Hindawi
Journal of Robotics
Volume 2021, Article ID 6644986, 12 pages
https://doi.org/10.1155/2021/6644986

Research Article

Underwater Depth Estimation for Spherical Images

Jiadi Cui, Lei Jin, Haofei Kuang, Qingwen Xu, and Sören Schwertfeger

Mobile Autonomous Robotic Systems Lab, School of Information Science and Technology, ShanghaiTech University, Shanghai, China

Correspondence should be addressed to Jiadi Cui; cuijd@shanghaitech.edu.cn

Received 15 December 2020; Accepted 29 May 2021; Published 18 June 2021

Academic Editor: L. Fortuna

This paper proposes a method for monocular underwater depth estimation, which is an open problem in robotics and computer vision. To this end, we leverage publicly available in-air RGB-D image pairs for underwater depth estimation in the spherical domain with an unsupervised approach. For this, the in-air images are style-transferred to the underwater style as the first step. Given those synthetic underwater images and their ground truth depth, we then train a network to estimate the depth. This way, our learning model is designed to obtain the depth up to scale, without the need of corresponding ground truth underwater depth data, which is typically not available. We test our approach on style-transferred in-air images as well as on our own real underwater dataset, for which we computed sparse ground truth depths data via stereopsis. This dataset is provided for download. Experiments with this data against a state-of-the-art in-air network as well as different artificial inputs show that the style transfer as well as the depth estimation exhibit promising performance.

1. Introduction

Underwater depth estimation is an open problem for marine robotics [1, 2], which is usually used for 3D reconstruction, navigation, and intermediate steps for underwater color correlation [3, 4]. Due to the properties of underwater environments, underwater perception is quite different from in-air perception. Images captured underwater usually look bluish because longer wavelengths of the visible sunlight are absorbed earlier than shorter wavelengths. Underwater images may also be more greenish, because of algae in the water. Besides, the underwater images are more blurred than those in-air captured by the same camera, due to turbidity. These reasons increase the difficulty of depth estimation from images. Thus, many researchers put effort on underwater image processing. For example, using dark channel priors is proposed to restore underwater images in [5, 6], inspired by [7] on removing haze in air. The study in [8] implemented underwater image stitching based on spectral methods, which are more robust to turbidity than feature-based methods. Besides image enhancement, some work focused on depth estimation. The study in [9] exploited the relationship between depth and blurriness of underwater images to estimate depth. In addition, deep learning was also applied to estimate the depth of underwater images, for example, the study in [4] used a convolution neural network (CNN) to generate relative depth, which was then one of the inputs for a color correction network. Learning-based methods are very popular these days, and there are many applications about depth estimation, for example also in some microsystems [10, 11].

Apart from normal pin-hole cameras, omnidirectional cameras are becoming popular, due to their large field of view (FOV). They have been widely used on ground robots [12–16]. Some research groups also studied omnidirectional cameras for underwater use since they provide more information than perspective ones on object detection, localization, and mapping. The study in [17] designed omnidirectional video equipment and put it on dolphins to capture data. The study in [18] improved on-land omnidirectional cameras for underwater use and proposed the method for camera calibration.

In addition, the sometimes long visible distances in water increase the region of undefined depth, especially compared to indoor scenes, which makes the depth estimation more difficult. Although there are several papers on active

methods for underwater 3D imaging [19], capturing omnidirectional underwater depth images remains a big challenge, which makes ground truth depth unavailable. This paper proposes to leverage publicly available in-air spherical images for depth estimation in the underwater domain. Specifically, our approach follows a two-stage pipeline. (i) Given in-air RGB-D spherical pairs from the Stanford 2D-3D-S dataset [20], we train a style-transfer network [21] to convert in-air images to the underwater domain. (ii) Given the generated underwater images and their depth maps, we train a depth estimation network which is specially designed for spherical images. During testing, we can generate depth directly from the input image. Our approach is unsupervised in that only underwater images (i.e., no ground truth underwater depth) are required for the whole training process.

Following our preliminary work [22], the main contributions of our paper are as follows:

(i) To the best of our knowledge, we are the first group to employ CycleGAN to spherical underwater images

(ii) This is also the first method to employ deep learning to estimate depth in spherical underwater images

(iii) We provide a spherical underwater dataset, which consists of 3,000 high-quality images from the Great Barrier Reef

(iv) We provide a benchmark of the proposed network with respect to handcrafted images

2. Related Work

2.1. Unsupervised Depth Learning. Learning-based methods for depth estimation are popular. However, for adversarial environments, such as underwater or forest scenarios, annotated data is difficult to obtain. Therefore, supervised learning has difficulties in achieving a good performance with the absence of a large amount of labeled data. Unsupervised learning and self-supervised learning are two methods for utilizing unlabeled data in the learning process. One reason for using unlabeled data is that producing a dataset with clear labels is expensive, but unlabeled data is being generated all time. The motivation is to make use of the much larger amount of unlabeled data. The main idea of self-supervised learning is to generate the labels from unlabeled data, according to the structure or characteristics of the data itself, and train with this unsupervised data through a supervised manner. Self-supervised learning is wildly used in representation learning to make a model learn the latent features of the data. These methods are wildly used in computer vision [23–27], video processing [28, 29], and robot control [30–32].

There is much previous work related to the self-supervised method for depth estimation. In 2017, [33] proposed the monodepth framework to exploit epipolar geometry constraints and proposed a novel training loss to train their model along a self-supervised way. After that, there are some methods related to using geometry constraints to achieve self-supervision. The study in [34] utilized epipolar geometry constraints to estimate both depth and surface normals. The study in [35] investigated the multimodality depth completion task with a self-supervised method by constructing a loss function with photometric constraints, and their method achieved the state of the art (SOTA) on the KITTI depth completion benchmark. The study in [36] exploited the bilateral cyclic relationship between stereo disparities and proposed an adaptive regularization scheme to handle covisible and occluded problems in a stereo pair.

Different from geometric constraints-based methods, there are some approaches that try to exploit the constraint between different modalities, called the wrapped-based method. The study in [37] proposed a wrapped-based method to estimate both depth and pose. They designed a loss based on wrapping nearby views to the target using the computed depth and pose. The study in [38] proposed monodepth2 to combine depth and camera pose with geometry constraints. To improve the robustness of the model, they also proposed the minimum reprojection loss and utilized a multiscale sampling method in their framework. Currently, monodepth2 achieves SOTA results on the KITTI benchmark. Because these methods can predict both depth and camera pose, they are wildly used in robotics and self-driving cars as a visual odometry (VO) system. Zhan et al. investigated the end-to-end unsupervised depth-VO [39] and also integrated the depth with Perspective-n-Point (PnP) method to achieve high robustness [40].

