

# Robust Real-Time AUV Self-Localization Based on Stereo Vision-Inertia

Yangyang Wang, *Student Member, IEEE*, Dongbing Gu, *Senior Member, IEEE*, Xiaorui Ma, *Member, IEEE*, Jie Wang, *Senior Member, IEEE*, and Hongyu Wang, *Member, IEEE*

**Abstract**—Autonomous underwater vehicles (AUVs) play an important role in deep-sea exploration, in which AUV self-localization is a key component. However, due to poor visibility caused by challenging marine environments, AUVs are often equipped with high-cost and heavy-weight acoustic sensors to accomplish localization tasks. We propose a robust real-time AUV self-localization method based on stereo camera and inertial sensor, which merges point and diagonal features, as well as inertial measurements to overcome the challenges of poor visibility. Our method also includes an underwater loop detection algorithm based on the combination of points and diagonal segments, which can extract effective binary descriptors in low-textured underwater scenarios. Furthermore, we develop an AUV self-localization system based on a real-time, portable, low-cost, and small volume sensor suite. Finally, we test the proposed method in a real underwater environment using our sensor suite, and the experimental results demonstrate the effectiveness of the proposed method under dramatically changing underwater scenarios.

**Index Terms**—Localization, AUV, stereo vision-inertia, underwater.

## I. INTRODUCTION

**A**UTONOMOUS underwater vehicles (AUVs) have demonstrated significant potentials in the exploitation and utilization of marine resources in recent years, for instance, deep-sea exploration, subaqueous construction, underwater rescue, routine seafood products monitoring, pipeline detecting and real-time seabed mapping [1], [2], [3], [4]. For complex and dynamic marine environments, self-localization function of AUVs is the foundation for accomplishing potential applications. Self-localization capability could significantly improve the AUV autonomy while performing tasks in an unknown environment [5].

This work was supported in part by the National Natural Science Foundation of China under Grants 62271102, and 62071081, and in part by the Fundamental Research Funds for the Central Universities under Grant DUT21GF204, and in part by the Dalian Science and Technology Innovation Foundation under Grant 2022JJ11CG002. Yangyang Wang is currently a Visiting Fellow with the University of Essex, supported by the China Scholarship Council program.

Yangyang Wang, Xiaorui Ma, and Hongyu Wang are with the School of Information and Communication Engineering, Dalian University of Technology, Dalian, 116024, P.R. China (e-mail: yyw@mail.dlut.edu.cn, maxr@dlut.edu.cn, whyu@dlut.edu.cn).

Dongbing Gu is with the School of Computer Science and Electronic Engineering, University of Essex, CO4 3SQ Colchester, U.K. (e-mail: dgu@essex.ac.uk)

Jie Wang is with the School of Information Science and Technology, Dalian Maritime University, Dalian 116026, P.R. China (e-mail: wang\_jie@dlmu.edu.cn).

The underwater environments for sensors to be used for localization and mapping tasks is harsh. Although a few high-precision sensors could lead to good results, the high-power and heavy-weight limit their wider applications. Radio waves are extremely strongly attenuated in saltwater because seawater which contains a tremendous amount of salt is a conductive transmission medium and could significantly attenuate the electromagnetic waves. Acoustic sensors, such as underwater acoustic sensor networks (UASNs) [6], Doppler velocity log (DVL) [7], sonar [8], could also provide effective localization capability, but the high-cost and low-bandwidth hinder their application scenarios. In particular, many breakwaters have been built up near ports, where the magnetic interference is strong, and some acoustic positioning devices like ultra-short baseline (USBL) is prone to fail. And UASNs need to go through a cumbersome installation process before deployment. LiDAR sensors have been successfully applied in self-driving cars due to their high precision in range. However, they are bulky on small underwater vehicles and the propagating energy attenuates rapidly in water.

Recently visual simultaneous localization and mapping (SLAM) technology has demonstrated a certain level of accuracy and robustness on ground mobile vehicles with the benefits of light weight and low power consumption. Mur-Artal et al. [9] proposed a feature-based SLAM system (ORB-SLAM) for monocular, stereo and RGB-D cameras, which demonstrates real-time performance. Gálvez-López et al. [10] used the BoW technology to retrieve images, which can effectively solve the loop detection problem. Subsequently, a number of approaches [11], [12], [13] based on ORB-SLAM extensions have been proposed. Most vision-based localization algorithms [14], [15], [16], [17] were used to estimate the pose of ground mobile robots or unmanned drones. Direct application in large-scale underwater environments still poses significant challenges due to the nature of complex dynamic underwater scenes, such as texture-less images, changing viewpoints, and motion blurs. In this paper, we propose to use an inexpensive inertial measurement unit (IMU) sensor together with stereo camera to implement a robust real-time AUV self-localization algorithm to address the issues posed by underwater challenging scenes.

