

Article

UV3D: Underwater Video Stream 3D Reconstruction Based on Efficient Global SFM

Yanli Chen ¹, Qiushi Li ¹, Shenghua Gong ² , Jun Liu ^{2,3} and Wenxue Guan ^{2,*} 

¹ Key Laboratory of CNC Equipment Reliability, Ministry of Education, School of Mechanical and Aerospace Engineering, Jilin University, Changchun 130022, China; chenyanli@jlu.edu.cn (Y.C.); liqs19@mails.jlu.edu.cn (Q.L.)

² College of Computer Science and Technology, Jilin University, Changchun 130012, China; gongsh18@mails.jlu.edu.cn (S.G.); liujun2019@buaa.edu.cn (J.L.)

³ College of Electronic Information Engineering, Beihang University, Beijing 100191, China

* Correspondence: guanwx18@mails.jlu.edu.cn; Tel.: +86-153-3077-1569

Abstract: With the increasing demand for underwater resource exploration, three-dimensional (3D) reconstruction technology is important for searching for lost underwater civilizations, underwater shipwrecks, or seabed structures. However, faced with the limitations of underwater unmanned systems in terms of energy, bandwidth, and transmission delay, 3D reconstruction technology based on video streams as direct data will not work well. We propose a terminal image processing strategy to save data transmission time and cost and to obtain 3D scene information as soon as possible. Firstly, we propose an adaptive threshold key frame extraction algorithm based on clustering, which extracts key frames from the video stream as structure from motion (SFM) image sequences. On this basis, we enhance the underwater images with sufficient and insufficient illumination to improve the image quality and obtain a better visual effect in the 3D reconstruction step. Additionally, we choose global SFM to construct the scene and propose a faster rotation averaging method, the least trimmed square rotation averaging (LTS-RA) method, based on the least trimmed squares (LTS) and L1RA methods. It is proven to reduce 19.97% of the time through experiments. Finally, our experiments demonstrate that the dense point cloud saves about 70% of the transmission cost compared to video streaming.

Keywords: key frame extraction; clustering; image enhancement; LTS rotation averaging; global SFM-PMVS



Citation: Chen, Y.; Li, Q.; Gong, S.; Liu, J.; Guan, W. UV3D: Underwater Video Stream 3D Reconstruction Based on Efficient Global SFM. *Appl. Sci.* **2022**, *12*, 5918. <https://doi.org/10.3390/app12125918>

Academic Editors: Hongli Xu, Qichuan Ding, Junyi Wang and Hao Wang

Received: 26 March 2022

Accepted: 7 June 2022

Published: 10 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As is known to us, the ocean contains 80 percent of all living resources on earth. Along with this comes a strong demand for marine resource exploration. The 3D reconstruction is a crucial step in some underwater tasks, such as exploration, archaeology, and freshwater ecological environment investigations. Scientists have completed some underwater missions, such as quantifying the habitat complexity of freshwater ecosystems and exploring and modeling underwater historical sites based on 3D reconstruction technology [1–3]. In the area of underwater detection, autonomous underwater vehicles (AUVs) with detection devices are widely used as underwater mobile detection equipment. AUV can operate for longer periods and over larger areas to complete those tasks beyond human capabilities. The detection devices carried on the AUVs can be divided into two types: acoustics (e.g., side scan sonar) and optics (e.g., monocular camera). However, due to their different imaging principles, there are strong and weak points to both of them. Regarding sonar imaging systems, they can propagate over long distances, even in turbid water. For example, Song et al. made a seabed terrain 3D reconstruction using two-dimensional (2D) forward-looking sonar in 2019 [4]. However, sonar images lack color information and are black and white binary images; additionally, the side scan sonar images are very narrow and contain too little information. Optical cameras are also widely used as external sensors underwater;

optical images are colorful and full of detailed information compared to acoustic sensors. They are used on land more commonly, such as with drone navigation or self-driving technology. The limitation of optical camera applications underwater mainly comes from the poor visibility underwater due to the much stronger attenuation and scattering of light in water than that in the atmosphere, meaning optical sensors are more suitable for exploring close-range targets [5].

In general, research on underwater 3D reconstruction is mainly based on the SFM (structure from motion) system—a method used for calculating both camera poses and structures from a set of images [6], which can be divided into two categories, incremental SFM and global SFM. Incremental SFM [7,8] starts from a two-view 3D reconstruction as the initial structure and add views one-by-one with bundle adjustment [9] every time. Reconstructing the model of the scene sequentially can be robust and accurate; however, with the repeated registration and triangulation process, the cumulative error becomes larger and larger, which may result in scene drift [10]. Additionally, repetitively solving nonlinear bundle adjustments results in poor run-time efficiency. Regarding global SFM [11], the pipeline usually solves the problem in 3 steps. The first step solves all pairwise relative rotations through the epipolar geometry, and constructs a view graph whose vertexes represent cameras and whose edges represent epipolar geometry constraints. The second step involves rotation averaging [12] and translation averaging [13], which separately solve camera orientation and motion issues. The last step is the bundle adjustment, which aims to minimize the reprojection error and refine both the scene structure and camera poses. Compared to incremental SFM, the global method avoids cumulative error and is more efficient. Regarding the disadvantages of global SFM, it is not robust enough to outliers.

According to a large number of studies, most underwater stereo vision systems are based on incremental SFM [14,15]. However, the incremental approach is strongly dependent on the choice of the initial image pair and the selection of the next frame. Due to the complexity of the underwater environment coupled with the surge in water, the instability of the device will increase and the image may be intermittent, meaning that the selection of the next frame will be more difficult. Although the incremental approach is more robust, the efficiency and accuracy are not as good as the global approach, and the cumulative error also easily causes scene drift. If the robustness of global SFM can be improved, we could obtain a better reconstruction effect at a faster speed. The application of global SFM in underwater 3D reconstruction currently is still very limited, so in this paper we will use the more rapid and robust global SFM for the 3D reconstruction of underwater targets, and we will also propose a complete reconstruction process for underwater environments.

For underwater 3D reconstruction tasks, sonar images lack color information, meaning optical images are more conducive to obtaining the complete picture of the target. There are many challenges for underwater 3D reconstruction. On the one hand, the quality of the image significantly influences the reconstruction results; however, the underwater images may have problems such as noise, blur, and color degradation [16,17], meaning image enhancement and color correction are necessary prior to reconstruction [18]. To restore images degraded by the underwater environment, many scientists have proposed many solutions. For instance, an unsupervised method was raised by Galdran et al. [19] in 2015, which is robust and simple at the same time. Additionally, some supervised methods [20,21] based on training data have been presented in recent years. On the other hand, the propagation speed of an acoustic wave is five times slower than that on land; along with the large amount of noise and narrow bandwidth [22], the efficiency of underwater data transmission is extremely low.

To address the problems described above, and because of the rich detailed information available from optical imaging and superior transmission efficiency of acoustic systems, we present a terminal image processing strategy based on an underwater acousto-optical fusion imaging system [23]. After the sonar on the leader ship detects the target, the AUV carrying the optical camera will approach the target and obtain the three-dimensional

structure of the target from the video stream, which is completed on the terminal AUV to implement the smart ocean concept [24]. After this, the processed data are sent to the leader ship through the acoustic wave. In general, the core of this article is focused on how to reconstruct the three-dimensional structure of the target object and compress the data from the continuous optical images collected by the optical camera mounted on the AUV. The process is as follows. Firstly, to obtain the image sequence for SFM, the key frame extraction method is used to obtain key frames from the video stream. The following step is to improve the quality of the images, which is essential for the feature detection of underwater images. In this paper, we use different methods to enhance the images with various disadvantages. In addition, we propose a robust method for rotation averaging during the global SFM process, which improves the efficiency of the 3D point estimation. Finally, we summarize an SFM-PMVS pipeline, and every detail is enumerated. Under this strategy, we improve the data transmission efficiency and make it easier to obtain an overview of the underwater scene.