This idea was also extended to combine more computer vision tasks. The study in [41] exploited the content consistency between the depth and semantic information. The study in [42] proposed the GeoNet to utilize the geometric relationships between depth, optical flow, and camera pose and use an unsupervised learning framework to predict them. The study in [43] proposed a competitive collaboration framework to predict depth, pose, optical flow, and motion segmentation parallel with an unsupervised method.

Currently, unsupervised depth estimation is successful in an indoor or urban scenario. But there are still few applications in adversarial scenarios. The study in [44] proposed a generative model and exploited cycle-consistent constraints to train the model in an unsupervised fashion. Their method achieves the SOTA on their dataset, but it is also hard to implement in real underwater applications and the amount of available data is also not enough for training.

2.2. Underwater Depth Estimation and Color Correction. In contrast to on-land scenarios, underwater depth estimation is more challenging due to scattering and absorption effects [9, 45], as mentioned above. For that, several methods jointly optimize depth estimation and color correction. In other words, accurate depth helps restore image colors and depth can also be estimated from the information of color distortion. For example, the authors of [9, 46] presented an image formulation model to estimate depth from image blurriness. In [5], a dark channel prior is used for underwater depth estimation and image restoration to dismiss the attenuation, backscattering effects. The study in [47]

presented adaptive image dehazing based on the depth information.

As introduced in Section 2.1 (Unsupervised Depth Learning), there are many successful learning methods to estimate depth for in-air images. Thus, a naive way to estimate underwater depth is to restore underwater images to in-air style so that this depth learning strategy can be applied. In [48], such a strategy proves to be efficient in underwater depth estimation. Both deep learning and mathematical methods are very popular for image restoration. In [49], they use the Jaffe-McGlamery model [50, 51], a mathematical method, to handle the problems, which decreases the absorption and scattering effects based on irradiance and depth. In [52], a learning-based method was proposed to solve depth estimation and color correction in spherical domains at the same time by solving left-right consistency under a multicamera setting. However, deep learning usually requires a large amount of data, which is not available for the underwater field. To overcome this problem, the study in [4] proposed a generative adversarial network to generate synthetic underwater images from in-air datasets.

Our work is inspired by WaterGAN [4], but also different from it. WaterGAN requires depth as input to simulate the attenuation and scattering effect, while our underwater GAN only needs underwater and in-air images as input. Our preliminary work is reported in [22], where we proposed the two-stage pipeline to solve underwater omnidirectional depth estimation. In the first perspective image pipeline, the WaterGAN [4] was used to transfer RGB-D images to underwater RGB-D images. Then, a fully convolutional residual network (FCRN) [53] depth estimation network was trained with the underwater image as input. In the second omnidirectional stage, we synthesized images from in-air equirectangular images to underwater equirectangular images by decreasing the values in the red channel (due to its short wavelength nature in the underwater environment) and blurring the image based on its distance to the camera origin. Finally, inspired by [54], a distortion-aware convolution module replaced the normal convolution in the FCRN based on the spherical longitude-latitude mapping. In this work, we replace the simple operations in the red channel with a learning method to generate synthetic underwater omnidirectional images. In addition, we improve the method to estimate underwater depth. Finally, we are more thoroughly evaluating the results of our algorithm, by estimating ground truth depths for distinctive feature points. In [54], the FCRN [53] was identified as the state-of-the-art (SOTA) network for omnidirectional CNNs, and we thus adopt it and compare to it in this paper.

We want to emphasize that, in general, depth estimation from a single RGB image is a very challenging problem. As our experiments later will show, our approach does not give very accurate estimates, neither do the other depth estimation approaches mentioned in this section. Also, as with any monocular vision problem, our results are up to an unknown scale factor. Nevertheless, we believe this work to be worthwhile because it paves a path towards potentially more successful approaches (see the future work) and, even not being very accurate, has potential use cases, for example in navigation or color correction.

3. Methodology

Figure 1 demonstrates our two-stage pipeline. (i) Given in-air RGB-D spherical pairs from the Stanford2D-3D-S dataset [20], we train CycleGAN [21] to convert in-air images to the underwater domain. (ii) Given the generated underwater images and their depth maps, we train a depth estimation network to learn depth. In the following, we introduce the two parts separately.

3.1. Style Transfer. Generative adversarial nets (GANs) are designed for data augmentation and are now widely used in style-transfer tasks. GANs are two-player mini-max games between a generative model G and a discriminative model D [55]. The value function about this adversarial process is

$$\min_G \max_D V(D,G) = \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))], \tag{1}$$

where p_{data} denotes the features in the data and $p_{\mathbf{z}}$ holds random values at first. This value function is also the loss function for the deep neural network.

The underwater style-transfer algorithm CycleGAN [21] consists of two networks, a network G for forward mapping and a network F for inverse mapping. Given input images, network G converts to the target domain and network F converts back to the original domain. A cycle consistency is enforced as $F(G(X)) \approx X$ and vice versa, to ensure the mappings will be constrained well. Thus, the loss function of the forward mapping function G: $X \longrightarrow Y$ is

$$\mathscr{L}_{\text{GAN}}(G, D_Y, X, Y) = \mathbb{E}_{y\sim p_{\text{data}}(y)}[\log D_Y(y)] + \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log(1 - D_Y(G(x))]. \tag{2}$$

We use X as input to domain D_X and Y as input to domain D_Y. Examples of our input images from the two domains are demonstrated in Figures 2 and 3. Since both our input and output operate under the spherical domain, we directly adopt the network with no modification to the convolution operators.

Moreover, CycleGAN applies a new idea about cycle consistency, which is $y \longrightarrow F(y) \longrightarrow G(F(y)) \approx y$. And the loss function on this step is

$$\mathscr{L}_{\text{cyc}}(G,F) = \mathbb{E}_{x\sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] + \mathbb{E}_{y\sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1]. \tag{3}$$

Finally, the full objective for CycleGAN is

$$\begin{aligned}\mathscr{L}(G,F,D_X,D_Y) &= \mathscr{L}_{\text{GAN}}(G,D_Y,X,Y) + \mathscr{L}_{\text{GAN}}(F,D_X,Y,X) + \lambda\mathscr{L}_{\text{cyc}}(G,F),\\ G^*,F^* &= \arg\min_{G,F}\max_{D_x,D_Y}\mathscr{L}(G,F,D_X,D_Y).\end{aligned} \tag{4}$$

Because the method is pixel-to-pixel, the dataset is preprocessed by resizing the images into a reasonable size.