Due to the ability to achieve high accuracy in localization, visual SLAM with IMU sensors has been applied for ground mobile platforms [18], [19]. However, when deployed directly in marine environments, the robustness of those methods usually decreases in low texture underwater environments. It is not easy to find a large number of keypoint features

and sometimes they failed to complete the planned path. Our algorithm utilises new combined features with IMU sensors and we evaluated that they are effective and efficient in underwater environments. Specifically, we first extract the ORB feature points, and then we continue to extract diagonal segments around ORB features. In this way, some potential linear shape elements can be extracted in a low texture environment. Afterwards, the obtained feature points and line segments are fused with IMU readings. Our motivation is that the pose estimation in some challenging underwater scenes could be compensated for with IMU information in short intervals. Our loop detection component also benefits from our new combined features in some challenging underwater scenes. We have successfully implemented the proposed method on a NVIDIA TX2 platform in the Yellow Sea area. Our system is designed as a real-time, portable, low-cost, and small volume unit for AUV self-localization. Experimental results demonstrate that the proposed method could produce a small error of 0.39 m under dramatically changing underwater scenarios and achieve real-time performance. We summarize our contributions as follows:

- We propose an underwater self-localization scheme based on stereo vision-inertia to solve the robustness problem in low-textured dynamic underwater environments.
- We propose an underwater loop detection algorithm based on the combination of points and diagonal segments, which can extract effective binary descriptors in challenging marine environments.
- We develop a real-time, portable, low-cost, and small volume sensor suite based AUV self-localization system, and carry out extensive underwater experiments to test its performance.

The paper is structured as follows. Section II gives a brief introduction to some related works. Section III presents an overview of the proposed robust real-time self-localization framework for AUVs. In Section IV, we elaborate the details of our proposed algorithm. Section V describes our experimental results in the fire pool and the Yellow Sea. Finally, the conclusions and directions for future work are summarized in Section VI.

## II. RELATED WORKS

Self-localization is an important technology in AUVs that could be deployed to conduct exploration, underwater rescue and oceanographic survey. In the early years, magnetic compass was usually adopted for voyage. However, it can only determine the course angle and cannot be used for navigation alone. It needs to be combined with speed, initial position and other information. At present, most underwater navigation algorithms are based on acoustic sensors such as sonar, USBL, DVL [20], and long baseline (LBL) [4], [21], [22]. Teixeira et al. [23] proposed a submap-based technique that utilizes sonar to map underwater structures with complex geometries. In [24], the authors described a system that uses sonar and navigation sensors to acquire features of interest in a shallow-water ocean environment. Xu et al. [25] proposed a filtering method for cooperative localization under compass failure

and non-Gaussian measurement noise. In [26], the authors proposed a filtering framework, in which the factor related to the speed of propagation of the acoustic waves in the medium is explicitly considered and estimated. However, using acoustic sensors to collect data in aquaculture is costly, and sometimes deploying the underwater acoustic sensor node equipment requires significant time to be invested.

Recently, with the rapid development of underwater communication technologies, UASNs have been emergent and are attracting the interest of numerous researchers. More specifically, Yan et al. [6] proposed an AUV-aided localization scheme for UASNs with current field estimation, including AUVs, surface buoys, active and passive sensor nodes. In [27], the authors proposed a Q-learning based delay-aware routing approach to extending the lifetime of underwater sensor networks. This route recalculation in UASNs is very difficult within intermediate nodes due to low intelligent sensors and additional energy consumption. Afterwards, Lin et al. [28] employed software-defined networking (SDN) technology and proposed an SDN-enabled distributed architecture for delay-sensitive spatiotemporal routing in UASNs. Nevertheless, the long cruise distance of an AUV and limited bandwidth could increase the transmission delay [29]. In addition, sensor nodes often have passive motions caused by ocean currents or tides, which could also affect the accuracy of positioning.

For terrestrial environments, self-driving cars and unmanned aircrafts are equipped with high-precision 3D LiDAR. Rozenberszk et al. [30] introduced a LiDAR-only odometer and localization system that could deal with the challenges of globally localizing a vehicle in urban scenarios. In [31], the authors proposed a global localization scheme using 3D LiDAR scan, which relies on learning-based descriptions of point cloud segments and computes the full 6-DOF pose in a map. In order to achieve better performance in road estimations, some researchers adopted the strategy of fusing multiple sensors. For example, Gu et al. [32] proposed a road detection method based on a multi-modal conditional random field, which combines both sparse LiDAR point clouds with dense camera images for road detection in real-time. However, underwater vehicles in civil fields are always equipped with low-cost sensors and low computational resources. As the size of LiDAR is large, it is not suitable for small water spaces.

