IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 11 May 2022, accepted 15 June 2022, date of publication 23 June 2022, date of current version 20 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3185753

Intelligent Underwater Stereo Camera Design for Fish Metric Estimation Using Reliable Object Matching

NAOMI A. UBINA[1,2], SHYI-CHYI CHENG[1], (Member, IEEE), CHIN-CHUN CHANG[1], (Member, IEEE), SIN-YI CAI[1], HSUN-YU LAN[3], AND HOANG-YANG LU[4]

[1]Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung 202, Taiwan
[2]College of Computing Studies, Information and Communications Technology, Isabela State University, Isabela 3309, Philippines
[3]Department of Aquaculture, National Taiwan Ocean University, Keelung 202, Taiwan
[4]Department of Electrical Engineering, National Taiwan Ocean University, Keelung 202, Taiwan

Corresponding author: Shyi-Chyi Cheng (csc@mail.ntou.edu.tw)

This work was supported in part by the Ministry of Science and Technology under Grant MOST 110-2221-E-019-048; and in part by the Fisheries Agency, Council of Agriculture, Taiwan, under Grant 111AS-6.2.1-FA-F6.

ABSTRACT Precise fish metric estimation is essential in providing intelligent aquaculture farm decisions. Stereo vision has been widely used for size estimation. Still, many factors affect fish metrics accuracy using a low-cost underwater stereo camera, such as distance, ambient lighting, water velocity, and turbidity. Although such a system is affordable and energy-efficient, they are less accurate in estimating depths than its active counterparts. Since power source is always a problem in offshore aquaculture sites, energy-efficient devices are important. To deal with the accuracy problems of the camera, we propose an effective deep-learning-based object matching to optimize the fish metric estimation. In terms of the challenges of the underwater environment, an analysis of the accuracy of the fish 3D position calculation in the aquaculture cage based on the captured stereo camera images is performed. The analysis assumes a known geometrical configuration of the rectified camera system. The critical factor limiting the 3D fish metric estimation accuracy is the resolution of the computed depth maps of fish. An object-based matching is proposed for underwater fish tracking and depth computing to address this issue using reliable convolutional neural networks (CNNs). For each stereo video frame, an object classification and instance segmentation CNN separates the fish objects from their background. The fish objects are then cropped and matched using sub-pixel disparity computation of the video interpolation CNN. The calculated fish disparities and depth values are used for fish metric estimations. We also tracked each fish and computed the metrics across frames. The median metrics are calculated as the final result to reduce the noises introduced by the different gestures of the fish. Furthermore, underwater stereo video datasets with the actual metrics of sampled fish measured by humans are also constructed to verify the effectiveness of our approach. Our proposed method has less than a 5% error rate for fish length estimation.

INDEX TERMS Convolutional neural network, object tracking, object-based stereo matching.

I. INTRODUCTION

One of the applications of machine learning in computer vision focuses on the computer's ability to see and understand captured digital images and videos from cameras and sensors. It involves the acquisition and processing digital images and uses extracted data to solve problems such as navigation, tracking, medical visualization, and object recognition without human intervention. It is programmed to process as a human visual system to understand, detect, and identify

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wei.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

objects based on their appearance and environment [1]. Recent advances in computer vision combined with AI and AR are promising areas for research and integration into underwater environment monitoring and surveillance. According to Lepetit [2], computer vision (CV) has great potential for applications in augmented reality. Its ability to use visual features automatically captured by the camera makes it possible to create a virtual world taken and based on a real-world scene, providing high precision and accuracy. Three-dimensional (3D) information has been utilized to improve augmented reality applications. A 3D model of the environment offers richer interactions with higher-level engagement. Objects in the 3D model provide greater accuracy in being represented in computer-generated elements and give the users a better sense of these elements in the real world [3]. 3D modeling in computer vision aims to recognize 3D objects from visual input. These inputs depend on the viewer's direction, the illumination conditions, and the geometric representations [4]. A 3D model can be used for measurement and visualizations or a combination of both, making them applicable to various computer vision problems [5]. Also, 3D reconstruction deals with producing or extracting the 3D information of an object through a 3D point cloud and a depth map or a disparity map. Applications of 3D reconstruction are evident in robotics [6], [7], [8]], self-driving cars and navigation [9], [10], object detection [11], [12], and facility simulations [13], [14], [15]. However, 3D modeling for living or continuously moving fish objects remains challenging due to the limitation of capturing stereo images in an underwater environment.

There are two types of 3D reconstruction techniques. The 'active' approach uses an optical sensor and usually uses LIDAR, which is relatively expensive but requires less processing to acquire 3D information. The passive technique uses optical sensors for its 3D reconstruction, but with many challenging requirements, requiring more sophisticated methods [16]. Therefore, many explorations were made using inexpensive sensors specifically for stereo vision, which is much cheaper but requires high computational cost. The challenge for stereo image processing is to provide an accurate but efficient computational requirement with a faster result to build the 3D model of the target object or scene.

Stereo vision mimics how the human eye captures and processes two different views of an object. First, each eye captures its view of the scene and sends the information to the brain. Then, it matches and combines the similarities of the two images and adds the slight differences called disparities to provide the depth perception [17] needed to generate the 3D information. Using the 3D data and distance from the camera, the size of an object can be measured. But there are also many challenges to stereo image processing, such as estimating the depth of a stereo image pair, occlusions, large saturate areas, and repetitive patterns [5], [18]. In addition, the focal axes for camera calibration and image rectification must be considered to ensure that the epipolar lines of the two input images are parallel [16]. The generation of accurate disparity maps in real-time is one of the increasing demands for stereo image processing. Many existing methods generate disparity maps at a slow speed but with higher accuracy. In addition, others generate quickly but with a lower accuracy [19]. Nevertheless, the use of stereo systems can be promising, as they reduce the presence of multipath interference [20], handle occluded and non-texture regions [21], and provide robustness over triangulation-based structured light systems [22]. In stereo vision, there are four stages: cost matching, cost aggregation, disparity selection, and disparity refinement [23]. The stereo matching problem states that, given a pair of images taken from a stereo camera and its epipolar constraints, find the most appropriate matching patch using the other image [24].

The field of aquaculture has been one of the great contributors to strengthening and ensuring food security, most especially for high protein sources. As part of aquaculture production, monitoring fish growth in aquaculture ponds and cages based on size is very important. The size of the fish is a crucial parameter for fish stock assessment since it provides information on the best time for selling, whether the target fish growth is achieved, and whether fast-growing fish can be separated from the slow ones [25]. It also helps predict the daily feed intake to avoid underfeeding or overfeeding, thus maximizing fish production and farm profit. When combined with relevant sensors and historical data, images from different sites can be used to analyze the fish size, the number of fish, fish feeding intensity of fish schools, and assessment of fish diseases.

Traditional approaches to fish length estimation are invasive since it involves directly sampling the fish using a measuring board (ruler) with a scaling unit such as millimeters or centimeters to measure fish length. Aside from being laborious, such a method is also prone to inconsistencies and bias since results are dependent on one's expertise and eye direction. Furthermore, invasive procedures to measure length requires physical capturing and handling of the fish before adversely affecting their health, growth rate, and quality of the harvested product due to injury and stress [26]. With the advancement of technology and the integration of computerized systems, specifically computer vision techniques, fish measurement is now automated. It no longer requires manual measurement of fish size or length. In the work of Rahim *et al.* [27], to lessen the invasive method, the authors used different types of digital cameras with different camera positions to perform automatic measurements of fish. The authors used the fish length from the digital images (FiLEDI) framework. They used optical theory and image processing techniques to identify the actual fish size from the captured image's pixel value and obtained it. But 2D images are flat and do not provide the holistic view or volume of the object and only use length and height as its dimensions; they cannot accurately measure curved objects. Using an RGB camera, depth camera, and a remote computer with a centralized database installed on fishing vessels, Maia *et al.*, [28] integrated fish measurement by auto-detecting the fish boxes

by acquiring the 3D images. Stereo vision cameras provide simultaneous views and consider different positions to estimate fish size even in a free-swimming environment such as ponds/tanks or open cages [29], [30], [31]. Although such a method works, the cost of equipment, time for image processing, and complexity of the camera system are enormous challenges.

Deep learning methods for stereo image processing are gaining popularity. Many studies have been involved in stereo imaging [5], [9], [32], but only a few applications have been applied to modeling underwater scenes specifically for fish size estimations. The focus of this paper is the application of underwater imagery using a low-cost stereo camera system to capture stereo images from aquaculture farms and generate 3D objects using deep learning techniques, specifically convolutional neural networks (CNNs) for fish length estimation. For stereo images, depth information and disparity map information are essential for quality 3D modeling [33], [34], [35]. Underwater images have more significant challenges, as images that were taken generally have problems with image degradation, poor contrast, blurring, color deviation [36], poor visibility, light attenuation, and water turbidity [37]. To deal with these challenges, a low-cost underwater stereo camera system captures stereo images and matches each fish in the left and right images using an unsupervised stereo matching neural network. The dense disparities between the left and right fishes are computed to obtain the depth map of the 3D model of each fish. Using this 3D model, the fish's body length, height, and width are estimated more accurately. The estimated values combined with various sensors and weight regression formulas can establish the growth curve of the fish.

One potential problem of a low-cost stereo image camera system is it could be incompletely or incorrectly synchronized, which causes the object's pose in the left image to be slightly different from that of the right image. Figure 1 shows the diagram of our proposed stereo matching for underwater object reconstruction using the left and right images as the inputs. The stereo image rectification is a pre-processing technique to obtain the correct intrinsic and extrinsic parameters by calibrating the stereo camera system. Then, each corrected image is inputted to the instance segmentation neural network to transform each image frame into a set of fish objects and background objects. Next, the correspondence in the left image is searched in the right image to generate the disparity map for object matching. But a single disparity value cannot accurately restore the pixel depths of the 3D object. Also, a more difficult challenge for fish length estimations is that the target fish object has multiple gestures since it freely swims in the underwater environment, which brings additional noise in measuring the exact 3D information of the fish. Lastly, the fish might overlap with other fish in the captured images, degrading the mask accuracy of the fish objects even with well-designed instance segmentation CNNs.

In solving these difficulties, the left and right image objects are cropped and aligned to form the input pair. These images were further processed using the video interpolation CNN (VICNN) [38] based stereo matching algorithm, which calculates the residual disparity of each pixel in the left object. The core of our stereo matching algorithm is VICNN, which synthesizes the intermediate object to establish the pixel correspondences between the left and right objects. Instead of using a single frame image, based on the proposed object matching scheme, we tracked each fish across frames and calculated a sequence of 3D models to reduce the biometric noise introduced by the gesture variations of a freely swimming fish. This mechanism is nonintrusive and reduces manual handling of the fish to prevent stress [39] and disturbance and avoids injury caused by fish catching in estimating the biological information.