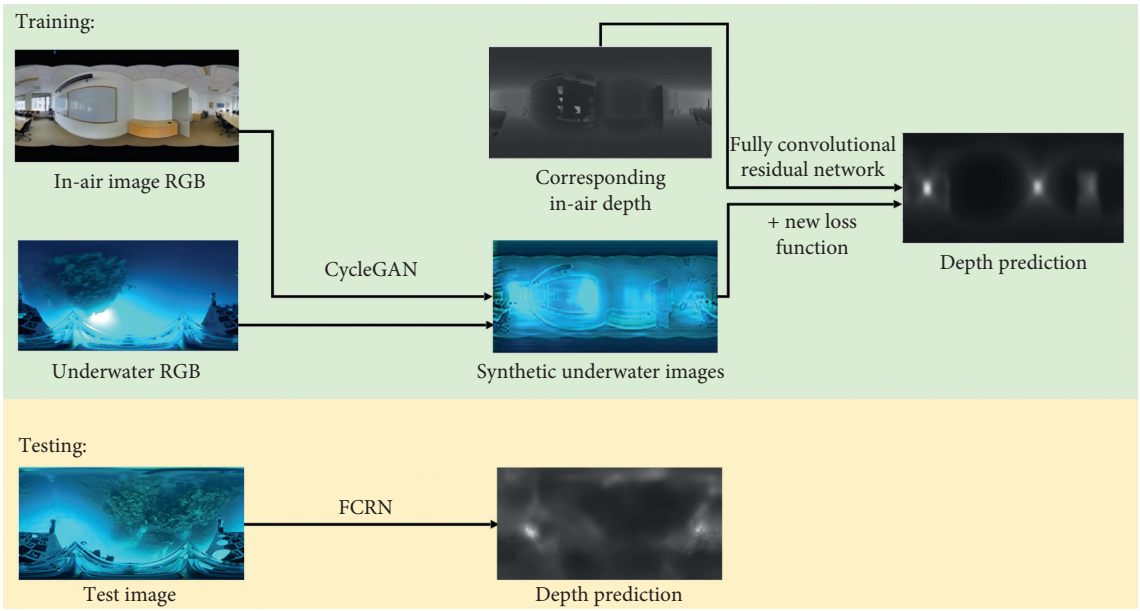

Figure 1: Full pipeline of our approach. We propose to leverage publicly available RGB-D datasets for style transfer and depth estimation in an unsupervised approach.

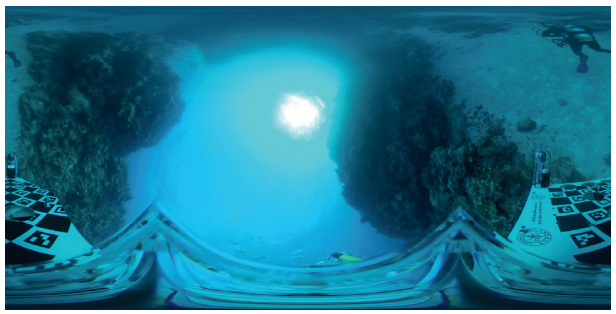

Figure 2: A typical underwater omnidirectional image.

Compared with WaterGAN, the CycleGAN only needs underwater and in-air images as input, whereas WaterGAN requires depth as input to simulate the attenuation and scattering effects.

3.2. Depth Estimation. With the recent success of convolutional neural networks, different CNN-based approaches are proposed to solve the supervised depth estimation task [53, 56]. However, most of the above approaches require large amounts of accurate image and ground truth depth pairs, currently unavailable in the spherical underwater domain. Instead, we propose to leverage an available in-air spherical dataset, the Stanford 2D-3D-S benchmark [20], and convert it to underwater style with StyleGAN. Specifically, given X_i, D_i pairs from the raw Stanford 2D-3D-S benchmark, we first convert X_i to the underwater domain X_i^w:

$$X_i^w = \text{CycleGAN}(X_i), \tag{5}$$

where X_i denotes the original in-air image from the dataset, D_i its corresponding depth, and X_i^w is the converted underwater image. We can then train our network with the converted X_i^w and D_i pairs.

Following the recent success of depth estimation in the spherical domain [57], we adopt FCRN, one of the state-of-the-art single models on NYUv2 [53]. The network consists of a feature extraction model and then several upconvolutions layers to increase the resolution. Here, an UNet [58] is used as the backbone in all our experiments. Finally, the $L1$ difference will be calculated between the output depth and ground truth depth maps:

$$L_{\text{depth}} = \sum_{d \in x,y} \left\| D_{\text{pred}} - D_{\text{gt}} \right\|_1, \tag{6}$$

where D_{pred} denotes the prediction of the network, D_{gt} denotes the ground truth depth map, and x, y enumerate all the pixels in the input image.

Smoothness regularization has been used frequently for depth estimation in planar images in previous research [33, 38] to encourage the estimated depths to be locally similar. For depth estimation in perspective images, the term is defined as follows:

$$L_{\text{sm}} = \sum_{p_t} \sum_{d \in x,y} \left\| \nabla_d D_t(p_t) \right\|_1, \tag{7}$$

where L_{sm} is a smoothness term that penalizes the $L1$ norm of first-order depth gradients along both the x and y directions in 2D space.

The equirectangular projection of a 360° image, however, is with distortion, and directly leveraging depth smoothness terms means we must impose larger weights for the point pairs with larger latitudes. Simply combining the above loss designed for perspective images into the training process might lead to suboptimal results. The reason is that the equirectangular projection of spherical images oversamples

