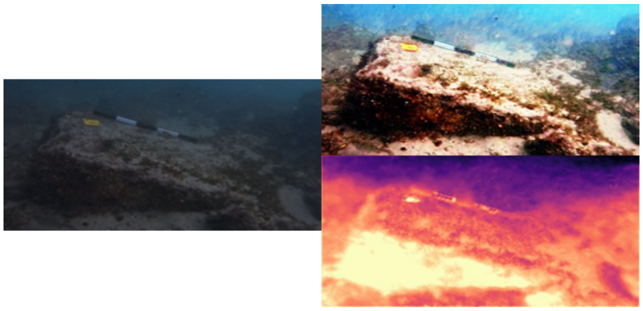
Sea-ing in Low-light

Supplementary Material

The figures and tables in this supplementary material are numbered using the prefix S, and are arranged as follows:

1. Sample outputs from SelfLUID-Net for an input video sequence.
2. The statistics of the proposed dataset ULVStereo.
3. Additional ablations on sequential processing of low light underwater (LLUW) images.
4. Discussion on underwater lowlight scenarios.
5. Comparison of network complexity.
6. Limitations.
7. Additional results.
8. The network architecture of the proposed SelfLUID-Net.

S1. Sample outputs from SelfLUID-Net



(a) Input LLUW video

(b) Restored image and depth map

Figure S1. Sample outputs (Restored image and depth map) from SelfLUID-Net for an input LLUW video sequence. *Best viewed when the document is opened in Adobe Reader.*

SelfLUID-Net can process a single image in 16ms. i.e., it can process around 62 frames per second. For a given input image of size 512x512, our network will give the restored image and depth map in 16 ms. For a video, each frame should be sequentially passed to the network to get the corresponding restored image and depth map to form a restored output video. Such an output for an input video of frame rate 30 fps with image dimensions 1024x512 is given in Fig. S1. Please use Adobe Reader to view the video.

S2. Statistics of ULVStereo

Underwater Low-light Stereo Video (ULVStereo) dataset contains stereo pairs of low-light and normally lit underwater (UW) images. It has 10 pairs of videos (ULV1 to ULV10) where each pair (low-light and normally lit videos) is captured around a submerged UW structure (natural or man-made rock). Each 3D structure differs in shape and size (span) and presents varied appearances from different viewpoints, providing ULVStereo with a rich scene diversity. The number of frames and a sample stereo image pair from each set are given in Table S1. Low-light and normally-lit stereo image pairs share the same scene but with different levels of illumination. The table contains information on which videos are utilized for training and which are allocated for testing. The captured images are of resolution 1920×1080 pixels. There are a total of 20956 frames in the ULVStereo dataset. We did calibration in-air for clear visibility of checkerboard corners and pattern stability. Our calibration did not account for refraction in water. However, the GoPro camera we used has a short focal length (f∼16-34mm). Moreover, we trained our network with central image patches (800×800), where rays remain largely parallel to optical axis as the objects we imaged were at distances significantly greater (3-4m) than f, thus reducing refraction effects. The three main features of our dataset are 1) The image pairs are of the same scene taken simultaneously and with different illumination; 2) Images in a pair contain disparity that provides depth information; and 3) It contains videos that cover different underwater structures. These features of ULVStereo should enable researchers in UW domain to harness it for many applications like underwater low-light image enhancement, depth estimation, 3D UW structure recovery, lowlight underwater neural radiance fields and Gaussian splatting, etc.

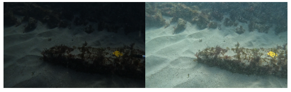
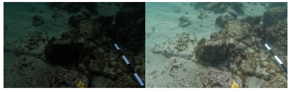
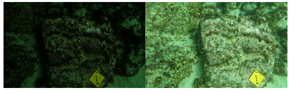
ULV1, 1293×2 = 2586, Training 	ULV2, 1402×2 = 2804, Training
ULV3, 927×2 = 1854, Training 	ULV4, 1566×2 = 3132, Training
ULV5, 741×2 = 1482, Testing 	ULV6, 899×2 = 1798, Testing
ULV7, 1110×2 = 2220, Training 	ULV8, 891×2 = 1782, Testing 
ULV9, 941×2 = 1882, Training 	ULV10, 708×2 = 1416, Testing

Table S1. Dataset summary of ULVStereo with sample frames from the stereo pairs (lowlight, normally-lit) of each video. Video name, Number of frames, and whether they are used for training or testing are specified for each video.

