UWStereo: A Large Synthetic Dataset for Underwater Stereo Matching

Qingxuan Lv, Junyu Dong, *Member, IEEE,* Yuezun Li, *Member, IEEE,* Sheng Chen, *Life Fellow, IEEE* Hui Yu, *Senior Member, IEEE,* Shu Zhang, Wenhan Wang

***Abstract*—Despite recent advances in stereo matching, the extension to intricate underwater settings remains unexplored, primarily owing to: 1) the reduced visibility, low contrast, and other adverse effects of underwater images; 2) the difficulty in obtaining ground truth data for training deep learning models, i.e. simultaneously capturing an image and estimating its corresponding pixel-wise depth information in underwater environments. To enable further advance in underwater stereo matching, we introduce a large synthetic dataset called UWStereo. Our dataset includes 29,568 synthetic stereo image pairs with dense and accurate disparity annotations for left view. We design four distinct underwater scenes filled with diverse objects such as corals, ships and robots. We also induce additional variations in camera model, lighting, and environmental effects. In comparison with existing underwater datasets, UWStereo is superior in terms of scale, variation, annotation, and photo-realistic image quality. To substantiate the efficacy of the UWStereo dataset, we undertake a comprehensive evaluation compared with eleven state-of-the-art algorithms as benchmarks. The results indicate that current models still struggle to generalize to new domains. Hence, we design a new strategy that learns to reconstruct cross domain masked images before stereo matching training and integrate a cross view attention enhancement module that aggregates long-range content information to enhance the generalization ability.**

***Index Terms*—Underwater stereo matching dataset Stereo matching Masked image learning**

I. Introduction

TOWARD building an intelligent agent in real world, the ability of visual understanding occupies a significant position in the whole blueprint, especially for parsing 3D scene structure, since it helps the agent locate itself and interact with surroundings [1], [2]. As a primary task for restoring spatial geometry from 2D images, stereo matching have been attracting more and more interests in recent years [3]–[5], and facilitating the development of many down-stream tasks like Multi-view Stereo [6], [7], RGBD-SLAM [8], 3D reconstruction [9], and etc.

Existing methods show two main branches for the development of stereo matching models: one focuses on direct disparity prediction through cost aggregation extension, exemplified by works such as [3], [10]–[12], while the other employs iterative optimization to refine disparity estimates, demonstrated in contributions like [4], [5], [13], [14]. Most of them adhere to the simulation-to-real (Sim2Real) training paradigm to avoid the data scarcity. This paradigm involves initially training the model on extensive synthetic datasets, *e.g.* SceneFlow [15] and Nerf-Stereo [16], for depth recovery, followed by fine-tuning on real-world datasets like KITTI [17], [18], MiddleBury2014 [19], and ETH 3D [20]. This training approach has been proven to be effective (left part of Fig. 1), yet its application in underwater settings, as discussed in [21]–[24], faces considerable challenges.

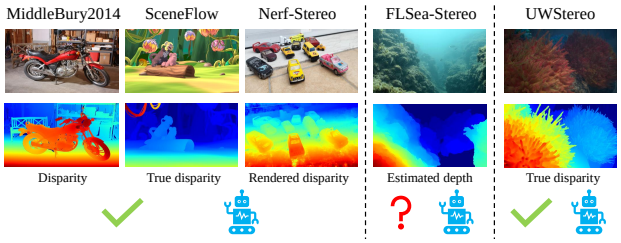

Fig. 1. Illustration of the dilemma for underwater stereo matching. *Left*: With sufficient datasets, stereo matching models can be easily trained, evaluated, and applied on aquatic environments. *Middle*: The accurate depth information is hard to acquired in real underwater scenes. *Right*: Our UWStereo is able to provide accurate depth information for all pixels and synthesize photo-realistic underwater images.

Basically, the quality of images captured in underwater scenes is significantly degraded due to factors like scattering, light absorption, and refraction [28]. These issues result in reduced visibility, low contrast, and various adverse effects, thereby limiting the practical utility of underwater data in oceanic engineering applications. Additionally, the lack of effective sensors and solutions for accurately estimating depth information in underwater environments makes it impossible to obtain images with corresponding ground truth disparity annotations. These formidable challenges underscore the current predicament in underwater stereo matching research [29]. As illustrated in the middle portion of Fig. 1, the estimated depth map, serving as the ground truth in the FLSea-Stereo dataset [23], exhibits compromised accuracy when compared to analogous data in existing datasets.

To address the challenges of scarcity of underwater stereo matching data, we present a novel dataset named **UWStereo** in this paper. To simulate intricate underwater environments, we

Corresponding authors: Yuezun Li and Junyu Dong

Q Lv, J Dong, Y Li, S Zhang, and W Wang are with the Department of Computer Science and Technology, Ocean University of China, Qingdao, Shandong Province, 266100 China.

S. Chen is with School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K., and also with the College of Computer Science and Technology, Ocean University of China, Qingdao 266100, China (E-mail: sqc@ecs.soton.ac.uk).

H. Yu is with the cSCAN Centre, University of Glasgow, G12 8QB, UK, U.K. (E-mail: Hui.Yu@glasgow.ac.uk).

TABLE I
STATISTICS OF EXISTING STEREO MATCHING DATASETS. THE UPPER PART PRESENTS EXISTING REAL-WORLD STEREO DATASETS, WHILE THE LOWER PART PRESENTS SYNTHETIC DATASETS. THE REAL-WORLD DATASETS ARE CONSTRAINED IN TERMS OF DATA AMOUNT AND SUFFERS FROM DEGRADED DISPARITY ANNOTATIONS. IN CONTRAST, THE SYNTHETIC DATASET CAN OFFER SUFFICIENT DATA AMOUNT WITH HIGHLY ACCURATE DISPARITY ANNOTATIONS. THE DATASET PROPOSED IN THIS STUDY PROVIDES A LARGE-SCALE, TRAINABLE DATASET FOR UNDERWATER STEREO MATCHING, FACILITATING FURTHER RESEARCH IN UNDERWATER DEPTH ESTIMATION.

Dataset	Training Images	Testing Images	Resolution	Stereo Disparity	Underwater	Baseline	Focal Length
KITTI-2012 [17]	194	195	1226×370	Sparse	No	-	-
KITTI-2015 [18]	200	200	1242×375	Sparse	No	-	-
MiddleBury2014 [19]	23	10	2880×1988	Dense	No	-	-
Nerf-Stereo [16]	65148	-	1160×522	Rendered	No	$0.5, 0.3, 0.1(cm)$	-
HIMB [21]	4047	15	-	No	Yes	-	-
FLSea-Stereo [23]	7337	-	1280×720	Estimated Depth	Yes	-	≈ 1800
UWSLAM [25]	3110	-	1355×1002	No	Yes	-	1273.77
MPI Sintel [26]	1064	564	1024×436	Dense	No	$10(unit)$	-
VAROS [27]	4713	-	1280×720	No	Yes	-	-
SceneFLow [15]	35454	4370	960×540	Dense	No	$1(unit)$	$1050, 450$
UWStereo	26612	2956	1280×720	Dense	Yes	$6, 12, 18, 24, 30(unit)$	1400

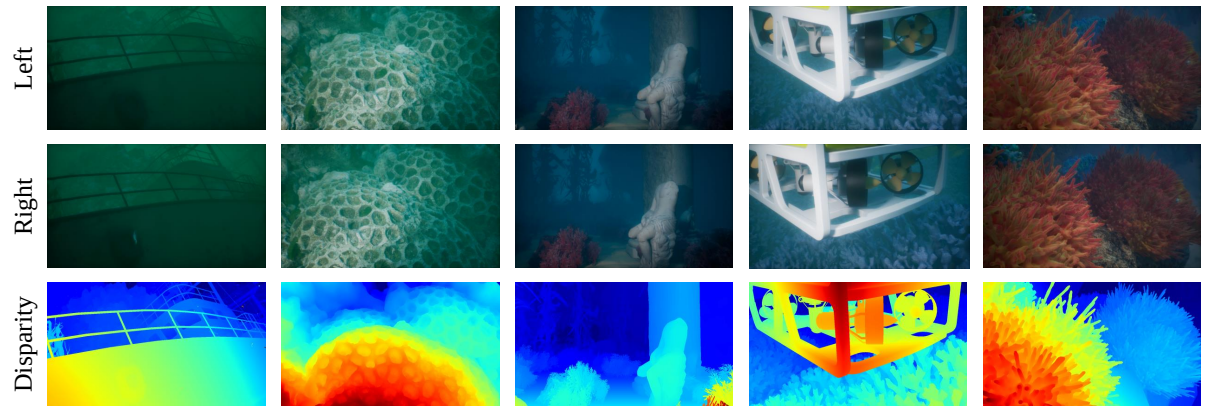

Fig. 2. Stereo image examples from the UWStereo dataset. The proposed dataset encompasses a diverse range of underwater objects, including ships, reefs, sculptures, robots, and corals, providing abundant training data for underwater disparity estimation.

utilize Unreal Engine 5 (UE5[1]) to create four distinct synthetic scenes, namely coral, default, industry, and ship, and fill them with diverse objects including corals, rocks, ships, and etc. In detail, the UWStereo comprises 29,568 stereo image pairs with accurate disparity annotations. Like MPI Sintel [26] and Nerf-Stereo [16], we also incorporate extra variations in camera baseline, light sources, and other environmental effects. Compared to existing datasets, UWStereo is the first synthetic dataset toward underwater scenes and has several advantages: 1) it has a large number of images and low redundancy; 2) it contains rich samples at a close view to underwater objects for underwater scenes; 3) the pixel-level annotations are dense and accurate. A detailed comparison between UWStereo and other existing datasets is presented in Table I and we show some examples in Fig. 2.