2. System Design and Process Introduction

The core aim of this study is to design an underwater AUV terminal processing strategy based on an optical vision system that can improve the transmission efficiency by transmitting point clouds instead of transmitting image data, so as to achieve the purpose of compressing the image data. The system performs calculations from the video stream to the 3D point cloud at the terminal AUV, and the main flow chart of the system is shown in Figure 1.

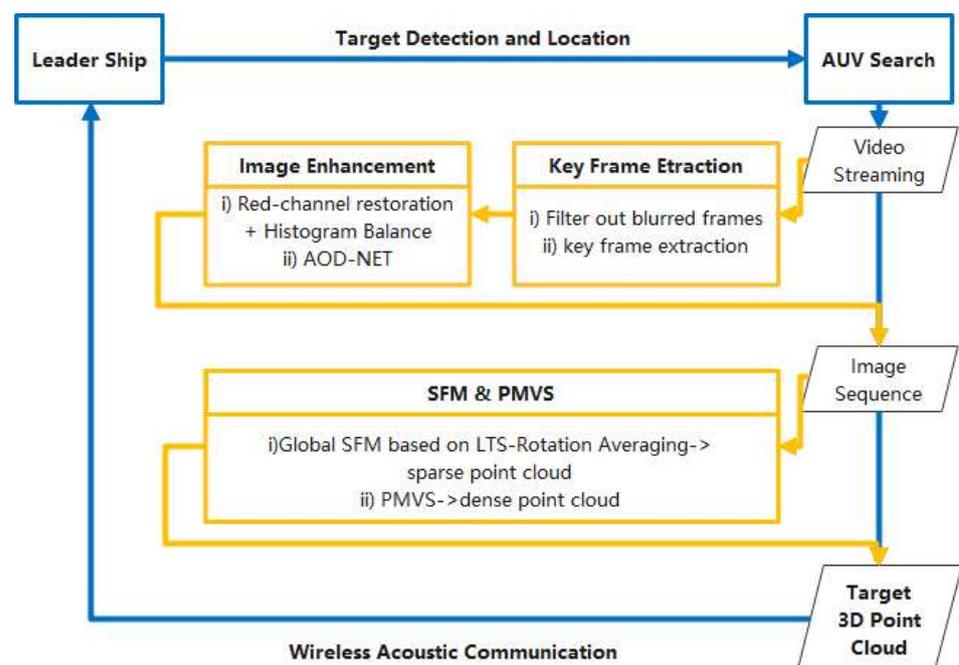


Figure 1. Underwater AUV terminal processing strategy based on optical vision system.

First of all, the leader ship with the sonar positioning and navigation system achieves the detection and location of the target, then the AUV's route is planned and driven from the main ship to collect the video stream of the target.

Next, the video stream is preprocessed to obtain the image sequence through 'key frame extraction' and 'image enhancement' processes. We firstly filter the blurred frames with the Laplace variance algorithm and extract key frames using the clustering algorithm. The key frame extraction step allows more efficient 3D reconstruction based on images. A detailed description of this stage can be seen in Sections 3.1 and 3.2. Next is the image enhancement module. In this step, different enhancement methods are used for images with

various problems, including color degradation and too much noise. This step reduces the number of incorrect feature matches and allows for more robust global rotation estimations. We give a detailed description of this stage in Section 3.3.

The last step is the 3D reconstruction, whereby the point clouds can be generated from the image sequence through the ‘SFM and PMVS’ process. We present a more efficient rotation-averaging method based on L1RA [25] raised in 2013 by Chatterjee in our algorithm, namely LTS (least trimmed squares), which is a robust regression method that is integrated into the rotation averaging. Our experiment show its efficiency, while a detailed description is shown in Section 6.1. Then, we introduce a clear global SFM-PMVS pipeline in Section 5.

3. SFM Preliminaries

We obtain a video stream from the AUV’s camera, although we need to extract key frames from the video stream as the image sequence for the 3D reconstruction. However, building the structure from all of the frames is not realistic; therefore, we develop the process shown in Figure 2 to convert the video stream to an image sequence that describes the preliminaries for the SFM system. Firstly, we filter out the blurred frames from the video stream and obtain the initial frames. Next, a key frame extraction method based on clustering is used to extract key frames from initial frames. At last, we perform the image enhancement step on the key frames to obtain the final image sequence.

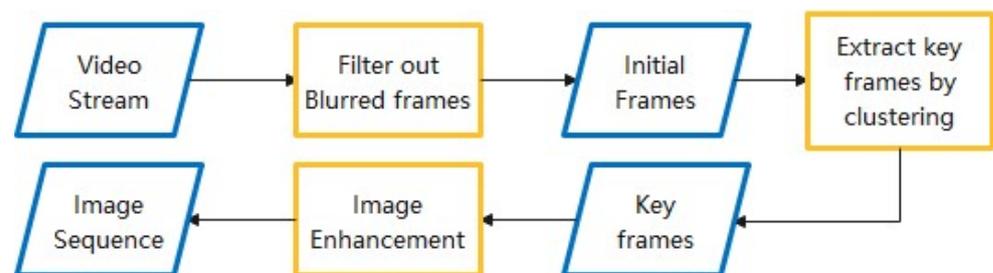


Figure 2. SFM preliminary flow chart.

3.1. Filter Out Blurred Frames

Firstly, we take the underwater environment into consideration. Due to the instability of underwater equipment caused by water flow, a few frames in the video may be blurred, so we hope to obtain clearer image frames as much as possible for subsequent 3D reconstruction efforts. The first step is to filter out the blurred frames. In this paper, we measure the degree of ambiguity with the Laplace variance algorithm.

The Laplacian operator is the second-order derivative of the image, which can detect rapid changes in the gray value of the image. It highlights the region in the image that contains fast gradient changes. Referring to Equation (1), $I(x, y)$ is the grayscale image, $L(x, y)$ is the Laplacian-based image, and Equation (2) is the discrete form of the Laplacian operator:

$$L(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \quad (1)$$

$$L(x, y) = I(x + 1, y) + I(x - 1, y) + I(x, y + 1) + I(x, y - 1) - 4I(x, y) \quad (2)$$

This process can be calculated using a convolution kernel and can generate an image that describes abrupt changes in the grayscale. As early as 2000, an algorithm [26] based on Laplacian variance was proposed to measure sharper focus. The boundary in a clear image is clear, and the Laplacian variance is large; the boundary information of a blurred image is less clear, and the variance is small. Therefore, the Laplacian operator can be used for image blur detection, and in this paper we choose this method to measure the blur degree of underwater images.

The algorithm steps are as follows. Firstly, we convolve a certain channel of the picture (generally use gray values) with a Laplacian mask and then calculate the variance (the square of the standard deviation) of the response as the ambiguity of the image. We set the average blur of all frames as the threshold for judging clarity, and then filter out the frames that are lower than the threshold.