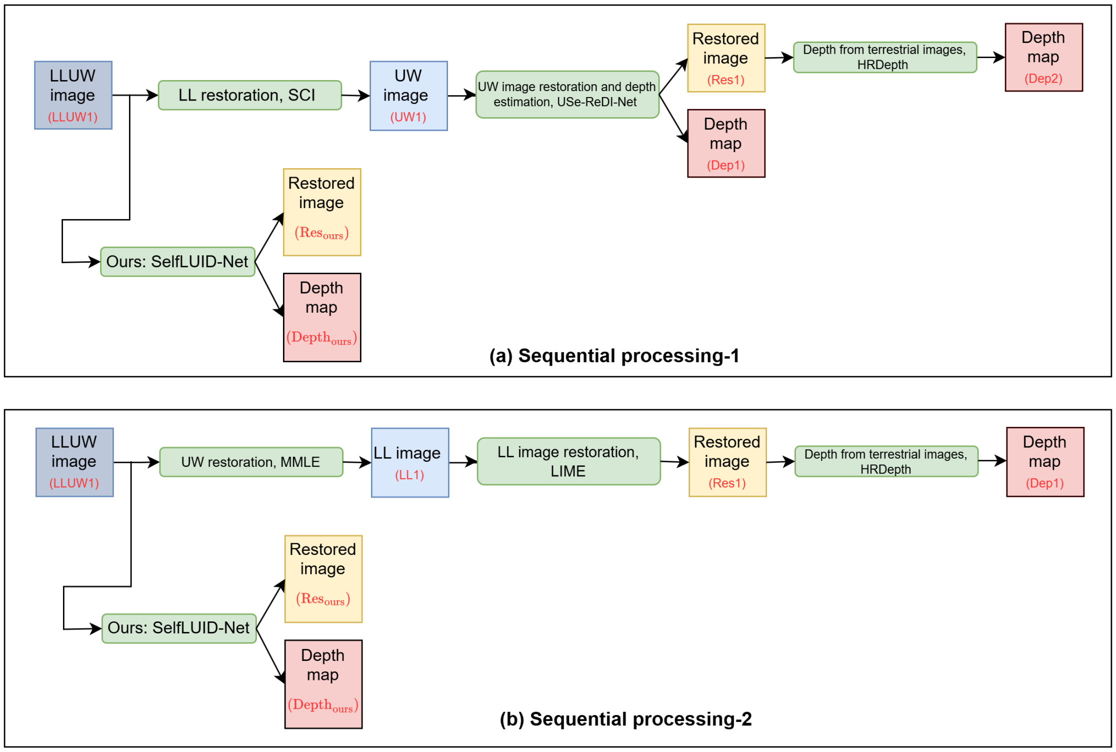

LLUW image (LLUW1)
LL restoration, SCI
UW image (UW1)
UW image restoration and depth estimation, USe-ReDI-Net
Restored image (Res1)
Depth from terrestrial images, HRDepth
Depth map (Dep2)
Depth map (Dep1)
Ours: SelfLUID-Net
Restored image (Res$_{\text{ours}}$)
Depth map (Depth$_{\text{ours}}$)
(a) Sequential processing-1
LLUW image (LLUW1)
UW restoration, MMLE
LL image (LL1)
LL image restoration, LIME
Restored image (Res1)
Depth from terrestrial images, HRDepth
Depth map (Dep1)
Ours: SelfLUID-Net
Restored image (Res$_{\text{ours}}$)
Depth map (Depth$_{\text{ours}}$)
(b) Sequential processing-2

Figure S2. Two sequential processing methods ((a) and (b)) have been done on a LLUW image input (LLUW1) to get restored image (Res1) as well as depth map (Dep1 and Dep2). SelfLUID-Net returns the restored image (Res$_{\text{ours}}$) and depth map (Depth$_{\text{ours}}$). Please refer to the names of outputs (in red) from each stage to see the visual outputs in Fig. S3 and S4.

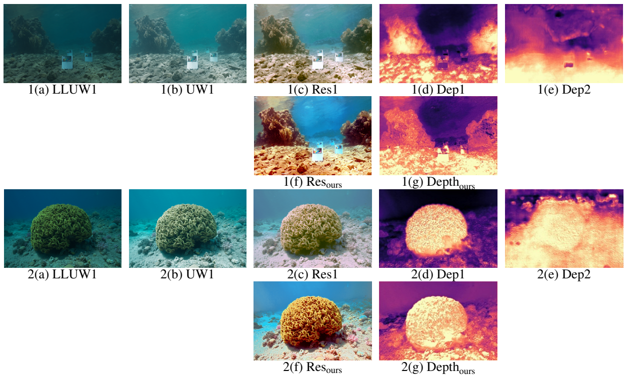

1(a) LLUW1 1(b) UW1 1(c) Res1 1(d) Dep1 1(e) Dep2
1(f) Res$_{\text{ours}}$ 1(g) Depth$_{\text{ours}}$
2(a) LLUW1 2(b) UW1 2(c) Res1 2(d) Dep1 2(e) Dep2
2(f) Res$_{\text{ours}}$ 2(g) Depth$_{\text{ours}}$

Figure S3. The outputs of each stage from sequential processing-1 (names are given in Fig. S2 in red) for input LLUW images from Seathru dataset [1]. Note that our restored image and depth map are better than those obtained from the sequential processing.

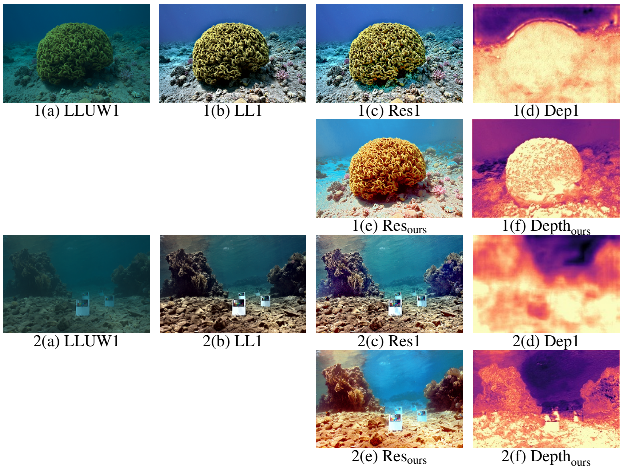

1(a) LLUW1 1(b) LL1 1(c) Res1 1(d) Dep1
1(e) Res$_{\text{ours}}$ 1(f) Depth$_{\text{ours}}$
2(a) LLUW1 2(b) LL1 2(c) Res1 2(d) Dep1
2(e) Res$_{\text{ours}}$ 2(f) Depth$_{\text{ours}}$

Figure S4. The outputs of each stage from sequential processing-2 (names are given in Fig. S2 in red) for input LLUW images from Seathru dataset [1]. Note that our restored image and depth map are better than those obtained from the sequential processing.