Leveraging this dataset, we conduct an extensive benchmark by selecting eleven recently developed stereo matching algorithms and retraining them. The results underscore the superiority of iterative optimization methods over other approaches. We further evaluate the generalization ability of these models. The comparison reveals that existing stereo matching methods are still hard to generalize to underwater scenes. To overcome this, we first refine an iterative stereo matching model by integrating a cross view attention enhancement module to enable the model to aggregate long-range content information from stereo images. Then, we draw inspiration from masked image learning and design a new training strategy by employing a paired masked image reconstruction pretraining task before stereo matching training. The results indicate that recovering paired masked images with a small mask ratio will enhance the generalization ability. We hope our work can inspire further research interests for underwater stereo matching and other down-stream tasks. The contributions of this paper are summarized as:

1) We introduce a large synthetic stereo matching dataset

[1]UE5: https://www.unrealengine.com/

containing 29,568 stereo images with dense and accurate disparity annotations, aiming to facilitate the researches toward developing stereo matching networks for real-world underwater scenes.
2) We select and retrain eleven recent stereo matching algorithms for the benchmarking purpose and perform evaluation for their generalization ability.
3) To enhance the generalization ability between terrestrial and underwater synthetic datasets, we induce a cross view enhancement module that enable the model to aggregate long-range context information, and design a new strategy that couples the stereo matching with a paired masked image reconstruction pretraining stage.

The remainder of this paper is organized as follows. In Section II, some related works is briefly introduced. Section III presents the details of synthesizing the underwater stereo matching dataset. Section VI demonstrates the training strategy for developing generalized stereo matching network. Section V contains the experimental results and discussions about the value of the proposed dataset. Finally, Section VI concludes this paper.

II. Related Works

A. Stereo Matching

Traditional stereo matching algorithms typically consist of four key steps including matching cost computation, cost aggregation, optimization, and disparity refinement [30]. With the success of deep neural networks, early attempts focused on how to compute the accurate matching cost [31], [32]. Further studies turned to consider the post-processing of the disparity estimation [33], [34]. Recently, many works had emerged with an efficient end-to-end structure. One prominent way is to employ 3D convolutional networks to regularize and filter the cost volume to improve the representative ability of the cost volume [3], [10]–[12], [35]–[37]. Differently, inspired by the optical flow estimation [38], another branch introduced a novel iterative optimization strategy to iteratively optimize the disparity field using recurrent convolutional structure [4], [5], [13], [14]. Specifically, RAFT-Stereo [4] firstly introduced the multi-level GRU units to update the disparity map. Then, IGEV [5] extended RAFT-Stereo with a combined geometry encoding volume, which provides concise initial disparity for updating, rather than starting with a blank map. HSMNet [13] proposed a decouple LSTM module to keep high-frequency information during the iterative updating stage. CREStereo [14] proposed a cascaded recurrent network to independently refine disparities in different cascade level.

Notably, some works tried to incorporate self-supervised learning into the framework of stereo matching [16], [39], [40]. Nerf-Stereo [16] rendered a large amount of image triplets by leveraging neural radiance field (Nerf) [41] and developed triplet photometric loss and rendered disparity loss to train the model with self-supervised signals. Rao *et al.* [39] formed a multi-task learning paradigm by simultaneously training the model with stereo matching and masked image reconstruction. Croco-Stereo [40] performed self-supervised cross-view completion pretraining to encourage the model to learn dense geometric information.

B. Stereo Matching Datasets

In the field of stereo matching, several widely used datasets, such as Middlebury2014 [19], KITTI [17], [18], and ETH3D [20], had become important benchmarks in research. These datasets had advanced the development and optimization of algorithms by providing high-precision depth information. However, a common limitation of these datasets is their relatively small size, primarily due to the challenges of obtaining pixel-wise depth information. In real-world scenarios, manually annotating depth information is not only time-consuming and labor-intensive but also prone to environmental variations such as lighting, reflections, and occlusions, which further complicate and increase the cost of annotation.

To address these challenges, researchers had begun exploring the emerging simulation-to-real paradigm. This approach leverages rendered images and depth information in synthetic scenes to overcome the limitations of scarce real-world data. MPI Sintel [26] is one of the pioneers in this field, synthesizing over 1,000 stereo images with depth maps from open-source animation clips. Following this, the SceneFlow [15] dataset marked a significant advancement in the field by introducing a synthetic dataset containing more than 35,000 stereo frames, involving over 30,000 objects extracted from ShapeNet [42]. The diverse appearances of these objects greatly enrich the variety of training data, effectively addressing the challenge of training models with limited real-world data.

These synthetic datasets not only surpass traditional datasets in terms of data volume but also provide highly controllable environments, allowing researchers to systematically explore the performance of various algorithms. However, despite the clear advantages of synthetic datasets in enriching data diversity, the domain gap between synthetic and real-world scenarios remains a critical issue. Consequently, many studies are now focusing on how to transfer the knowledge learned from synthetic data to real-world scenes, further narrowing the gap between simulation and reality. These efforts not only help enhance the robustness of models in real-world environments but also lay the foundation for the future development of more broadly applicable stereo matching systems.

C. Underwater Image Datasets

Several datasets have been developed to support various vision tasks within the underwater domain. For example, the UIEB [24] dataset was introduced specifically for underwater image enhancement, providing a benchmark for improving the visual quality of underwater imagery. The SUIM [43] dataset goes further by offering underwater images with pixel-level semantic annotations, making it useful for tasks such as object detection and segmentation in underwater environments. Additionally, the SQUID [22] and Sea-thru [44] datasets provide raw underwater images along with estimated distance maps, which are crucial for depth estimation and 3D reconstruction tasks.

Building on these efforts, the FLSea dataset [23] advances the field by offering both monocular and binocular images paired with estimated depth information, making it a valuable resource for stereo matching and depth prediction tasks. The

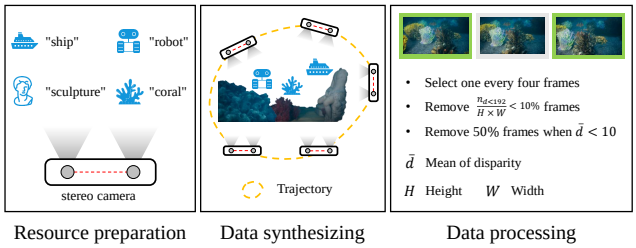

Fig. 3. Synthesizing workflow. We first collected a large number of model assets and designed a virtual camera based on a stereo camera system. Subsequently, we edited camera motion trajectories within the scenes to generate synthetic data. Finally, we design data processing rules to reduce data redundancy.

VAROS dataset [27], another significant contribution, extends the synthetic pipeline for underwater environments by providing 4,713 synthetic monocular images with corresponding depth maps and surface normal information. These datasets collectively highlight the importance of recovering depth information from underwater images for various vision tasks, including image restoration, reconstruction, and 3D modeling.

Despite these advancements, existing underwater datasets still face several limitations. Many datasets, such as UIEB, SQUID, and HIMB, are relatively small in scale, which limits the diversity and complexity of the scenes they cover. Datasets like VAROS and FLSea, while larger, often feature simpler scene structures, which may not fully capture the complexity of real-world underwater environments. Additionally, some datasets, including SQUID, FLSea, and Sea-thru Nerf [44], suffer from compromised or less accurate distance annotations, which can impact the performance of depth-related tasks.

To address these challenges, we propose a new synthetic stereo matching dataset specifically designed for underwater scenes. Our dataset is characterized by its substantial size, diverse scene variations, and highly accurate annotations, setting it apart from previous datasets. By offering a more comprehensive and reliable resource, our dataset aims to stimulate further research in the field of underwater stereo matching, ultimately contributing to more robust and effective underwater vision systems.

III. The UWStereo Dataset

A. The Importance of The UWStereo

The core challenge of underwater stereo matching lies in obtaining precise pixel-wise depth annotations, which significantly increases the difficulty of constructing trainable datasets. Early real-world datasets were limited by the inherent challenges of depth acquisition, resulting in scarce training samples and noisy disparity annotations [17]–[19], as shown in Table I. To address this issue, subsequent research leveraged synthetic data with high-precision depth annotations, leading to substantial improvements in the accuracy of disparity estimation in real-world scenarios [15], [26].

However, the lack of dedicated datasets remains a major bottleneck in advancing research on underwater stereo matching. This limitation primarily arises from the absence of high-precision underwater depth sensing technologies or reliable depth estimation algorithms. On one hand, underwater optical imaging suffers from severe degradation caused by scattering and absorption, making it challenging for conventional stereo matching algorithms to generate continuous and accurate disparity estimations [29], [45]. On the other hand, although acoustic-based depth sensing devices are effective over longer distances, they typically operate beyond the visual range of optical sensors, thereby posing significant challenges for multi-sensor fusion [46]–[48].

Given these challenges, the development of high-quality synthetic underwater stereo matching datasets becomes critically important. Synthetic data not only enables the generation of large-scale training samples with accurate ground-truth disparity annotations but also facilitates progress in underwater stereo matching, 3D reconstruction, and other related domains.