3.2. Extract Key Frames

In this section, the key frame extraction algorithm based on clustering is elaborated. In machine learning, clustering is an unsupervised learning method used for data grouping. Clustering divides a data set into different classes or clusters according to a certain standard (such as a distance criterion). When it comes to key frame extraction, the crucial issue of the clustering process is the choice of clustering criteria [27,28]. The key frame extraction method based on clustering can be divided into 2 steps, and the pseudocode of the key frame extraction algorithm is shown in Algorithm 1:

Algorithm 1 Key Frame Extraction

Input: $F_{initial} = \{F_1, F_2, \dots, F_n\}$ Initial frames

Output: $F_{key} = \{F_1, F_2, \dots, F_k\}$ Key frames

procedure1: Dendrogram Construction

1: features of initial frames: $f = \{f_1, f_2, \dots, f_n\}$;

2: $\{D_{ij} | i, j = 1, 2, \dots, n\}$ Distances \leftarrow pdist (features);

3: while (num of clusters > 1):

merge the clusters with smallest distance;

calculate distances between new cluster and old clusters;

end procedure1

procedure2: Key frame extraction

4: $t \leftarrow$ OTSU (D_{ij})

5: num of clusters $k \leftarrow$ fcluster (dendrogram, t)

6: $F_{key} = \{F_1, F_2, \dots, F_k\} \leftarrow$ extract clustering centers

end procedure2

procedure1: Dendrogram Construction

Step 1: Extract features of image sequence $f = \{f_1, f_2, \dots, f_n\}$ from initial frames $F_{initial} = \{F_1, F_2, \dots, F_n\}$.

Differing from other kinds of data, the image data are all high-dimensional. If we directly use the Euclidean distance between the image data as the standard for clustering, the calculation is too large and may lead to memory overflow. In step 1, instead of calculating Euclidean distances among images, we use LBP and HSV fusion features. An LBP (local binary pattern) [29] is an operator used to describe the local texture features of an image. It has significant advantages, such as gray invariance and rotation invariance, and it is easy and efficient to calculate. Additionally, the LBP feature is robust to illumination changes, and uneven illumination may occur underwater, so it is suitable for underwater image description. HSV (hue, saturation, value) is a kind of color space created by A. R. Smith in 1978 [30] according to the intuitive characteristics of color, and it is closer to people's experience and perception of color than RGB. In addition, HSV features describe the global features of the image, while LBP features describe the local texture information of the image, so their fusion can describe a frame more comprehensively. Due to the low saturation of underwater images, the saturation component in the HSV space is not convincing as an image feature. At the same time, due to the uneven illumination of the underwater environment, the V component in the HSV space can better represent the difference between frames. Therefore, we use the combination of H and V components in the HSV space with the image's LBP features to refine the local feature of the image.

Step 2: Compute distances $\{D_{ij} | i, j = 1, 2, \dots, n\}$ between frames with the pdist function in SciPy (an open-source Python algorithm library and math toolkit).

Step 3: Generate a dendrogram through the while loop.

Hierarchical clustering is an iterative process of continuously calculating the class spacing and merging the classes with the smallest distance until there is only one class left and results in a dendrogram. Which is shown in Figure 3, the abscissa of dendrogram represents the sample number, and the ordinate represents the distance between clusters.

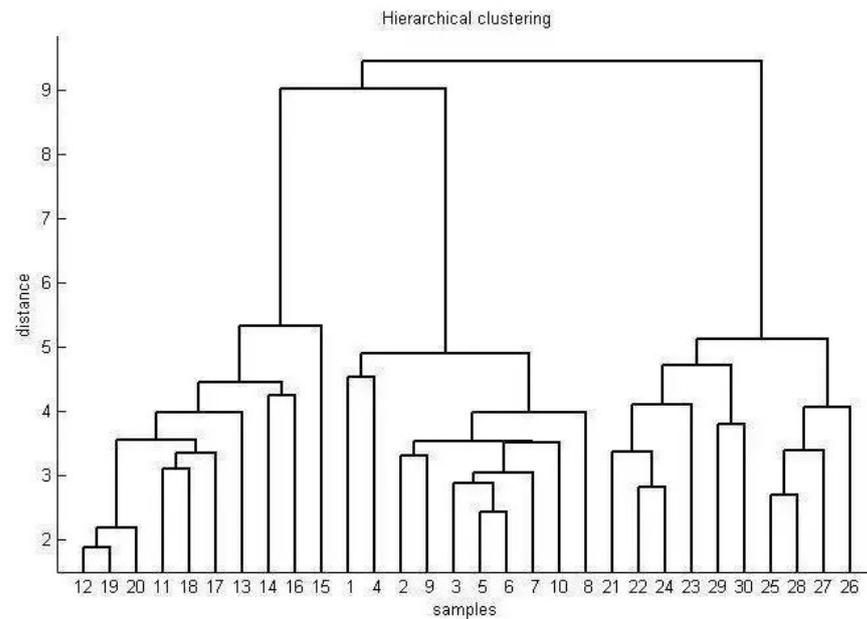


Figure 3. Clustering dendrogram.

procedure2: Key frame extraction

Step 4: Obtain the adaptive threshold t to split the dendrogram with the OTSU algorithm.

Different clustering results can be obtained via segmentation at a specific level according to a threshold, so the determination of the segmentation threshold is very important.

In this paper, we use the OTSU algorithm [31] to obtain an adaptive threshold to split the dendrogram, which is often used to determine the threshold during image binarization. The basic idea of the OTSU algorithm is to divide the samples into two categories based on the threshold, whereby the greater the variance between the two categories, the better the classification. First, we calculate the interclass variance with all sample values as the threshold and take the value corresponding to the maximum variance as the optimal threshold.

Step 5: Split the dendrogram by t with the `fcluster` function in SciPy and extract clustering centers $F_{key} = \{F_1, F_2, \dots, F_k\}$ as the output key frames.

3.3. UW Image Enhancement

Due to the complexity of the marine environment, the quality of images underwater is influenced by the attenuation and scattering of light. Regarding light attenuation, different propagation characteristics of light with different wavelengths in water result in issues such as color distortion, blurred vision, and reduced contrast in images collected by the underwater optical sensor. In addition, light scattering resulting from suspended particles may cause images to be noisy and smoggy, and this leads to problems such as blurred edges and a lack of detail. Additionally, when the light irradiates the object in the water, it will scatter when it meets the impurities in the water and it will be directly received by the camera, resulting in the low contrast of the image.

Considering images acquired in different environments, here we use two algorithms to enhance the image sequence automatically: method 1 targets underwater images with red channel degradation, and method 2 targets images with impaired sharpness.

Method 1: For shallow water environments with enough natural light, there is no need for external lighting equipment. Still, the longer-wavelength red light will decay faster than blue or green light, and blue-green tones will dominate the image. In this paper, we perform the image fusion based on a red channel prior method [18] with improved histogram equalization. The red channel prior method is improved based on the dark channel prior (DCP) method [19], and it recovers some of the visibility while correcting the color attenuation. The histogram equalization method can compensate for the difficulty in differentiating gray levels visually, making it a powerful tool for adaptive contrast enhancement. The complete algorithm flow is shown in Figure 4.