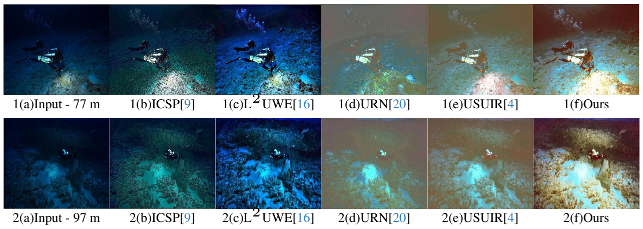
S3. Sequential processing vs SelfLUID-Net

SelfLUID-Net outputs the restored image and depth map simultaneously from a single LLUW image. UW image restoration methods devised for normally-lit UW images struggle due to low-light effects in LLUW images, while LL restoration methods devised for terrestrial images struggle due to haze present in the UW images. Instead of using SelfLUID-Net, another possibility is to i) sequentially process LLUW images (UW image restoration followed by LL image restoration and vice versa) using SoTA methods. Then ii) estimate the depth map from the restored image obtained from the earlier step using SoTA method for depth

	PSNR(dB)/SSIM	ρ/SI-MSE
Sequential processing-1	15.86/0.48	Dep1: 0.43/0.54 Dep2: 0.26/0.66
Sequential processing-2	16.78/0.52	0.35/0.60
Ours: SelfLUID-Net	17.21/0.58	0.52/0.40

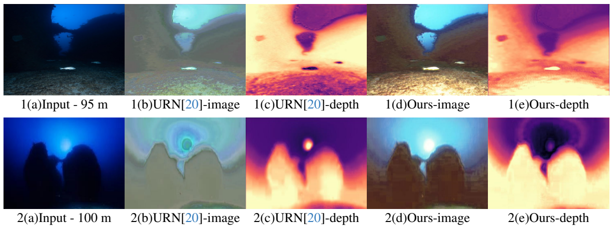
Table S2. Quantitative comparisons of image restoration and depth estimation accuracy for two sequential processing methods and our method. Average PSNR/SSIM is calculated on images from UIEB$_{\text{dark}}$ [11] dataset and ρ/SI-MSE is calculated on Seathru [1] dataset.

estimation. The two types of sequential processing (Sequential processing-1 with LL restoration followed by UW restoration, and Sequential processing-2 with UW restoration followed by LL restoration) are given in Fig. S2. All the Deep learning methods shown in Fig. S2 are trained using images from our ULVStereo dataset. SCI [15] is trained using LLUW images, USe-ReDI-Net [20] is trained using normally lit UW Images, and HRDepth [14] is trained using restored UW images (from USe-ReDI-Net). In Fig. S2, the output of each stage and the input image are named in red. The corresponding images from Sequential processing 1 and 2 are provided in Fig. S3 and Fig. S4, respectively. From Fig. S3, it can be seen that the LL image restoration method removes lowlight effects from the image, but is unable to remove haze. It can also be seen that some color information is lost after LL image restoration. Hence, UW image restoration after accounting for low light is not satisfactory. The depth map estimated from the sequential processing is also inferior. Fig. S4 shows that UW image recovery followed by LL restoration also gives poor restored results and the depth maps estimated from the restored outputs of this sequential method are again not good. In both cases, our results are better. The metric values for both the sequential methods are given in Table S2. Our SelfLUID-Net has the best metric scores for both depth estimation and restoration. Sequential processing of LLUW images provides suboptimal results as compared to SelfLUID-Net.


1(a)Input - 77 m 1(b)ICSP[9] 1(c)L^{2}UWE[16] 1(d)URN[20] 1(e)USUIR[4] 1(f)Ours

2(a)Input - 97 m 2(b)ICSP[9] 2(c)L^{2}UWE[16] 2(d)URN[20] 2(e)USUIR[4] 2(f)Ours

Figure S5. (a) Input UW images from deep water depths (depth from the sea-surface is also mentioned) and (b-f) the enhanced images obtained from different methods. URN: USe-ReDI-Net. Note that our output results are visually good.


1(a)Input - 95 m 1(b)URN[20]-image 1(c)URN[20]-depth 1(d)Ours-image 1(e)Ours-depth

2(a)Input - 100 m 2(b)URN[20]-image 2(c)URN[20]-depth 2(d)Ours-image 2(e)Ours-depth

Figure S6. (a) Input UW images from deep water depths (depth from the sea-surface is also mentioned) and the enhanced images (b and d) and depth map (c and e) obtained from URN: USe-ReDI-Net [20] and our method. Note that only our enhanced outputs are visually good and we have better depth maps than USe-ReDI-Net [20].

S4. Underwater lowlight scenarios

In underwater imaging, lowlight conditions occur due to insufficient lighting during image capture. This mainly happens in two cases.

1. Near-shore situations where
   (a) Natural light is insufficient at low depths, e.g., at night time or on cloudy days, or
   (b) Under normal ambient lighting but at relatively higher depths (around 10m).
2. Off-shore situations where the available sunlight fails to reach the deep waters.

Our method is mostly applicable to the former case in near-shore situations. This is a very practically relevant scenario as many UW expeditions without the use of external light sources are undertaken near-shore. Our ULVStereo dataset was captured at a depth of 4-7m to image 3D artifacts submerged at that depth. The lowlight videos in our dataset by adjusting the exposure settings of the camera can be considered as the data captured at a lower-depth scenario (around 4 m) with insufficient natural light (case 1(a)). Considering the attenuation of light intensity as it travels from the sea-surface to UW objects, our captured lowlight videos can also be interpreted as data captured at a relatively higher-depth (around 10m) with normal ambient lighting (case 1(b)). We have successfully tested our method on real LLUW datasets captured with normal camera settings in these actual low light conditions (1(a) and 1(b)), i.e., (a) at a lower depth of 5m (FLSea [19] dataset) in less ambient light and (b) at a higher depth of 10m (Seathru [1] dataset). It is to be noted that, even though the training images from our ULVStereo dataset has lowlight images captured by adjusting the camera exposure settings, our method works well on real LLUW images captured in actual lowlight conditions.