B. Synthesizing Workflow

Previous works synthesized underwater images by using Blender[2] rendering engine [15], [27] or generative models like Nerf [16] and Generative Adversarial Network (GAN) [49]. However, it's not trivial to capture underwater stereo images with accurate dense depth information for above methods, as the insufficient ability of generative models in rendering depth information and the distribution discrepancy between synthetic and real-world images. Thus, toward providing dense and accurate depth information for underwater scenes, we employ UE5 as the simulator because of its powerful ability in simulating real-world visual effects [3]. The synthesizing workflow contains three steps and is shown in Fig. 3.

Resource preparation: MPI Sintel [26] utilizes open-source animation sequences as simulation environments, while SceneFlow [15] uses digital objects provided by some animations or ShapeNet [42]. These virtual assets empower the synthetic data to provide sufficient variation of disparity distributions. Inspired by these works, we invested substantial efforts in collecting numerous virtual resources that could potentially appear in underwater scenes to enrich the disparity distributions. The collected resources includes *sunken shipwrecks, scanned coral, simple sculpture models, dilapidated cars, wind turbine generators, and underwater robots*. Based on these diverse objects, we constructed four distinct underwater scenes named coral, default, industry, and ship respectively, denoting as S_i.

we also design another crucial component, a binocular camera, simulated by using two separate camera components. Similar to Sceneflow, we set the film back size to $32mm \times 18mm$ and focus plane $f_p = 150$ units in simulator. The two camera components have the same intrinsic parameters, denoting as:

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1400 & 0 & 640 \\ 0 & 1400 & 360 \\ 0 & 0 & 0 \end{bmatrix}. \quad (1)$$

We only consider translation between two camera, where the translation vector is $\mathbf{t} = [b, 0, 0]$. Differently, we regard the

[2]Blender: https://www.blender.org/

[3]We use an open source plugin named EasySynth https://github.com/ydrive/EasySynth for data generation

TABLE II
DETAILS OF THE UWSTEREO.

Dataset	Total Images	Training Images	Testing Images
coral	4,494	4,045	449
default	4,720	4,248	472
industry	9,230	8,307	923
ship	11,124	10,012	1,112
all	29,568	26,612	2,956

baseline b as a variable rather than a fixed value, which is also examined in DSMNet [35] or Nerf-Stereo [16].

Data synthesizing: Having prepared the requisite virtual assets, we proceed to the data synthesizing phase. In the context of each S_i, an initial camera position p is selected in proximity to visible objects, aiming to facilitate the subsequent camera motion. We then create an animation sequence along with an manually edited motion trajectory that emulates the movement of a binocular camera. Notably, we observed that the sequence lengths in both MPI Sintel (up to 50) and SceneFlow (up to 300) datasets are relatively limited. In contrast, we set the frame rate to 15 frame per second and edit a long enough sequence with at least 1500 frames for each scene. The ground truth disparity is generated according to:

$$disparity = \frac{b \cdot f_x}{depth}, \tag{2}$$

where b and f_x can be retrieved from the camera parameters, and depth information is synthesized during moving.

Furthermore, to introduce additional variations, we exercise control over synthesized data by manipulating key factors such as camera baseline, light sources, and volumetric fog. Specifically, under the assumption of camera sensor has a restricted visual range in underwater scenes, we assign the baseline b to a certain value within the set $\{6, 12, 18, 24, 30\}$ to generate balance disparity distribution. The utilization of additional light sources, represented as $E_l \in \{0, 1\}$, is also considered since binocular camera sensors are commonly integrated into remotely operated vehicles (ROVs) equipped with additional lighting sources. Moreover, volumetric fog, a significant feature of UE5, is harnessed to simulate underwater light scattering conditions. Within this context, we diversify the visual attributes of the synthesized data by manipulating the density and color of the volumetric fog. The density of volumetric fog is specified as $E_d \in \{1.0, 2.0\}$, and the color variation includes blue and green, denoted as $E_c \in \{blue, green\}$. These diverse settings yield the synthesis of approximately $60,000$ consecutive stereo matching data instances derived from a single trajectory within a scene, calculated as $1500 \times 5 \times 2 \times 2 \times 2 = 60,000$. Given that we have incorporated four distinct scenes, the cumulative volume of synthesized data characterized by precise depth information approximates to around $240,000$ instances.

Data processing: Previous datasets employed continuous frames to guarantee the continuity. Despite the large size, those synthetic data are redundant in the temporal dimension leading to longer training costs. Another important problem is the synthetic disparity distribution may be imbalanced, as the camera motion trajectory need to be manually edited, which inevitably induces motion bias.

Considering to resolve the above perturbations, we decide to design a further data processing step to cleanup the synthetic data. Concretely, we design three rules as: 1. Set interval to 4 to sample a frame. 2. Remove 50% of the frames which satisfy $\bar{d} < 10$, where $\bar{d}$ represents the mean of disparities. 3. Remove the data when $\frac{n_{disp>192}}{H \times W} > 10\%$, where $n_{disp>192}$ represents the number of pixels that the disparity exceeds 192. H and W denote the height and width of the image. These rules help remove excessive imbalanced data, facilitating the convergence of training (the comparison of disparity distributions between our UWStereo and other datasets is presented in supplementary material.).

C. Dataset Details

The proposed UWStereo dataset contains a total of 29,568 stereo matching image pairs oriented to the underwater environment. All data contain dense disparity annotations for left view. The details of the dataset are presented in Table II. We randomly selected 10% of each scene to serve as testing images and the remaining 90% as training data. In comparison with existing stereo matching datasets, UWStereo is firstly designed for intricate underwater scenes and contains several advantages: 1) it has large number of images and low redundancy; 2) it contains rich samples at close view to underwater objects instead the distant views for underwater scenes; 3) the pixel-level annotations are dense and accurate.

IV. THE STRATEGY

We notice that some recent works [39], [40], [50] effectively relate the masked representation learning [51] to the task of cross-image matching, which suggests that pretraining the model with masked image reconstruction objectives improves the generalization ability. Inspired by these works, we aim to build up a paired masked image reconstruction pretext task on both terrestrial and underwater images to study whether the masked representation can enhance the generalization ability.

Network Structure We build our model based the IGEV [5]. One difference is that we insert a new cross view enhancement (CVE) module behind the feature extractor, since we observe the correspondence between two different views helps the model to perceive geometry structure of scene [50], [52]. Concretely, the CVE module consists of a Multi-Head Attention block (MHA) and n LoFTR blocks denoting as T_θ and T_{loftr} respectively. T_θ is used to adjust the CNN-extracted representation to patch-wise style. Following PMatch [50], we also add positional embeddings before feeding to T_{loftr} to encourage the model to track the positional information for each patch. Then, since the LoFTR block is able to aggregate long-range context information by self and cross attention layers, we employ multiple T_{loftr} layers to focus on searching cross view correspondence to enhance the feature representation. The left top part of Fig.4 illustrate the structure of CVE. The network structures for pretraining and stereo matching training are also presented in the left bottom and right parts of Fig. 4. The cost aggregation module is kept during pretraining, as the matching cost can also provide guidance to recover paired images. While, we drop the iterative

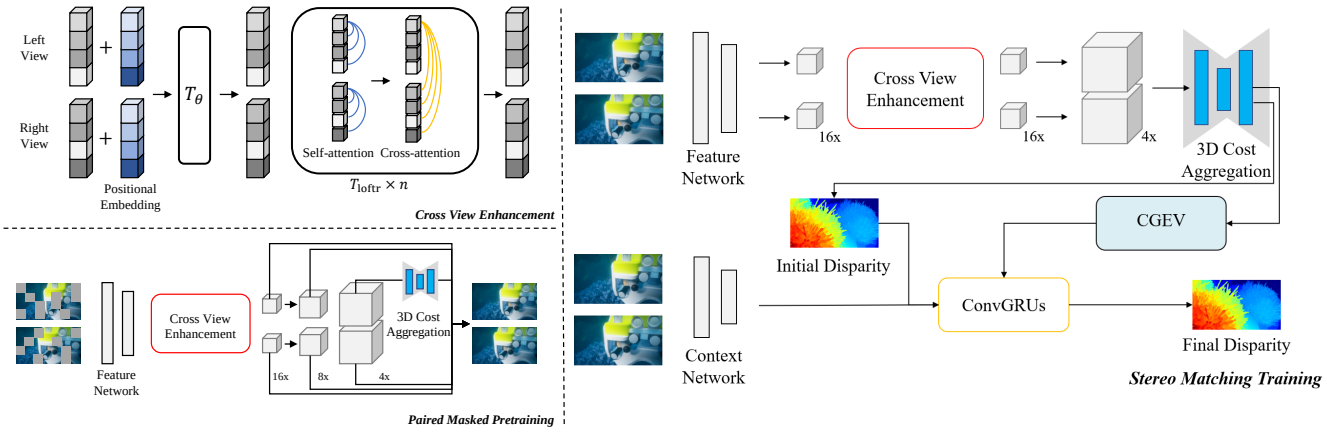

Fig. 4. *Left top*: The structure of Cross View Enhancement (CVE) module. *Left bottom*: The network structure employed during pretraining. *Right*: The network structure for stereo matching training.

updating module in pretraining stage, as it is not suit for paired masked image reconstruction pretext task. We design some lightweight decoders at different scales to reconstruct the masked images.

