Neural shape-from-shading for survey-scale self-consistent bathymetry from sidescan," 2022, *arXiv:2206.09276*.

[13] J. Cuschieri and M. Hebert, "Three-dimensional map generation from side-scan sonar images," *J. Energy Resour. Technol.*, vol. 112, pp. 96–102, 1990.

[14] J. Zhao, X. Shang, and H. Zhang, "Reconstructing seabed topography from side-scan sonar images with self-constraint," *Remote Sens.*, vol. 10, no. 2, 2018, Art. no. 201, doi: 10.3390/rs10020201.

[15] I. Dzieciuch, D. Gebhardt, C. Barngrover, and K. Parikh, "Non-linear convolutional neural network for automatic detection of mine-like objects in sonar imagery," in *Proc. Int. Conf. Appl. Nonlinear Dyn.*, Springer, 2017, pp. 309–314.

[16] G. Huo, Z. Wu, and J. Li, "Underwater object classification in sidescan sonar images using deep transfer learning and semisynthetic training data," *IEEE Access*, vol. 8, pp. 47407–47418, 2020, doi: 10.1109/ACCESS.2020.2978880.

[17] D. Einsidler, M. Dhanak, and P. Beaujean, "A deep learning approach to target recognition in side-scan sonar imagery," in *Proc. IEEE/MTS OCEANS Conf.*, Charleston, SC, USA, 2018, pp. 1–4, doi: 10.1109/OCEANS.2018.8604879.

[18] M. Rahnemoonfar and D. Dobbs, "Semantic segmentation of underwater sonar imagery with deep learning," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 9455–9458, doi: 10.1109/IGARSS.2019.8898742.

[19] M. Wu et al., "ECNet: Efficient convolutional networks for side scan sonar image segmentation," *Sensors*, vol. 19, no. 9, 2019, Art. no. 2009, doi: 10.3390/s19092009.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.

[21] F. Liu, C. Shen, and G. Lin, "Deep convolutional neural fields for depth estimation from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5162–5170.

[22] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, Curran Associates, 2017, pp. 6405–6416.

[23] S. Walz, T. Gruber, W. Ritter, and K. Dietmayer, "Uncertainty depth estimation with gated images for 3D reconstruction," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, 2020, pp. 1–8, doi: 10.1109/ITSC45102.2020.9294571.

[24] T. Gneiting and A. E. Raftery, "Strictly proper scoring rules, prediction, and estimation," *J. Amer. Statist. Assoc.*, vol. 102, no. 477, pp. 359–378, 2007, doi: 10.1198/016214506000001437.

[25] P. Blondel, *The Handbook of Sidescan Sonar*, 1st ed. Berlin, Germany: Springer, 2010, doi: 10.1007/978-3-540-49886-5.

[26] N. Bore and J. Folkesson, "Modeling and simulation of sidescan using conditional generative adversarial network," *IEEE J. Ocean. Eng.*, vol. 46, no. 1, pp. 195–205, Jan. 2021, doi: 10.1109/JOE.2020.2980456.

[27] T. Guerneve, K. Subr, and Y. Petillot, "Three-dimensional reconstruction of underwater objects using wide-aperture imaging sonar," *J. Field Robot.*, vol. 35, no. 6, pp. 890–905, 2018.

[28] J. Yamanaka, S. Kuwashima, and T. Kurita, "Fast and accurate image super resolution by deep CNN with skip connection and network in network," in *Proc. Int. Conf. Neural Inf. Process.*, Springer, 2017, pp. 217–225.

Yiping Xie received the B.S. degree in electrical engineering in 2017 from Beihang University, Beijing, China, and the M.Sc. degree in computer science in 2019 from the Royal Institute of Technology (KTH), Stockholm, Sweden, where he is currently working toward the Ph.D. degree with the Wallenberg AI, Autonomous Systems and Software Program at the Robotics Perception and Learning Lab.

His research interests include perception for underwater robots, bathymetric mapping, and localization with sidescan sonar.

Nils Bore received the M.Sc. degree in mathematical engineering from the Faculty of Engineering, Lund University, Lund, Sweden, in 2012, and the Ph.D. degree in computer vision and robotics from the Robotics Perception and Learning Lab, Royal Institute of Technology (KTH), Stockholm, Sweden, in 2018.

He is currently a Researcher with the Swedish Maritime Robotics project at KTH. His research interests include robotic sensing and mapping, with a focus on probabilistic reasoning and inference. Most of his recent work has been on applications of specialized neural networks to underwater sonar data. In addition, he is interested in system integration for robust and long-term robotic deployments.

John Folkesson (Senior Member, IEEE) received the B.A. degree in physics from Queens College, City University of New York, New York, NY, USA, in 1983, and the M.Sc. degree in computer science and the Ph.D. degree in robotics from the Royal Institute of Technology (KTH), Stockholm, Sweden, in 2001 and 2006, respectively.

He is currently an Associate Professor of robotics with the Robotics, Perception and Learning Lab, Center for Autonomous Systems, KTH. His research interests include navigation, mapping, perception, and situation awareness for autonomous robots.