Cameras have the potential to make localization systems more cost-effective. In the literature, some kinds of vision-based methods have been applied to the localization of unmanned equipment [33], [34], [35], [36]. Lippiello et al. [37] presented a pose tracking system based on a single camera to localize an aerial drone. Rublee et al. [38] proposed a fast and accurate binary descriptor based on BRIEF, called ORB, which is rotation invariant and resistant to noise. Then, Mur-Artal et al. [39] presented ORB-SLAM using ORB features. In order to further improve the robustness, some researchers have made extensions on top of ORB-SLAM. Zuo et al. [40] developed a robust efficient visual SLAM system based on ORB-SLAM, which utilizes point and line features. In [41], the authors proposed a stereo visual SLAM system that combines both points and line segments to improve the robustness. However, none of these methods is combined with IMU information and

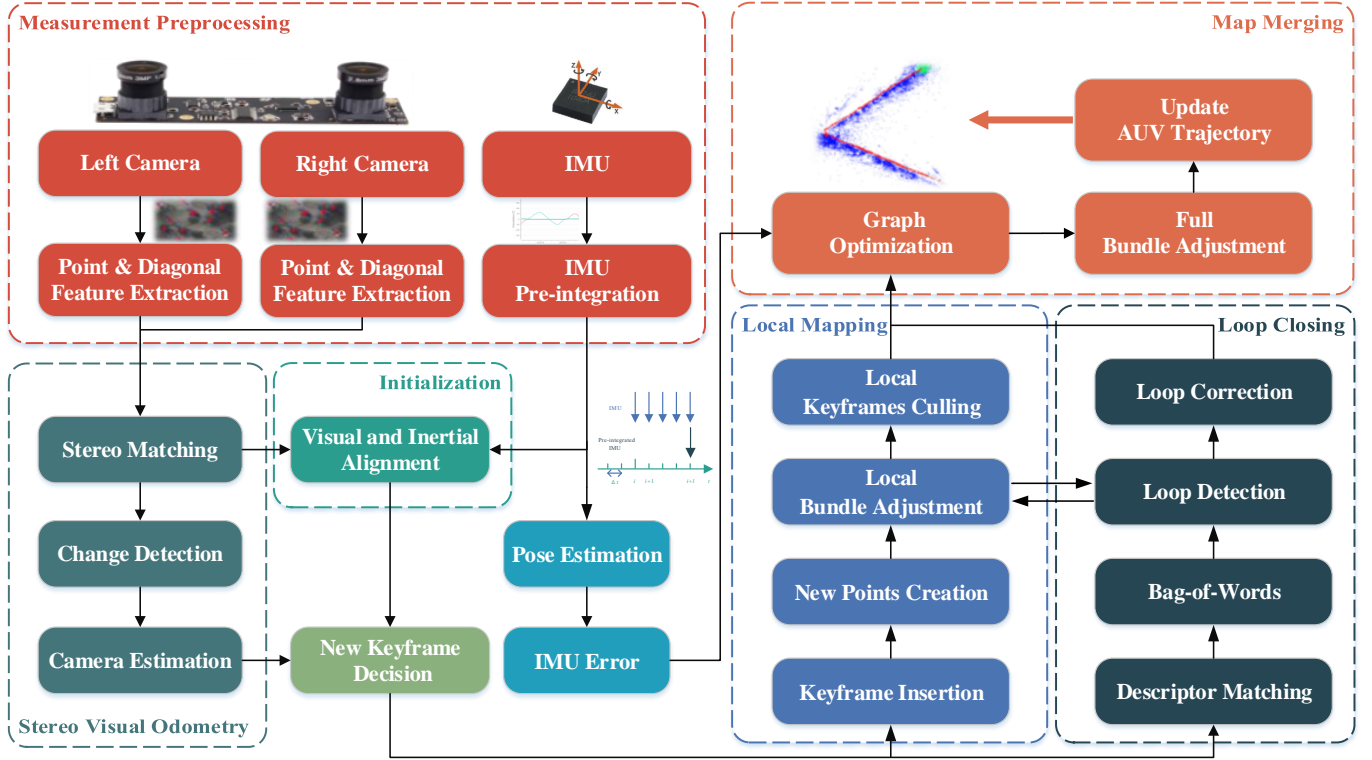


Fig. 1. The outline of the proposed algorithm.

they have not been applied for underwater applications.

Thanks to the low-cost IMU, vision camera is often combined with IMU and other sensors to improve the estimation of pose. An IMU consists of accelerometers measuring linear acceleration and gyroscopes measuring angular velocity. Due to the drift of accelerometers and gyroscopes, the estimation error will increase with time. In [18], the authors proposed a visual-inertial fusion methodology for state estimation with online calibration refinement. Wang et al. [42] proposed an underwater self-localization method based on Pseudo-3D vision-inertia for AUVs, which merges depth information with a 2D visual image to achieve continuous and robust localization. Eisele et al. [43] developed and experimentally validated a plenoptic navigation system on a low-cost central processing unit, which validates that vision-inertia is feasible in unstable marine environments.

Different from other existing studies that employ costly sonars, USBLs, or LiDARs to acquire measurement data, our proposed scheme, as described in the following section, includes a stereo camera, and a low-cost IMU to improve the performance of underwater self-localization estimation. A key feature of our localization method is to achieve the pose estimation for a miniature underwater vehicle with low-cost sensors and limited computational resources. Therefore, a robust underwater self-localization method for AUVs is proposed. Since our method is based on underwater vision and inertia, we would like to highlight the main differences with other papers. For example, compared with [13], we do not need a pose constraint from acoustic odometry to link

two submaps. Compared with [44], our optimization scheme is better than its extended Kalman filter (EKF) framework, i.e. we can reduce the accumulated error with time. In addition, we do not need to prepare artificial markers in advance. It is more suitable for unknown underwater environments.