Pretraining setting Unlike Masked-CFNet [39] which introduces a multi-task learning framework, we follow PMatch [50] to form a self-supervised pretext task by reconstructing the paired images. Given left and right images denoting as x_l and x_r, we feed them to feature extractor to extract features at scale $s = 2$ denoting as $f_l^{s=2}$ and $f_r^{s=2}$. We then use two predefined mask ratio r_1 and r_2 to randomly generate masks, where the patch size is 32×32 instead a small mask size in Masked-CFNet. The masked feature can be represented as:

$$f_l^{s=2'} = f_l^{s=2} * (1 - w) + m * w, \tag{3}$$

where m represents the mask tokens and w is the corresponding mask. The masked features are then fed to the following modules to extract features at $s = 4$, $s = 8$, and $s = 16$. Furthermore, we reconstruct the input images by regressing the raw pixel values for feature at each scale. It should be noticed that the cost aggregation is appended for left feature at $s = 4$, since we want to keep it to be consistent with the stereo matching training. The reconstruction objective can be formed as:

$$\mathcal{L}_M = \sum_s \frac{1}{N_{mask}} (||x_l^s - x_l^{s'}||_1 + ||x_r^s - x_r^{s'}||_1), \tag{4}$$

where N_{mask} denotes the number of masked image patches. $x_l^{s'}$ and $x_r^{s'}$ denote the reconstructed images at a scale s.

Stereo Matching Training Following previous works [3]–[5], [13], we employ l_1 loss as the objective of stereo matching. Like IGEV [5], we supervise the network on both initial disparity and updated disparity map. Hence, the training loss can be defined as:

$$\begin{aligned} \mathcal{L}_T = & \, SmoothL1(d_{init} - d_{gt}) \\ & + \sum_{i=1}^{N_{iter}} \gamma^{N_{iter}-i} ||d_i - d_{gt}||_1, \end{aligned} \tag{5}$$

where $\gamma = 0.9$ and N_{iter} denotes the number of iteration.

V. Experiments

A. Experimental Setting

Compared Methods For benchmarking purposes, we have selected eleven stereo matching methods, which are PSMNet [3], GwcNet [12], ACVNet [10], AANet [11], IGEV [5], RAFT-Stereo [57], CREStereo [14], HSMNet [13], LEAStereo [53], MoCha-Stereo [55], and Selective-Stereo [56]. Additionally, we have included two methods specifically designed for generalized stereo matching models, namely GraftNet [57] and DSMNet [35], in our comparative analysis. Nerf-Stereo [16] and Croco-Stereo [40] were also taken into account as their excellent generalization capabilities.

Implementation details For our model, we use four LoFTR layers to construct the CVE module. During the pretraining stage, we set $r_1 = r_2 = 0.5$ to generate masks. The training set consists of KITTI [17], [18], Middlebury2014 [19], SceneFlow [15], and the proposed UWStereo. We train the model for 100k steps with a batch size of 16 on a single RTX 4080 GPU. The crop size is set to 320×320, and the learning rate is set to 0.0001. For stereo matching training, the learning rate is set to 0.0002, and the model is trained for 200k steps with a batch size of 4. We randomly crop images to 320×736 and apply the same data augmentation strategy as IGEV. Following previous methods, we use 22 and 32 update iterations for training and evaluation, respectively.

Metrics The widely used end-point-error (EPE) and threshold error rate (>3px) were used as metrics for evaluation. We set the threshold to 3px for all experiments. We also employed several image quality assessment algorithms to compare the image quality of our proposed synthetic data with the previous datasets. The image quality comparison is presented in supplementary material.

B. Results and Analysis

Benchmarking For benchmarking purposes, we retrain eleven recently developed models on the UWStereo dataset and present the results in Table III. We observe that the Coral scene is the most challenging, while the Default scene is the easiest. This is mainly because the objects in the Coral scene are scanned from real-world environments and often exhibit

TABLE III
COMPARISON WITH STATE-OF-THE-ART METHODS.WE SELECT ELEVEN STATE-OF-THE-ART STEREO MATCHING MODELS TO CONDUCT BENCHMARK EXPERIMENTS. THE UPPER PART REPRESENTS COST-FILTERING-BASED ALGORITHMS, WHILE THE LOWER PART CORRESPONDS TO ITERATIVE OPTIMIZATION METHODS. ITERATIVE OPTIMIZATION ALGORITHMS DEMONSTRATE GREATER STABILITY ACROSS VARIOUS SCENARIOS COMPARED TO COST-FILTERING-BASED APPROACHES.

Method	Coral		Default		Industry		Ship		All	
	EPE	>3px	EPE	>3px	EPE	>3px	EPE	>3px	EPE	>3px
LEAStereo [53]	2.32	12.02	0.87	4.35	1.83	10.02	1.94	8.12	1.49	7.06
PSMNet [3]	2.65	13.88	1.26	5.96	1.68	8.42	1.62	7.50	1.32	6.75
AANet [54]	2.61	11.92	0.97	4.04	1.31	6.65	1.52	7.20	1.27	5.81
GwcNet [12]	2.25	10.27	0.71	2.88	1.05	4.43	1.06	4.27	1.07	4.23
ACVNet [10]	1.93	7.90	0.50	1.63	0.81	3.28	0.79	2.95	0.85	**3.13**
RAFT-Stereo [4]	1.55	7.86	0.43	1.57	0.66	3.22	0.67	3.24	0.93	4.64
CREStrereo [14]	2.07	12.47	0.37	1.34	-	-	**0.57**	**2.55**	-	-
HSMNet [13]	**1.09**	**6.26**	**0.36**	**1.29**	**0.44**	**1.94**	0.59	2.79	**0.68**	3.25
MoCha-Stereo [55]	1.53	19.49	0.51	10.26	0.70	12.83	0.75	14.83	0.94	15.11
IGEV [5]	**1.49**	**7.53**	0.50	1.51	0.51	2.12	0.68	3.23	0.73	3.45
IGEV [5] + Ours	1.53	7.91	**0.39**	**1.45**	**0.48**	**2.06**	**0.62**	**3.01**	**0.69**	**3.44**
Selective-Stereo [56]	1.57	8.44	0.41	1.69	0.53	2.47	0.66	3.24	0.80	4.11
Selective-Stereo [56] + Ours	1.59	8.48	**-**	**-**	**-**	**-**	-	-	**0.78**	**3.86**

TABLE IV
GENERALIZATION ABILITY COMPARISON. 'SF' REFERS TO THE SCENEFLOW DATASET. '*' MEANS THAT A MIXTURE OF EXISTING DATASETS ARE USED. THE BEST PERFORMANCE OF EACH PART IS SHOWN IN BOLD FONT.

Method	Training Set	EPE	Coral		Default		Industry		Ship		All	
			EPE	>3px	EPE	>3px	EPE	>3px	EPE	>3px	EPE	>3px
LEAStereo [53]	SF	0.78	2.50	12.99	**1.42**	**5.81**	**1.88**	**9.22**	**3.56**	**14.52**	**2.53**	**11.26**
PSMNet [3]	SF	1.09	2.68	13.17	2.36	7.21	2.85	10.58	4.95	15.70	3.54	12.36
AANet [11]	SF	0.87	2.94	12.74	**1.42**	7.70	**1.88**	12.82	**3.56**	19.25	**2.53**	14.34
GwcNet [12]	SF	0.76	**2.26**	**11.50**	1.62	6.70	1.92	9.41	4.49	16.52	2.89	11.97
ACVNet [10]	SF	**0.48**	2.88	13.70	3.40	8.24	3.46	11.41	7.45	18.48	4.86	13.91
RAFT-Stereo [4]	SF	0.61	2.23	11.60	1.12	6.25	2.23	8.96	2.88	13.10	2.30	10.49
CREStrereo [14]	*	-	**2.10**	**9.75**	1.19	6.05	**1.19**	**6.25**	3.48	12.39	**2.19**	**9.06**
HSMNet [13]	SF	0.48	2.61	13.97	5.38	10.47	4.18	12.33	10.61	21.52	6.55	15.74
Selective-Stereo [56]	SF	0.44	2.31	13.05	**1.07**	**5.66**	2.86	9.20	**2.74**	**12.29**	2.44	10.34
MoCha-Stereo [55]	SF	**0.41**	2.90	16.47	1.90	12.76	3.35	18.62	4.83	22.87	3.62	17.85
IGEV [5]	SF	0.47	**2.29**	**12.60**	**1.09**	**5.56**	1.99	7.58	**2.65**	12.28	**2.14**	9.74
IGEV [5] + Ours	SF	**0.46**	2.38	13.27	1.18	6.03	**1.56**	**6.75**	2.73	**12.16**	**2.14**	**9.45**
Graft-PSMNet [57]	SF	-	3.27	15.31	2.87	3.23	3.91	10.48	6.37	16.13	4.57	12.82
Croco-Stereo [40]	*	-	2.43	12.52	2.19	9.40	2.03	9.40	4.51	16.12	3.05	12.21
Nerf-PSMNet [16]	Nerf-rendered	-	2.48	11.87	2.26	7.82	2.07	9.47	4.63	15.82	3.13	11.96
Nerf-RAFT [16]	Nerf-rendered	-	2.29	**10.76**	1.41	5.74	1.95	8.12	3.46	12.49	2.48	9.78
DSMNet [35]	SF+Carla [58]	-	**2.19**	11.42	**1.00**	**4.56**	**1.39**	**6.89**	**2.36**	**10.78**	**1.81**	**8.67**

complex structures and diverse textures, which significantly increase the difficulty of disparity estimation.

The compared methods can be categorized into two main types: **cost-filtering-based methods** [3], [10]–[12], [53] and **iterative refinement methods** [4], [5], [13], [14], [55], [56]. Among the cost-filtering-based models, ACVNet [10] outperforms others by a large margin. For instance, when trained on the full UWStereo dataset, ACVNet achieves an EPE of 0.85, compared to 1.49 for LEAStereo. This improvement is mainly attributed to its attention-based cost volume construction, which enhances the extraction of matching-relevant information.