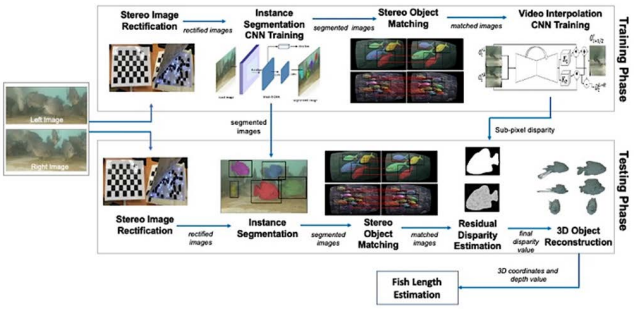

FIGURE 1. Block diagram of our stereo matching-based underwater scene modeling.

The contributions of the proposed approach are as follows. First, we proposed a deep neural network that establishes a real-time pixel correspondence between stereo images. Second, our system directly trains the raw data video, which reduces the deep neural networks training complexity for 3D model reconstruction, and large set of human-annotated label data requirements for training is eradicated. Third, using traditional stereo matching algorithms, it is difficult to establish precise pixel correspondences from the texture-less stereo images. The integration of interpolated signals of the matched object pairs ensures the correctness of the computed disparity image from precise correspondences with minimal object matching error. Next, the object-based stereo matching optimization algorithm contributes to designing the image warping of the disparity to minimize the resulting smoothness. Also, incorporating fish tracking enables our approach to measure the length of a freely swimming fish directly from the aquatic pool or aquaculture on-site location. Finally, we successfully integrated a low-cost stereo and power-efficient camera system as our sensor for our data collection.

The remainder of this paper is as follows. Section 2 provides the Related Works, Section 3 contains the details of

our Methods, and Section 3 is on the Experimental Results. Finally, Section 4 is our Conclusions and Future Works.

II. RELATED WORKS

Traditional stereo matching methods use the low-level features of the image patches around the pixel to evaluate the difference. In addition, many use local methods, where the disparity with the lowest matching cost is selected. The disparity obtained from this method is high-quality but time-consuming. On the other hand, a semi-global method trades off the computational requirement and the time the results. But these two traditional methods are still limited and yield a poor depth map quality [40]. Also, conventional matching methods are problematic regarding wide baseline image feature extraction and the reliability of its feature descriptors and matching measures [41].

Many successful works progress stereo image processing systems using neural networks and deep learning to improve the traditional methods. Deep learning methods have significantly gained popularity by contributing impressive results to improve various computer vision tasks. Although it requires vast training data sets and high computational power to produce better results, the availability and improvement of computing resources make it less of a barrier [40]. Convolutional neural networks (CNNs) have achieved great success in stereo matching as they can learn more complex, robust, and powerful deep feature representations than conventional stereo matching methods. Its robust feature representation is very suitable for the stereo matching problem as it measures the similarities between pixels of two images using deep features for more powerful and accurate matching results compared to handcrafter features [42].

In recent years, challenges involving various stereo matching algorithms for high performance in terms of matching and reduced processing time apply to real-time requirements. There have been trade-offs in quality and processing time, but developing an algorithm that can do both would be very promising. In the work of Zhang *et al.* [43], the authors proposed an improved binocular stereo matching algorithm based on Minimum Spanning Tree (MST) cost aggregation. The height of the MST is used for parallel processing to speed up computation. Although the processing time has been reduced, one has to consider the quality of the matching results as an essential attribute. In implementing stereo matching techniques, an adaptive window has been widely used. The selection of a matching window can affect the performance of the matching algorithm. In the work of [44], an adaptive window and semi-global matching algorithm and the sum of absolute differences calculate the matching cost. Many works also focused on determining the size of window-based to perform stereo-matching requirements [45], [46], [47].

Many works also supported CNN stereo matching to improve the accuracy of the matching results [48]. Xia *et al.* [49] optimized the 3D convolution kernel of the Pyramid Stereo Matching and reduced the computational complexity without losing its accuracy. A two-branch convolutional sparse representation model is proposed by Cheng *et al.* [50] to reduce the heavy load of labeling ground truth disparities. Their proposed approach learns the convolutional filter from stereo image pairs and does not rely on ground truth disparity maps. AdaStereo aims to align multi-level representations for deep stereo matching networks. A non-adversarial progressive color transfer algorithm is integrated for input image-level alignment. The authors also designed an efficient parameter-free cost normalization layer for internal feature-level alignment achieving state-of-the-art disparity networks fine-tuned with target-domain ground truths [51]. Wang *et al.* [52] created a pyramid voting module (PVM) by first building its multi-scale cost volume and later adopting a recurrent unit to iteratively update disparity estimations at high resolution. An improvement in segmentation accuracy of 3% by [53] by extending DeepLabv3+ for the image segmentation network. It improved the appearance of the segmented target objects and retained more feature information. In the work of Jia *et al.* [54], the authors used a 2D encoder-decoder network to generate a rough disparity map and construct a disparity range for the 3D aggregation network. Their work showed a significant improvement in accuracy and reduced memory costs simultaneously. The stacked hourglass structure also refined the disparity from coarse to fine. Also, a multi-cross attention model for stereo matching improved the matching accuracy and effectively provided an end-to-end disparity regression. The stacked hourglass was utilized to extract the characteristics of the low-resolution feature images [55]. The datasets used in testing and training the networks were publicly available stereo image datasets. There is a need to implement underwater datasets for training and testing to find their appropriateness for such an environment to generate 3D models for fish and length estimations.

Some works also integrated deconvolution networks as an application in stereo matching. Deconvolution methods are beneficial for eliminating image defects and increasing resolution. In the work of Cheng *et al.* [56], their method help improves the performance of conventional deconvolution networks (DN) and reduces the computational requirements. Also, in the paper of Ma *et al.* [57], a deconvolutional network was utilized to enlarge the size of the input feature map. The de-convolved features of the left and right image patches are fed into the successive convolutional layers with max-pooling to obtain its compact features. Although DNs seem to provide improved results, they demand high computational requirements when performing multi-layer convolution sparse decomposition on images, require more powerful machines to perform efficient processing, and are prone to blur degradation.

Recently, works on underwater stereo-matching algorithms have also been available. Based on the belief propagation (BP) network, Xu *et al.* [58] utilized the Markov random field (MRF) model. The estimated disparity map is calculated using BP on an MRF model consisting of an observed

and hidden node representing the matching cost and disparity value, respectively. To ensure that the stereo-matching method reflects the underwater environment, an energy function based on the degradation of images was added to represent the brightness change caused by light illumination. Even though such an approach stimulates an underwater environment, it still does not cover other factors such as turbulent water where the image geometry is being distorted, and the objects are fixed and non-moving, which does not represent a natural environment for aquaculture farms.

Disparity refinement helps eliminate mismatches caused by occlusion, low texture, and many others. Popular refinement methods are based on the consistency check of the left and right two disparity maps [59] to ensure that noise was reduced or eradicated to achieve a higher accuracy rate. However, images for stereo vision are prone to texture-less regions and false matches due to the uniqueness of their area pixels, which affect the accuracy of disparity maps. Regularization helps smoother disparity maps by eliminating and filtering the image noise in the map. Adding TVL1 as an additional term help eradicates noise and edges for optical flow optimization. Also, including VICNN specifically interpolated signals in our approach helps compute disparity with increased accuracy using precise correspondence to lessen object matching error.

Deep learning has been integrated into fish length estimation, especially morphological features [60]. A deep learning-based segmental analysis was used [61] to determine the length feature of fish by analyzing the completely visible fish (CVF) segments such as head, body, and tail. Fish length is estimated using the CVF. The work of Yu *et al.* [62], [63] used Mask-RCNN to implement pixel-level instance segmentation to segment fish morphological features. The fish body image is first pre-processed and augmented and then fed into the Mask-RCNN for training to obtain the segmented image and parameters of the target frame. The fish length is generated by calculating the number of pixels in the detection frame and mapping it to the actual fish feature parameters. Also, a CNN classifier was developed to detect the regions of the fish head, tail fork, and color plate. The fish body length is estimated using the distance between the snout and fork points using a pixel-to-distance ratio and an accuracy of 98.78% [64]. Another morphological feature was used to estimate fish measurement using U-Net. In the work of [65], the U-Net structure was improved by using 3×3 dilated convolution with a dilation rate of 3 and 1×1 convolution that replaced the 3×3 convolution of the original network. The result shows that the improved U-net has a better segmentation effect on the fish edge and reached 97.6% accuracy. The results of these approaches are promising, but the authors used an image of a captured and un-moving fish in training their neural networks. The underwater environment with a free-swimming fish was not considered.

On the other hand, three-dimensional (3D) references have also been utilized to estimate the fish size. Risholm *et al.* [66] used an underwater 3D ranged-gated camera to perform fish length estimation of free-swimming fish with an algorithmic pipeline to detect, track and estimate fish length stages with a length estimation error of 1%. DBScan clustering algorithm takes the depth frame and produces a segmentation of the detected fish. The results are expected to have a high accuracy rate since they used a ranged-gated underwater 3D camera with high-resolution underwater intensity and depth images, which requires a much higher implementation cost and more energy to operate, which is a challenge in the open sea aquaculture environment. Our approach used a low-cost and power-efficient stereo camera system to estimate fish length.

III. MATERIALS AND METHODS

A. RELIABLE OBJECT MATCHING WITH SEMANTIC SEGMENTATION CNN

As shown in Figure 1, our approach integrates camera calibration for preprocessing as the first stage to obtain the stereo system's correct intrinsic and extrinsic parameters. Then, the internal and external parameters are inputted into the image rectification process to obtain the rectified images. This rectification process projects images using a common horizontal image plane as it twists back the left and right image pixels to have the exact coordinates in the horizontal plane. Image rectification yields all epipolar lines to be parallel in the image plane. For illustration, we assumed that the input stereo images were rectified before being applied to the proposed system for further processing.

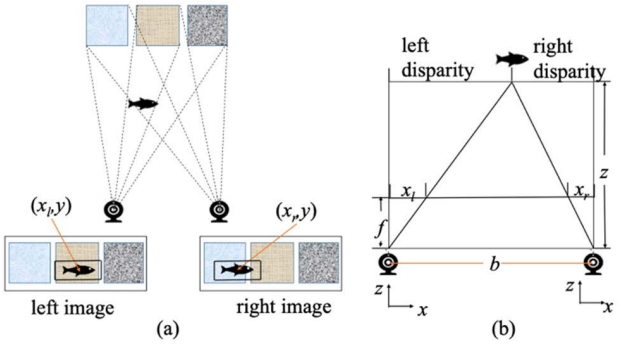

FIGURE 2. The capturing theory of the stereo camera system: (a) an example of the captured stereo images; (b) the theory to compute the disparity and depth of the object. The *y*-axis is perpendicular to the page [67].