In off-shore deep sea situations (case 2), UW images are extremely dark. There are no publicly available real datasets for these extreme conditions. We have collected some underwater images from internet which are captured very deep (around 100 m) and are included in Fig. S5 and S6. It is to be noted that these images contain illumination from external light sources carried by the divers. However, they still appear relatively darker overall. We tested our SelfLUID-Net and other four methods (two self-supervised

UW methods: USUIR [4] and USe-ReDI-Net [20]; two traditional LLUW methods: ICSP [9] and L^{2}UWE [16]) on these deep water images. Comparison of restored outputs from different methods is given in Fig. S5. It can be seen that, compared to other methods, our method gives better restored image outputs, even though the output contains some overexposed area at the portions of external light source. Along with enhanced image, USe-ReDI-Net [20] returns depth map also. We have included a comparison of depth map and restored image returned from USe-ReDI-Net [20] and our SelfLUID-Net in Fig. S6. It can be seen that the restored image outputs from USe-ReDI-Net [20] are not good. Also, our depth map is better than USe-ReDI-Net. Even though our network is not trained with dark deep UW images, it performs reasonably well on such deep water images while other methods struggle.

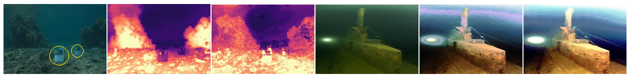
S5. Comparison of network complexity during training and testing

We use multiple constraints to make our self-supervision stronger. From ablation studies, we observed that utilizing fewer constraints results in poorer network performance. The individual subnetworks for R-Net, L-Net, TD/TB-Net, and beta-Net are relatively smaller networks with three to four Conv-Norm-ReLU blocks. PoseNet has a ResNet encoder and a lighter decoder. But after combining all the subnetworks, our model becomes bulky with a total trainable parameter count of around 16.4 M. Training our network using image patches of size 800x800 with a batch size of 1 takes around 22 GB GPU memory. It is to be noted that the entire network is used only during training. During inference, the single network R-Net is used to output the restored image, and TD/TB-Net and beta-Net are used to output the depth map requiring only around 3.1M parameters and 16ms to execute a 512x512 image.

Table S3. Number of trainable parameters (M) and Execution time in milliseconds for a 512x512 image.

	Mono2 [5]	USUIR [4]	USe-ReDI-Net [20]	Ours
Parameters (M)	14.2	2.2	24.7	16.4
Exec. time (ms)	25	14	18	16

The number of trainable parameters and the execution time during testing for our method and 3 baseline methods (Mono2 [5] which is a depth estimation method, USUIR [4] which is an UW image restoration method, and USe-ReDI-Net [20] which gives both restored image and depth map from normally-lit UW image) are given in Table S3. Our method has a higher number of trainable parameters, but during test time, since we use a small part of the network, our execution time is less. The number of parameters for USUIR [4] is very less, but it performs only image restoration. Mono2 [5] performs only depth estimation.



1(a)Input 1(b)URN [20] 1(c)Ours 2(a)Input 2(b)USUIR [4] 2(c)Ours

Figure S7. Failure cases for (1) depth estimation on Seathru dataset [1], and (2) restoration on NUID dataset [9]. URN: USe-ReDI-Net [20].

S6. Limitations

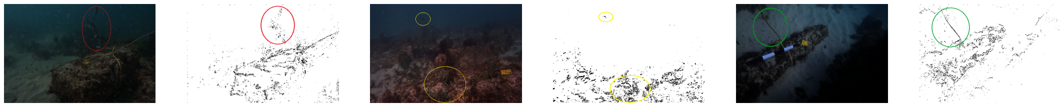
As given in Sec. S5, SelfLUID-Net contains a number of subnetworks with a total of around 16.4 M trainable parameters. To train the network using image patches of size 800x800 with a batch size of 1, it takes around 22 GB GPU memory. There are several hyperparameters to be tuned which we have done using grid-search.

Figure S7 shows failure example cases for 1) depth estimation (on an image from Seathru dataset [1]) and 2) image restoration (on an image from NUID dataset [9]). Results are given for SelfLUID-Net and the closest SoTA method (USe-ReDI-Net [20] for depth and USUIR [4] for restoration). For objects with reflective surfaces, the depth returned by both USe-ReDI-Net and our SelfLUID-Net are not good (see the encircled regions in Fig. S7:(1)). USe-ReDI-Net struggles more as it returns a higher depth for the whole object placed closed by, whereas our depth map is wrong only at the color checkerboard portions which span only a small area. In Fig. S7:(2)(a), the LLUW image contains a portion of a light source. Our restored images have artifacts around the light source since the LLUW model that we followed is not valid in such image areas. USUIR [4] also struggles and its restoration quality at other portions of the image is inferior to ours.

S7. Additional results

Additional qualitative results

In additional to the qualitative results given in the main paper, comparison results on more LLUW images are given in Fig. S10 and Fig. S11 for image restoration (on two images from ULVStereo, Seathru [1], and NUID [9], and one image from UIEB$_{\text{dark}}$ [11]) and depth estimation (on two images from Seathru [1], FLSea [19], and ULVStereo datasets), respectively. In the main paper, due to space constraints, we have not included the restored outputs of LL restoration method ZeroDCE [6]. But we have included its results in Fig. S10. Even though ZeroDCE [6] removes low light effects, UW haze is still present in their restored images (see Fig. S10:(1,3,5,7)(k)). For Seathru [1] and FLSea [19] datasets, we have now included the depth maps returned from Mono2 [5] (it has not been included in the main paper). Similar to other depth estimation methods for terrestrial images (HRDepth [14] and Manydepth [22]), Mono2 also struggles due to lowlight as well as UW haze (see Fig.



1(a)target image1 1(b)mask1 2(a)target image2 2(b)mask2 3(a)target image3 3(b)mask3

Figure S8. The predicted masks (b) to remove moving pixels in monocular videos are given for three target images (1,2,3)(a).