Furthermore, we find that iterative methods generally outperform cost-filtering-based methods across most scenes. This superiority is likely due to their ability to iteratively refine disparity predictions by retrieving and correcting uncertain regions based on local cost cues [4], [5], [14], [56]. Notably, HSMNet [13] achieves the best EPE in several scenes. Its performance can be attributed to the use of high-frequency features, which are more robust to the low image quality and blurry textures commonly found in underwater imagery.

In addition, our proposed model consistently outperforms IGEV in several scenarios. This demonstrates the effectiveness of our method in enhancing feature representations for more accurate cross-view feature matching in challenging underwater environments.

Generalization Benchmarking To investigate the generalization ability of stereo matching models between terrestrial

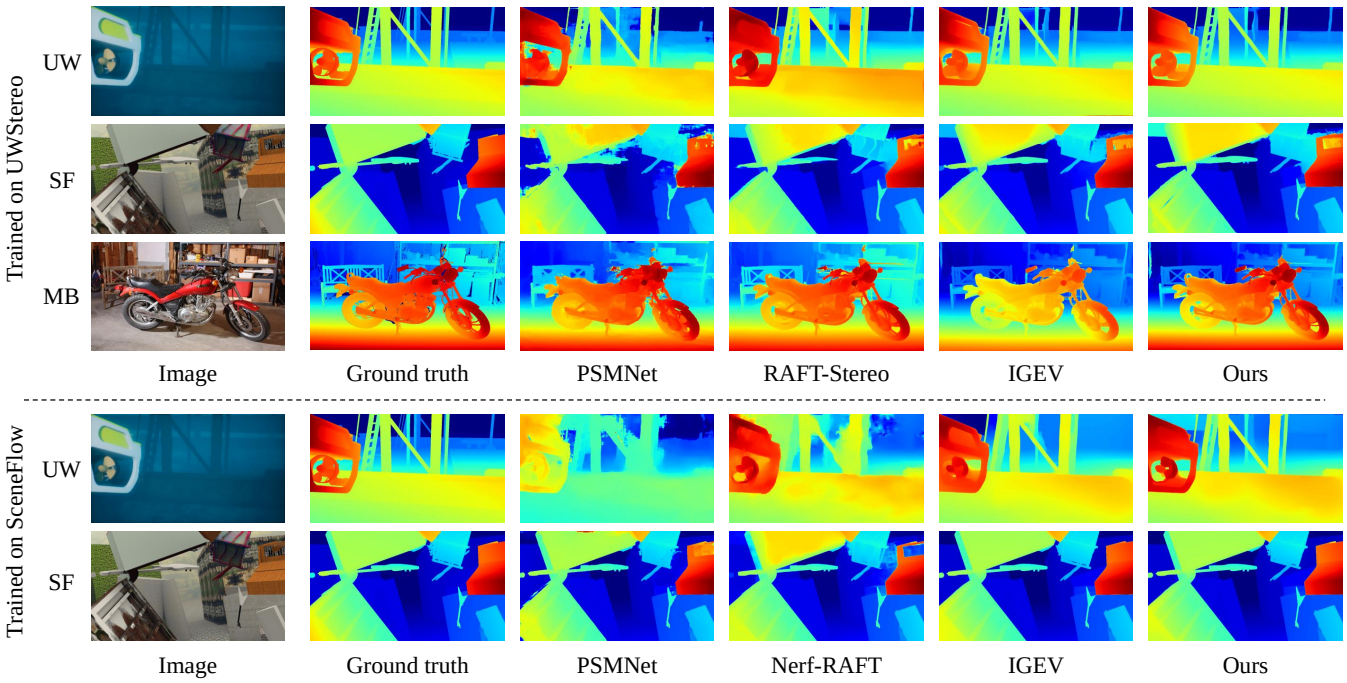

Fig. 5. Visualization results. "UW", "SF", and "MB" represent UWStereo, SceneFlow, and MiddleBury2014 respectively. **Top part:** the models are trained on UWStereo and evaluated on other datasets. **Bottom part:** the models are trained on SceneFlow and evaluated on UWStereo.

TABLE V
ABLATION STUDY FOR MASK RATIO.

Model	CVE	SceneFlow		UWStereo	
		EPE	>3px	EPE	>3px
SceneFlow → UWStereo					
IGEV		0.52	**2.65**	2.48	10.92
Baseline	✓	0.52	2.70	2.47	11.00
0.25	✓	0.52	2.72	2.21	10.18
0.5	✓	**0.52**	2.72	**2.15**	**9.95**
0.75	✓	0.52	2.75	2.97	11.29
UWStereo → SceneFlow					
IGEV		2.14	**8.00**	0.73	3.45
Baseline	✓	2.18	8.38	0.72	**3.29**
0.25	✓	2.21	8.81	0.73	3.55
0.5	✓	**2.06**	8.05	**0.69**	3.44
0.75	✓	2.30	8.52	0.74	3.48

TABLE VI
ABLATION STUDY FOR THE CVE MODULE.

Method	Training Set	EPE	All	
			EPE	>3px
IGEV [5]	SF	0.47	2.14	9.74
without CVE	SF	0.48	**2.10**	9.57
Ours	SF	**0.46**	2.14	**9.45**

and underwater synthetic data, we conduct a cross-domain evaluation using models pretrained on SceneFlow. Notably, we also include DSMNet [35], Graft-PSMNet [57], Nerf-RAFT [16], Nerf-PSMNet [16], and Croco-Stereo [40] in our comparison, as these models are specifically designed to improve generalization.

TABLE VII
EFFICIENCY COMPARISON ON 'CORAL' SCENE. ELFNET AND PCWNET ARE EVALUATED WITH $(368, 320)$ AND $(640, 960)$ RESOLUTION RESPECTIVELY. OTHER METHODS ARE EVALUATED ON $(736, 1280)$ RESOLUTION.

Model	EPE	>3px	Run-time
GwcNet [12]	2.25	10.27	0.28
ELFNet* [59]	2.97	15.36	1.08
PCWNet* [60]	2.98	15.33	0.42
ACVNet [10]	1.93	7.90	0.32
RAFT-Stereo [4]	1.55	7.86	0.56(0.35)
IGEV [5]	**1.49**	**7.53**	0.38(**0.22**)
Ours	1.53	7.91	0.50(0.35)

As shown in Table IV, most iterative methods exhibit better generalization performance than cost-filtering-based methods. However, HSMNet fails to generalize well to underwater scenes, achieving an EPE of 6.55 on the entire UWStereo dataset. This could be due to its high-frequency features being overfitted to the terrestrial domain.

Interestingly, models specifically designed for domain generalization do not always perform well when facing the domain gap between terrestrial and underwater environments. For instance, Graft-PSMNet obtains an EPE of 4.57 and a ¿3px error rate of 12.82 on all UWStereo scenes, which is even worse than the original PSMNet. In contrast, Nerf-PSMNet shows improved performance, reducing the EPE from 3.54 to 3.13, likely due to the large number of NeRF-rendered images used during training, which enhance generalization.

Among iterative optimization methods, IGEV demonstrates strong generalization performance. From the experimental

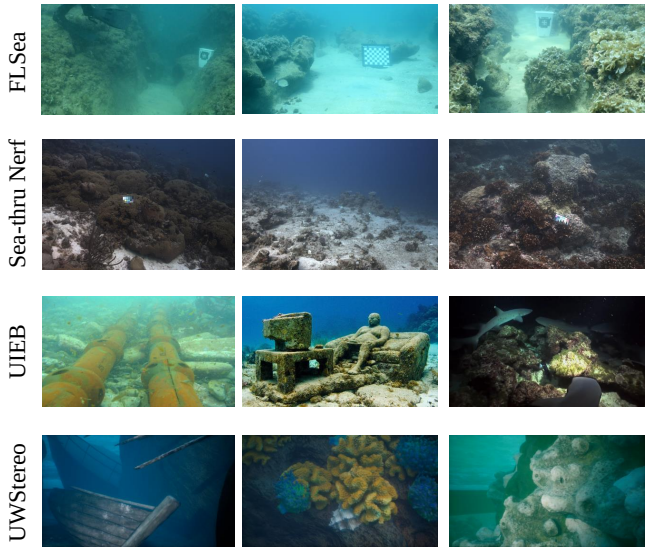

Fig. 6. Comparison with underwater datasets. Other datasets, such as FLSea and Sea-thru NeRF, primarily contain empty underwater scene data. In contrast, the proposed UWStereo dataset includes a greater number of close-range objects. Therefore, UWStereo is better suited for underwater object depth estimation tasks.

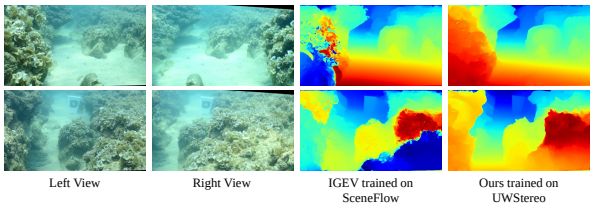

Fig. 7. Real-world qualitative evaluation on FLSea-Stereo *rock garden2* part. In real underwater scenarios, training with UWStereo can enhance the generalization performance of stereo matching models on underwater data.

results, we observe that our method achieves better generalization in certain scenarios (e.g., Industry, Ship, and All scenes) compared to IGEV, particularly in terms of the ¿3px error rate. Moreover, under the same training settings, our method achieves a slightly better EPE on the SceneFlow dataset than IGEV (0.46 *vs.* 0.47), which further demonstrates the effectiveness of our proposed self-supervised pretraining strategy in improving generalization across domains.