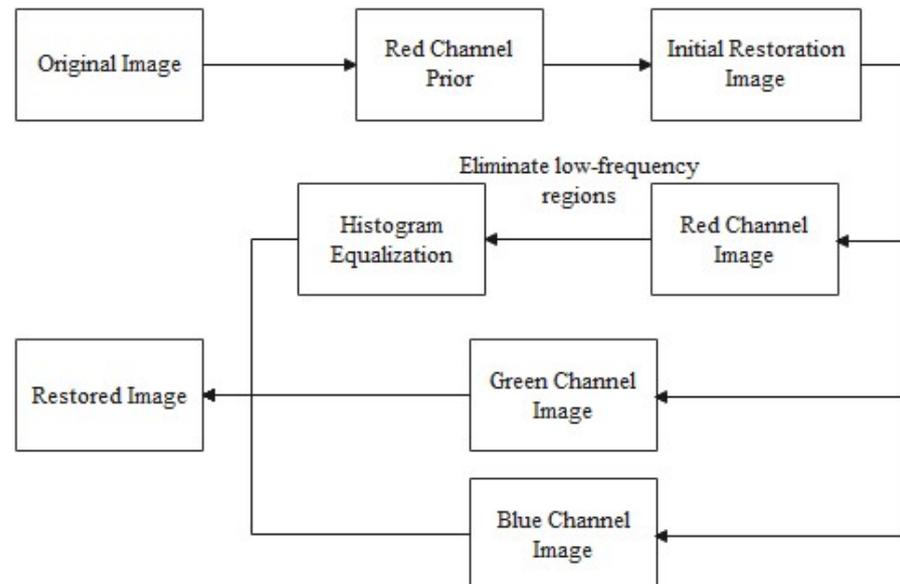


Figure 4. Image enhancement algorithm flow chart.

Firstly, we restore the original image with the red channel prior method and obtain the initial restoration image. Secondly, as our method specifically aims at the red channel, we eliminate the low-frequency regions before histogram equalization. In our method, we use the gradient as the standard of filtration, whereby the area where the gradient value is close to 0 is the low frequency region, so we note the gray value of the area as 0 and obtain a new grayscale histogram. In the end, histogram equalization is applied to further enhance the red channel, and then we remerge the three channels to obtain the final restored image.

Our results are shown in Figure 5, (b) and (d) are histograms of the original image (a) and enhanced image (c) respectively, the R/G/B histograms of (a) and (c) has pixels in the range of 0–255, and the pixel values are evenly distributed on the x-axis, and the y-axis represents the number of pixels corresponding to each pixel value. We can see the color histogram is more balanced after enhancement, and the red component is significantly increased. From another point of view, it can be seen from (c) that the image quality and color is improved comparing with (a), and there are more details in image (c) so more feature points can be detected. We also use UIQM [32], which is a linear combination of a color measurement index (UICM), clarity measurement index (UISM), and contrast measurement index (UIConM). It targets the degradation mechanism and imaging characteristics of underwater images, whereby the larger the value, the better the color balance, clarity, and contrast of the image.

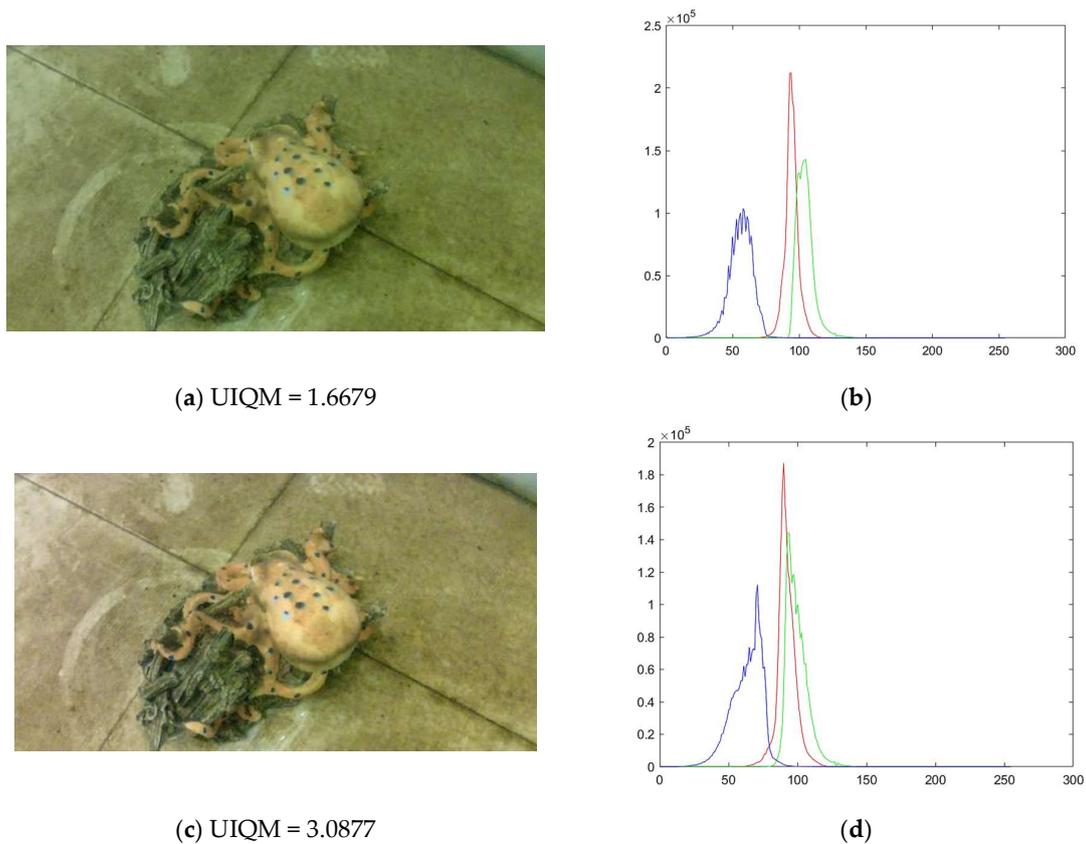


Figure 5. (a,b) The original image and its color histogram. (c,d) The enhanced image and its color histogram.

Method 2: As for water without enough light, it is necessary to use external lighting equipment, and the image quality will be reduced due to the scattering of suspended objects. In such situations, we switch to another mode [33] to enhance the images. The method proposed by Liu and Gong et al. improves the quality of the whole picture; enhances the quality of the most important, useful, and task-related information; and weakens the information of areas of no interest. The result is shown in Figure 6, where it can be seen that the quality in (b) is greater than the original image, which not only improves the quality of the pixels, but also enhances the target’s information and suppresses irrelevant background information effectively. We can see in Figure 6 that the UIQM of image (b) is higher than image (a), which means that the quality of the image underwater is greatly enhanced.



Figure 6. (a) Original image and (b) enhanced image.

4. LTS-L1RA

For the reconstruction of an underwater scene, global SFM is more suitable for the following reasons. On the one hand, it is more efficient for the whole terminal processing system described in Figure 1. Additionally, there is no need to consider the choice of initial pairs or the adding orders of cameras, which is an indispensable step in the incremental method, especially in such a complicated underwater environment.

In this paper, we use the camera translation registration method based on the ACransac [34] proposed in 2013 [6] to solve the translation averaging problem, and we come up with a more efficient method for robust rotation averaging based on robust regression. In addition, we give an introduction to the existing methods used for rotation averaging in Section 4.1 and elaborate on our method in Section 4.2.