### III. SYSTEM OVERVIEW

The architecture of the proposed robust real-time AUV self-localization system based on stereo vision-inertia is shown in Fig. 1. The system starts with a measurement preprocessing module, in which point and diagonal segment features are extracted and tracked, and IMU measurements between two consecutive frames are pre-integrated. The stereo visual odometry module matches the stereo keypoints, and then estimates the pose by detecting the changes. After inserting a new keyframe, the local mapping and loop closing modules begin to perform in parallel. In both processes, we use all IMU measurements, local visual measurements and feature correspondences retrieved from loop closure to optimize the AUV pose. Finally, the map merging module performs global optimization to update the 3D trajectory of the AUV after eliminating the drift. The aforementioned modules run concurrently in a multithread setting. Each module has different running rates and real-time guarantee to ensure that the whole system runs reliably at all times.

Table I shows the main notations used in this paper. We define the IMU coordinate system as the right-handed, which is the same as the body frame. The axis directions are shown in Fig. 2.

TABLE I  
NOTATION DEFINITIONS

Name	Description
$(\cdot)^w$	The world frame
$\mathbf{b}_a$	The accelerometer biases
$\mathbf{b}_w$	The gyroscope biases
$\mathbf{g}^w$	The gravity vector in the world frame
$\mathbf{n}_a$	The additive noise in acceleration measurements
$\mathbf{n}_w$	The additive noise in gyroscope measurements
$\mathbf{R}$	The rotation matrices
$\mathbf{p}_k$	The position states of the $k^{th}$ frame
$\mathbf{v}_k$	The velocity states of the $k^{th}$ frame
SO(3)	The special orthogonal group
SE(3)	The special Euclidean group
Exp	$\mathbb{R}^3 \rightarrow \text{SO}(3)$
Log	$\text{SO}(3) \rightarrow \mathbb{R}^3$
$\Delta t$	The sampling period of the IMU

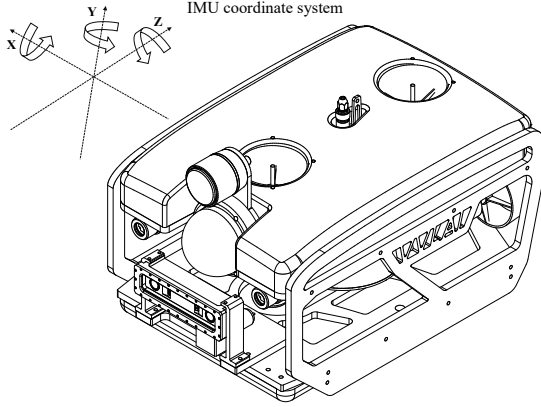


Fig. 2. The IMU coordinate system of the underwater vehicle.

#### IV. UNDERWATER STEREO VISUAL-INERTIAL SELF-LOCALIZATION METHOD

This section describes the proposed real-time self-localization algorithm. The main modules in Fig. 1 will be described with details in the subsections below.

##### A. Measurement Preprocessing

The raw accelerometer  $\hat{\mathbf{a}}$  and gyroscope  $\hat{\boldsymbol{\omega}}$  measurements from IMU are represented as:

$$\begin{aligned}\hat{\mathbf{a}}_t &= \mathbf{a}_t + \mathbf{b}_{a_t} + \mathbf{R}_t^t \mathbf{g}^w + \mathbf{n}_a, \\ \hat{\boldsymbol{\omega}}_t &= \boldsymbol{\omega}_t + \mathbf{b}_{w_t} + \mathbf{n}_w.\end{aligned}\quad (1)$$

We consider  $\hat{\mathbf{a}}_t$  and  $\hat{\boldsymbol{\omega}}_t$  are measured in the body frame, and we define the IMU frame to be the same as the body frame. The IMU measurements are affected by acceleration bias  $\mathbf{b}_a$ , gyroscope bias  $\mathbf{b}_w$ , and additive noise. We assume that the

additive noise in acceleration and gyroscope measurements are Gaussian,  $\mathbf{n}_a \sim \mathcal{N}(\mathbf{0}, \sigma_a^2)$ ,  $\mathbf{n}_w \sim \mathcal{N}(\mathbf{0}, \sigma_w^2)$ :

$$\begin{aligned}\dot{\mathbf{b}}_{a_t} &= \mathbf{n}_{b_a}, \\ \dot{\mathbf{b}}_{w_t} &= \mathbf{n}_{b_w}.\end{aligned}\quad (2)$$

All the state information for the visual-inertial optimization is defined as follows:

$$\mathbf{X}_k \simeq \{\mathbf{R}_k, \mathbf{p}_k, \mathbf{v}_k, \mathbf{b}_a, \mathbf{b}_w\}, \quad (3)$$

where  $\mathbf{X}_k$  represents the IMU state when taking the  $k^{th}$  image. The above formula includes the orientation, position and velocity of an IMU measurement, as well as the acceleration bias and gyroscope bias in the IMU body frame.