Each rectified left and right image pair is first inputted to a semantic segmentation neural network [68] with excellent object detection and segmentation performance. The semantic segmentation CNN transforms each frame image into a set of reliable fish objects and the background object. Let $\boldsymbol{O}_t^L$ ($\boldsymbol{O}_t^R$) and $\boldsymbol{O}_{t+1}^L$ ($\boldsymbol{O}_{t+1}^R$) be the tracked object in frames $\boldsymbol{t}$ and $\boldsymbol{t}+1$ of the left (right) video, respectively. The object pair ($\boldsymbol{O}_t^L$, $\boldsymbol{O}_t^R$) is a stereo object if the matched cost between $\boldsymbol{O}_t^L$ and $\boldsymbol{O}_{t+1}^L$ is small, and their motion vectors are similar. Given the object $\boldsymbol{O}_t^L$, the basic processing of object matching is to search the corresponding object $\boldsymbol{O}_t^R$ in the right image along the x-axis since the images have been rectified to have a

horizontal epipolar geometry. Figure 2 depicts the basic concept of image capturing using a stereo camera system. In Figure 2, the foreground object, i.e., the fish, is overlapped with different backgrounds in the rectified left and right images, which decreases the accuracy of object matching. Thus, we perform the semantic segmentation CNN to obtain the masks of O_t^L and O_t^R, in which the backgrounds are removed.

The object matching scheme first computes the motion vector between $O_t^L(O_t^R)$ and $O_{t+1}^L(O_{t+1}^R)$ based on the computation of the matching cost with the object pair (O_t^L, O_{t+1}^L) $((O_t^R, O_{t+1}^R))$ as the input. The matching cost between objects O_t and O_{t+1} is measured by aggregating pixel-wise matching costs in both objects with the support weights [69]. Let x_t (x_{t+1}) be the center of the object O_t (O_{t+1}). The support weight between pixels x_1 and x_2 is defined as

$$w(x_t, x_{t+1}) = \exp(-((\Delta c_{x_t,x_{t+1}}/\sigma_c) + (\Delta g_{x_t,x_{t+1}}/\sigma_g))) \tag{1}$$

where $\Delta c_{x_t,x_{t+1}}$ and $\Delta g_{x_t,x_{t+1}}$ represents the color difference and the spatial distance between pixels x_t and x_{t+1}, respectively; σ_c is the variance of color difference; σ_g is determined according to the size variance of all the objects. The value of $w(x_t, x_{t+1})$ measures the strength of the pixel correspondence (x_t, x_{t+1}). Notice that the motion vector of the pixel x_t can be computed as $u_{x_t} = x_{t+1} - x_t$. To assume every pixel in O_t would have a similar motion vector, we can compute the matching cost of the object pair (O_t, O_{t+1}) by combining the pixel-wise support-weights in both objects:

$$E(x_t, x_{t+1}) = \frac{\sum_{p_t \in O_t, p_{t+1} \in O_{t+1}} w(x_t, p_t)\, w(x_{t+1}, p_{t+1})\, \Delta c_{p_t,p_{t+1}}}{\sum_{p_t \in O_t, p_{t+1} \in O_{t+1}} w(x_t, p_t)\, w(x_{t+1}, p_{t+1})} \tag{2}$$

where the pixel pair (p_t, p_{t+1}) is constrained to have the motion vector u_{x_t}. Using (2), for each object O_t in frame t, we can define the matched object O_{t+1}^* with the center at x_{t+1}^* in frame $t+1$ as

$$O_{t+1}^* = \arg \max_{O_{t+1} \in S_{t+1(x_t)}} E(x_t, x_{t+1}) \tag{3}$$

where $S_{t+1(x_t)}$ is the set of all possible objects within the search window with x_t as the center in frame $t+1$.

Once the motion vector u_t^L (u_t^R) of the object O_t^L (O_t^R) is determined, the matching cost defined in (2) for stereo object searching can be refined as

$$E_S\left(x_t^L, x_t^R\right) = E\left(x_t^L, x_t^R\right) + \lambda \left\| u_t^L - u_t^R \right\|_2 \tag{4}$$

where x_t^L and x_t^R be the center pixels of O_t^L and O_t^R, respectively; $\left\| u_t^L - u_t^R \right\|_2$ is the L_2 distance between motion vectors u_t^L and u_t^R; $\lambda > 0$ is the Lagrange multiplier. Using (4), for each object O_t^L in the left image, we can define the best-matched object O_t^{R*} with the center at $(x_t^{R,*}, y)$ in the right image as

$$O_t^{R*} = arg \min_{O_t^R \in S_O} E_S\left(x_t^L, x_t^R\right) \tag{5}$$

where S_O is the set of all possible objects in the right image for matching the object O_t^L. Notice the disparity of object O_t^L with the center pixel x_t^L can be computed as $d_{O_t^L} = x_t^L - x_t^{R,*}$.

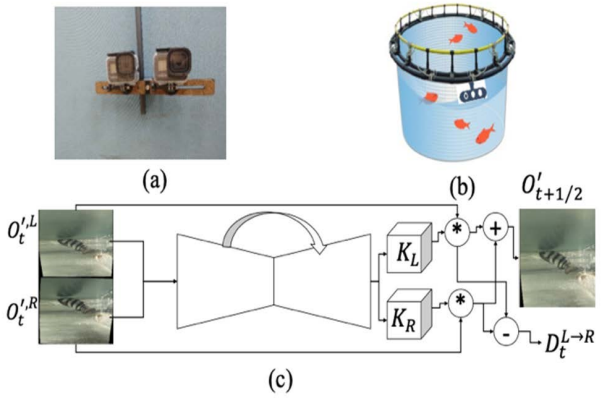

FIGURE 3. Establishing the stereo matching algorithm for residual disparity computation based on the video interpolated CNN: (a) the underwater stereo imaging system; (b) the schematic diagram of using the stereo imaging system to monitor underwater fish; (c) given a pair of rectified stereo underwater images $(O_t'^{,L}, O_t'^{,R})$, the video interpolation CNN (VICNN) [9] produces a set of pixel-wise kernels K_L and K_R for synthesizing the middle object $O'_{t+1/2}$ and the displacement image $D_t^{L\to R}$.

B. PIXEL-WISE RESIDUAL DISPARITY ESTIMATION WITH VIDEO INTERPOLATION CNN

Obviously, the pixel-wise motion vectors (optical flow) in an object are similar but not the same because individual parts of the object might perform different actions. Similarly, the pixel-wise disparities of an object are not the same as that of center pixel since the depth information of a real-world 3D object is not the same everywhere. As mentioned above, given a detected object pair (O_t^L, O_t^R), Equation (5) can compute the basic disparity for each pixel in O_t^L. Obviously, for each pixel x in O_t^L, this disparity difference Δd_x should be estimated to model the disparity of x as

$$d_x = d_{O_t^L} + \Delta d_x \tag{6}$$

where $d_{O_t^L}$ is the disparity value of the center pixel x_t^L defined by Eq. (5). Suppose we translate the centers of matched objects into the common original point (0,0). In that case, the left and the right object are aligned with each other and form a new object pair $(O_t'^{,L}, O_t'^{,R})$ which can be used to estimate the pixel disparity difference Δd_x of the pixel $x \in O_t^L$.

Figure 3 shows the proposed stereo matching algorithm based on the video interpolated CNN (VICNN) [38], synthesizing the middle object $O'_{t+1/2}$ with the object pair $(O_t'^{,L}, O_t'^{,R})$ as the input. For each pixel x in $O'_{t+1/2}$, the

VICNN computes a pixel-wise kernel pair $(\boldsymbol{K_L}(x), \boldsymbol{K_R}(x))$ to interpolate the pixel value of $x = (\boldsymbol{x}, \boldsymbol{y})$ in $\boldsymbol{O'_{L,R}}$ using the following equation:

$$\boldsymbol{f}(x) = < \boldsymbol{K_L}(x), \boldsymbol{P_L}(x) > + < \boldsymbol{K_R}(x), \boldsymbol{P_R}(x) > \quad (7)$$

where $\langle \cdot, \cdot \rangle$ is the inner product operator; $\boldsymbol{P_L}(x) \in \boldsymbol{O'^{,t}_t}$ and $\boldsymbol{P_R}(x) \in \boldsymbol{O'^{,R}_t}$ are patches with the common center x. The kernel pair can also be used to compute the disparity difference of x:

$$\Delta \boldsymbol{d}_x = \prec \boldsymbol{K_R}(x), \boldsymbol{U_x}(x) > - \prec \boldsymbol{K_L}(x), \boldsymbol{U_x}(x) \quad (8)$$

where $\boldsymbol{U_x}$ is the x-displacement matrix with $\boldsymbol{U_x}(x') = x' - x$. Eq. (8) defines the displacement image as

$$\boldsymbol{D_t^{L \to R}} = \{\Delta \boldsymbol{d}_x\}_{x \in \boldsymbol{O'^{,L}_t}}. \quad (9)$$

Although CNN-based video interpolation can generate accurate interpolated images for both uniform regions and edges, it cannot ensure the correctness of displacement vectors for pixels in the uniform areas. Therefore, instead of proposing a new architecture for VICNN, we modified the loss function used by VICNN by adding additional metrics to improve or optimize and further re-train VICNN using our own set of underwater training videos to precisely generate the displacement image $\boldsymbol{D_t^{L \to R}}$ for the underwater image pair $(\boldsymbol{O'^{,L}_t}, \boldsymbol{O'^{,R}_t})$.