Furthermore, both DSMNet and CREStereo achieve impressive results on several underwater scenes. We speculate that this is because they are trained on synthetic data generated by the Unreal Engine simulator, such as Carla [58], which may share structural and visual characteristics with our UWStereo dataset. Lastly, we note the limited performance of Croco-Stereo, which suggests that the integration of self-supervised learning with stereo matching remains an open and underexplored direction for future research.

Visualization In Fig. 5, we present the visualization results of several stereo matching models by separately training the model on SceneFlow or UWStereo. In the top part, we observe that our method achieves more stable disparity estimation results when trained on UWStereo dataset than other compared methods. Furthermore, when only trained on SceneFlow (bottm part), our approach can still generalize well to underwater scene. These results demonstrate that the proposed self-supervised pretraing strategy is able to improve the cross-domain generalization ability for stereo matching networks.

TABLE VIII
REAL-WORLD QUANTITATIVE EVALUATION ON FLSEA-STEREO *rock garden2* PART.

Model	Training Set	Mean Error (m)
IGEV	SceneFlow	1.10
IGEV	UWStereo	1.22
Our	UWStereo	**0.87**

Efficiency In Table VII, we compare the efficiencies by feeding them with data with fixed $(736, 1280)$ resolution on one RTX 4080 GPU card. Due to the limitation of GPU memory, we feed ELFNet [59] and PCWNet [60] with $(368, 320)$ and $(640, 960)$ data. The run-time is averaged by performing 100 runs. For iterative methods, we report the metrics by using 32 and 16 updates. We observe that our model achieves comparable efficiency.

Ablation Study for the CVE Module. To assess the impact of the CVE module, we train the model on the SceneFlow dataset and evaluate its generalization performance on the UWStereo dataset. As shown in Table VI, we observe that pretraining without the CVE module leads to improved performance on both the EPE and >3px metrics in terms of the UWStereo. While the EPE on SceneFlow experiences a slight decline (e.g., 0.48 compared to 0.47), the overall trend suggests better generalization. Additionally, the model trained with the CVE module outperforms IGEV in terms of EPE on SceneFlow and the >3px metric on UWStereo. These results indicate that learning self-supervised representations through paired image reconstruction enhances generalization across diverse environments. Furthermore, the CVE module improves matching accuracy by capturing long-range cross-view correspondences.

Ablation Study for Mask Ratio. In Table V, we conduct an ablation study for the mask ratio by training the model on one domain and evaluating on another domain. Notably, we use small batch size for training than other methods in Table IV due to the hardware limitation. Hence the generalization performances will degenerate to some extent. We observe our model produces best generalization performances when setting $r_1 = r_2 = 0.5$. These demonstrate that an appropriate mask ratio encourages the model to learn correspondence between left and right views. However, a higher or lower one may hinder the model to generalize to another domain.

C. Discussion

The Practical Value of UWStereo. In Fig. 6, we present samples from previous underwater datsets and our UWStereo. We observe that most of our samples are synthesized with close views instead of containing many distant views like FLSea [23] or Sea-thru Nerf [61]. This implies the model trained with UWStereo may be suitable for the reconstruction of underwater objects instead of blank underwater scenes.

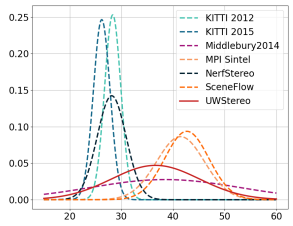

Fig. 8. MUSIQ(↑).

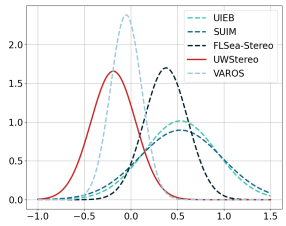

Fig. 9. UIQM(↑).

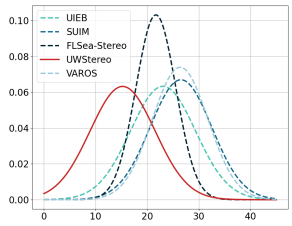

Fig. 10. UCIQE(↑).

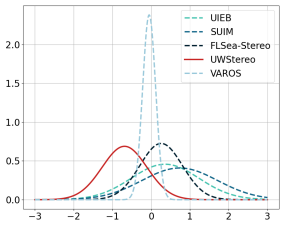

Fig. 11. URanker(↑).

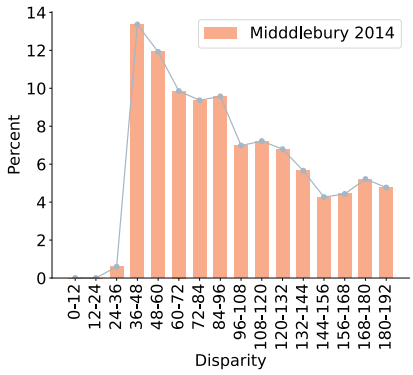

Fig. 12. MiddleBury2014.

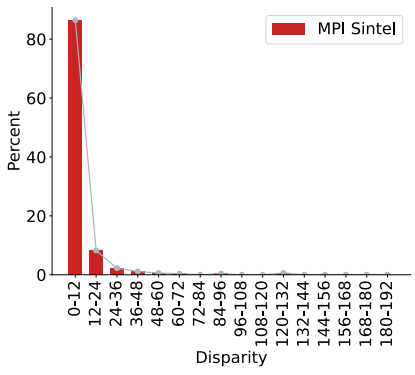

Fig. 13. MPI Sintel.

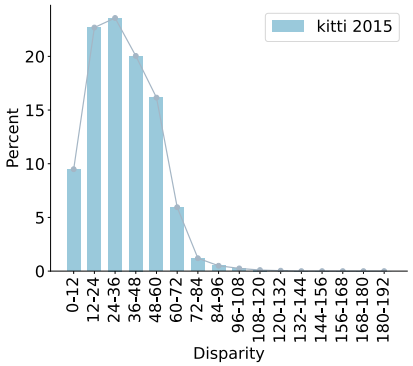

Fig. 14. KITTI 2015.

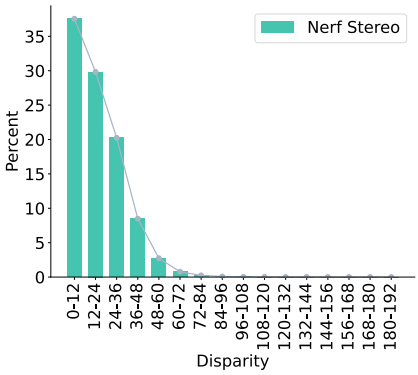

Fig. 15. Nerf-Stereo.

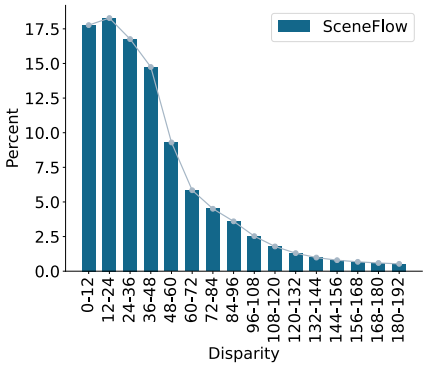

Fig. 16. SceneFlow.

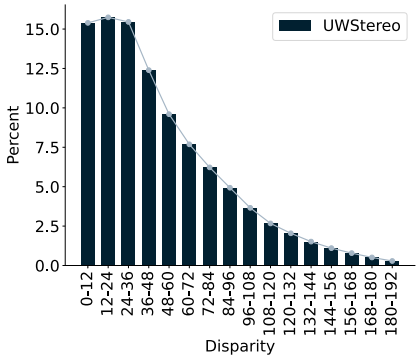

Fig. 17. UWStereo.

For verification, we further conduct real-world quantitative and qualitative evaluations on part of FLSea-Stereo dataset, as the dataset provides estimated depth information and real-world underwater stereo images. We select the *rock garden2* part for evaluation. In detail, this split totally contains 305 image pairs. As seen in Table VIII, the results show that our method achieves a better mean error than IGEV trained with SceneFlow dataset, e.g. 0.87 v.s. 1.02. Meanwhile, when trained with the UWStereo dataset, IGEV performs worse than that trained with SceneFlow. This is mainly because simulated underwater effects hinder the learning of stereo matching models. Whereas, with our proposed pretraining strategy and CVE, the generalization ability of model for real-world underwater scene improves considerably, e.g. 1.22 v.s. 0.87. We also visualize the stereo matching results in Fig. 7, where the results suggest that our method performs more stable than IGEV trained on SceneFlow. These demonstrates that the UWStereo dataset and our method are profitable for solving real-world underwater stereo matching.

Detailed Disparity Distribution Comparison We present separate visualizations of the disparity distribution for six stereo matching datsets on Fig.12-Fig.17. Notably, the Middle-Bury2014 (Fig.12) dataset exhibits numerous samples of large disparity annotations, forming a distinctive pattern. In contrast, the MPI Sintel (Fig.13) dataset displays a distribution that inversely corresponds to that of MiddleBury2014. Practical datasets, including Kitti 2015 (Fig.14) , Nerf-Stereo (Fig.15), and SceneFlow (Fig.16), showcase a significant concentration of disparities within the $0-72$ interval. Interestingly, our proposed UWStereo (Fig.17) mirrors a similar distribution pattern to these practical datasets. This observation underscores the effectiveness of UWStereo as a valuable resource for training stereo matching models in terms of underwater environments.