We denote  $\Delta t$  as the sampling period of the IMU,  $i$  and  $i+1$  as two consecutive IMU measurement time instants,  $n$  and  $n+1$  as two consecutive keyframe time instants. Fig. 3 shows the different rates for the IMU and camera. The motion estimate between time  $n$  and  $n+1$  can be computed as:

$$\begin{aligned}\mathbf{R}_{n+1} &= \mathbf{R}_n \prod_{t=i}^{i+l-1} \text{Exp}((\hat{\boldsymbol{\omega}}_t - \mathbf{b}_{w_t} - \mathbf{n}_w) \Delta t), \\ \mathbf{p}_{n+1} &= \mathbf{p}_n + \sum_{t=i}^{i+l-1} \left[ \mathbf{v}_t \Delta t + \frac{1}{2} \mathbf{g} \Delta t^2 + \frac{1}{2} \mathbf{R}_t (\hat{\mathbf{a}}_t - \mathbf{b}_{a_t} - \mathbf{n}_a) \Delta t^2 \right], \\ \mathbf{v}_{n+1} &= \mathbf{v}_n + \sum_{t=i}^{i+l-1} \mathbf{g} \Delta t + \sum_{t=i}^{i+l-1} \mathbf{R}_t (\hat{\mathbf{a}}_t - \mathbf{b}_{a_t} - \mathbf{n}_a) \Delta t,\end{aligned}\quad (4)$$

where  $l$  is the number of IMU measurement steps between two keyframes.

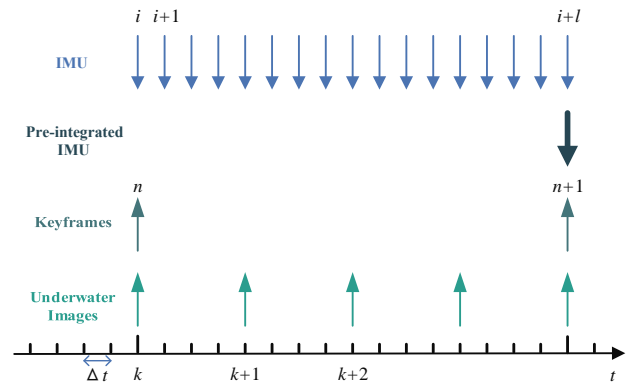


Fig. 3. Illustration of the different rates for IMU and camera.

We preintegrate the IMU measurements between two consecutive underwater images. Then we can obtain the rotation, position and velocity states of the  $(k+1)^{th}$  frame as the initial value for the visual odometry. They can be written as  $\mathcal{B}_{k,k+1} = (\Delta \mathbf{R}_{k,k+1}, \Delta \mathbf{p}_{k,k+1}, \Delta \mathbf{v}_{k,k+1})$ , and the information matrix for the measurements is  $\sum_{\mathcal{B}_{k,k+1}}$ . The IMU measurement residual  $\mathbf{r}_{\mathcal{B}_{k,k+1}}$  can be stated as:

$$\begin{aligned}
\mathbf{r}_{\mathcal{B}_{k,k+1}} &= \left[ \mathbf{r}_{\Delta \mathbf{p}_{k,k+1}}^T, \mathbf{r}_{\Delta \mathbf{v}_{k,k+1}}^T, \mathbf{r}_{\Delta \mathbf{R}_{k,k+1}}^T \right]^T, \\
\mathbf{r}_{\Delta \mathbf{R}_{k,k+1}} &= \text{Log}(\Delta \mathbf{R}_{k,k+1}^T \mathbf{R}_k^T \mathbf{R}_{k+1}), \\
\mathbf{r}_{\Delta \mathbf{p}_{k,k+1}} &= \mathbf{R}_k^T (\mathbf{p}_{k+1} - \mathbf{p}_k - \mathbf{v}_k \Delta t - \frac{1}{2} \mathbf{g} \Delta t^2) - \Delta \mathbf{p}_{k,k+1}, \\
\mathbf{r}_{\Delta \mathbf{v}_{k,k+1}} &= \mathbf{R}_k^T (\mathbf{v}_{k+1} - \mathbf{v}_k - \mathbf{g} \Delta t_{k,k+1}) - \Delta \mathbf{v}_{k,k+1}. \quad (5)
\end{aligned}$$

### B. Stereo Visual Odometry

We extract point and diagonal segment features in stereo underwater images and search for matches for each left feature in the right image. After establishing the corresponding relationship between two stereo frames, we project the keypoints and their diagonal features from the current frame to the next frame. We estimate the camera motion iteratively by using Gauss-Newton least-square [45], [46] to minimize the projection errors of the keypoints and their diagonal features. The motion estimation between the  $k^{th}$  and  $(k+1)^{th}$  frames can be expressed by  $\mathbf{T}_{k,k+1}$ .