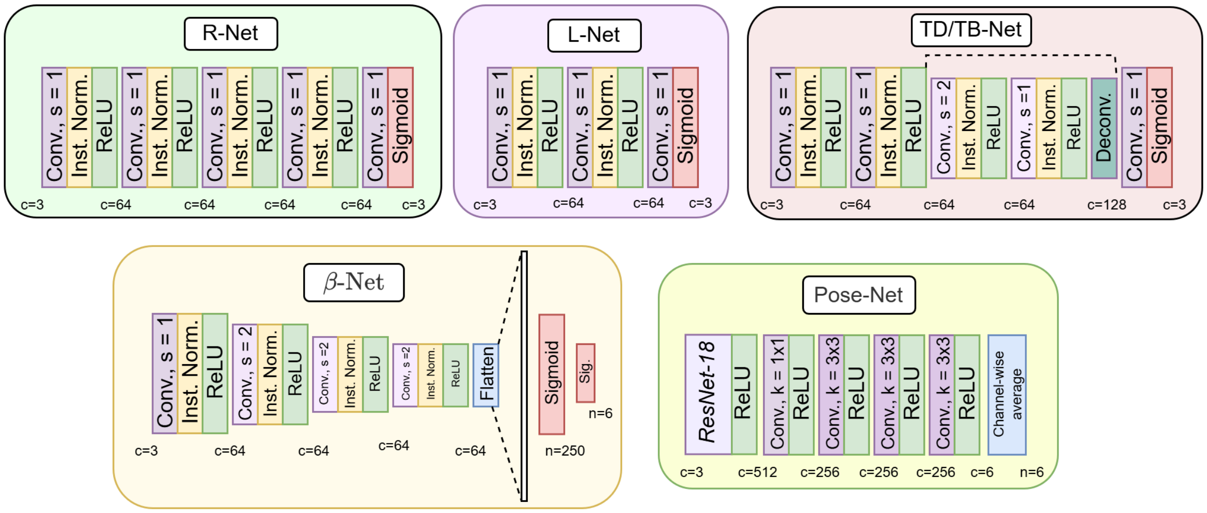


Figure S9. Network structure of each block in SelfLUID-Net. Inst. Norm.: Instance normalization, c: Number of channels in input or output, n: Number of output nodes. Values for either n or c are specified at the bottom of the network blocks. The size of the kernel, k is 3x3 unless specified.

S11:(1,2)(f),(3,4,5,6)(g)). For all the real UW datasets that we used for comparison, our method gives restored images with good quality and plausible depth maps. Our consistent and superior results with training on ULVStereo and testing on other datasets show its generalization ability.

Moving pixel masking

In main paper (Fig. 5), we have included one example of the predicted mask to remove the adverse effect of moving pixels on reprojection loss. In Fig. S8, we have included additional examples of predicted masks for three more target images. It can be seen that pixels correspond to floating plants (Fig. S8(1)), moving fish (Fig. S8(2)), and moving rope (Fig. S8(3)), along with small plants on the rocks are masked out not to affect the reprojection loss.

Additional ablations on loss terms

In the paper, we gave ablation studies mainly for our contributions. i.e., for $\mathcal{L}_R$ & its components and $\mathcal{L}_{\text{spa}}$. $\mathcal{L}_{\text{rec}}$ and $\mathcal{L}_{\text{dc}}$ cannot be excluded from loss calculation since they enforce the physics of image formation. $\mathcal{L}_{\text{ds}}$, $\mathcal{L}_{\text{is}}$, and $\mathcal{L}_{\text{clr}}$ have been used in literature. Our network (n/w) with ablations for these 3 losses has (PSNR/SSIM)(ρ/SI-MSE): No $\mathcal{L}_{\text{ds}}$:(16.9/0.54)(0.65/0.20), No $\mathcal{L}_{\text{is}}$:(16.4/0.51)(0.59/0.21), No $\mathcal{L}_{\text{clr}}$:(17.0/0.55)(0.69/0.19).

S8. SelfLUID-Net network architecture

In the main paper, the block diagram for SelfLUID-Net is given. The detailed structure of each block is given in Fig. S9. The input LLUW image I_L is disentangled into its latent components (global background light A, reflectance R_L, illumination L_L, transmission maps T_D and T_B). A is estimated analytically from I_L using a Gaussian blur-kernel. The reflectance R_L is estimated using the reflectance network (R-Net), illumination L_L is estimated from the illumination network (L-Net), transmission maps T_D and T_B are estimated from the transmission map networks TD-Net and TB-Net, respectively. R-Net uses only stride-1 convolutions to avoid missing any details. As in [20], transmission map networks use stride-2 convolutions and skip connections in TD-Net and TB-Net. L-Net has a small network structure with two conv-normalization-ReLU blocks and returns a three-channel illumination output. The outputs of R-Net and TD/TB-Net have three channels. Depth D is estimated from T_D, T_B, and the channel-wise extinction-coefficient β which is estimated using β-Net [20] from a 100x100 input image patch. β-Net returns a six-valued vector where a set of three β values corresponds to either T_D or T_B. For training using monocular video frames, we have used PoseNet to return the pose between neighboring frames. For PoseNet, we use the same network structure which was followed by monocular depth estimation methods for terrestrial images [5, 25]. PoseNet outputs a 6-dimensional camera pose output consisting of 3 rotation angles and 3 translations.

References

[1] Derya Akkaynak and Tali Treibitz. Sea-thru: A method for removing water from underwater images. In *CVPR*, pages 1682–1691, 2019. 2, 3, 4, 7, 8

[2] Dana Berman, Deborah Levy, Shai Avidan, and Tali Treibitz. Underwater single image color restoration using haze-lines and a new quantitative dataset. *IEEE PAMI*, 43(8):2822–2837, 2021. 7, 8

[3] Paulo L.J. Drews, Erickson R. Nascimento, Silvia S.C. Botelho, and Mario Fernando Montenegro Campos. Underwater depth estimation and image restoration based on single images. *IEEE CG&A*, 36(2):24–35, 2016. 8

[4] Zhenqi Fu, Huangxing Lin, Yan Yang, Shu Chai, Liyan Sun, Yue Huang, and Xinghao Ding. Unsupervised underwater image restoration: From a homology perspective. *AAAI*, 36 (1):643–651, 2022. 3, 4, 7

[5] Clement Godard, Oisin Mac Aodha, Michael Firman, and Gabriel Brostow. Digging into self-supervised monocular depth estimation. In *ICCV*, pages 3827–3837, 2019. 4, 6, 8