Image Quality Discussion We select four different image

quality assessment methods named MUSIQ(↑) [62], UIQM(↑) [63], UCIQE(↑) [64], and Uranker(↑) [65] to study the image quality. MUSIQ is designed for terrestrial data while other three methods are for underwater images. By assessing each image with a quality score, we we calculate the mean and standard deviation to fit a Gaussian distribution. Fig. 8-Fig. 11 show all the comparisons.

Concretely, we compare UWStereo with other five terrestrial datasets in Fig. 8,. An observation is that the qualities of synthetic data (MPI Sintel [26] and SceneFlow [15]) are better than those of real-world data (KITTI [17], [18], MiddleBury2014 [19]). While, the qualities of Nerf-rendered images [16] are similar to those captured in real-world scenes. As for UWStereo, we see a medium image quality level which lies in the location that is higher than real-world data but lower than previous synthetic data. This demonstrates that our UWStereo is closer to real-world data than previous synthetic datasets.

Fig. 9, Fig. 10, and Fig. 11 show the comparisons between our UWStereo and other four underwater datasets (*e.g.* UIEB [24], VAROS [27], HIMB [21], and Flsea-Stereo [23]). Despite VAROS produces similar performances in UCIQE and URanker, the discrepancy between synthetic data and real-world underwater data is still exists because the rendering engine cannot perfectly simulate real-world visual effects. But since there is no effective way to acquire accurate depth information for real-world underwater environments, our UWStereo can be considered as a substitute for a real-world dataset to facilitate the future researches for underwater stereo matching task.

VI. Conclusion

In this paper, we present UWStereo, a comprehensive synthetic dataset designed specifically for underwater stereo matching. This dataset features a wide variety of objects and environmental variations, and it includes 29,568 stereo image pairs meticulously annotated with dense disparity maps for the left view. To assess the robustness of current approaches, we benchmark eleven state-of-the-art stereo matching methods against UWStereo and analyze their ability to generalize to underwater scenes. The results reveal that existing models face significant challenges when applied to underwater environments, highlighting a critical gap in the field. To address these challenges, we draw inspiration from masked image learning and propose a novel cross-view enhancement module. This module introduces a new training strategy that focuses on reconstructing cross-domain masked images prior to stereo matching training, thereby enhancing the models' generalization capabilities. We also explore the practical implications of our proposed dataset, discussing how it can serve as a valuable resource for the development of more robust underwater stereo matching algorithms.

In conclusion, UWStereo represents a significant advancement in the field of underwater stereo matching, offering extensive support for both current research and future innovation. By providing a large-scale, well-annotated dataset tailored to the unique challenges of underwater environments, UWStereo paves the way for the development of more effective and generalized stereo matching solutions in this specialized domain.

References

[1] M. Stoiber, M. Sundermeyer, and R. Triebel, "Iterative corresponding geometry: Fusing region and depth for highly efficient 3d tracking of textureless objects," in *CVPR*, June 2022, pp. 6855–6865.

[2] H. Lee and J. Park, "Instance-Wise Occlusion and Depth Orders in Natural Scenes," in *CVPR*, 2022, pp. 21 210–21 221.

[3] J.-R. Chang and Y.-S. Chen, "Pyramid Stereo Matching Network," in *CVPR*, 2018, pp. 5410–5418.

[4] L. Lipson, Z. Teed, and J. Deng, "Raft-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching," in *3DV*, 2021, pp. 218–227.

[5] X. Gangwei, X. Wang, X. Ding, and X. Yang, "Iterative Geometry Encoding Volume for Stereo Matching," in *CVPR*, 2023.

[6] F. Wang, S. Galliani, C. Vogel, and M. Pollefeys, "Itermvs: Iterative Probability Estimation for Efficient Multi-View Stereo," in *CVPR*, 2022, pp. 8606–8615.

[7] Z. Zhang, R. Peng, Y. Hu, and R. Wang, "Geomvsnet: Learning Multi-View Stereo with Geometry Perception," in *CVPR*, 2023.

[8] H. Wang, J. Wang, and L. Agapito, "Co-SLAM: Joint Coordinate and Sparse Parametric Encodings for Neural Real-Time SLAM," in *CVPR*, 2023.

[9] Y. Ren, F. Wang, T. Zhang, M. Pollefeys, and S. Süsstrunk, "Volrecon: Volume Rendering of Signed Ray Distance Functions for Generalizable Multi-View Reconstruction," in *CVPR*, 2023.

[10] G. Xu, J. Cheng, P. Guo, and X. Yang, "Attention Concatenation Volume for Accurate and Efficient Stereo Matching," in *CVPR*, 2022, pp. 12 981–12 990.

[11] H. Xu and J. Zhang, "Aanet: Adaptive Aggregation Network for Efficient Stereo Matching," in *CVPR*, 2020, pp. 1956–1965.

[12] X. Guo, K. Yang, W. Yang, X. Wang, and H. Li, "Group-Wise Correlation Stereo Network," in *CVPR*, 2019, pp. 3273–3282.

[13] H. Zhao, Z. Huizhou, Y. Zhang, J. Chen, Y. Yang, and Y. Zhao, "High-frequency Stereo Matching Network," in *CVPR*, 2023.

[14] J. Li, P. Wang, P. Xiong, T. Cai, Z. Yan, L. Yang, J. Liu, H. Fan, and S. Liu, "Practical Stereo Matching via Cascaded Recurrent Network With Adaptive Correlation," in *CVPR*, 2022, pp. 16 263–16 272.

[15] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation." in *CVPR*, 2016, pp. 4040–4048.

[16] F. Tosi, A. Tonioni, D. Gregorio, and M. Poggi, "Nerf-Supervised Deep Stereo," in *CVPR*, 2023.

[17] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012, pp. 3354–3361.

[18] T. Schops, J. L. Schonberger, S. Galliani, T. Sattler, K. Schindler, M. Pollefeys, and A. Geiger, "A multi-view stereo benchmark with high-resolution images and multi-camera videos," in *CVPR*, July 2017.

[19] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nešić, X. Wang, and P. Westling, "High-resolution stereo datasets with subpixel-accurate ground truth," in *Pattern Recognition*. Springer International Publishing, 2014, pp. 31–42.

[20] T. Schops, J. L. Schonberger, S. Galliani, T. Sattler, K. Schindler, M. Pollefeys, and A. Geiger, "A multi-view stereo benchmark with high-resolution images and multi-camera videos," in *CVPR*, 2017, pp. 3260–3269.

[21] K. A. Skinner, J. Zhang, E. A. Olson, and M. Johnson-Roberson, "Uwstereonet: Unsupervised learning for depth estimation and color correction of underwater stereo imagery," in *ICRA*, 2019, pp. 7947–7954.

[22] D. Berman, D. Levy, S. Avidan, and T. Treibitz, "Underwater Single Image Color Restoration Using Haze-Lines and a New Quantitative Dataset," *IEEE TPAMI*, pp. 1–1, 2020.

[23] Y. Randall and T. Treibitz, "Flsea: Underwater Visual-Inertial and Stereo-Vision Forward-Looking Datasets," *arXiv*, vol. abs/2302.12772, 2023.

[24] C. Li, C. Guo, W. Ren, R. Cong, J. Hou, S. Kwong, and D. Tao, "An underwater image enhancement benchmark dataset and beyond," *IEEE TIP*, vol. 29, pp. 4376–4389, 2020.

[25] G. Billings, R. Camilli, and M. Johnson-Roberson, "Hybrid visual slam for underwater vehicle manipulator systems," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6798–6805, 2022.

[26] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *ECCV*, Oct. 2012, pp. 611–625.