We revised the training procedure of the original VICNN by integrating the total variation of the detected displacement vectors to ensure that the estimated displacement vectors will be very smooth. The authors of VICNN [9] used two input receptive patches $\boldsymbol{R_{i,1}} \boldsymbol{R_{i,2}}$ at the center of $(\boldsymbol{x_i}, \boldsymbol{y_i})$ with the corresponding input patches $\boldsymbol{P_{i,2}} \boldsymbol{P_{i,2}}$ which are smaller than the receptive field patches where both are also centered in the same location. $\tilde{\boldsymbol{C}}_{\boldsymbol{i}}$ is the ground truth color and $\boldsymbol{G_i}$ is the ground-truth gradient at $(\boldsymbol{x_i}, \boldsymbol{y_i})$. Initially, the loss function measures the difference between the interpolated pixel color and its corresponding ground-truth defined as:

$$\boldsymbol{E_C} = \sum_{\boldsymbol{i}} \left\| \left[\boldsymbol{P_{i,1}} \boldsymbol{P_{i,2}} \right] * \boldsymbol{K_i} - \tilde{\boldsymbol{C}}_{\boldsymbol{i}} \right\|_{\boldsymbol{L_1}} \quad (10)$$

where subscript i is the i^{th} training example and $\boldsymbol{K_i}$ is the output of the neural network's convolutional kernel. Using only the color loss and even with the integration of L_1 norm, which is the sum of the absolute values of the distances in the original space to preserve the edges of the image [70], still leads to a blurry result. The integration of gradients in the loss function corrects the shortcoming of the color loss. The gradients of the input patches are first computed, followed by convolution using the estimated kernel, which generates the gradient of the interpolated image at the pixel interest. Based on the eight immediate neighboring pixels, eight versions of gradients were computed using finite difference and all added into the gradient loss function defined as:

$$\boldsymbol{E_g} = \sum\nolimits_i \sum\nolimits_{k=1}^{8} \left\| \left[\boldsymbol{G^k_{i,1}} \boldsymbol{G^k_{1,2}} \right] * \boldsymbol{K_i} - \tilde{\boldsymbol{G}}^{\boldsymbol{k}}_{\boldsymbol{i}} \right\|_{l_1} \quad (11)$$

where k belongs to the eight ways to compute the gradient. Meanwhile, $\boldsymbol{G^k_{i,1}}$ and $\boldsymbol{G^k_{i,2}}$ are the gradients based on input patches $\boldsymbol{P_{i,2}} \boldsymbol{P_{i,2}}$ and the input ground-truth gradient is in $\tilde{\boldsymbol{G}}^{\boldsymbol{k}}_{\boldsymbol{i}}$. The final loss of VICNN combines the color and gradient loss as

$$E_f = E_c + \lambda \cdot E_g \quad (12)$$

where ($\lambda = 1$) determines the smoothness of the output. To verify the quality of the displacement vector estimation of the original loss function ($\boldsymbol{E_f}$) of VICNN, we also add TVL1 [71] factor as an additional term of (12), a popular approach to remove impulse noise and preserve image edges [70] to optimize the optical flow detection. We integrated the sum of the gradient values of the displacement vectors, *i.e.*, $\left|\nabla \boldsymbol{D_x^{L \to R}}\right| + \left|\nabla \boldsymbol{D_y^{L \to R}}\right|$ as the total variation function for x, y to smoothen the results of the detected displacement vectors. The final loss function $\boldsymbol{E_F}$ is denoted as

$$\boldsymbol{E_F} = (\boldsymbol{E_c} + \lambda \cdot \boldsymbol{E_g}) + (\left|\nabla \boldsymbol{Du^{l \to r}_{t,x}}\right| + \left|\boldsymbol{Du^{l \to r}_{t,y}}\right|). \quad (13)$$

The gradient provides shape information and TVL1 for temporal information and will increase its displacement vector estimation's overall robustness and reliability.

In training the neural network, image ground truth is needed to train the parameters of the neural network. We follow the concept of VICNN using three consecutive video frames of both the left $(\boldsymbol{I^L_t}, \boldsymbol{I^L_{t+1}}, \boldsymbol{I^L_{tC2}})$ and right $(\boldsymbol{I^R_t}, \boldsymbol{I^R_{t+1}}, \boldsymbol{I^R_{tC2}})$ incorporating the second frame $(\boldsymbol{I^R_{t+1}})$ as our ground truth and will help determine the final parameters of VICNN to generate the displacement vectors using (8). Figure 4 shows an example to estimate the final disparity image of the target based on the object-based stereo matching algorithm.

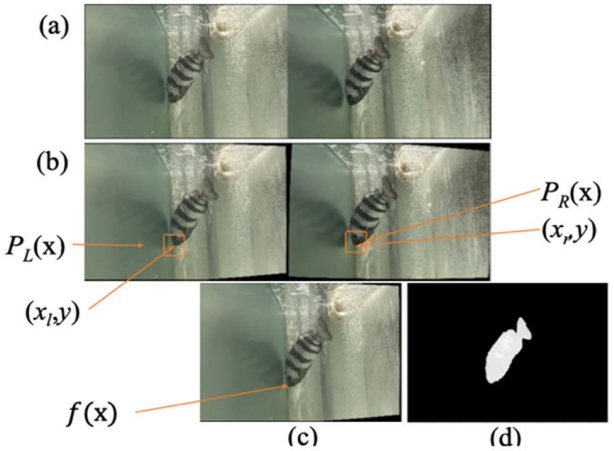

FIGURE 4. An example of the underwater fish object disparity estimation using the proposed approach: (a) the original stereo images; (b) the rectified stereo images; (c) the interpolated image of (b); (d) the computed disparity image of the fish object in (c) [67].

C. 3D OBJECT RECONSTRUCTION FOR FISH METRIC ESTIMATION

Once the disparity value of pixel i in $\boldsymbol{O_t^L}$ has been computed, the depth value of the pixel and its 3D coordinates can be calculated as

$$[X_i, Y, Z_i] = \left[(x_i - c_x) * Z_i/_f, (y_i - c_y) * Z_i/_f, f * b/_{d_i}\right] \tag{14}$$

where f is the camera's focal length; b is the baseline, defined as the distance between the centers of the left and right cameras; (c_x, c_y) is the optical center in the image plane; d_i is the computed disparity value of pixel i. For each pixel in the object $\boldsymbol{O_t^L}$, we use Eq. (14) to compute its corresponding 3D point. This also defines the point cloud $\boldsymbol{P_t^L} = \{(X_i, Y_i, Z_i)\}_{i=1}^{|O_t^L|}$ of the fish contained in $\boldsymbol{O_t^L}$. Then we perform Singular Values Decomposition (SVD) for the 3D point cloud $\boldsymbol{P_t^L}$ and the matrix can be disassembled to generate the following:

$$\boldsymbol{P_t^L} = \boldsymbol{U}\Sigma\boldsymbol{V^T} \tag{15}$$

where Σ is a diagonal matrix composed of three eigenvalues $\lambda_i = 1, \ldots, 3$; the columns of $\boldsymbol{U}$ **and** $\boldsymbol{V}$ are called the left-singular vectors and right-singular vectors of $\boldsymbol{P_t^L}$, respectively. Without loss of the generality, we often have $V = [\mathbf{v_i}]_{i=1}^3$, where $\mathbf{v_i}$ is the i-th eigenvector of the matrix $\boldsymbol{P_t^L}$.

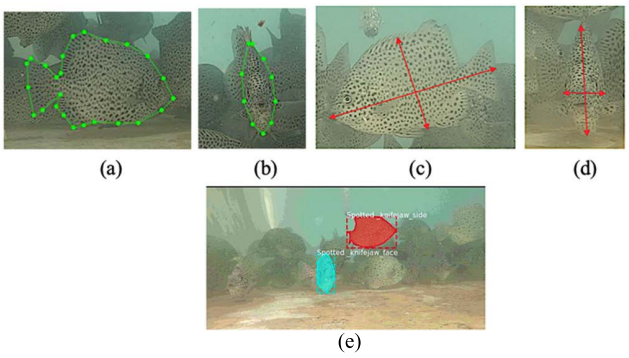

FIGURE 5. The fish objects are labeled either 'side' or 'front' view for training our semantic segmentation CNN: (a) example of the 'side view' fish; (b) example of the 'front view' fish; (c) the body length L and body height H of (a); (d) body height H and body width W of (b); (e) segmented fish objects with correct posture labels from the input image in the testing phase.

For each point $\mathbf{p} \in \boldsymbol{P_t^L}$, we project p onto the three eigenvectors **v1, v2, v3** (defined by the matrix $\boldsymbol{V}$) to get the new 3D coordinate points:

$$[X_i', Y_i', Z_i'] = \left[< X_i', \mathrm{v}_1 >, < Y_i', \mathrm{v}_2 >, < Z_i', \mathrm{v}_3 >\right], \quad i = 1, \ldots, |O_t^L|. \tag{16}$$

The converted coordinate points estimate the body length L, body height H, and body width W of the fish based on the fish posture recognition result. As shown in Figure 5, the training fish objects are labeled either 'side view' or 'front view' to train our semantic segmentation CNN to segment fish objects with correct posture labels from the input image in the testing phase. Based on the posture label and the converted coordinate points using (16), the formula to estimate the fish metrics is as follows:

$$\begin{bmatrix} L \\ H \\ W \end{bmatrix} = \begin{bmatrix} \max\limits_i(x_i) - \min(x_i) \\ \max\limits_i(y_i) - \min(y_i) \\ \max\limits_i(z_i) - \min(z_i) \end{bmatrix}, \textit{ if } O_t^L \text{ is' } \textit{side view'}. \tag{17}$$

In this work, the objects identified as 'side view' class are considered the reliable ones for fish metric estimation, while the 'front view' objects are skipped to avoid extra noise in the fish metric measurement results.

Another factor that affects the estimation accuracy of underwater fish metrics is the distance b between the left and right camera lenses. The larger the value of b, the more accurate the fish metrics will be. However, increasing the distance requires a large stereo camera which is difficult to use in offshore cages. Thus, we used a small value of b ($b = 11.4$ cm) to set up the camera though it is limited in measuring the metrics of the fish far away from the camera.

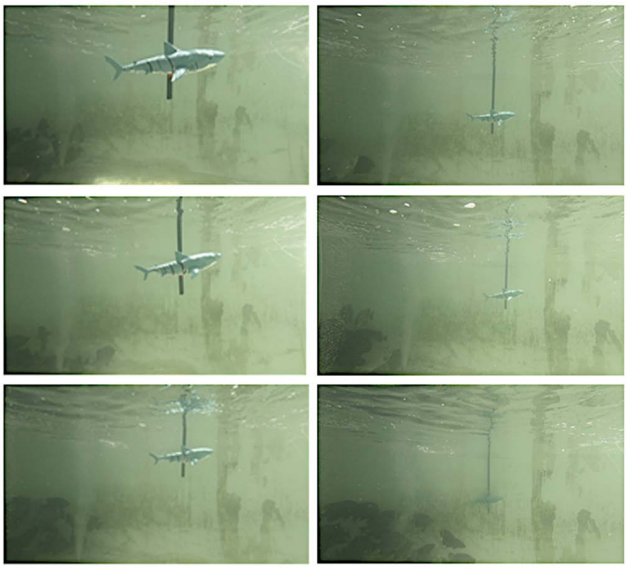

FIGURE 6. Images of fake fish at distances of 70, 90, 120, 150, 180, and 200 cm from the stereo camera lens.

We perform different experiments with this set of cameras to determine effective ranges. First, we used a fake fish with a body length of 30 cm with a distance of 70 cm and 90 cm from the lens. To test further, we also used shooting distances of 120 cm, 150 cm, 180 cm, and 200 cm, as shown in Figure 6. After calibrating the captured images, we used the method proposed in this paper to estimate the body length and use the error between the estimated body length and the actual body length for comparison. Finally, we compared the results of the estimated body length of the fish using our captured fake fish images using different distances to get the best result. Based on Table 1, the highest error rate (9.6%) is at a 200 cm distance and is considered the maximum effective distance range from the camera lens. Therefore, fish more than 200 cm

away from the camera lens are discarded in the body length estimation due to a higher error rate.