[6] Chunle Guo Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, and Runmin Cong. Zero-reference deep curve estimation for low-light image enhancement. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1780–1789, 2020. 4, 7

[7] Xiaojie Guo, Yu Li, and Haibin Ling. Lime: Low-light image enhancement via illumination map estimation. *IEEE Transactions on Image Processing*, 26(2):982–993, 2017. 7

[8] Honey Gupta and Kaushik Mitra. Unsupervised single image underwater depth estimation. In *ICIP*, pages 624–628, 2019. 8

[9] Guojia Hou, Nan Li, Peixian Zhuang, Kunqian Li, Haihan Sun, and Chongyi Li. Non-uniform illumination underwater image restoration via illumination channel sparsity prior. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2023. 3, 4, 7

[10] Manvi Jha and Ashish Kumar Bhandari. Cbla: Color-balanced locally adjustable underwater image enhancement. *IEEE Transactions on Instrumentation and Measurement*, 73:1–11, 2024. 7

[11] Chongyi Li, Chunle Guo, Wenqi Ren, Runmin Cong, Junhui Hou, Sam Kwong, and Dacheng Tao. An underwater image enhancement benchmark dataset and beyond. *TIP*, 29:4376–4389, 2020. 3, 4, 7

[12] Chongyi Li, Chunle Guo, and Chen Change Loy. Learning to enhance low-light image via zero-reference deep curve estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):4225–4238, 2022. 7

[13] Risheng Liu, Long Ma, Jiaao Zhang, Xin Fan, and Zhongxuan Luo. Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10556–10565, 2021. 7

[14] Xiaoyang Lyu, Liang Liu, Mengmeng Wang, Xin Kong, Lina Liu, Yong Liu, Xinxin Chen, and Yi Yuan. Hr-depth: High resolution self-supervised monocular depth estimation. *AAAI*, 35(3), 2021. 3, 4, 8

[15] Long Ma, Tengyu Ma, Risheng Liu, Xin Fan, and Zhongxuan Luo. Toward fast, flexible, and robust low-light image enhancement. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5627–5636, 2022. 3, 7

[16] Tunai Porto Marques and Alexandra Branzan Albu. L2uwe: A framework for the efficient enhancement of low-light underwater images using local contrast and multi-scale fusion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2286–2295, 2020. 3, 4, 7

[17] Yan-Tsung Peng and Pamela C. Cosman. Underwater image restoration based on image blurriness and light absorption. *TIP*, 26(4):1579–1594, 2017. 8

[18] Yan-Tsung Peng, Keming Cao, and Pamela C. Cosman. Generalization of the dark channel prior for single image restoration. *TIP*, 27(6):2856–2868, 2018. 8

[19] Yelena Randall and Tali Treibitz. Flsea: Underwater visual-inertial and stereo-vision forward-looking datasets, 2023. 3, 4, 8

[20] Nisha Varghese, Ashish Kumar, and A. N. Rajagopalan. Self-supervised monocular underwater depth recovery, image restoration, and a real-sea video dataset. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12214–12224, 2023. 3, 4, 5, 7, 8

[21] Kun Wang, Zhenyu Zhang, Zhiqiang Yan, Xiang Li, Baobei Xu, Jun Li, and Jian Yang. Regularizing nighttime weirdness: Efficient self-supervised monocular depth estimation in the dark. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16035–16044, 2021. 8

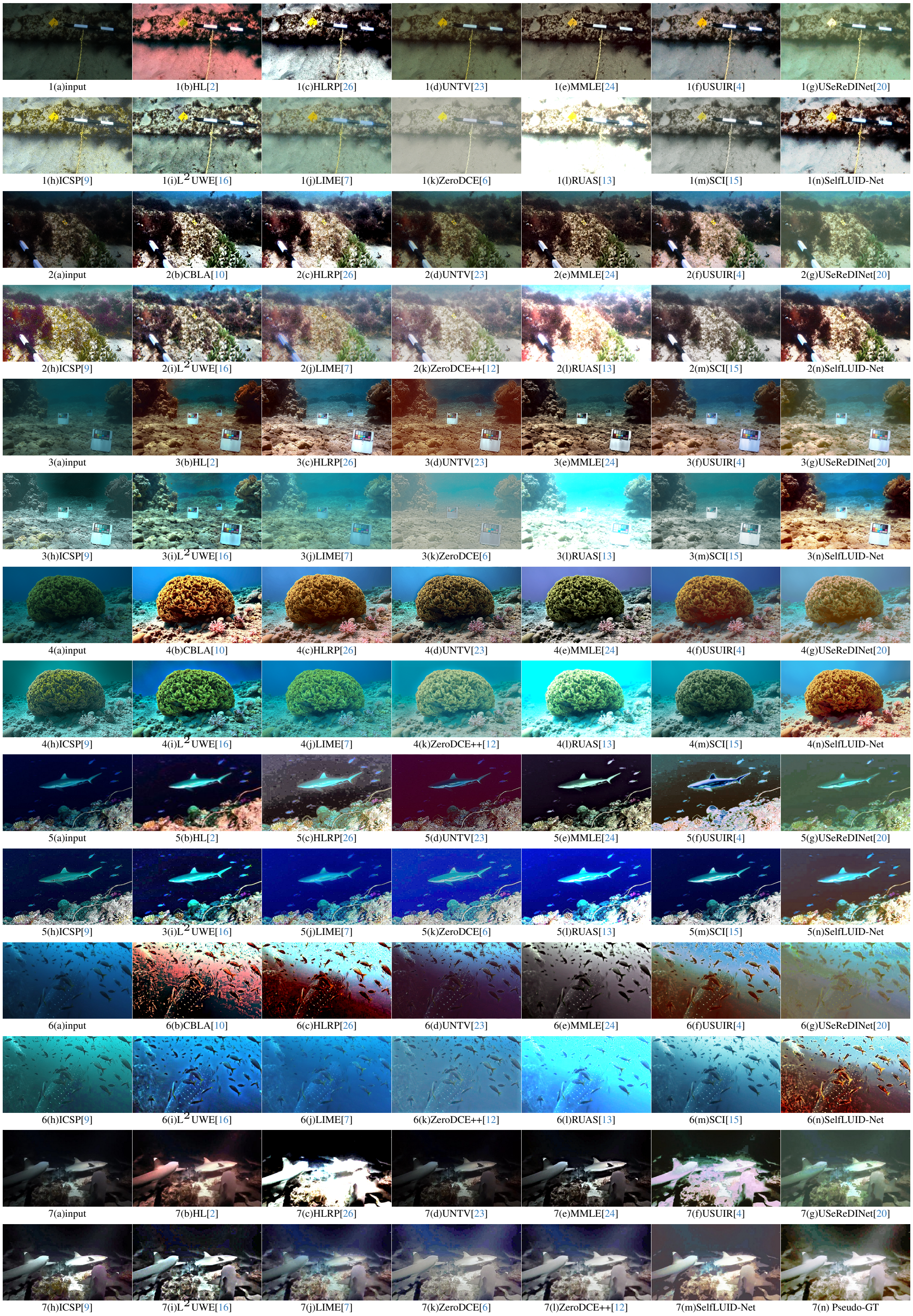
[22] Jamie Watson, Oisin Mac Aodha, Victor Prisacariu, Gabriel Brostow, and Michael Firman. The temporal opportunist: Self-supervised multi-frame monocular depth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1164–1174, 2021. 4, 8