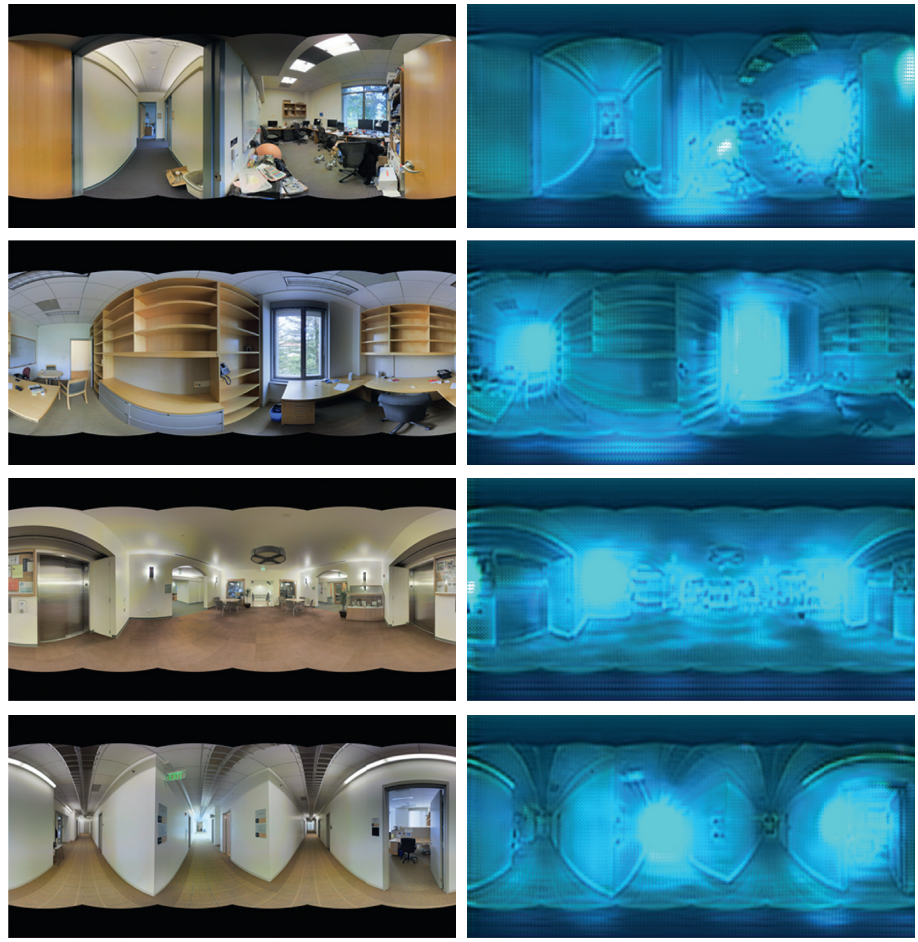

Figure 3: Generated images with our CycleGAN. (a) On the left are examples from Domain I (in-air). (b) On the right are our generated images. We are able to produce the lightening color effects from the original underwater dataset.

the image in the polar regions. Taking inspiration from the recent work of learning in the spherical domain [59], we propose that the weight of the distance of two points is based on their spherical distance, after which we arrive at the following spherical depth smoothness regularizer:

$$L_{\text{sm}}^{\text{sph}} = \sum_{p_t} \sum_{\theta=0,\phi=0}^{\Theta,\Phi} \omega_{\theta,\phi} \left\| \nabla_d D_t\left(p_t\right) \right\|_1, \tag{8}$$

where $\omega_{\theta,\phi}$ are the weights for each point and $\omega_{i,j} \propto \Omega(\theta,\phi)$. $\Omega(\theta,\phi)$ is the solid angle corresponding to the sampled area on the depth map located at (θ,ϕ). $L_{\text{sm}}^{\text{sph}}$ is a spatial smoothness term that penalizes the $L1$ norm of second-order depth gradients along both the θ and ϕ directions in 2D space.

Our final loss is a weighted combination of the above factors with λ_1 as the weighting factor:

$$L = L_{\text{depth}} + \lambda_1 L_{\text{sm}}^{\text{sph}}. \tag{9}$$

4. Experimental Details

We evaluate our approach with two experiments. Firstly, we use the synthetic underwater Stanford 2D-3D-S dataset with exact ground truth to quantitatively evaluate the algorithm. Here, we also compare to the SOTA algorithm for in-air spherical images: FCRN [53], in two setups. We test FCRN with the synthetic (GAN) images, as well as with the original RGB images as input. All algorithms are trained using the synthetic underwater images. The second experiment uses real omnidirectional underwater images and sparse ground truth points estimated via bundle adjustment to test the algorithm with in situ data.

In the following, we first introduce the datasets, hyperparameters, and evaluation metrics used in the experiments.

4.1. Datasets. Stanford 2D-3D-S [20] is one of the standard benchmarks for in-air datasets. The dataset provides omnidirectional RGB images and corresponding depth information, which is necessary data for depth estimation training. Furthermore, it also provides semantics in 2D and 3D, 3D mesh, and surface normals.

In addition, we use a dataset that we collected by scuba diving in the Great Barrier Reef. We use this for training our CycleGAN with original, spherical underwater images as well as for testing our approach. This omnidirectional dataset for style transfer and testing was collected with an

Insta360 ONE *X* (https://www.insta360.com/product/insta360-onex) camera at depths between 1 m and 25 m.

To evaluate the final results from our two-stage pipeline, the ground truth depth of the underwater scenario is generated based on epipolar geometry. The generation steps are as follows: firstly, a pair of stereo images with a known baseline are used to estimate sparse map points by feature matching, five-point algorithm [60], and triangulation [61].

Then, two pairs of stereo images, taken at different times, with big enough spatial disparity, including the one for map points, are used to fine-tune the position of the map points with bundle adjustment. Finally, the depth of these map points is normalized to 0 to 255 and used as up-to-scale ground truth.

Figure 4 shows an example of points (green dots) that are used as ground truth. It can be seen that most of these points are on the reef instead of water because the open water and the surface do not have feature points. Though only sparse points are generated, we believe that they are sufficient for the evaluation of our depth results. On the underwater dataset used for evaluation, we generate about 100 points for each image.

4.2. Hyperparameters. The hyperparameters for the style transfer include the resolution of input images, which is set to 512×256 pixel. We then train the CycleGAN [21] with these hyperparameters: learning rate ($2e$-4) and number of epochs (8).

We implement the FCRN for depth estimation with the PyTorch framework and train our network with the following hyperparameters settings during pretraining: mini-batch size (8), learning rate ($1e$-2), momentum (0.9), weight decay (0.0005), and number of epochs (50). We gradually reduce the learning rate by 0.1 every 10 epochs. Finally, we tune the whole network with learning rate ($1e$-4) for another 20 epochs. λ_1 is set to $1e$-4 in all our experiments.

4.3. Metrics. For our depth estimation network, we adopt FCRN [53] and compare the model with the initial loss function and our new loss function. Apart from these two networks, we also use FCRN based on the original in-air images, which are not processed by CycleGAN. For evaluation, we use the following common metrics for the comparisons on the datasets mentioned above: root mean square error (RMS) $\sqrt{(1/T)\sum_p (g_p - z_p)^2}$, mean relative error (Rel) $(1/T)\sum_p (\|g_p - z_p\|/g_p)$, mean log 10 error (log 10) $(1/T)\sum_p \|\log_{10} g_p - \log_{10} z_p\|$, and pixel accuracy as the percentage of pixels with $\max((z_i/z_i^{\text{gt}}), (z_i^{\text{gt}}/z_i)) < \delta$ for $\delta \in [1.25, 1.25^2, 1.25^3]$. T denotes the numbers of pixels and g_p and z_p represent the ground truths and the depth map predictions, respectively.