TABLE 1. The percentage error in estimating the body length of the fake fish using different distance.

Distance between the lens and the fake fish	Fake fish body length estimation	Error (%)
70 cm	30.7 cm	2%
90 cm	30.5 cm	1.6 %
120 cm	30.2 cm	0.6 %
150 cm	30.6 cm	2 %
180 cm	31.6 cm	5 %

IV. EXPERIMENTAL SET-UP AND DATASETS

We used two GoPro cameras to attain the stereo camera lens setup, as shown in Figure 3(a). The computer used for training the neural network for object detection, image matching, and 3D reconstruction is an Intel Core i7 -8700 3.2GHz CPU, 32.0GB RAM, and NVIDIA GeForce GTX 1080ti GPU using the Python environment. The Pool 13 of the University Aquatic Center is located at the National Taiwan Ocean University in Keelung City, with 150 porphyry sculptures, and the Pingtung Hengchun Aquaculture site with approximately 40,000 golden pomfret was the experimental environment. Two environments were considered to ensure that our approach works for less dense and highly dense aquaculture tanks or cages. The video collected from these locations was used to train and test the neural network using the stereo camera to capture the left and right images.

Figure 7 shows the labeled images for training our semantic segmentation CNN and the video interpolation CNN. Four videos with a total of 2 hours, 25 minutes, and 3 seconds were taken from the A13 Aquatic Center Pond, while four videos with a total of 1 hour, 29 minutes, and 41 seconds from taken from the Henchun Ocean open-sea cage for the experiment. Leave-one-out cross-validation technique was employed for the distribution of our training and testing datasets which is very appropriate for small datasets. We utilized n - 1 video data for training while the remaining video was used for testing. The same process is repeated until the last video is utilized. The video used for test data in the previous iteration is no longer used for the next iteration.

V. EXPERIMENTAL RESULTS

The stereo camera captured the left and right underwater images and passed through the Mask_RCNN neural network, where the fish and the background are segmented through instance segmentation. The experiment uses two fish species: the ponds contain porphyria sea bream, and the Hengchun ocean has the golden pomfret. In training the Mask-RCNN neural network to perform instance segmentation, we manually labeled the images and utilized 200 images for the training data and 500 images for the testing data taken from the Aquatic Center with an accuracy rate of 90%. The segmentation results are shown in Figure 8. Meanwhile, from the data collected from the Hengchun aquaculture site, 500 images were used for the training and 800 images for the testing with an accuracy rate of 85%, and segmentation results are shown in Figure 8(b).

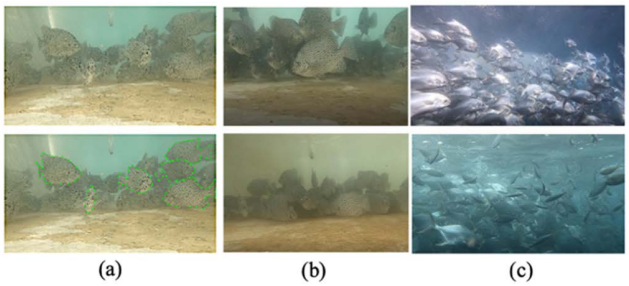

FIGURE 7. Stereo video datasets for training and testing: (a) images from the Aquatic Center; the upper row is the original image, and the bottom is the labeled data; (b) images captured from A13 Aquatic Center Pond; and (c) images captured from Henchun Ocean open-sea cage.

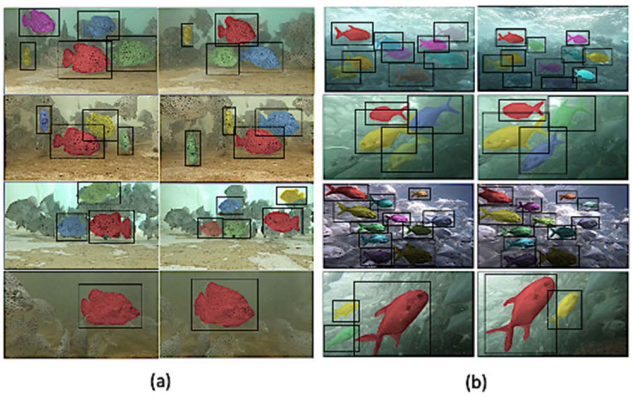

FIGURE 8. Segmentation results: (a) A13 pool in the Aquatic Center and (b) from the Hengchun offshore fish cage.

The Mask-RCNN training and testing loss result is shown in Figure 9. As reflected in the loss curves for training and testing, the loss value close to zero is at the 200th iteration. Segmenting underwater target objects from their background is challenging considering fish's continuous or active movement, varied textures, quality of water, and luminosity. These problems should be regarded as a requirement to create a robust and accurate fish detection for fish length and density estimation.

The detection accuracy results for the two data collection sites for Mask-RCNN are shown in Table 2. Accuracy has different results for the two environments since the quality of the collected video data varies regarding water quality and turbidity and affects the identification of the target objects. Video collected from the pool environment with fewer fish populations has a higher accuracy rate of 95% compared to the open fish cage with a dense fish population of 90%.

After rectification, the proposed object matching scheme tracks objects across video frames. It searches the matched right object for each left object, whose initial disparity value

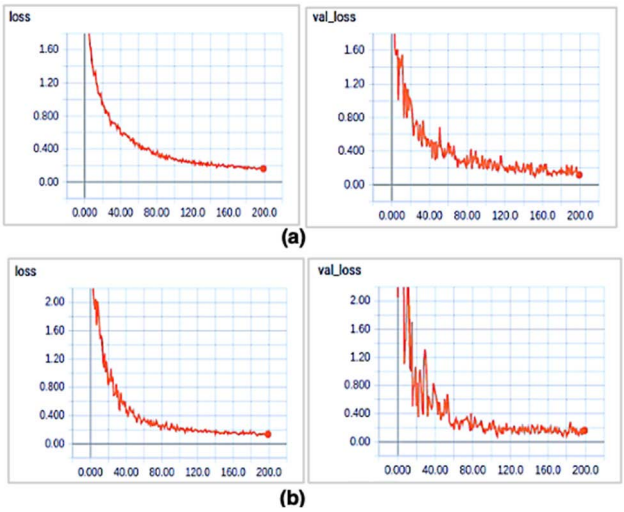

FIGURE 9. Mask-RCCNN (a) training loss and (b) test loss for the A13 pool at the Aquatic Center and (a) training loss and (b) test loss for Hengchun offshore fish cage.

TABLE 2. Mask-RNN detection accuracy results.

Field Area	Video Data	Accuracy
Aquatic Center A13 Pool	Video 1	92%
	Video 2	90%
	Video 3	95%
	Video 4	87%
Hengchun Offshore Fish Cage	Video 1	90%
	Video 2	84%
	Video 3	88%
	Video 4	87%

is determined by the displacement vector between the centers of the matched objects. The results of pairing the 3D object using the left and right images are shown in Figure 10. The matching accuracy for the images collected from the A13 pool of the Aquatic Center is 90%, while the Hengchun open-sea cage is 80%. The lower result from Hengchun is due to the highly dense fish population, making it hard to perform matching due to high fish object overlaps.

We used the Mask-RCNN to segment the fish object and separate it from the background, followed by image cropping. The segmented cropped image is used for initial disparity computation, and then the 3D object is obtained by subtracting the object's center point. But using such an approach will cause all pixels of the target object to be the same, with a disparity result. Furthermore, when such an approach is integrated into the 2D measurement method, there is a considerable loss of information with the original three-dimensional object.

We integrated interpolated signals using the cropped segmented images as input to the VICNN [38] for disparity fine-tuning to improve the result and achieve a minimal matching error. First, the VICNN will do the video warping and object interpolation, followed by optical flow estimation. Then, the resulted optical flow is used to obtain the disparity.

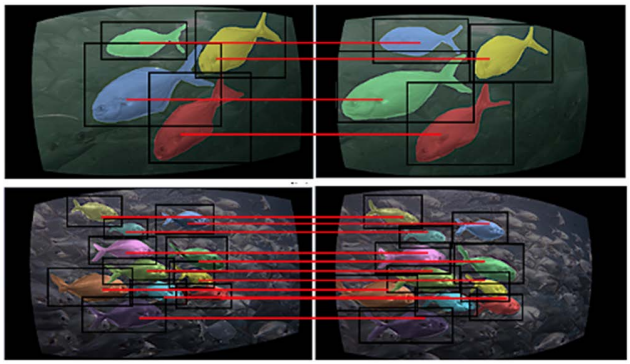

FIGURE 10. Matching results of 3D objects where the top is in the A13 pool in the Aquatic Center and the bottom is in Hengchun offshore fish cage.

The VICCNN is trained in advance, and the parameters have been optimized to get high-quality synthetic interpolated results. The neural network also produces an optimized pixel-wise kernel for the interpolation work, thus improves generates a pixel-wise residual disparity result. The comparison using the two fish cages of the initial disparity from the fine-tuned disparity is in Figures 11 and 12, which show changes in the original objects after disparity fine-tuning.

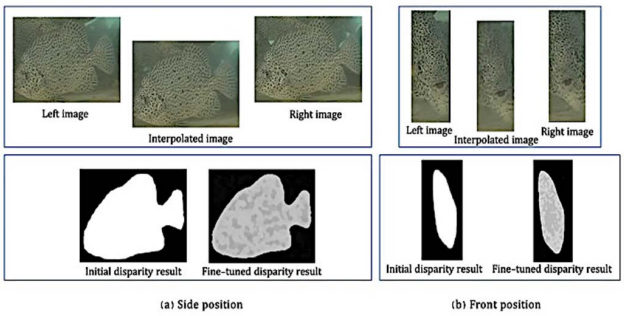

FIGURE 11. Initial and fine-tuned disparity results with the side and front fish positions in the A13 pool of the Aquatic Center.

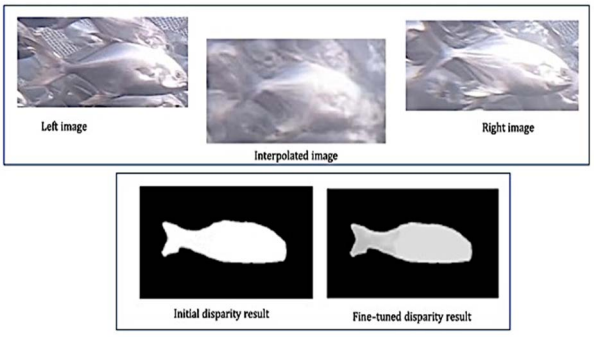

FIGURE 12. Initial and fine-tuned disparity results with the side and front fish positions in the Hengchun ocean open sea cage.