[23] Jun Xie, Guojia Hou, Guodong Wang, and Zhenkuan Pan. A variational framework for underwater image dehazing and deblurring. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(6):3514–3526, 2022. 7

[24] Weidong Zhang, Peixian Zhuang, Hai-Han Sun, Guohou Li, Sam Kwong, and Chongyi Li. Underwater image enhancement via minimal color loss and locally adaptive contrast enhancement. *TIP*, 31:3997–4010, 2022. 7

[25] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, pages 6612–6619, 2017. 6

[26] Peixian Zhuang, Jiamin Wu, Fatih Porikli, and Chongyi Li. Underwater image enhancement with hyper-laplacian reflectance priors. *IEEE Transactions on Image Processing*, 31:5442–5455, 2022. 7



1(a)input 1(b)HL[2] 1(c)HLRP[26] 1(d)UNTV[23] 1(e)MMLE[24] 1(f)USUIR[4] 1(g)USeReDINet[20]

1(h)ICSP[9] 1(i)L^{2}UWE[16] 1(j)LIME[7] 1(k)ZeroDCE[6] 1(l)RUAS[13] 1(m)SCI[15] 1(n)SelfLUID-Net

2(a)input 2(b)CBLA[10] 2(c)HLRP[26] 2(d)UNTV[23] 2(e)MMLE[24] 2(f)USUIR[4] 2(g)USeReDINet[20]

2(h)ICSP[9] 2(i)L^{2}UWE[16] 2(j)LIME[7] 2(k)ZeroDCE++[12] 2(l)RUAS[13] 2(m)SCI[15] 2(n)SelfLUID-Net

3(a)input 3(b)HL[2] 3(c)HLRP[26] 3(d)UNTV[23] 3(e)MMLE[24] 3(f)USUIR[4] 3(g)USeReDINet[20]

3(h)ICSP[9] 3(i)L^{2}UWE[16] 3(j)LIME[7] 3(k)ZeroDCE[6] 3(l)RUAS[13] 3(m)SCI[15] 3(n)SelfLUID-Net

4(a)input 4(b)CBLA[10] 4(c)HLRP[26] 4(d)UNTV[23] 4(e)MMLE[24] 4(f)USUIR[4] 4(g)USeReDINet[20]

4(h)ICSP[9] 4(i)L^{2}UWE[16] 4(j)LIME[7] 4(k)ZeroDCE++[12] 4(l)RUAS[13] 4(m)SCI[15] 4(n)SelfLUID-Net

5(a)input 5(b)HL[2] 5(c)HLRP[26] 5(d)UNTV[23] 5(e)MMLE[24] 5(f)USUIR[4] 5(g)USeReDINet[20]

5(h)ICSP[9] 3(i)L^{2}UWE[16] 5(j)LIME[7] 5(k)ZeroDCE[6] 5(l)RUAS[13] 5(m)SCI[15] 5(n)SelfLUID-Net