It is assumed that the stereo image is corrected so that the epipolar line is horizontal. Then, we generate the stereo keypoint with the coordinates of the left feature and the horizontal coordinate of the right match. In the extraction of point features, we leverage the ORB features method [9] which provides good invariance to changes in viewpoint and illumination and allows for a fast, efficient keypoint matching. We define  $\mathbf{r}_{k,j}$  as the reprojection errors at the  $k^{th}$  frame for a position  $p_j$ :

$$\mathbf{r}_{k,j} = \mathbf{u}_{k+1,j} - \pi(\mathbf{T}_{k,k+1}, \mathbf{u}_{k,j}), \quad (6)$$

where  $\mathbf{u}_{k,j}$  is the visual observation of a point  $p_j$  at the  $k^{th}$  image,  $\mathbf{u}_{k+1,j}$  is the visual observation of the same point at the  $(k+1)^{th}$  image,  $j \in [1, m]$ .  $\pi$  is the back projection function, which projects a pixel at current frame into a pixel at next frame using camera intrinsic parameters and  $\mathbf{T}_{k,k+1}$ . The information matrix is defined as  $\Sigma_{k,j}$ .

When all ORB features are extracted in the frame, we extract the line segments similar to four-rectangle features around its points. As shown in Fig. 4, we take 5 pixels up and down the diagonal direction of the feature point to form a new line segment, which looks very similar to a diagonal feature.

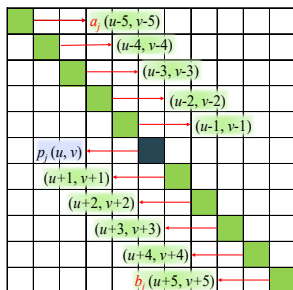


Fig. 4. Illustration of a point and its diagonal segment feature.

We define  $\mathbf{a}_{k,j}$ ,  $\mathbf{b}_{k,j}$  as the 2D endpoints of a diagonal segment  $\mathbf{d}_{k,j}$  in the  $k^{th}$  frame,  $\mathbf{a}_{k+1,j}$ ,  $\mathbf{b}_{k+1,j}$  as the 2D endpoints located on the corresponding diagonal segment in the  $(k+1)^{th}$  frame. The normalized line can be written as:

$$\mathbf{d}_{k+1,j} = \frac{\mathbf{a}_{k+1,j} \times \mathbf{b}_{k+1,j}}{|\mathbf{a}_{k+1,j} \times \mathbf{b}_{k+1,j}|}, \quad (7)$$

where  $\mathbf{d}_{k+1,j}$  is the  $j^{th}$  diagonal segment in the  $(k+1)^{th}$  frame.

We then define the error function  $e_{k,j}$  to represent the distance between the endpoints of the diagonal segment in the  $k^{th}$  frame and its corresponding line in the  $(k+1)^{th}$  frame. The line reprojection error  $e_{k,j}$  can be expressed as:

$$e_{k,j} = \left[ \frac{\mathbf{d}_{k+1,j} \cdot \pi(\mathbf{T}_{k,k+1}, \mathbf{a}_{k,j})}{|\mathbf{d}_{k+1,j} \cdot \pi(\mathbf{T}_{k,k+1}, \mathbf{b}_{k,j})|} \right]. \quad (8)$$

### C. Local Mapping

In order to ensure the real-time performance of the system, the local mapping and loop closure detection module described in the next subsection will be performed only when a new keyframe is inserted. The keyframe includes the observed stereo features and their descriptors, and the visual descriptors corresponding to the left image calculated by the visual vocabulary. After a new keyframe is inserted, the system starts to perform the bundle adjustment procedure of the local map.

Given a set of keyframes  $N$  with their state vectors  $\{\mathbf{X}_0, \mathbf{X}_1, \dots, \mathbf{X}_k, \dots, \mathbf{X}_N\}$ , a set of observation points  $m$  with their state vectors  $\{\mathbf{u}_{0,0}, \mathbf{u}_{0,1}, \dots, \mathbf{u}_{k,j}, \dots, \mathbf{u}_{N,m}\}$ , and a set of observation positions of the endpoints for diagonal segments  $m$  with their state vectors  $\{(\mathbf{a}_{0,0}, \mathbf{b}_{0,0}), (\mathbf{a}_{0,1}, \mathbf{b}_{0,1}), \dots, (\mathbf{a}_{k,j}, \mathbf{b}_{k,j}), \dots, (\mathbf{a}_{N,m}, \mathbf{b}_{N,m})\}$ , we can define the vector  $\chi$ , which contains all the above state vectors to be optimized. The vision-inertia optimization function can be written as:

$$\min_{\chi} \left\{ \sum_{k=1}^N \|\mathbf{r}_{\mathcal{B}_{k-1,k}}\|_{\Sigma_{\mathcal{B}_{k,k+1}}}^2 + \sum_{k=1}^N \sum_{j=0}^m \rho(\|\mathbf{r}_{k,j}\|_{\Sigma_{k,j}}) + \sum_{k=1}^N \sum_{j=0}^m \rho(\|e_{k,j}\|_{\Sigma_{k,j}}) \right\}, \quad (9)$$

where  $\rho$  is used to decrease the impact of spurious matchings in the reprojection error.

### D. Loop Closing

We extend a DBow2 method [10], a bag-of-words place recognition approach, to improve the underwater loop closure detection process using points and diagonal segments. This procedure is performed in parallel with the local mapping module. We extract the visual descriptor information of key points and diagonal segments from each underwater image, then stack them into a word vector, and finally construct a vocabulary from different underwater datasets. When running the loop closure detection module, the system searches the



database for the image which contains the word most similar to the current keyframe.