4.4. Metric for Real Experiment. To evaluate the final results of our two-stage approach, we rely on the sparse ground truth points captured with the approach described in Section 4.1. (Datasets). For all nonzero points, whose positions will be denoted by (i, j), we find the corresponding depth in the ground truth and estimated depth. The result of our estimation is up to an unknown scale factor. We thus minimize the error by calculating the best fitting scale factor for the ground truth. To do so, we calculate the scale parameter between each pair of ground truth and result and then get the median factor. To be more specific, in one pair of ground truth and result, there is the ratio of the ground truth value $P_{\text{gt}}(i, j)$ to the result value $P(i, j)$ for each point pairs. Then, using these ratios for one image, we can calculate their median s to simulate the optimization procedure, like the least-square method, and set the median s as the scale parameter between the ground truth and result. Finally, we rescale the result and compute the error E about each point. The error E about each image is calculated by

$$E = Q_{1/2}\left(\frac{\left|P_{gt}(i, j) - s \cdot P(i, j)\right|}{P_{gt}(i, j)}\right), \quad \text{if } P_{\text{gt}}(i, j) \neq 0. \tag{10}$$

Here, the operation $Q_{1/2}$ is to calculate the median of all cases for the ground truth points and the result points.

Figure 4: An example of ground truth points. The picture is captured by Insta360 ONE X camera at real ocean scenarios. Green points represent the interest points, whose depths are calculated by stereopsis.

5. Results

In this section, we will demonstrate the results on the converted Stanford 2D-3D-S dataset and real underwater images collected in the Great Barrier Reef.

5.1. Evaluation of Synthetic Images. Since there are few underwater datasets with ground truth depth, we synthesize underwater style images from the Stanford 2D-3D-S dataset. CycleGAN [21] is used to generate synthetic underwater images in this work. Figure 4 shows several examples of the synthetic images. It can be seen the generated images successfully transfer the in-air images to underwater style, especially with respect to color.

One interesting phenomenon during the transfer is that if we attempt to train for many epochs in the style-transfer network, a lot of unnecessary and unreasonable features are also learned. However, in most cases, we just need to transfer some specific features, like color. The testing on our own underwater dataset revealed that the estimation results for some water-only parts are not accurate enough. This may

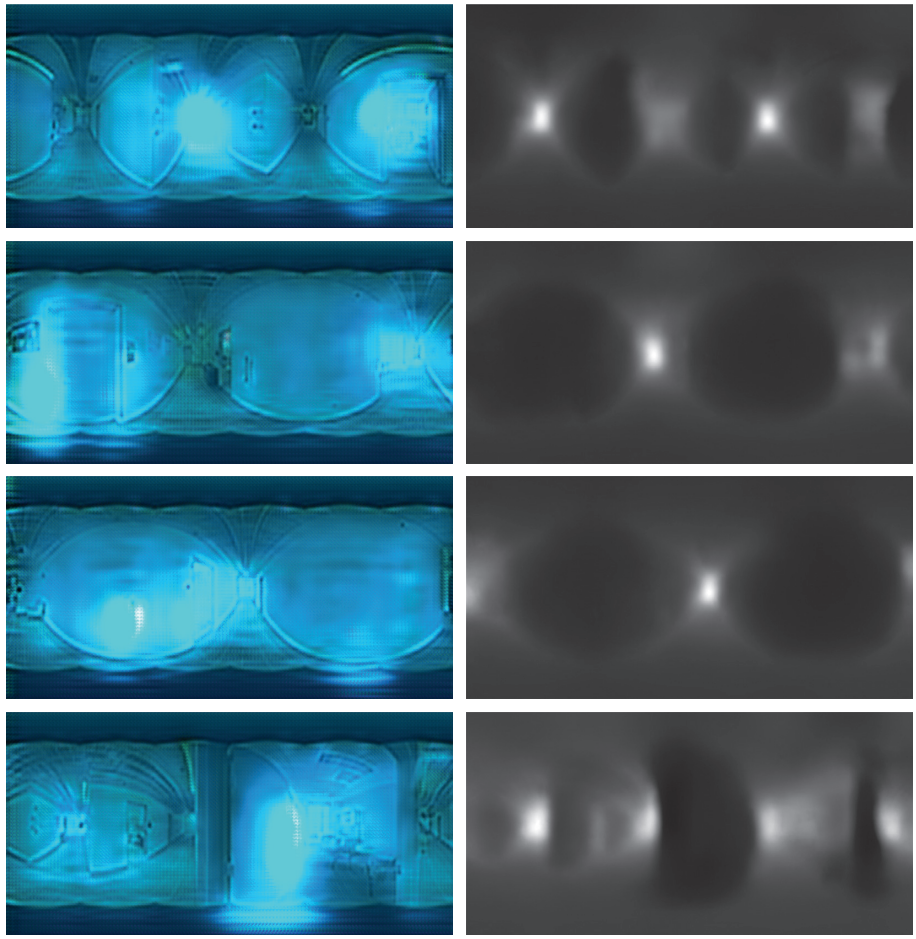

Figure 5: Generated depth from style-transferred underwater Stanford 2D-3D-S dataset. (a) On the left are the input images. (b) On the right are the corresponding predicted depth maps.

also be due to the fact that indoor scenarios are too different from the underwater domain.

Figure 5 presents the results of the estimated depth from the synthetic underwater Stanford 2D-3D-S dataset, where brighter pixels represent a larger depth and darker pixels are closer. It can be seen that the estimated depths on the right of Figure 5 corresponding to the left image are acceptable, especially the further area. Additionally, Table 1 gives a more rigorous evaluation of the results. Comparing to the classic FCRN network, our improved loss function gives slightly better results as indicated by the smaller RMS, Rel, and log10.

It can also be seen from the FCRN RGB experiment that using RGB images for training the SOTA network gives far worse results compared to ours and also to FCRN trained with GAN images. Because the style-transferred images mainly imitate the color information, the network was adopted to estimate the depth information from these images.

5.2. Evaluation of Real Underwater Images. After achieving acceptable results on the synthetic dataset, we also evaluate the results on the real underwater images. Note that we cannot compare to any other methods here, since, to the best of our knowledge, we are the first to propose an algorithm for depth estimation on spherical underwater images. Figure 6 demonstrates the estimated depth on our underwater dataset. Similarly, it can be seen that the brighter parts on the right correspond to areas more far away on the right of Figure 6, which implies that the network at least estimates the depth correctly in some regions.

Because our network is based on the Stanford 2D-3D-S dataset, in which the original images are all lacking the upper and lower parts (15.6% of the image height for each part), these parts are filled with pure black pixels. Therefore, the upper and lower parts in the final results about underwater depth estimation are also not evaluated. In the other words, we only use panorama images instead of spherical images actually.

Though our underwater dataset does not have ground truth depth maps, we can evaluate the results with the sparse map points. We randomly choose 20 images to test with the corresponding ground truth calculated by stereopsis.

According to the metric presented, the results are shown in the first row of Table 2. There, each column shows results averaged over all images. In the first column, we take the median of the errors of all pixels for which we have ground truth in that image, in the second column we take the mean

Table 1: Performance comparison on 1412 images from the Stanford 2D-3D-S dataset.