[27] P. G. O. Zwilgmeyer, M. Yip, A. L. Teigen, R. Mester, and A. Stahl, "The VAROS Synthetic Underwater Data Set: Towards realistic multi-sensor underwater data with ground truth," in *ICCVW*, 2021, pp. 3715–3723.
[28] D. Akkaynak and T. Treibitz, "A Revised Underwater Image Formation Model," in *CVPR*, 2018, pp. 6723–6732.
[29] X. Ye, J. Zhang, Y. Yuan, R. Xu, Z. Wang, and H. Li, "Underwater depth estimation via stereo adaptation networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 9, pp. 5089–5101, 2023.
[30] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International journal of computer vision*, vol. 47, pp. 7–42, 2002.
[31] J. Zbontar, Y. LeCun *et al.*, "Stereo matching by training a convolutional neural network to compare image patches." *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2287–2318, 2016.
[32] A. Shaked and L. Wolf, "Improved stereo matching with constant highway networks and reflective confidence learning," in *CVPR*, 2017, pp. 4641–4650.
[33] F. Guney and A. Geiger, "Displets: Resolving stereo ambiguities using object knowledge," in *CVPR*, 2015, pp. 4165–4175.
[34] A. Seki and M. Pollefeys, "Sgm-nets: Semi-global matching with neural networks," in *CVPR*, 2017, pp. 231–240.
[35] F. Zhang, X. Qi, R. Yang, V. Prisacariu, B. Wah, and P. Torr, "Domain-Invariant Stereo Matching Networks," in *ECCV*, 2020, pp. 420–439.
[36] H. Dai, X. Zhang, Y. Zhao, H. Sun, and N. Zheng, "Adaptive disparity candidates prediction network for efficient real-time stereo matching," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 3099–3110, 2022.
[37] Q. Chen, B. Ge, and J. Quan, "Unambiguous pyramid cost volumes fusion for stereo matching," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2023.
[38] Z. Teed and J. Deng, "Raft: Recurrent All-Pairs Field Transforms for Optical Flow," in *ECCV*, 2020, pp. 402–419.
[39] Z. Rao, B. Xiong, M. He, Y. Dai, R. He, Z. Shen, and X. Li, "Masked representation learning for domain generalized stereo matching," in *CVPR*, 2023.
[40] P. Weinzaepfel, T. Lucas, V. Leroy, Y. Cabon, V. Arora, R. Brégier, G. Csurka, L. Antsfeld, B. Chidlovskii, and J. Revaud, "Croco v2: Improved cross-view completion pre-training for stereo matching and optical flow," in *ICCV*, October 2023, pp. 17 969–17 980.
[41] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *ECCV*, 2020.
[42] M. Savva, A. X. Chang, and P. Hanrahan, "Semantically-enriched 3d models for common-sense knowledge," in *CVPR*, 2015, pp. 24–31.
[43] M. J. Islam, C. Edge, Y. Xiao, P. Luo, M. Mehtaz, C. Morse, S. S. Enan, and J. Sattar, "Semantic Segmentation of Underwater Imagery: Dataset and Benchmark," in *IROS*. IEEE, 2020.
[44] D. Akkaynak and T. Treibitz, "Sea-Thru: A Method for Removing Water From Underwater Images," in *CVPR*, 2019.
[45] Y. Xu, D. Yu, Y. Ma, Q. Li, and Y. Zhou, "Underwater stereo-matching algorithm based on belief propagation," *Signal, Image and Video Processing*, vol. 17, no. 4, pp. 891–897, 2023.
[46] Z. Qu, O. Vengurlekar, M. Qadri, K. Zhang, M. Kaess, C. Metzler, S. Jayasuriya, and A. Pediredla, "Z-splat: Z-axis gaussian splatting for camera-sonar fusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–12, 2024.
[47] Z. Tang, Y. Li, and C. Wang, "Multiangle sonar imaging for 3-d reconstruction of underwater objects in shadowless environments," *IEEE Journal of Oceanic Engineering*, pp. 1–12, 2025.
[48] M. Qadri, K. Zhang, A. Hinduja, M. Kaess, A. Pediredla, and C. A. Metzler, "Aoneus: A neural rendering framework for acoustic-optical sensor fusion," in *ACM SIGGRAPH 2024 Conference Papers*, 2024.
[49] J. Li, K. A. Skinner, R. M. Eustice, and M. Johnson-Roberson, "Watergan: Unsupervised generative network to enable real-time color correction of monocular underwater images," *IEEE Robotics and Automation letters*, vol. 3, no. 1, pp. 387–394, 2017.
[50] S. Zhu and X. Liu, "Pmatch: Paired Masked Image Modeling for Dense Geometric Matching," in *CVPR*, 2023.
[51] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked Autoencoders Are Scalable Vision Learners," in *CVPR*, 2022, pp. 16 000–16 009.
[52] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "Loftr: Detector-Free Local Feature Matching With Transformers," in *CVPR*, 2021, pp. 8922–8931.
[53] X. Cheng, Y. Zhong, M. Harandi, Y. Dai, X. Chang, H. Li, T. Drummond, and Z. Ge, "Hierarchical Neural Architecture Search for Deep Stereo Matching," in *NeurIPS*, 2020.
[54] F. Zhang, V. Prisacariu, R. Yang, and P. H. Torr, "Ga-Net: Guided Aggregation Net for End-To-End Stereo Matching," in *CVPR*, 2019, pp. 185–194.
[55] Z. Chen, W. Long, H. Yao, Y. Zhang, B. Wang, Y. Qin, and J. Wu, "Mocha-stereo: Motif channel attention network for stereo matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 27 768–27 777.
[56] X. Wang, G. Xu, H. Jia, and X. Yang, "Selective-stereo: Adaptive frequency information selection for stereo matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 19 701–19 710.
[57] B. Liu, H. Yu, and G. Qi, "Graftnet: Towards Domain Generalized Stereo Matching With a Broad-Spectrum and Task-Oriented Feature," in *CVPR*, 2022, pp. 13 012–13 021.
[58] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, S. Levine, V. Vanhoucke, and K. Goldberg, Eds., vol. 78. PMLR, 13–15 Nov 2017, pp. 1–16.
[59] J. Lou, W. Liu, Z. Chen, F. Liu, and J. Cheng, "Elfnet: Evidential local-global fusion for stereo matching," in *ICCV*, October 2023, pp. 17 784–17 793.
[60] Z. Shen, Y. Dai, X. Song, Z. Rao, D. Zhou, and L. Zhang, "Pcw-net: Pyramid combination and warping cost volume for stereo matching," in *ECCV*, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds., 2022, pp. 280–297.
[61] D. Levy, A. Peleg, N. Pearl, D. Rosenbaum, D. Akkaynak, S. Korman, and T. Treibitz, "Seathru-NeRF: Neural Radiance Fields in Scattering Media," in *CVPR*, vol. abs/2304.07743, 2023.
[62] J. Ke, Q. Wang, Y. Wang, P. Milanfar, and F. Yang, "Musiq: Multi-scale image quality transformer," in *ICCV*, 2021, pp. 5128–5137.
[63] K. Panetta, C. Gao, and S. Agaian, "Human-visual-system-inspired underwater image quality measures," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 3, pp. 541–551, 2016.
[64] M. Yang and A. Sowmya, "An Underwater Color Image Quality Evaluation Metric," *IEEE TIP*, vol. 24, no. 12, pp. 6062–6071, 2015.
[65] C. Guo, R. Wu, X. Jin, L. Han, W. Zhang, Z. Chai, and C. Li, "Underwater Ranker: Learn Which Is Better and How to Be Better," in *AAAI*, 2023, pp. 702–709.

VII. Biography Section

Qingxuan Lv was born in Shanxi, China, in 1996. He received his bachelor's degree in Computer Science and Technology from the Shanxi University of Finance and Economics in 2018. He received his master's degree in Computer Science and Technology from the Ocean University of China (OUC) in 2021. He is currently a candidate of a doctor's degree at the ocean group of VisionLab OUC. His research interests include computer vision and machine learning. Specifically, he is interested in universal domain adaptation and semantic segmentation.

Junyu Dong received the B.Sc. and M.Sc. degrees in applied mathematics from the Department of Applied Mathematics, Ocean University of China, Qingdao, China, in 1993 and 1999, respectively, and the Ph.D. degree in image processing from the Department of Computer Science, Heriot-Watt University, Edinburgh, U.K., in November 2003. He is currently a Professor and the Head of the Department of Computer Science and Technology. His research interests include machine learning, big data, computer vision, and underwater image processing.

Yuezun Li (Member, IEEE) received the B.S. degree in software engineering from Shandong University in 2012, the M.S. degree in computer science in2015, and the Ph.D. degree in computer science from University at Albany–SUNY, in 2020. He was a Senior Research Scientist with the Department of Computer Science and Engineering, University at Buffalo–SUNY. He is currently a Lecturer with the Center on Artifcial Intelligence, Ocean University of China. His research interests include computer vision and multimedia forensics. His work has been published in peer reviewed conferences and journals, including ICCV, CVPR, TIFS, TCSVT, etc.

Sheng Chen (IEEE Life Fellow) received the B.Eng. degree in control engineering from the East China Petroleum Institute, Dongying, China, in 1982, the Ph.D. degree in control engineering from City University, London, in 1986, and the higher doctoral (D.Sc.) degree from the University of Southampton, Southampton, U.K., in 2005. From 1986 to 1999, he held research and academic appointments with the University of Sheffield, the University of Edinburgh, and the University of Portsmouth, U.K. Since 1999, he has been with the School of Electronics and Computer Science, University of Southampton, where he is a Professor of Intelligent Systems and Signal Processing. His research interests include adaptive signal processing, wireless communications, modeling and identification of nonlinear systems, neural network and machine learning, intelligent control system design, evolutionary computation methods, and optimization. Professor Chen has authored over 700 research papers. He have 20,000+ Web of Science citations with h-index 61, and 39,000+ Google Scholar citations with h-index 83. Dr Chen was elected to a fellow of the United Kingdom Royal Academy of Engineering in 2014. He is a fellow of Asia-Pacific Artificial Intelligence Association (FAAIA), a fellow of IET, and an original ISI Highly Cited Researcher in engineering (March 2004).

Hui Yu received the PhD degree from Brunel University London in 2009. His research interests include visual and cognitive computing, social vision, social robot and machine learning. His research particularly focuses on 3D/4D facial expression reconstruction and perception, image and video analysis for human-machine and social interaction as well as intelligent vehicle applications. He leads the Visual Computing and Social Robot Group (VCSR) in cSCAN at the University of Glasgow. He has been awarded the Industrial Fellowship project by the Royal Academy of Engineering. He also serves as an Associate Editor for Neurocomputing, IEEE Transactions on Human-Machine Systems, IEEE Transactions on Intelligent Vehicles, and IEEE Transactions on Computational Social Systems journal. More publication information can be found on Google Scholar.

Shu Zhang Shu Zhang is currently an Associate Professor and Postgraduate supervisor at Ocean University of China, Qingdao, China. He received his PhD in Computer Application Technologies from Ocean University of China. He was previously a research associate at the University of Portsmouth, Portsmouth, UK. His main research interests include computer vision, feature analysis, 3D reconstruction, video processing, underwater image analysis, and deep learning among others.

Wenhan Wang Wenhan Wang is a postgraduate student majoring in computer technology at Ocean University, China, Qingdao, China. Her main research interests include computer vision, SLAM, underwater image analysis and among others.