Once a loop closure is successfully detected, we correct the error distributed along the loop, which is solved by the pose graph optimization.  $\mathbf{T}_{n,n+1} \in \text{SE}(3)$  is the motion estimation between the  $n^{\text{th}}$  and  $(n+1)^{\text{th}}$  keyframes. During the optimization, the corresponding element  $\xi_{n,n+1} \in \mathfrak{se}(3)$  of the associated Lie-algebra is given. We can write the error function as follows:

$$\mathcal{L}_{n,n+1}(\xi_n, \xi_{n+1}) = \log(\exp(\xi_{n,n+1}) \cdot \exp(\xi_{n+1}) \cdot \exp(\xi_n)^{-1}), \quad (10)$$

where  $\log : \text{SE}(3) \mapsto \mathfrak{se}(3)$  is the matrix logarithm which maps a transformation matrix to an element on the tangent space,  $\exp : \mathfrak{se}(3) \mapsto \text{SE}(3)$ .

We leverage the general graph optimization (g2o) library [47] to solve the pose graph optimization problem and obtain the optimized pose of keyframes. The landmarks (both keypoints and diagonal segments) observed are updated while updating the pose of the keyframes. We estimate their relative motion by fusing the matched landmarks. Concretely, we seek a match between the features from both keyframes while also searching for a new correspondence between the current keyframe and the local map associated with the old keyframe. Finally, we merge the local maps on both sides of the loop.

### E. Map Merging

In this paper, we fuse the features of points and diagonal segments extracted from stereo underwater images, as well as the measurements from IMU. We use the sliding window of keyframes and their points as the optimizable variables. All underwater features seen by all these keyframes, and keyframe poses observing these features are also optimized. Here, the features and keyframes are associated through reprojection errors. In the map merging process, the pose graph optimization is able to propagate the loop correction to the rest of the map. Then, after considering the mid-term and long-term matching of the loop closure, the global bundle adjustment is used to find the pose estimation. Note that the global bundle adjustment is implemented only when the number of keyframes is lower than the threshold to avoid the huge computational costs. After the graph optimization and global bundle adjustment, the system will update the 3D trajectory of the AUV.

## V. EXPERIMENTS AND ANALYSIS

### A. Experimental Setup

We test the proposed method in real dynamic underwater scenarios to evaluate its performance. The self-design underwater vehicle for algorithm verification consists of the following components:

- NVIDIA Jetson TX2,
- Forward-looking stereo global shutter camera with  $752 \times 480$  resolution,
- Inertial measurement unit (ADXL326 and ADXRS620, package alignment error:  $\pm 1$  degrees, interaxis alignment error:  $\pm 0.1$  degrees),

- Lithium-ion rechargeable battery (24VDC, 10Ah).

The custom-made sensor suite was designed for underwater scenario localization and mapping as the target application. Extensive experiments were carried out in a fire pool and the Yellow Sea. The size of whole underwater vehicle system is  $550 \times 400 \times 300 \text{ mm}^3$  in dimensions. It consists of a forward-looking stereo global shutter camera, an IMU, a pair of LEDs, a lithium-ion battery (Charging Voltage: 220VAC), etc. The AUV is driven by the lithium battery and is equipped with four thrusters, including two plane omnidirectional thrusters and two vertical thrusters, as shown in Fig. 5.



Fig. 5. The picture of the underwater vehicle testbed that we used for the experiment. It includes a forward-looking stereo global shutter camera, an IMU, a depth gauge, a pair of LEDs, etc.

The underwater real scene experiment is an arduous task in reality, which is limited by a variety of conditions. We first set up water tank experiments to test the motion performance of the AUV, as shown in Fig. 6 (a). To quickly evaluate the performance of the algorithm in real-time, we use a display emulator on NVIDIA TX2 to debug the underwater system through a remote desktop so that we can obtain a high-resolution remote desktop for operation.

In order to fairly evaluate the proposed method and facilitate the comparison with the ground truth, we have also carried out experiments in the Yellow Sea area, as shown in Fig. 6 (b). It can be seen from Fig. 6 (b) that there are some stones scattered along the coast of the port. We manipulated the AUV to move along the parallel direction of the stones, and its movement direction is shown by the red arrow in Fig. 6 (b). Due to the influence of waves in the ocean, the AUV was easily failed to move even in the straight line. Here we used a PID algorithm to mediate the control system of the AUV so that it can navigate in the predefined direction. At the same time, we use an Unmanned Aerial Vehicle (UAV) to track the trajectory of the AUV in real-time. Since we have no ground truth in the Yellow Sea experimental environment, we evaluate our method using the GPS data of the UAV as the ground truth.