Methods	RMS (m) ↓	Rel (m) ↓	log10 ↓	$\delta < 1.25\uparrow$	$\delta < 1.25^2\uparrow$	$\delta < 1.25^3\uparrow$
Ours: + $L_{\text{grad}}^{\text{sph}}$	0.683	0.177	0.075	0.744	0.919	0.972
FCRN GAN	0.687	0.181	0.078	0.737	0.920	0.972
FCRN RGB	1.281	0.327	0.181	0.387	0.648	0.801

All tests use images transformed with GAN as input. Our approach and FCRN GAN were trained with synthetic images, while FCRN RGB uses, for comparison, RGB images as training data. The terms are explained below. The arrows indicate that smaller (↓) or bigger (↑) values are better.

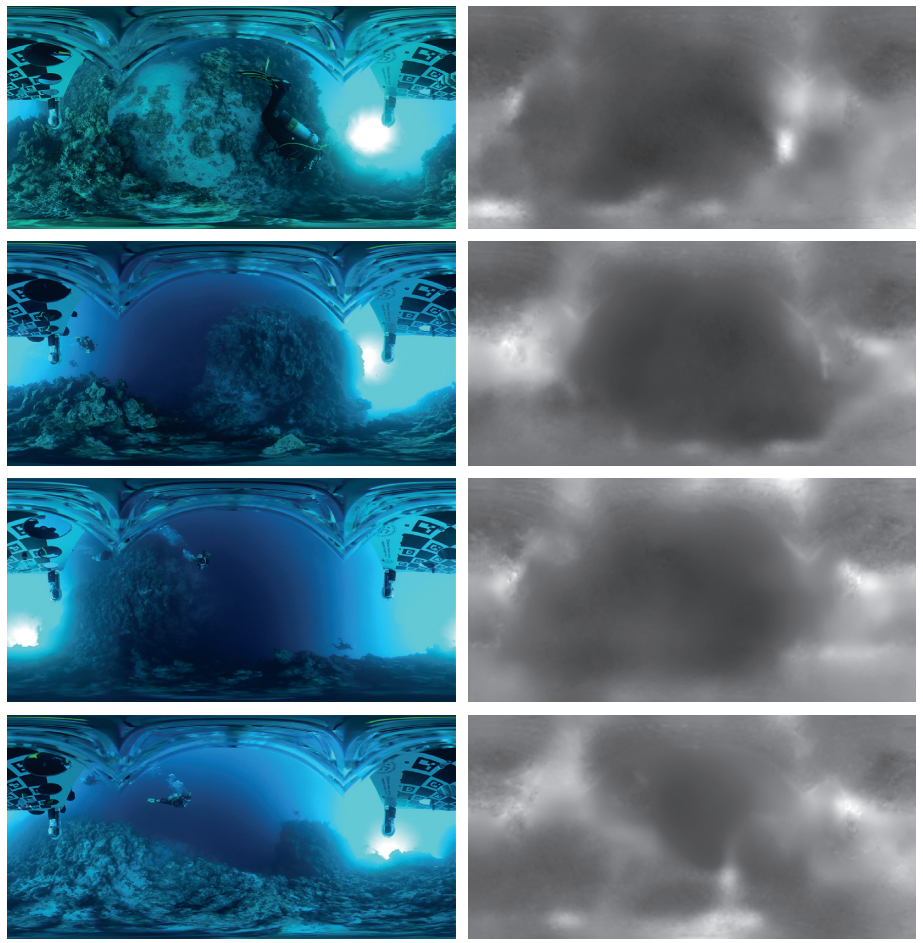

Figure 6: Generated depth from our underwater dataset. (a) On the left are the input images. (b) On the right are the corresponding predicted depth maps. We can find the upper and lower parts (15.6% of the image width for each part) are not good, and the reasons are shown in Evaluation Section.

Table 2: Performance comparison between the ground truth and various results.

Results types	Average median error	Average mean error	Average standard deviation
Ours	0.22	0.40	0.62
FCRN (trained with RGB)	0.30	3.76	7.16
Black result	1.00	1.00	0.00
White result	0.95	1.10	0.65
Random noise result	0.96	2.83	3.31
Gray-scale result	0.95	1.10	7.12
Black input	0.27	3.75	7.18
White input	0.31	3.70	6.91
Random noise input	0.32	3.77	7.00
Gray-scale input	0.24	0.51	1.26

More details are shown in the Supplementary Material.

error in each pixel, and the last column shows the standard deviation in each image, each averaged over all images. We can see that the average median error is 22% of the estimated depth, with a mean error of 40% and a standard deviation of 62%. Of course, those values show that the estimated depth is quite inaccurate. Nevertheless, we believe that they are still somewhat useful for certain applications, for example, navigation, colorization, dehazing, or location fingerprinting. Furthermore, we hope that, in the future, those values can be improved, for example by better and more training data and by providing a few consecutive or stereo frames as input.

In order to better understand the properties of our approach and put the evaluation results for our method into perspective, we use the same test frames to compare with three other cases. The new row in Table 2 shows the results of the original FCRN, trained with the normal RGB images from Stanford 2D-3D-S. When testing this network with our real underwater data, we see that the average mean error and the average standard deviation are very big, compared to our proposed approach. This shows that using the CycleGAN synthetic images during training is very advantageous. Even though this does not prove that the CycleGAN provides a very realistic underwater transfer, it is a very strong indication towards it.

The other two cases we show in Table 2 aim to show that our approach is indeed doing something useful and not just giving some random values. Firstly, we make four different fake depth results for comparison. The "black result" depth image is all black (0 distance), the "white result" depth image is all white, and the "random noise result" depth image has random distances. Finally, there is also a depth image called "gray-scale result," which is simply the input underwater image in grey scale. Please note that, in the "black result" case, there are all 0's in the image, so the scale parameter s cannot be obtained by the metric presented above. However, any scale that acts on 0 is itself. Thus, we just change the metric to a specific way, that is, setting scale parameter $s = 1$. Then, the error in that case is always 1; thus, the standard deviation is 0. We can see that the evaluations of all those fake results are much worse than our result.

Secondly, we used the same data as above (black, white, random noise, gray-scale input image) as the input to our approach. This can be regarded as a test to see if the network is overfitting too much. Generating good results on meaningless data would be a clear indication of overfitting, for example, because the training data is not diverse enough. We can see that the average median error is in the range of our result. We think this is due to two reasons: (i) provided with meaningless data, the network seems to generate depth images that somewhat resemble typical depth images; thus, it might be overfitting a bit. (ii) The rescaling process of our evaluation is optimizing the generated depth maps, such that they best fit the ground truth (for the underwater image that is not being used here). The median error of that ground truth may be quite small for those "typical" depth images generated for meaningless data. But looking at the average mean error and standard deviation, we see that those generated depth maps have a very big error, thus showing that our result is clearly much better.