Figure 13 shows the results for comparing the disparity of the stereo image matching using our approach with

semi-global block matching (SGBM) [72] that integrates pixel-wise matching based on mutual information and the approximation of the global smoothness constraint. SGBM detects occlusions and disparities using subpixel accuracy. The results using our video interpolated optical flow, even with poor image quality due to water turbidity, have better disparity results with the results generated by SGBM. This is because the structure or appearance of the fish using our method is more visible and apparent. To calculate the disparity of pixel x in the object O_l, the depth value of pixel $\boldsymbol{x}$ and its 3D coordinates X, Y, Z are $[X, Y, Z] = [(\boldsymbol{x}-\boldsymbol{c_x}) * Z/f, (y - c_y) * Z/f, f * b/\boldsymbol{d}_x]$, where f is the focal length of the camera and b is the baseline or the distance between the center of the left lens and the center of the lens. The said formula is used to convert the 2D coordinate of the object $\boldsymbol{O_l}$ into a 3D point cloud coordinate. To perform filtering for better results, we modified the outlier point cloud.

Based on our fake fish experiment results in Table 2, we set the most effective stereo camera lens distance at 200 cm. Fish beyond that distance are discarded since they will obtain a significant error affecting our fish length estimation accuracy. The stereo camera system is an affordable device to get the image depth and can be used to estimate object size, but it is only limited to a certain distance; such intervention was made to address its limitations and ensure the best optimization results.

We also considered the continued movement of the fish, where their body and tails swing while swimming, which makes their estimated body length different for each frame. We used a tracking method to estimate the fish body length in each frame and calculate the average with the accumulated body lengths of other frames. The obtained average value will be used as the final estimated body length.

To verify the error value with the actual body length of the fish, we track a single fish and measure its body length in full frames. We first manually measured the exact body length of the single fish (30.8 cm), placed it in the water tank, and took a video using our low-cost stereo camera. We tracked this fish by capturing the left and the right images and then estimated the body length in full frames. Since the estimated body length of the fish is not consistent for each frame due to its continuous movement, we considered the final body length by getting the average of the body length for each frame.

The body length estimation of the single fish using 20 frames is in Figure 14, where the average body length is 20.895 cm, with an average error of 2.38%. The maximum error is in Frame 2 at 5.52% error. The average body length is computed as $\frac{1}{n}\sum_{i=1}^{n} X_i$ while the average error is calculated using $\frac{1}{n}\sum_{i=1}^{n} E_i$ where X_i is the estimated body length and E_i is the error value for each frame n. To calculate the length error value (e) for each frame n, we use

$$e = \frac{\text{actual body length estimated body length}}{\text{actual body length}} \times 100.$$

Figure 15 is an illustrative diagram of our proposed effective range filtering, executed after the 3D estimation.

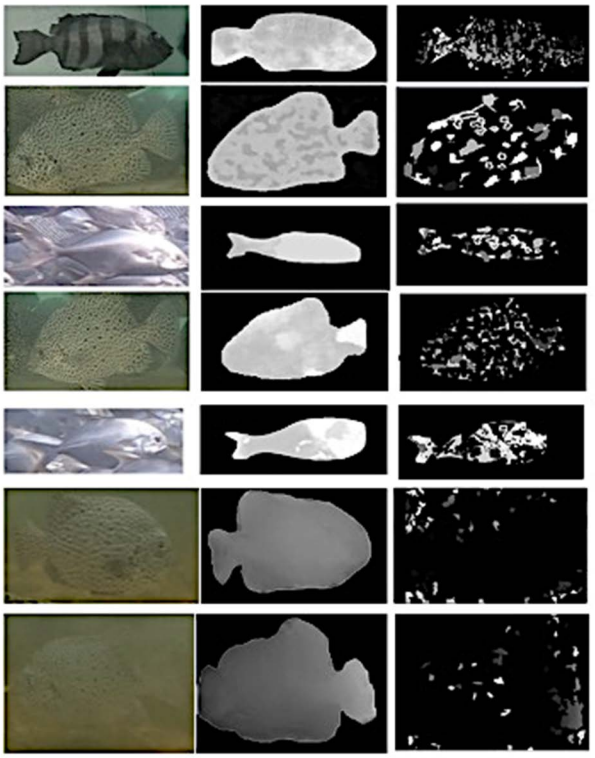

FIGURE 13. Image matching disparity comparison where (a) is the original image, and the generated disparity results are (b) our video interpolated optical flow and (c) the semi-global block matching [72].

Frame No.	Actual body length	Estimated body length	Length error value
1	30.8	31.8	3.25
2	30.8	32.5	5.52
3	30.8	30.4	1.3
4	30.8	29.9	2.92
5	30.8	30.6	0.65
6	30.8	31.2	1.30
7	30.8	31.9	3.57
8	30.8	30.5	0.97
9	30.8	30	2.60
10	30.8	31.4	1.95
11	30.8	32.1	4.22
12	30.8	31.5	2.27
13	30.8	30.4	1.3
14	30.8	30.1	2.27
15	30.8	29.8	3.25
16	30.8	30.3	1.62
17	30.8	31.2	1.30
18	30.8	31.6	2.60
19	30.8	31.1	0.97
20	30.8	29.6	3.90

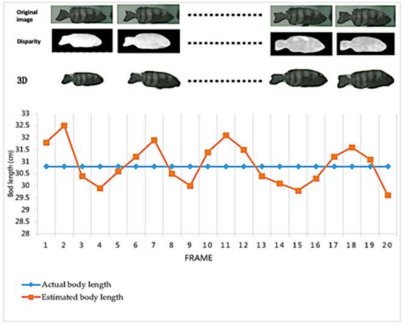

FIGURE 14. Tracking result of a single fish using 20 frames.

Figure 15(a) is the result of the instance segmentation from the collected video from Cage-1; (b) is the disparity image result, and the blue dots in (c) represent 12 segmented fish images from (a). The depth value of the fish is the 3D point cloud depth value where the fish is closest to the lens. Therefore, the effective range is filtered based on the depth value, and only the fish within the 50 - 200 cm range was estimated in terms of body length.

Estimating the stereo-correspondence in the underwater environment is very difficult and is affected by water turbidity and varying ambient lighting. Water current is also a factor in an open sea cage where fish freely swims in the broader area. Fish also deform as they swim, which may cause

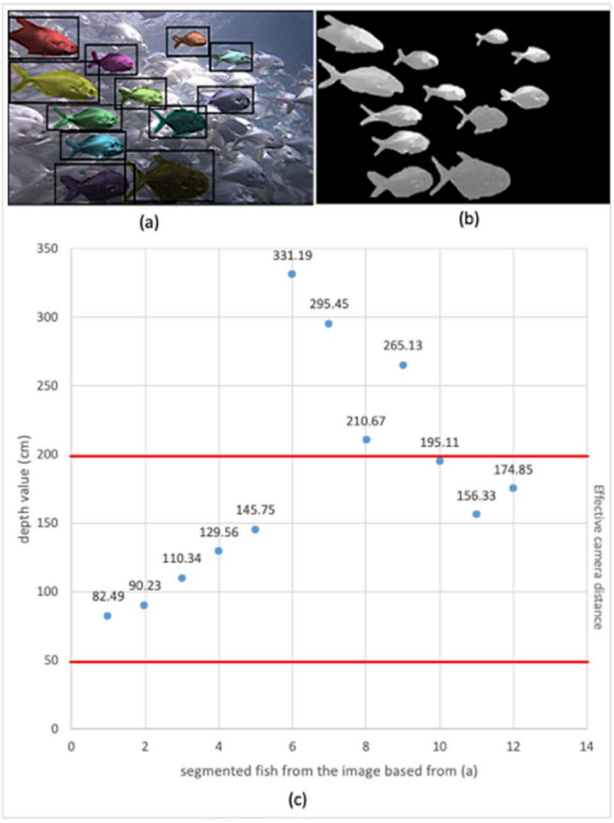

FIGURE 15. 3D estimation using an effective range where (a) is the instance segmentation result of one frame, (b) is the disparity image, and (c) is the chart representing the depth value of each fish detected and the depth value of the 3D point cloud where the fish is closest to the lens. The effective range is filtered based on the depth value using the range 50 to 200 cm within the red line in the graph; only those within the effective lens distance are estimated in body length.

self-occlusions [66]. To make the system robust, tracking the fish for a more extended period is essential. We performed a tracking mechanism estimating a precise length of a fish that was previously measured and are swimming freely in a tank using 20 frames to consider the different fish positions. The fish length is estimated in each frame and aggregated all the length estimates in each track.

The information in Table 3 shows the manual measurement of the body length of the fish and the estimation results. Its corresponding estimation error using 2D, 3D, and estimation using the most effective camera lens distance (50 cm to 200 cm). The fish outside the effective range were discarded due to a high error rate based on the earlier experiment.

For the 2D measurement, we used the flat left and right images and combined them to generate a single image as the source image. The fish is first segmented, and the straight line of the segmented fish body part is used as the fish length. The fish length is calculated from its nose to the tail fork with the longest axis using the pixel-wise measurement of the segmented fish image. The 2D measurement has some problems since the body of the free-swimming fish is not straight, which makes the length measurement inaccurate. For the 3D measurement, we used the depth and 3D coordinate positions since there are instances where the fish body is in a curve form or different posture. We measured the length of the fish by the distance of the fish from the camera. The results show that the error of the estimated body length after the 3D reconstruction is significantly reduced compared to the results from the 2D estimation. Integrating the effective camera distance for the Porphyry seabream fish located in a smaller pond area seems insignificant in the error reduction since it has almost the same results as the 3D estimation, unlike the results for the golden pomfret with a significant error reduction. The effectiveness of camera distance is more relevant to large fish cages, as manifested in the result for golden pomfret located in an open cage with more transparent water and a larger fish cage size. When the water is clear, the fish can still be seen even far from the stereo camera lens; thus, discarding the fish outside the effective camera distance will significantly reduce error for large cages.

TABLE 3. Manual fish size measurement and the different methods used to estimate the fish body length.