4.1. Existing Methods for Rotation Averaging

Before the rotation averaging, we obtain the relative rotations R_{ij} and relative translations t_{ij} from image pairs by decomposing the essential matrix E_{ij} . Equation (3) shows the construction of the essential matrix, the product of the antisymmetric matrix of the translation vector t_{ij} , and the rotation matrix R_{ij} .

$$E_{ij} = t_{ij} \hat{R}_{ij} \tag{3}$$

With the relative rotations, we can construct a view graph $G = \{V, \epsilon\}$, where V represents cameras and ϵ represents epipolar constraints between pairwise cameras. Global rotations $R_{global} = \{R_1, R_2 \dots, R_n\}$ are solved from a set of relative rotations R_{ij} between camera coordinate systems, and this process is called rotation averaging (RA). In the case of feature matching with outliers, the problem is converted to an appropriate minimization problem as Equation (5), which satisfies Equation (4). In Equation (5), $d(R_{ij}, R_j R_i^{-1})$ means the distance measure between observations R_{ij} and the current global estimate R_i, R_j .

$$R_j = R_i R_{ij} \tag{4}$$

$$\operatorname{argmin}_{R_{global}} \sum_{(i,j) \in \epsilon} d(R_{ij}, R_j R_i^{-1})^2 \tag{5}$$

Many scientists have proposed different methods to solve the rotation averaging problem described above. In 2004 [35], Govindo firstly proposed to solve the rotation problem using the Lie algebra, while in 2007 [36] Mcartinec and Pajdla estimated absolute rotations simply and quickly by solving the problem linearly using the least squares method. After that in 2013 [25], a more efficient and accurate method was raised, which used a modern $l1$ -norm [37] optimization method to solve a linear system of equations such as in Equation (9), which is based on the Lie group and Lie algebra. The group formed by the rotation matrix R is the Lie group $SO(3)$. Each Lie group has corresponding Lie algebra $so(3)$, and the method for mapping the Lie group to the Lie algebra is logarithmic mapping (Equation (6)). In contrast, the mapping of the Lie algebra to the Lie group involves exponential mapping (Equation (7)). The Lie algebraic form of Equation (4) can be expressed as Equation (8). According to Equation (8), Equation (9) can be regarded as a set of simultaneous equations. In Equation (9), A is a sparse matrix that is constructed of 0 and 1, whereby the meaning of every row represents a pair of cameras; $\omega_{global}, \omega_{relative}$ correspond to the Li algebra of global rotations and relative rotations. In [25], the results were refined using the iteratively reweighted least squares (IRLS) method [38].

$$R = \exp(\hat{\omega}) \in SO(3) \tag{6}$$

$$\hat{\omega} = \log(R) \in so(3) \tag{7}$$

$$R_i R_{ij} = R_j \Rightarrow e^{\hat{\omega}_i} e^{\hat{\omega}_{ij}} = e^{\hat{\omega}_j} \Rightarrow \omega_{ij} = \omega_j - \omega_i = \underbrace{[\dots - 1 \dots 1 \dots]}_{A_{ij}} \omega_{global} \tag{8}$$

$$A \omega_{global} = \omega_{relative} \tag{9}$$

In this paper, we enhance the efficiency of the rotation averaging based on $l1$ -norm optimization and the robust regression method, and the details of our algorithm are shown in Section 4.2.

4.2. LTS-L1RA Algorithm

In this paper, we come up with a more efficient method based on the L1RA ($l1$ – norm rotation averaging) and c-step methods.

In 2013 [35], Govindo firstly minimized and estimated the global rotations robustly. In the L1RA algorithm, the initial guesses for global rotations $R_{global} = \{R_1, R_2, \dots, R_n\}$ are firstly calculated according to Equation (4) and then the sparse matrix A is constructed. After this comes the iterative optimization process, whereby each iteration recalculates the error of the relative rotation vector ω_{rel} (step 1: calculate ΔR_{ij} , the discrepancy between the observations R_{ij} , and the current global estimate; step 2: convert ΔR_{ij} to $\Delta \omega_{ij}$ due to Equation (7)) and then minimizes $\|A \Delta \omega_{global} - \Delta \omega_{rel}\|_{l1}$ with $l1$ minimization to obtain $\Delta \omega_{global}$ (step 3) and update the global rotations (step 4). The iteration stops once the error meets the requirement.

A review of the L1RA method is shown as Algorithm 2.

Algorithm 2 $l1$ – norm Rotation Averaging

Input: $R_{ij} = \{R_{ij1}, R_{ij2}, \dots, R_{ijk}\}$ Relative Rotations

Output: $R_{global} = \{R_1, R_2, \dots, R_n\}$ Global Rotations

Initialization: Initial guess

Function name and argument: $L1(R_{relatives}, R_{guess})$

$A \leftarrow$ compute sparse matrix from relative rotations

while $\|\omega_{rel}\| < \epsilon$ do:

1: $\Delta R_{ij} = R_j^{-1} R_i R_{ij}$

2: $\Delta \omega_{ij} = \log(\Delta R_{ij})$

3: solve $A \Delta \omega_{global} = \Delta \omega_{ij}$ (minimize $\|A \Delta \omega_{global} - \Delta \omega_{rel}\|_{l1}$)

4: $R_{new} = R_{old} \exp(\Delta \omega_{global})$

end while

The LTS (least trimmed squares) method is a robust regression method that was proposed by Rousseeuw in 2006 [38]. They raised a new basic idea that makes it possible to compute a more approximate solution starting from any approximation to the LTS regression coefficients, and this process is called the c-step, where c stands for the ‘concentration’. The c-step is an iterative process, which iteratively fits the h-subset (a subset of h cases out of n samples) with the smallest error until reaching convergence, while the pseudocode of each iteration in the c-step is shown as Algorithm 3.

Algorithm 3 Pseudocode of C-step

1: estimate the regression coefficient ω_{old} based on the complete sample set H

2: compute the residuals based on H and sort according to ascending order:

$e_{old(i)}, i = 1, 2, \dots, n$

3: put H_{new} as h samples with the smallest error

4: estimate the regression coefficient ω_{new} based on H_{new}

In the c-step, the regression coefficient ω_{old} is firstly estimated based on set H , then we compute the residuals $e_{old(i)}, i = 1, 2, \dots, n$ based on H and sort them according to ascend-

ing order. Then, we extract h samples H_{new} with the smallest error, and the breakdown value h is always set as $n/2$ to n , which is the ratio of dirty data that can exist before making an incorrect estimate. In step 4, the new regression coefficient ω_{new} is estimated based on H_{new} . Additionally, it was demonstrated in [39] that often the robust solution is obtained after two or three c-steps; that is, step 2 and step 3 are repeated in the pseudocode above, which makes the convergence faster.

The details of the LTS-L1RA method are shown in Algorithm 4.

Algorithm 4 Least trimmed squares (LTS)-L1RA

Input: $R_{ij} = \{R_{ij1}, R_{ij2}, \dots, R_{ijk}\}$ Relative Rotations

Output: $R_{global} = \{R_1, R_2, \dots, R_n\}$ Global Rotations

Initialization: Initial guess, $h = 0.75k$

procedure1: c-steps

$R_{new} =$ Initial guess

for I in range (3):

1: $R_{old} = R_{new}$

2: compute errors of R_{old} in degrees:

for all $(i, j) \in R_{ij}, R_i, R_j \in R_{old}$ do:

$\Delta R_{ij} = R_j^{-1} R_i R_{ij}$

$\theta_{ij} = \arccos(\text{tr}(\Delta R_{ij}) - 1)/2$

end for

3: $H \leftarrow$ sort (errors of R_{old} in degrees) [h]

4: compute $R_{new} :=$ L1RA based on $R_{ij}[H]$:

5: $R_{new} \leftarrow$ L1 ($R_{ij}[H], R_{old}$)

end for

end procedure1

procedure2: L1RA on subsets of k cases

6: compute $R_{global} :=$ L1RA based on $R_{ij} = \{R_{ij1}, R_{ij2}, \dots, R_{ijk}\}$ until convergence

end procedure2

In our method, we firstly use L1RA ($l1 - norm$ rotation averaging) based on the h -subset with the smallest errors in degrees three times due to the fast convergence character of the c-steps described above. We set the breakdown value h as $0.75k$ (the parameter k corresponds to the total number of samples, while h corresponds to the coverage of the total samples, which means a subset of h cases) in our method based on [39] in order to obtain a good balance between the breakdown value and statistical efficiency. In this step, we can quickly estimate the initial guess of the global rotations R_{new} based on small-scale samples with the lowest errors in degrees through 3 c-steps (step 1–step 5). After this, in procedure 2, we minimize the $l1 - norm$ based on all samples until we reach algorithm convergence and obtain the final global estimate R_{global} (step 6).