In the last row, we use the gray-scale version of the color frame as the input. As could be expected, this has reasonable, second-best results. Nevertheless, it is still worse than the color input, so the color seems to be important. Comparing the result of our method to all other tests, we see that the average median error, average mean error, and average standard deviation are much better for our approach, clearly showing that our approach does work to a certain extend.

6. Conclusions

This paper presented a supervised depth learning method for underwater spherical images. Firstly, we implemented style transfer based on CycleGAN to synthesize the underwater images. The results show that CycleGAN learned the features of underwater scenarios and synthesizes nice images in the underwater style. Those images are then used in a second network, a Fully Convolutional Residual Network (FCRN), to train underwater spherical depth estimation. The network is trained in a supervised manner. Our first experiment was using the synthetic images from CycleGAN for evaluation and comparison with FCRN. Furthermore, we tested our method on real underwater data from the Great Barrier Reef, for which we estimated sparse ground truth depth points using stereopsis and bundle adjustment. We also compared our results to artificial input and output data, to show that the network is indeed performing depth estimation. The experiments demonstrated that the style transfer, as well as the depth estimation results, is convincing. Our method achieves better results than training without GAN. It achieves slightly better results than FCRN trained with GAN, so our updated loss function is beneficial. The experiments also showed that the estimated depth on real underwater images is somewhat reasonable and better than all other methods and options we compared to.

Nevertheless, the approach is far from perfect, especially regarding the accuracy of the estimated depth. This is mainly due to the fact that estimating the depth from a single image is a very challenging task. Our approach is also not very general. The underwater dataset was taken only at one location with very good visibility. There are many more underwater scenarios with differing styles. So, more underwater training data is needed. In the future, we plan to work on a unified approach that can work in all kinds of different underwater situations. In addition, for testing in the real underwater environments, we also plan to mask water-only areas by a segmentation process. Collecting an in-air dataset with depth that looks closer to the underwater images might also further improve our performance. Those might be some canyons or deserts. Since the underwater data we collected actually also contains spherical videos from two more cameras, we will

investigate using this stereo data for depth training. Furthermore, more complicated network structures that take previous frames into account may provide even better results.

Data Availability

The images of the underwater dataset, including the data for the ground truth evaluation, can be found on https://robotics.shanghaitech.edu.cn/static/datasets/underwater/UW_omni.tar.gz (780 MB).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Supplementary Materials

Tables S1, S2, and S3 show the median, mean, and standard deviation of the error between the ground truth and results estimated from different methods. The column "ours" is the result estimated by the proposed method. The "gray-scale" is converted from the input RGB image. The remaining "random noise," "white" and "black," is generated manually. The column with "result" is calculated by comparing the ground truth and the image directly whereas that with "input" is computed by firstly taking the image as the input of the proposed network and then comparing the output with ground truth. The "ours without GAN" denotes the result about the model trained by the original in-air dataset, without CycleGAN. In addition, the "gt size" is the number of points provided by ground truth. (*Supplementary Materials*)

References

[1] A. Gomez Chavez, Q. Xu, C. A. Mueller, S. Schwertfeger, and A. Birk, "Adaptive navigation scheme for optimal deep-sea localization using multimodal perception cues," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, November 2019.

[2] J. Yuh and M. West, "Underwater robotics," *Advanced Robotics*, vol. 15, no. 5, pp. 609–639, 2001.

[3] C. Beall, B. J. Lawrence, I. Viorela, and D. Frank, "3d reconstruction of underwater structures," in *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4418–4423, IEEE, Taipei, Taiwan, September 2010.

[4] J. Li, K. A. Skinner, E. Ryan, and M. J.-R. Watergan, "Unsupervised generative network to enable real-time color correction of monocular underwater images," *IEEE Robotics and Automation Letters (RA-L)*, pp. 387–394, 2017.

[5] P. L. J. Drews, E. R. Nascimento, S. S. C. Botelho, and M. F. Montenegro Campos, "Underwater depth estimation and image restoration based on single images," *IEEE Computer Graphics and Applications*, vol. 36, no. 2, pp. 24–35, 2016.

[6] T. Łuczyński and A. Birk, "Underwater image haze removal with an underwater-ready dark channel prior," in *OCEANS 2017*, pp. 1–6, IEEE, Anchorage, AK, USA, September 2017.

[7] K. Kaiming He, J. Jian Sun, and X. Xiaoou Tang, "Single image haze removal using dark channel prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2341–2353, 2011.

[8] M. Pfingsthorn, A. Birk, S. Schwertfeger, H. Bülow, and K. Pathak, "Maximum likelihood mapping with spectral image registration," in *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*, pp. 4282–4287, Anchorage, AK, USA, May 2010.

[9] Y.-T. Peng, X. Zhao, and P. C. Cosman, "Single underwater image enhancement using depth estimation based on blurriness," in *Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP)*, pp. 4952–4956, Quebec, Canada, September 2015.

[10] P. Anandan, S. Gagliano, and M. Bucolo, "Computational models in microfluidic bubble logic," *Microfluidics and Nanofluidics*, vol. 18, no. 2, pp. 305–321, 2015.

[11] F. Cairone, P. Anandan, and M. Bucolo, "Nonlinear systems synchronization for modeling two-phase microfluidics flows," *Nonlinear Dynamics*, vol. 92, no. 1, pp. 75–84, 2018.

[12] A. A. Argyros, K. E. Bekris, S. C. Orphanoudakis, and L. E. Kavraki, "Robot homing by exploiting panoramic vision," *Autonomous Robots*, vol. 19, no. 1, pp. 7–25, 2005.

[13] R. Benosman, S. Kang, and O. Faugeras, *Panoramic Vision*, Springer-Verlag New York, Berlin, Germany, 2000.

[14] H. Kuang, Q. Xu, X. Long, and S. Schwertfeger, "Pose estimation for omni-directional cameras using sinusoid fitting," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, November 2019.

[15] T. Lemaire and S. Lacroix, "Slam with panoramic vision," *Journal of Field Robotics*, vol. 24, no. 1-2, pp. 91–111, 2007.

[16] Q. Xu, A. Gomez Chavez, H. Bülow, A. Birk, and S. Schwertfeger, "Improved fourier mellin invariant for robust rotation estimation with omni-cameras," in *Proceedings of the 2019 26th IEEE International Conference on Image Processing. IEEE*, Taipei, Taiwan, September 2019.