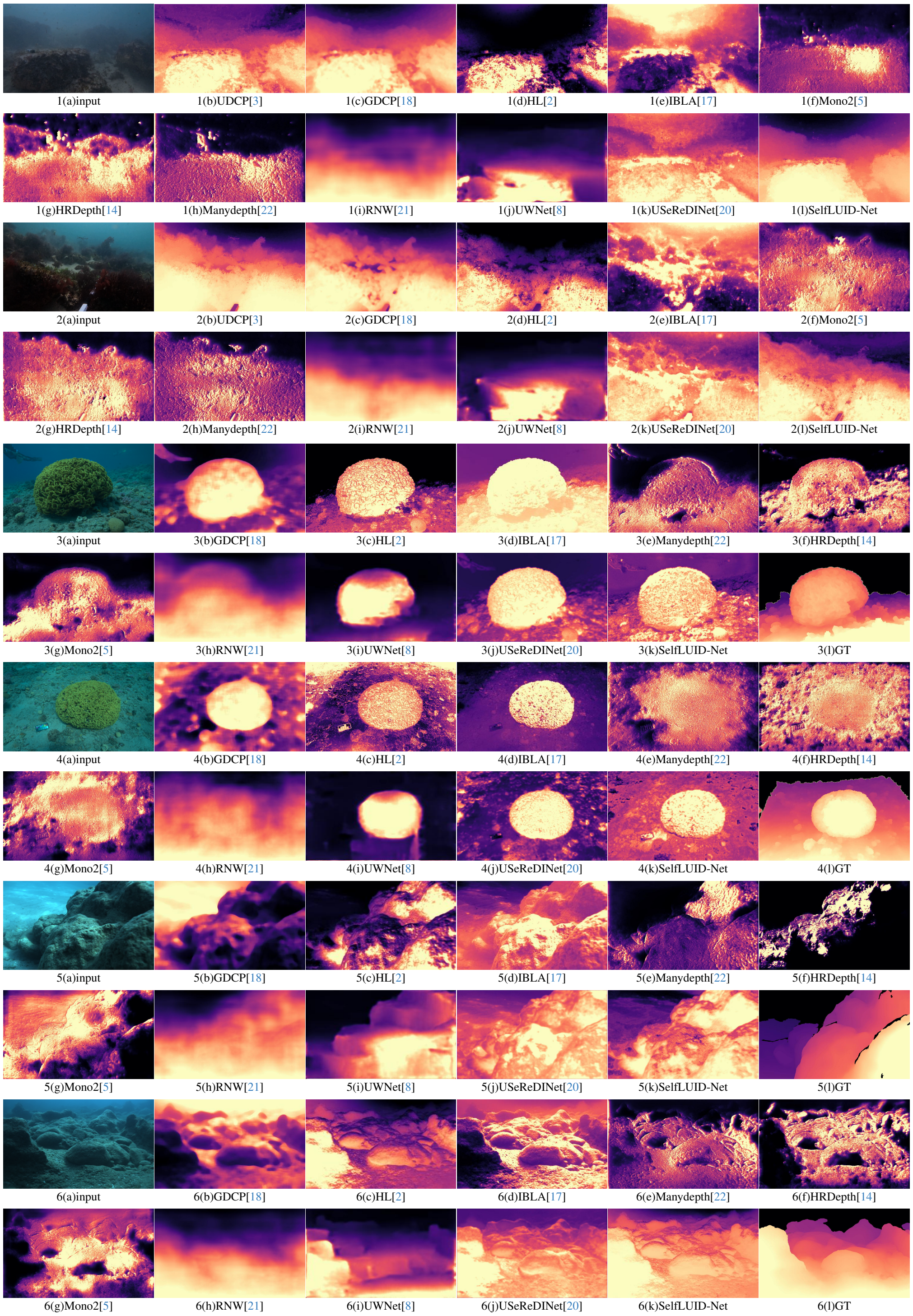
6(a)input 6(b)CBLA[10] 6(c)HLRP[26] 6(d)UNTV[23] 6(e)MMLE[24] 6(f)USUIR[4] 6(g)USeReDINet[20]

6(h)ICSP[9] 6(i)L^{2}UWE[16] 6(j)LIME[7] 6(k)ZeroDCE++[12] 6(l)RUAS[13] 6(m)SCI[15] 6(n)SelfLUID-Net

7(a)input 7(b)HL[2] 7(c)HLRP[26] 7(d)UNTV[23] 7(e)MMLE[24] 7(f)USUIR[4] 7(g)USeReDINet[20]

7(h)ICSP[9] 7(i)L^{2}UWE[16] 7(j)LIME[7] 7(k)ZeroDCE[6] 7(l)ZeroDCE++[12] 7(m)SelfLUID-Net 7(n) Pseudo-GT

Figure S10. Input UW image (a) from datasets: (1,2) - ULVStereo, (3,4) - Seathru [1], (5,6) - NUID [9], (7) UIEB$_{\text{dark}}$ [11] with pseudo ground truth (7(n)) for UIEB and the enhanced images obtained from different methods. Note that our output results are visually good.



1(a)input 1(b)UDCP[3] 1(c)GDCP[18] 1(d)HL[2] 1(e)IBLA[17] 1(f)Mono2[5]

1(g)HRDepth[14] 1(h)Manydepth[22] 1(i)RNW[21] 1(j)UWNet[8] 1(k)USeReDINet[20] 1(l)SelfLUID-Net

2(a)input 2(b)UDCP[3] 2(c)GDCP[18] 2(d)HL[2] 2(e)IBLA[17] 2(f)Mono2[5]

2(g)HRDepth[14] 2(h)Manydepth[22] 2(i)RNW[21] 2(j)UWNet[8] 2(k)USeReDINet[20] 2(l)SelfLUID-Net

3(a)input 3(b)GDCP[18] 3(c)HL[2] 3(d)IBLA[17] 3(e)Manydepth[22] 3(f)HRDepth[14]

3(g)Mono2[5] 3(h)RNW[21] 3(i)UWNet[8] 3(j)USeReDINet[20] 3(k)SelfLUID-Net 3(l)GT

4(a)input 4(b)GDCP[18] 4(c)HL[2] 4(d)IBLA[17] 4(e)Manydepth[22] 4(f)HRDepth[14]

4(g)Mono2[5] 4(h)RNW[21] 4(i)UWNet[8] 4(j)USeReDINet[20] 4(k)SelfLUID-Net 4(l)GT

5(a)input 5(b)GDCP[18] 5(c)HL[2] 5(d)IBLA[17] 5(e)Manydepth[22] 5(f)HRDepth[14]

5(g)Mono2[5] 5(h)RNW[21] 5(i)UWNet[8] 5(j)USeReDINet[20] 5(k)SelfLUID-Net 5(l)GT

6(a)input 6(b)GDCP[18] 6(c)HL[2] 6(d)IBLA[17] 6(e)Manydepth[22] 6(f)HRDepth[14]

6(g)Mono2[5] 6(h)RNW[21] 6(i)UWNet[8] 6(j)USeReDINet[20] 6(k)SelfLUID-Net 6(l)GT

Figure S11. Input UW image (a) from datasets: (1,2) - ULVStereo, (3,4) - Seathru [1], (5,6) - FLSea [19], with ground truth ((3,4)(l) and (5,6)(l)) for Seathru and FLSea datasets and the depth map obtained from different methods. Note that SelfLUID-Net returns plausible depth maps [see depth maps (3,4,5,6)(k) are closer to GT (3,4,5,6)(l)].