5. Global SFM-PMVS Pipeline

In this section, we review the whole 3D reconstruction pipeline. The SFM-PMVS method consists of the following steps: (1) extract features and match features; (2) construct a view graph; (3) compute camera motions using rotation averaging and translation averaging; (4) obtain the sparse structure and optimize it with bundle adjustments, then finally compute the dense point cloud with PMVS.

Step 1: feature extraction and feature matching.

We use SIFT [40] to extract image features, which is a scale-invariant feature descriptor with advantages in terms of stability, efficiency, and abundance (a large number of features even though the object is extremely small). The quality of an image may be greatly reduced because of poor contact and spectral distortion in underwater scenes, whereby SIFT can extract enough features. The 128-dimensional feature descriptor obtained by the SIFT algorithm can be used to match features between two images. We use KNN-match (a

feature matching method based on the K nearest neighbor algorithm [41] provided by opencv) to obtain matching points, and filter the matched points using the essential matrix.

Step 2: View graph construction.

We construct the view graph by finding the epipolar geometry (internal projective relationship between two views, which depends on the relative position and camera's internal parameters) between images. In the view graph, each vertex represents a camera and each edge means that relative motions between two cameras can be estimated. In our strategy, we use the rotation angle to represent the weight of the edge and construct a minimum spanning tree as the final view graph.

Step 3: Robust Rotation Averaging and Translation Averaging.

After step 1 and step 2, we can begin to compute the global rotation with the relative rotations. As described in Section 4.2, we use the LTS-L1RA method to compute global rotations. Then, we use the ACRANSAC method for the camera pose registration described in [6] to fulfill the translation averaging.

Step 4: Triangulation, Bundle Adjustment, and PMVS.

Through the robust estimation of global rotations and poses in step 3, we can recover the scene structure from the motion. In this step, we compute the 3D point positions via triangulation and then refine the structure and motions via bundle adjustment. After this, in order to provide a better visual effect, we use the PMVS [42] toolkit to obtain a dense point cloud.

6. Experiments

6.1. LTS-L1RA Confirmation Experiment

In order to prove the advantage of the LTS-L1RA in terms of efficiency, we provide a comprehensive evaluation of the accuracy and efficiency, which shows the performance of the L1RA-IRLS (l1-norm rotation averaging–iteratively reweighted least squares) and LTS-L1RA methods.

We use the 'Notredame' dataset as experimental data, which was published by Wilson and Snavely [13] in 2014, and includes 64,678 relative rotations and 715 cameras.

We set 5 indicators to analyze and compare the algorithms, measuring the running time in second, mean errors in degrees, RMS error in degrees, processing rate, and mean error/processing rate. The error in degrees calculation formula is shown in Equation (10), where $R_j R_i^{-1} R_{ij}^{-1}$ is the discrepancy between observations R_{ij} and the global estimate R_i, R_j , $\Delta\theta$ is the error in degrees, and the Equation (11) is the mean error in degrees. Equation (12) is the RMS error in degrees, where the number of camera pairs divided by the running time is the processing rate. The experiment was performed on Ubuntu 16.04 (CPU frequency: 2.50 GHz) and the result of test is shown in Table 1:

$$\Delta\theta = \arccos\left(\frac{\text{tr}(R_j R_i^{-1} R_{ij}^{-1}) - 1}{2}\right) \quad (10)$$

$$\Delta\theta_{\text{mean}} = \sum_{i=1}^k \Delta\theta_i / k \quad (11)$$

$$\Delta\theta_{\text{RMS}} = \sqrt{\sum_{i=1}^k \Delta\theta_i^2 / k} \quad (12)$$

Due to the experimental data shown above, we can see that the LTS-L1RA method has a shorter running time compared to L1RA-IRLS, with a slightly higher mean error and lower RMS error. We can see that the processing rate is increased by 24.96%. In order to evaluate the algorithm synthetically, we include a new parameter in our experiment to measure the overall efficiency of the algorithm, the mean error/processing rate. We can see that it is 19.11% lower than for L1RA-IRLS, meaning our method performs better in efficiency than L1RA-IRLS.

Table 1. Efficiency verification test.

Parameter	Method		
	L1RA-IRLS	LTS-L1RA	Comparison
Running time (s)	106.712	85.400	−19.97%
Mean error (degree)	3.394	3.432	+1.12%
RMS error (degree)	7.274	7.161	−1.55%
Processing rate (Amount of data/s)	606.099	757.354	+24.96%
Mean error/processing rate	5.6×10^{-3}	3.7×10^{-3}	−19.11%

6.2. Pool Experiment

The experiment was carried out in a pool measuring 3 m × 4 m × 1 m. The camera was equipped with a waterproof case and lighting device, and can be seen in Figure 7. In order to simulate the AUV's shooting effect, the camera was fixed to the rod to shoot around the target. As shown in Figure 6, the experimental equipment includes a camera with a lighting device, which is shown on the left, while the experimental environment is shown on the right, where red lines represent the motion trajectory of the camera. We respectively experiment with lights on and off in the same route. Additionally, the parameter list for the experiment is shown in Table 2.

**Figure 7.** Experimental equipment and environment.**Table 2.** Experiment parameters.

Name	Configuration
Pool	3 m × 4 m × 1 m
Medium	Water with suspended matter
Calibration board	GP100
Camera	With waterproof case and lighting device
Frame Width	1920 pixels
Frame Height	1080 pixels
Frame Rate	60 FPS (frames per second)
Environment	Python 3.5, Ubuntu 16.04
Visualization tool	MeshLab

We firstly obtain the image sequence by using the key frame extraction algorithm proposed in Section 3.2, and the 3D reconstruction results based on the image sequence are shown in Figure 8.