Fish Species	Videos	Body length (cm)						
		Manual Measurement	2D estimation		3D estimation		Effective range 3D estimation	
			Est. value	Error	Est. value	Error	Est. value	Error
Porphyry seabream	A13-1	20.485	26.43	29.02%	20.25	1.15%	20.25	1.15%
	A13-2	20.485	26.54	29.56%	21.13	3.15%	21.13	3.15%
	A13-3	20.485	25.22	23.11%	21.78	6.32%	20.61	0.61%
	A13-4	20.485	28.86	40.88%	21.22	3.59%	21.22	3.59%
Golden pomfret	Cage-1	30.3	43.42	43.3%	37.82	24.81%	28.87	4.71%
	Cage-2	30.3	45.64	50.62%	37.11	22.47%	32.14	6.07%
	Cage-3	32.374	47.41	46.44%	37.22	14.97%	31.34	3.19%
	Cage-4	32.374	46.96	45.05%	39.56	22.2%	30.23	6.62%

VI. CONCLUSION AND FUTURE WORKS

Our proposed approach provides fish metric estimation for the fish species Porphyry seabream and Golden pomfret with only less than a 5% error rate when compared to the manual measurement when the fish is tracked using multiple frames. For our effective distance range, we established a distance from the camera range to be 50 cm to 200 cm only since fish outside this range tend to give high accuracy error. The distance limitations are one of the drawbacks of using a low-cost camera system. For the 3D metric estimation in the natural aquaculture environment, incorporating an effective range has the lowest maximum error rate of 3.59% and 6.62% for the low and high dense cage, respectively. These results significantly improved the 3D metric estimation, especially for highly dense fish cages. Meanwhile, the 2D measurement has the highest error rate of 28.86% and 47.41% for these two types of cages. With these results, our proposed stereo camera system.

To deal with the limitations of the distance or range of the low-cost stereo camera system, more mechanisms should be incorporated to increase the accurate range or distance for the measurement to ensure that more fish objects are covered for better accuracy results. For the next step of our research, we will add more video datasets to train our instance segmentation neural network, especially for highly dense fish cages,

to improve its accuracy performance. Increasing the fish segmentation result will significantly affect the performance of our fish length estimation. We also plan to combine a stereo camera with a sonar system to generate a more accurate and precise method for measuring fish body length and weight. Fish species identification using the sonar enables the relative sonar to provide a depth reference value for 3D images, so combining them will improve the estimation accuracy.

Despite the limitations of the low-cost stereo camera system, its capabilities are already promising to provide an automatic and non-invasive fish metric estimation that reduces fish stress and can help farmers determine their current fish farm conditions. In addition, the information provided by the fish metric estimation can be integrated to establish fish weight conversion and growth curve, and incorporating the feeding data can derive the meat exchange rate.

ACKNOWLEDGMENT

The authors would like to acknowledge Prof. Yi-Zeng Hsieh of the Department of Electrical Engineering, National Taiwan Ocean University, for his valuable contribution during the early stage of this research work.

REFERENCES

[1] Immersed.Io. *What Computer Vision Means to AR and VR*. Accessed: Jun. 19, 2022. [Online]. Available: https://immersed.io/computer-vision-means-ar-vr/

[2] V. Lepetit, "On computer vision for augmented reality," in *Proc. Int. Symp. Ubiquitous Virtual Reality*, Jul. 2008, pp. 13–16.

[3] V. Thomas, S. Daniel, and J. Pouliot, "3D modeling for mobile augmented reality in unprepared environment," in *Advances in 3D Geo-Information Sciences* (Lecture Notes in Geoinformation and Cartography). Berlin, Germany: Springer, 2011, pp. 163–177.

[4] Y. Shirai, "3D computer vision and applications," in *Proc. 11th IAPR Int. Conf. Pattern Recognit.*, 1992, pp. 1–14.

[5] W. Luo, A. G. Schwing, and R. Urtasun, "Efficient deep learning for stereo matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5695–5703.

[6] A. Krishnan and J. Kollipara, "Cost-effective stereo vision system for mobile robot navigation and 3D map reconstruction," *Comput. Sci. Inf. Technol.*, vol. 4, pp. 75–86, Jan. 2014.

[7] D. Murray and J. J. Little, "Using real-time stereo vision for mobile robot navigation," *Auto. Robots*, vol. 8, no. 2, pp. 161–171, 2000.

[8] H. Ghazouani, M. Tagina, and R. Zapata, "Robot navigation map building using stereo vision based 3D occupancy grid," *J. Artif. Intell., Theory Appl. (JAITA), HyperSci.*, vol. 1, no. 3, pp. 63–72, 2011.

[9] A. Milella and G. Reina, "3D reconstruction and classification of natural environments by an autonomous vehicle using multi-baseline stereo," *Intell. Service Robot.*, vol. 7, pp. 72–92, Apr. 2014.

[10] J. Muhovič and J. Perš, "Correcting decalibration of stereo cameras in self-driving vehicles," *Sensors*, vol. 20, no. 11, p. 3241, Jun. 2020.

[11] L. Priya and S. Anand, "Object recognition and 3D reconstruction of occluded objects using binocular stereo," *Cluster Comput.*, vol. 21, no. 1, pp. 29–38, 2018.

[12] S. Khan. *Multi-View Approaches to Tracking, 3D Reconstruction and Object Class Detection*. Accessed: Jun. 19, 2022. [Online]. Available: https://stars.library.ucf.edu/etd/3506

[13] F. Simoes, M. Almeida, M. Pinheiro, R. D. Anjos, A. D. Santos, R. Roberto, V. Teichrieb, C. Suetsugo, and A. Pelinson, "Challenges in 3D reconstruction from images for difficult large-scale objects: A study on the modeling of electrical substations," in *Proc. 14th Symp. Virtual Augmented Reality*, May 2012, pp. 74–83.

[14] M. Scaioni, J. Crippa, L. Longoni, M. Papini, and L. Zanzi, "Mage-based reconstruction and analysis of dynamic scenes in a landslide simulation facility," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. IV-5/W1, pp. 63–70, Dec. 2017.

[15] C. Popescu, B. Täljsten, T. Blanksvärd, and L. Elfgren, "3D reconstruction of existing concrete bridges using optical methods," *Struct. Infrastruct. Eng.*, vol. 15, no. 7, pp. 912–924, 2019.

[16] L. Puglia and C. Brick, "Deep learning stereo vision at the edge," 2020, *arXiv:2001.04552*.

[17] H. Rosas, "Perception and reality in stereo vision: Technological applications," in *Advances in Stereo Vision*. London, U.K.: IntechOpen, 2011.

[18] J. Ashworth and K. Brasher, *3D Reconstruction: Methods, Applications and Challenges*. Hauppauge, NY, USA: Nova Science, 2011.

[19] Y. L. Z. Wang, G. Huang, B. Wang, L. van der Maaten, M. Campbell, and K. Weinberger, "Anytime stereo image depth estimation on mobile devices," in *Anytime Stereo Image Depth Estimation Mobile Devices*. Montreal, QC, Canada, Canada, 2019.

[20] D. Freedman, Y. Smolin, E. Krupka, I. Leichter, and M. Schmidt, "SRA: Fast removal of general multipath for ToF sensors," in *Proc. Eur. Conf. Comput. Vis.* Zurich, Switzerland, 2014, pp. 234–249.

[21] M. Bleyer, C. Rhemann, and C. Rother, "PatchMatch stereo–stereo matching with slanted support windows," in *Proc. Brit. Mach. Vis. Conf.*, 2011, pp. 1–11.

[22] D. A. Butler, S. Izadi, O. Hilliges, D. Molyneaux, S. Hodges, and D. Kim, "Shake'n'sense: Reducing interference for overlapping structured light depth cameras," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, May 2012, pp. 1933–1936.

[23] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, pp. 7–12, Apr. 2002.

[24] S. R. Fanello, J. Valentin, C. Rhemann, A. Kowdle, V. Tankovich, P. Davidson, and S. Izadi, "UltraStereo: Efficient learning-based matching for active stereo systems," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6535–6544.

[25] M. Hao, H. Yu, and D. Li, "The measurement of fish size by machine vision—A review," in *Proc. Int. Conf. Comput. Comput. Technol. Agricult.* Jilin, China, 2016, pp. 15–32.

[26] D. Li, Y. Hao, and Y. Duan, "Nonintrusive methods for biomass estimation in aquaculture with emphasis on fish: A review," *Rev. Aquaculture*, vol. 12, no. 3, pp. 1390–1411, Aug. 2020.

[27] M. Rahim, N. Abdullah, I. Amin, Z. Mohammad Zaidi, M. Man, and N. Othman, "A new approach in measuring fish length using FiLeDI framework," *Int. Arab J. Inf. Technol.*, vol. 9, no. 2, pp. 1683–3198, 2012.

[28] C. Maia, A. Ferreira, M. Oroszlányová, M. Azevedo, P. Ipma, P. Lisbon, R. Gaspar, C. Silva, S. Santos, and G. Menezes, "FishMetrics: A new integrated solution for fisheries automatic and remote size data collection for fish stock assessment," in *Proc. 3rd Int. Conf. Elect., Electron., Eng. Trends, Commun., Optim. Sci.* Andhra Pradesh, India, 2016, pp. 1–7.

[29] C. Shi, Q. Wang, X. He, X. Zhang, and D. Li, "An automatic method of fish length estimation using underwater stereo system based on LabVIEW," *Comput. Electron. Agricult.*, vol. 173, Jun. 2020, Art. no. 105419.

[30] J. L. Boldt, K. Williams, C. N. Rooper, R. H. Towler, and S. Gauthier, "Development of stereo camera methodologies to improve pelagic fish biomass estimates and inform ecosystem management in marine waters," *Fisheries Res.*, vol. 198, pp. 66–77, Feb. 2018.

[31] D. Perez, F. J. Ferrero, I. Alvarez, M. Valledor, and J. C. Campo, "Automatic measurement of fish size using stereo vision," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (I2MTC)*, May 2018, pp. 1–6.

[32] M. O'Byrne, V. Pakrashi, F. Schoefs, and A. B. Ghosh, "Semantic segmentation of underwater imagery using deep networks trained on synthetic imagery," *J. Mar. Sci. Eng.*, vol. 6, no. 3, p. 93, Aug. 2018.

[33] B. G. Teo and S. K. Dhillon, "An automated 3D modeling pipeline for constructing 3D models of MONOGENEAN HARDPART using machine learning techniques," *BMC Bioinf.*, vol. 20, no. S19, pp. 1–21, Dec. 2019.

[34] W. Chen, J. Gao, H. Ling, E. Smith, J. Lehtinen, A. Jacobson, and S. Fidler, "Learning to predict 3D objects with an interpolation-based differentiable renderer," in *Proc. Annu. Conf. Neural Netw. Process. Syst. (NeurIps)* Vancouver, BC, Canada, 2019, pp. 9609–9619.

[35] X.-F. Han, H. Laga, and M. Bennamoun, "Image-based 3D object reconstruction: State-of-the-art and trends in the deep learning era," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1578–1604, May 2021.

[36] M. Yang, J. Hu, C. Li, G. Rohde, Y. Du, and K. Hu, "An in-depth survey of underwater image enhancement and restoration," *IEEE Access*, vol. 7, pp. 123638–123657, 2019.