[17] B. Terry, "Dove: dolphin omni-directional video equipment," in *Proceedings of the International Conference on Robotics and Automation*, pp. 214–220, Paris, France, May 2000.

[18] J. Bosch, N. Gracias, P. Ridao, and D. Ribas, "Omnidirectional underwater camera design and calibration," *Sensors*, vol. 15, no. 3, pp. 6033–6065, 2015.

[19] F. Bruno, G. Bianco, M. Muzzupappa, S. Barone, and A. V. Razionale, "Experimentation of structured light and stereo vision for underwater 3d reconstruction," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 4, pp. 508–518, 2011.

[20] I. Armeni, S. Sax, A. R. Zamir, and S. Savarese, "Joint 2d-3d-semantic data for indoor scene understanding," 2017, https://arxiv.org/abs/1702.01105.

[21] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, October 2017.

[22] H. Kuang, Q. Xu, and S. Schwertfeger, "Depth estimation on underwater omni-directional images using a deep neural network," 2019, https://arxiv.org/abs/1905.09441.

[23] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1422–1430, Santiago, Chile, December 2015.

[24] J. Donahue, P. Krähenbühl, and Trevor Darrell, "Adversarial feature learning," 2016, https://arxiv.org/abs/1605.09782.

[25] A Dosovitskiy, P. Fischer, J. Tobias Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with exemplar convolutional neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 9, pp. 1734–1747, 2015.

[26] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Proceedings of the International Conference on Learning Representations*, Vancouver, Canada, April 2018.

[27] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *Proceedings of the European conference on computer vision*, pp. 649–666, Amsterdam, Netherlands, October 2016.

[28] C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. Murphy, "Tracking emerges by colorizing videos," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 391–408, Munich, Germany, September 2018.

[29] X. Wang and A. Gupta, "Unsupervised learning of visual representations using videos," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2794–2802, Santiago, Chile, December 2019.

[30] E. Jang, C. Devin, V. Vincent, and S. Levine, "Grasp2vec: learning object representations from self-supervised grasping," in *Proceedings of the Conference on Robot Learning*, Zurich, Switzerland, October 2018.

[31] A. Nair, S. Bahl, K. Alexander, P. Vitchyr, G. Berseth, and S. Levine, "Contextual imagined goals for self-supervised robotic learning," in *Proceedings of the Conference on Robot Learning*, Osaka, Japan, October 2019.

[32] X. Zhi, X. He, and S. Schwertfeger, "Learning autonomous exploration and mapping with semantic vision," in *Proceedings of the International Conference on Image, Video and Signal Processing. IVSP*, Shanghai China, February 2019.

[33] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 270–279, Honolulu, HI, USA, July 2017.

[34] H. Zhan, C. S. Weerasekera, R. Garg, and I. Reid, "Self-supervised learning for single view depth and surface normal estimation," in *Proceedings of the 2019 International Conference on Robotics and Automation (ICRA)*, pp. 4811–4817, Montreal, Canada, May 2019.

[35] F. Ma, G. V. Cavalheiro, and S. Karaman, "Self-supervised sparse-to-dense: self-supervised depth completion from lidar and monocular camera," in *Proceedings of the 2019 International Conference on Robotics and Automation (ICRA)*, pp. 3288–3295, Montreal, Canada, May 2019.

[36] A. Wong and S. Soatto, "Bilateral cyclic constraint and adaptive regularization for unsupervised monocular depth prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5644–5653, Long Beach, CA, USA, June 2019.

[37] 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*, pp. 1851–1858, Honolulu, HI, USA, July 2017.

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

[39] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. Reid, "Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 340–349, Long Beach, CA, USA, June 2019.

[40] H. Zhan, C. S. Weerasekera, J. Bian, and I. Reid, "Visual odometry revisited: what should be learnt?," 2019, https://arxiv.org/abs/1909.09803.

[41] P.-Y. Chen, H. Alexander, Y.-C. Liu, and Y.-C. F. Wang, "Towards scene understanding: unsupervised monocular depth estimation with semantic-aware representation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2624–2632, Long Beach, CA, USA, June 2019.

[42] 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*, pp. 1983–1992, Long Beach, CA, USA, June 2019.

[43] A. Ranjan, V. Jampani, L. Balles et al., "Competitive collaboration: joint unsupervised learning of depth, camera motion, optical flow and motion segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 12240–12249, Long Beach, CA, USA, June 2019.

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

[45] D. Paul, E. Nascimento, F. Moraes, S. Botelho, and M. Campos, "Transmission estimation in underwater single images," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 825–830, Sydney, Australia, April 2013.

[46] 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.

[47] X. Ding, Y. Wang, J. Zhang, and X. Fu, "Underwater image dehaze using scene depth estimation with adaptive color correction," in *OCEANS 2017*, pp. 1–5, Aberdeen, Scotland, June 2017.

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

[49] K. A. Skinner, E. Iscar, and M. Johnson-Roberson, "Automatic color correction for 3d reconstruction of underwater scenes," in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5140–5147, Singapore, May 2017.

[50] J. S. Jaffe, "Computer modeling and the design of optimal underwater imaging systems," *IEEE Journal of Oceanic Engineering*, vol. 15, no. 2, pp. 101–111, 1990.

[51] B. L. McGlamery, "Computer analysis and simulation of underwater camera system performance," *SIO Reference*, vol. 75, no. 2, 1975.

[52] K. A. Skinner, J. Zhang, E. A. Olson, and M. J.-R. Uwstereonet, "Unsupervised learning for depth estimation and color correction of underwater stereo imagery," in *Proceedings of the 2019 International Conference on Robotics and Automation (ICRA)*, pp. 7947–7954, Singapore, May 2019.

[53] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *Proceedings of the 2016 Fourth*

International Conference on 3D Vision (3DV), pp. 239–248, Stanford, California, October 2016.

[54] K. Tateno, N. Navab, and F. Tombari, "Distortion-aware convolutional filters for dense prediction in panoramic images," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 707–722, Munich, Germany, September 2018.

[55] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial nets," *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014.

[56] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," *Advances in Neural Information Processing Systems*, pp. 2366–2374, 2014.

[57] L. Jin, Y. Xu, Z. Jia et al., "Geometric structure based and regularized depth estimation from 360 indoor imagery," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 889–898, Seattle, WA, USA, June 2020.

[58] O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation," in *Proceedings of the International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Munich, Germany, October 2015.

[59] Z. Zhang, Y. Xu, J. Yu, and S. Gao, "Saliency detection in 360 videos," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 488–503, Munich, Germany, September 2018.

[60] H. Stewenius, D. Nister, F. Kahl, and F. Schaffalitzky, "A minimal solution for relative pose with unknown focal length," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, pp. 789–794, San Diego, California, June 2005.

[61] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, UK, 2003.