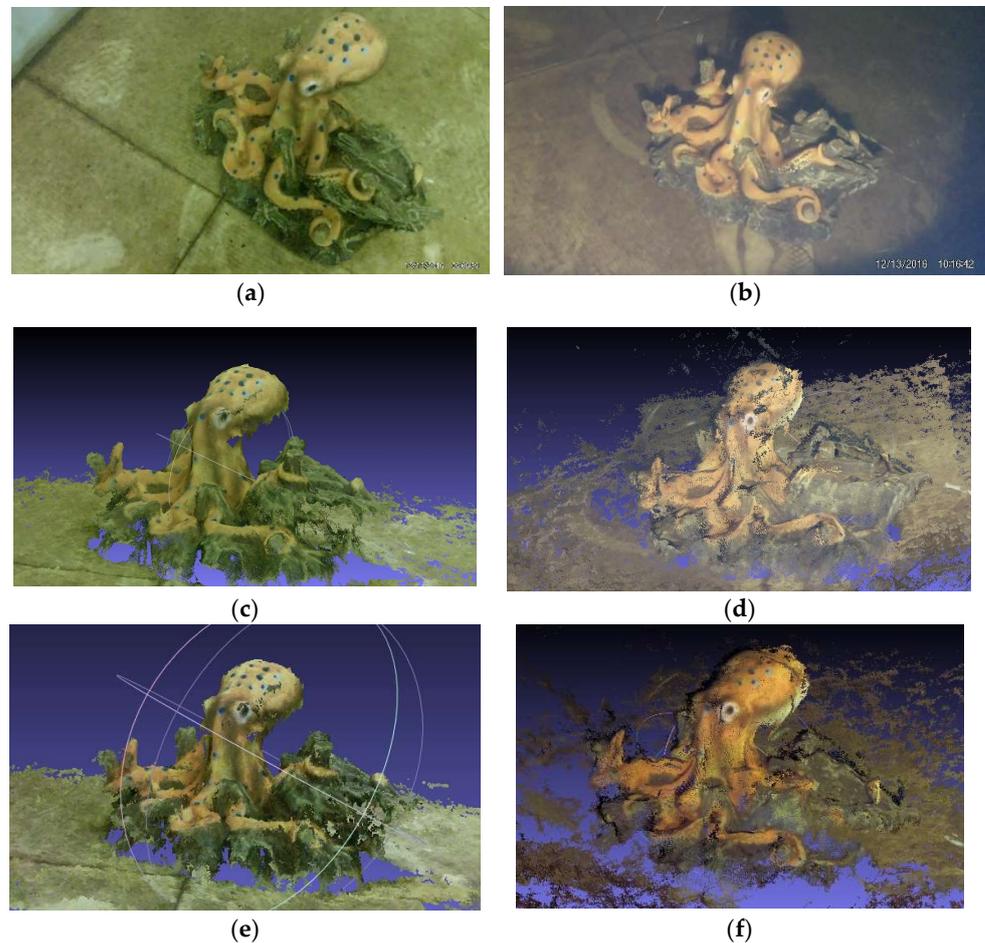


Figure 8. (a) Underwater image with natural light. (b) Underwater image with a lighting device. (c) Point cloud based on original image sequence (natural light). (d) Point cloud based on original image sequence (external lighting device). (e) Point cloud based on enhanced image sequence (natural light). (f) Point cloud based on enhanced image sequence (external lighting device).

In Figure 8, image (a), (b) are original underwater images with natural light and external light device, and image (c), (d) are the point cloud calculated based on original image sequence. The point cloud based on the image sequence enhanced by the image enhancement method described in Section 3.3 are shown in image (e) and (d). We can see that after image enhancement the red channel recession has been greatly reduced and there are more point clouds in image (e). Also, we can see that there is much noise caused by suspended matter scattering in image (d), it is shown in image (f) that after enhancement there are much less noise and better color performance.

In Figure 9, we show an independent experiment performed in the air. The result shows better color information; in other words, our image enhancement algorithm is still insufficient. In Table 3, we compare the number of bytes from the video streams and the corresponding point cloud data. We can see our method reduces the transmission costs by 74.21% and 64.22% in the underwater experiment above, meaning it can greatly save on transmission costs when data are transferred through wireless acoustic communication.



Figure 9. (a) Model in the air. (b) Reconstruction result based on images in the air.

Table 3. Data compression results.

Environment	Video	Point Cloud	Comparison
Enough light	140 MB	36.1 MB	−74.21%
Insufficient light	72.1 MB	25.8 MB	−64.22%

7. Result

In this work, we have presented a new pipeline based on an underwater acousto-optical fusion imaging system, specifically designed based on an underwater optical system. Firstly, a hierarchical clustering algorithm with adaptive threshold was proposed, which automatically determines the threshold of clustering by using the maximum inter-class variance method, which is more convenient for automatically extracting key frames. Secondly, in order to enhance the quality of the image underwater, we proposed an image enhancement algorithm based on the red channel prior and histogram equilibrium methods, which was verified as being effective. In addition, a more efficient rotation averaging algorithm was devised in our paper, and the efficiency of our method was verified by comparing it with the L1RA-IRLS method on the same dataset. At last, we performed an experiment in a swimming pool using our method, and the dense point cloud of the model was given in the article.

8. Future

Regarding the complex and changeable environment underwater, there are various degrees of degeneration with underwater images. Additionally, as for large-scale scenes, a wireless distribution system [43] is needed to detect targets underwater. For a more efficient and detailed system, in the future we will integrate an image analysis module to judge different image problems and use different image enhancement algorithms to improve the image quality. Additionally, in order to improve the efficiency of the system, we will drive the AUV cluster to complete the target detection and calculate the point clouds separately with the AUV positioning technology [44], and then send the data back to the leader ship and perform the point cloud registration and fusion processes. For another typical problem with underwater images, due to waterproof housing being needed for underwater cameras, light will pass through three mediums, namely water, glass, and air, which cause twice the refraction before arriving at the optical sensor. In the future, we will take the refraction into consideration, which can improve the accuracy of the scene structure estimation.

Author Contributions: Conceptualization, Y.C. and W.G.; Funding acquisition, Y.C. and J.L.; Investigation, S.G.; Project administration, Q.L. and S.G.; Resources, W.G.; Validation, Q.L.; Writing—original draft, Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 61971206, No. 61631008, No. U1813217, No. U19A2061, No. 61772228, and No. 61902143), the Fundamental Research Funds for the Central Universities 2017TD-18, the National Key Basic Research Program (2018YFC1405800), Jilin Scientific and Technological Development Program (No. 2020122208JC), the Education Department of Jilin Province (No. JJKH20211105KJ), the Key Research