[37] R. Schettini and S. Corchs, "Underwater image processing: State of the art of restoration and image enhancement methods," *EURASIP J. Adv. Signal Process.*, vol. 2010, no. 1, pp. 1–14, Dec. 2010.

[38] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive separable convolution," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 261–270.
[39] N. Ubina, S.-C. Cheng, C.-C. Chang, and H.-Y. Chen, "Evaluating fish feeding intensity in aquaculture with convolutional neural networks," *Aquacultural Eng.*, vol. 94, Aug. 2021, Art. no. 102178.
[40] K. Zhou, X. Meng, and B. Cheng, "Review of stereo matching algorithms based on deep learning," *Comput. Intell. Neurosci.*, vol. 2020, pp. 1–12, Mar. 2020.
[41] G. Yao, A. Yilmaz, F. Meng, and L. Zhang, "Review of wide-baseline stereo image matching based on deep learning," *Remote Sens.*, vol. 13, no. 16, p. 3247, Aug. 2021.
[42] Z. Liang, Y. Guo, Y. Feng, W. Chen, L. Qiao, L. Zhou, J. Zhang, and H. Liu, "Stereo matching using multi-level cost volume and multi-scale feature constancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 300–315, Jan. 2021.
[43] J. Zhang, Y. Zhang, C. Wang, H. Yu, and C. Qin, "Binocular stereo matching algorithm based on MST cost aggregation," *Math. Biosciences Eng.*, vol. 18, no. 4, pp. 3215–3226, 2021.
[44] "Semi-global stereo matching with adaptive window based on grayscale value," *J. Image Graph.*, vol. 24, no. 8, pp. 1381–1390, 2019.
[45] D. Jia, Z. Wang, and J. Liu, "Research on flame location based on adaptive window and weight stereo matching algorithm," *Multimedia Tools Appl.*, vol. 79, no. 11, pp. 7875–7887, 2020.
[46] Y. Han, W. Liu, X. Huang, S. Wang, and R. Qin, "Stereo dense image matching by adaptive fusion of multiple-window matching results," *Remote Sens.*, vol. 12, no. 19, p. 3138, Sep. 2020.
[47] X. Huang, W. Liu, and R. Qin, "A window size selection network for stereo dense image matching," *Int. J. Remote Sens.*, vol. 41, no. 12, pp. 4838–4848, Jun. 2020.
[48] G. Yao, A. Yilmaz, F. Meng, and L. Zhang, "Review of wide-baseline stereo image matching based on deep learning," *Remote Sens.*, vol. 13, no. 16, p. 3247, Aug. 2021.
[49] J. Xiao, D. Ma, and S. Yamane, "Optimizing 3D convolution kernels on stereo matching for resource efficient computations," *Sensors*, vol. 21, no. 20, p. 6808, Oct. 2021.
[50] C. Cheng, H. Li, and L. Zhang, "Two-branch convolutional sparse representation for stereo matching," *IEEE Access*, vol. 9, pp. 21910–21920, 2021.
[51] X. Song, G. Yang, X. Zhu, H. Zhou, Y. Ma, Z. Wang, and J. Shi, "AdaStereo: An efficient domain-adaptive stereo matching approach," *Int. J. Comput. Vis.*, vol. 130, no. 2, pp. 226–245, Feb. 2022.
[52] H. Wang, R. Fan, P. Cai, and M. Liu, "PVStereo: Pyramid voting module for end-to-end self-supervised stereo matching," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4353–4360, Jul. 2021.
[53] F. Liu and M. Fang, "Semantic segmentation of underwater images based on improved Deeplab," *J. Mar. Sci. Eng.*, vol. 8, no. 3, p. 188, Mar. 2020.
[54] X. Jia, W. Chen, Z. Liang, X. Luo, M. Wu, C. Li, Y. He, Y. Tan, and L. Huang, "A joint 2D–3D complementary network for stereo matching," *Sensors*, vol. 21, no. 4, p. 1430, Feb. 2021.
[55] M. Wei, M. Zhu, Y. Wu, J. Sun, J. Wang, and C. Liu, "A fast stereo matching network with multi-cross attention," *Sensors*, vol. 21, no. 18, p. 6016, Sep. 2021.
[56] C. Cheng, H. Li, and L. Zhang, "Two-branch deconvolutional network with application in stereo matching," *IEEE Trans. Image Process.*, vol. 31, pp. 327–340, 2022.
[57] X. Ma, Z. Zhang, D. Wang, Y. Luo, and H. Yuan, "Adaptive deconvolution-based stereo matching net for local stereo matching," *Appl. Sci.*, vol. 12, no. 4, p. 2086, Feb. 2022.
[58] Y. Xu, D. Yu, Y. Ma, Q. Li, and Y. Zhou, "Underwater stereo-matching algorithm based on belief propagation," *Signal, Image Video Process.*, to be published.
[59] H. Zhao and Y. Wan, "Disparity refinement based on feature classification and local propagation for stereo matching," in *Proc. Int. Conf. Comput. Vis., Appl., Design (CVAD)*, Dec. 2021, pp. 53–58.
[60] N. Petrellis, "Measurement of fish morphological features through image processing and deep learning techniques," *Appl. Sci.*, vol. 11, no. 10, p. 4416, May 2021.
[61] N. S. Abinaya, D. Susan, and R. K. Sidharthan, "Deep learning-based segmental analysis of fish for biomass estimation in an occulted environment," *Comput. Electron. Agricult.*, vol. 197, Jun. 2022, Art. no. 106985.
[62] C. Yu, X. Fan, Z. Hu, X. Xia, Y. Zhao, R. Li, and Y. Bai, "Segmentation and measurement scheme for fish morphological features based on mask R-CNN," *Inf. Process. Agricult.*, vol. 7, no. 4, pp. 523–534, 2020.
[63] C. H. Tseng, C.-L. Hsieh, and Y.-F. Kuo, "Automatic measure," *Inf. Process. Agricult.*, vol. 7, no. 4, pp. 523–534, 2020.
[64] C.-H. Tseng, C.-L. Hsieh, and Y.-F. Kuo, "Automatic measurement of the body length of harvested fish using convolutional neural networks," *Biosyst. Eng.*, vol. 189, pp. 36–47, Jan. 2020.
[65] C. Yu, Z. Hu, B. Han, P. Wang, Y. Zhao, and H. Wu, "Intelligent measurement of morphological characteristics of fish using improved U-Net," *Electronics*, vol. 10, no. 12, p. 1426, Jun. 2021.
[66] P. Risholm, A. Mohammed, T. Kirkhus, S. Clausen, L. Vasilyev, O. Folkedal, Ø. Johnsen, K. H. Haugholt, and J. Thielemann, "Automatic length estimation of free-swimming fish using an underwater 3D range-gated camera," *Aquacultural Eng.*, vol. 97, May 2022, Art. no. 102227.
[67] N. Ubina, S.-Y. Cai, S.-C. Cheng, C.-C. Chang, and Y.-Z. Hsieh, "Underwater 3D object reconstruction for fish length estimation using convolutional neural networks," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst. (ISPACS)*, Nov. 2021, pp. 1–2.
[68] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.* Venice, Italy, Oct. 2017, pp. 2961–2969.
[69] K.-J. Yoon and I. S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 650–656, Apr. 2006.
[70] V. Estrela, H. Magalhaes, and O. Saotome, "Total variation applications in computer vision," in *The Handbook of Research on Emerging Perspectives in Intelligent Pattern Recognition, Analysis, and Image Processing*, vol. 2016. Hershey, PA, USA: IGI Global, 2016, pp. 41–64.
[71] J. Sánchez, E. Meinhardt-Llopis, and G. Facciolo, "TV-L1 optical flow estimation," *Image Process. Line*, vol. 3, pp. 137–150, Oct. 2013.
[72] H. Hirschmuller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 807–814.

NAOMI A. UBINA received the B.S. degree in information technology from St. Paul University Philippines, in 2001, and the Master of Science degree in computer science from the University of the Philippines Los Baños, in 2012. She is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung, Taiwan. She has been a part of the Teaching Staff of Isabela State University, Philippines. Her research interests include machine learning and computer vision.

SHYI-CHYI CHENG (Member, IEEE) received the B.S. degree from the National Tsing Hua University, Hsinchu, Taiwan, in 1986, and the M.S. and Ph.D. degrees in electronics engineering and computer science and information engineering from the National Chiao Tung University, Hsinchu, in 1988 and 1992, respectively. From 1992 to 1998, he was a Technical Staff with Chunghwa Telecom Laboratories, Taoyuan, Taiwan. He was a Faculty Member of the Department of Computer and Communication Engineering, National Kaohsiung First University of Science and Technology, Kaohsiung, Taiwan, from 1999 to 2005. He is currently a Professor with the Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung, Taiwan. His research interests include computer vision, image/video compression, communications, machine learning, and big data analytics.

CHIN-CHUN CHANG (Member, IEEE) received the B.S. and M.S. degrees in computer science and the Ph.D. degree in computer science from the National Chiao Tung University, Hsinchu, Taiwan, in 1989, 1991, and 2000, respectively. From 2001 to 2002, he was a Faculty Member of the Department of Computer Science and Engineering, Tatung University, Taipei, Taiwan. In 2002, he joined the Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung, Taiwan, where he is currently an Associate Professor. His research interests include computer vision, machine learning, and pattern recognition. He is a member of the Institute of Electrical and Electronics Engineers (IEE).

SIN-YI CAI received the B.S. degree in computer and information engineering from Chung Hua University, Hsinchu, in 2019, and the M.S. degree in computer and information engineering from the National Taiwan Ocean University, Keelung, in 2021. Her research interests include machine learning, computer vision, and the Internet of Things.

HSUN-YU LAN received the B.S. degree in business management from Cheng Shiu University, Kaohsiung, Taiwan, in 2007, and the M.S. degree in aquaculture from the National Taiwan Ocean University, Keelung, Taiwan, in 2021, where she is currently pursuing the Ph.D. degree in aquaculture. She served as the Chairman's Assistant for Long Dian Marine Biotechnology Company Ltd., from 2014 to 2018. Her research interests include aquaculture economics, fish farm management, business management, and cross-disciplinary AI applications.

HOANG-YANG LU received the B.S., M.S., and Ph.D. degrees from the National Taiwan University of Science and Technology, Taipei, in 1991, 1993, and 2007, respectively, all in electronics engineering. From 1993 to 2007, he was a Lecturer with the Department of Electronics Engineering, Lee-Ming Institute of Technology, Taipei. He joined the Digital Technology Department, Kainan University, Taoyuan, Taiwan, in 2007; and the Department of Electronics Engineering, Huafan University, Taipei, in 2008, where he was an Associate Professor. Since 2009, he has been an Assistant Professor with the Department of Electrical Engineering, National Taiwan Ocean University (NTOU), where he is currently an Associate Professor. His research interests include signal processing, communication systems, and soft computing.