and Development Program of Jilin Province (No. 20210203175SF), Foundation of Education Bureau of Jilin Province under Grant(No.JJKH20220988KJ), Aeronautical Science Foundation of China under Grant(No.2019ZA0R4001), National Natural Science Foundation of China under Grant, (No. 51505174), Interdisciplinary integration innovation and cultivation project of Jilin university under Grant(No. JLUXKJC2020105).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Nakatani, T.; Ura, T.; Ito, Y. AUV “TUNA-SAND” and its Exploration of hydrothermal vents at Kagoshima Bay. In Proceedings of the OCEANS 2008—MTS/IEEE, Kobe Techno-Ocean, Kobe, Japan, 8–11 April 2008.
- Kalacska, M.; Lucanus, O.; Sousa, L.; Vieira, T.; Arroyo-Mora, J. Freshwater fish habitat complexity mapping using above and underwater structure-from-motion photogrammetry. *Remote Sens.* **2019**, *10*, 1912. [[CrossRef](#)]
- Nocerino, E.; Menna, F.; Remondino, F. Accuracy of typical photogrammetric networks in cultural heritage 3D modeling projects. *Int. Arch. Photogramm. Remote Sens.* **2014**, *XL-5*, 465–472. [[CrossRef](#)]
- Song, S.; Li, Y.; Li, Z.; Hu, Z.; Li, J.H. Seabed terrain 3D reconstruction using 2D forward-looking sonar: A sea-trial report from the pipeline burying project. *IFAC-Pap. Line* **2019**, *52*, 175–180. [[CrossRef](#)]
- Brandou, V.; Allais, A.G.; Perrier, M. 3D Reconstruction of Natural Underwater Scenes Using the Stereovision System IRIS. In Proceedings of the OCEANS 2007—Europe, Aberdeen, UK, 18–21 June 2007; IEEE: Piscataway, NJ, USA, 2007.
- Moulon, P.; Monasse, P.; Marlet, R. Global Fusion of Relative Motions for Robust, Accurate and Scalable Structure from Motion. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; IEEE: Piscataway, NJ, USA, 2013.
- Snavely, N.; Seitz, S.; Szeliski, R. Photo tourism: Exploring photo collections in 3D. *ACM Trans. Graph.* **2006**, *25*, 835–846. [[CrossRef](#)]
- Gao, X.; Hu, L.; Cui, H.; Shen, S.; Hu, Z. Accurate and efficient ground-to-aerial model alignment. *Pattern Recognit.* **2018**, *76*, 288–302. [[CrossRef](#)]
- Triggs, B.; Zisserman, A.; Szeliski, R. *Vision Algorithms: Theory and Practice*; Springer: Berlin, Germany, 2000.
- Wu, C. Towards Linear-Time Incremental Structure from Motion. In Proceedings of the 2013 International Conference on 3D Vision—3DV 2013, Seattle, WA, USA, 29 June–1 July 2013; IEEE Computer Society: Washington, DC, USA, 2013.
- Moulon, P.; Monasse, P.; Perrot, R. OpenMVG: Open Multiple View Geometry. In Proceedings of the International Workshop on Reproducible Research in Pattern Recognition, Cancún, Mexico, 4 December 2016.
- Hartley, R.; Trampf, J.; Dai, Y.; Li, H. Rotation averaging. *Int. J. Comput. Vis.* **2013**, *103*, 267–305. [[CrossRef](#)]
- Wilson, K.; Snavely, N. Robust Global Translations with 1DSfM. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014.
- Su, Z.; Pan, J.; Lu, L.; Dai, M.; He, X.; Zhang, D. Refractive three-dimensional reconstruction for underwater stereo digital image correlation. *Opt. Express* **2021**, *29*, 12131. [[CrossRef](#)]
- Su, Z.; Pan, J.; Zhang, S.; Wu, S.; Yu, Q.; Zhang, D. Characterizing dynamic deformation of marine propeller blades with stroboscopic stereo digital image correlation. *Mech. Syst. Signal Process.* **2022**, *162*, 108072. [[CrossRef](#)]
- Ba Nerjee, J.; Ray, R.; Vadali, S.; Shome, S.N.; Nandy, S. Real-time underwater image enhancement: An improved approach for imaging with auv-150. *Sadhana* **2016**, *41*, 225–238. [[CrossRef](#)]
- Soni, O.K.; Kumare, J.S. A Survey on Underwater Images Enhancement Techniques. In Proceedings of the 2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT), Gwalior, India, 10–12 April 2020.
- Galdran, A.; Pardo, D.; Picó, A.; Alvarez-Gila, A. Automatic red-channel underwater image restoration. *J. Vis. Commun. Image Represent.* **2015**, *26*, 132–145. [[CrossRef](#)]
- He, K.M.; Sun, J.; Tang, X.O. Single Image Haze Removal Using Dark Channel Prior. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 2341–2353. [[PubMed](#)]
- Li, C.; Guo, J.; Guo, C. Emerging from water: Underwater image color correction based on weakly supervised color transfer. *IEEE Signal Process. Lett.* **2018**, *25*, 323–327. [[CrossRef](#)]
- Li, J.; Skinner, K.A.; Eustice, R.M.; Johnson-Roberson, M. Watergan: Unsupervised generative network to enable real-time color correction of monocular underwater images. *IEEE Robot. Autom. Lett.* **2017**, *3*, 387–394. [[CrossRef](#)]
- Liu, L.; Zhou, S.; Cui, J.H. Prospects and problems of wireless communication for underwater sensor networks. *Wirel. Commun. Mob. Comput.* **2010**, *8*, 977–994.
- Liu, J.; Guan, W.; Wang, X.; Liu, J. Optical imaging study of underwater acousto-optical fusion imaging systems. In Proceedings of the Thirteenth ACM International Conference on Underwater Networks & Systems, Shenzhen, China, 3–5 December 2018.

24. Cui, J.H.; Kong, J.; Gerla, M.; Zhou, S. The challenges of building scalable mobile underwater wireless sensor networks for aquatic applications. *IEEE Netw.* **2006**, *20*, 12–18.
25. Chatterjee, A.; Govindu, V.M. Efficient and Robust Large-Scale Rotation Averaging. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; IEEE Computer Society: Washington, DC, USA, 2013.
26. Pech-Pacheco, J.L.; Cristobal, G.; Chamorro-Martinez, J.; Fernandez-Valdivia, J. Diatom autofocusing in brightfield microscopy: A comparative study. In Proceedings of the 15th International Conference on Pattern Recognition, Barcelona, Spain, 3–7 September 2000; IEEE Computer Society: Washington, DC, USA, 2000.
27. Nasreen, A.; Roy, K.; Roy, K.; Shobha, G. Key Frame Extraction and Foreground Modelling Using K-Means Clustering. In Proceedings of the 2015 7th International Conference on Computational Intelligence, Communication Systems and Networks, Riga, Latvia, 3–5 June 2015; IEEE Computer Society: Washington, DC, USA, 2015.
28. Gharbi, H.; Bahroun, S.; Massaoudi, M.; Zagrouba, E. Key frames extraction using graph modularity clustering for efficient video summarization. In Proceedings of the ICASSP 2017—2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; IEEE: Piscataway, NJ, USA, 2017.
29. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [[CrossRef](#)]
30. Smith, A.R. Color gamuts transform pairs. In *ACM-SIGGRAPH 78 Conference Proceedings*; Association for Computing Machinery: New York, NY, USA, 1978.
31. Otsu, N. Threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern.* **1979**, *9*, 62–66. [[CrossRef](#)]
32. Panetta, K.; Gao, C.; Agaian, S. Human-Visual-System-Inspired Underwater Image Quality Measures. *IEEE J. Ocean. Eng.* **2016**, *41*, 541–551. [[CrossRef](#)]
33. Liu, J.; Gong, S.; Guan, W.; Li, B.; Liu, J. Tracking and localization based on multi-angle vision for underwater target. *Electronics* **2020**, *9*, 1871. [[CrossRef](#)]
34. Moisan, L.; Moulon, P.; Monasse, P. Automatic homographic registration of a pair of images, with a contrario elimination of outliers. *Image Process. Line* **2012**, *2*, 329–352. [[CrossRef](#)]
35. Govindu, V.M. Lie-algebraic averaging for globally consistent motion estimation. In Proceedings of the IEEE Computer Society Conference on Computer Vision & Pattern Recognition, Washington, DC, USA, 27 June–2 July 2004.
36. Martinec, D.; Pajdla, T. Robust Rotation and Translation Estimation in Multiview Reconstruction. In Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition, Minneapolis, MN, USA, 18–23 June 2007; IEEE: Piscataway, NJ, USA, 2007.
37. Hartley, R.; Aftab, K.; Trunpf, J. L1 rotation averaging using the Weiszfeld algorithm. In Proceedings of the 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011.
38. Holland, P.W.; Welsch, R.E. Robust regression using iteratively reweighted least-squares. *Commun. Stat.* **1977**, *6*, 813–827. [[CrossRef](#)]
39. Rousseeuw, P.J.; Driessen, K.V. Computing lts regression for large data sets. *Data Min. Knowl. Discov.* **2006**, *12*, 29–45. [[CrossRef](#)]
40. Lowe, D. Distinctive image features from scaleinvariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
41. Cover, T.M.; Hart, P.E. Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* **1953**, *13*, 21–27. [[CrossRef](#)]
42. Furukawa, Y.; Ponce, J. Accurate, dense, and robust multiview stereopsis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 1362–1376. [[CrossRef](#)]
43. Wei, X.; Liu, Y.; Gao, S.; Wang, X.; Yue, H. An RNN-based delay-guaranteed monitoring framework in underwater wireless sensor networks. *IEEE Access* **2019**, *7*, 25959–25971. [[CrossRef](#)]
44. Liu, J.; Wei, X.; Bai, S. Autonomous underwater vehicles localisation in mobile underwater networks. *Int. J. Sens. Netw.* **2017**, *23*, 61. [[CrossRef](#)]