### B. Experimental Results

This section presents the experimental results. We set up two different experimental environments, and all data are collected by our custom-made underwater sensor suite. To collect datasets under different scenarios, ROS [48] has been

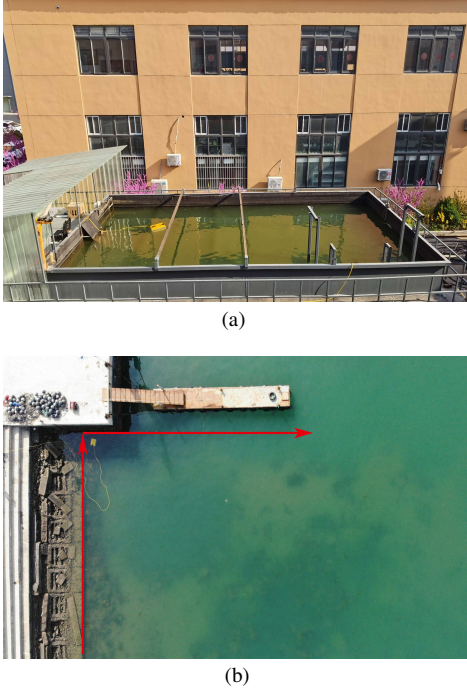


Fig. 6. Underwater experiment scene. (a) The picture of the water tank experiment. (b) Top view of the Yellow Sea experiment, taken by a UAV.

used to record data in bag files. The features tracked by the forward-looking camera in the dim underwater environment are shown in Fig. 7. As it can be seen in the example shown in Fig. 7, the proposed method can extract point and diagonal features in the low-textured dynamic underwater environment. The detailed results of the two experiments are as follows:

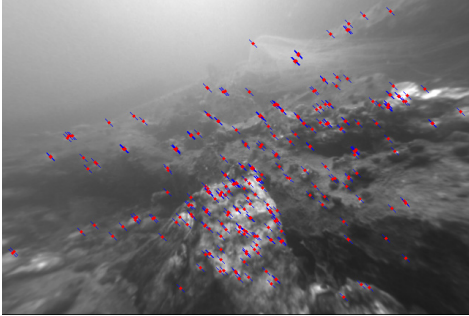


Fig. 7. Features tracked by the forward-looking camera in the Yellow Sea.

In the first experimental scenario of the port, we chose a place with a right angle for testing, as shown in Fig. 8 (a). We operated the AUV to move in the direction of the red arrow in Fig. 8 (a), and (b) shows the trajectory generated by Pangolin, which is a lightweight library for managing visualization and user interaction that wraps OpenGL functions. We also compare the results of the proposed method with the ground truth to evaluate our algorithm. Fig. 9 shows the detailed comparison results. Note that after losing the IMU compensation, the tracking is often lost at the corner, resulting in localization failure. Fig. 9 (c) shows the incomplete trajectory. Clearly, our visual-inertial scheme and loop closure detection are effective. In order to further verify the effectiveness of the proposed

method in the loop closure detection, we manipulated the AUV to move a rectangular trajectory. Fig. 10 shows the loop closure detection result, which demonstrates that our system successfully detects that the AUV returned to the starting position.

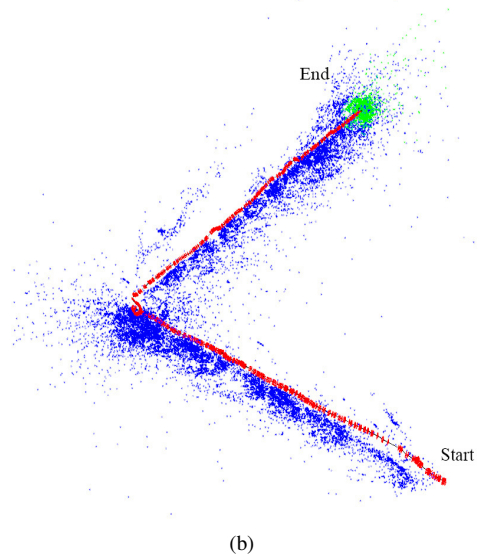
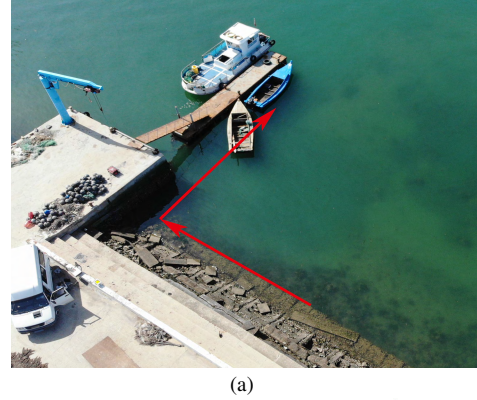


Fig. 8. Experimental results of the proposed method at the port. (a) The real scene of the experiment in the Yellow Sea, where the most data in this paper were collected. (b) Aerial view of real-time AUV localization and mapping at right angles. Both blue and green represent map points, where green represents current local map points and red represents keyframes.

As we have described in the Introduction section, many breakwaters have been built up near the port. The magnet interference there is strong, and some acoustic positioning equipment is prone to failure. We also tried to use SeaTrac X150 USBL micro positioning system to generate the trajectory at the right angle of the port, but unfortunately, the positioning trajectory is prone to drift. Fig. 11 shows the trajectory drift by the SeaTrac PinPoint-Underwater-Navigation Software. We can intuitively see that the SeaTrac X150 USBL positioning system is not suitable for this coastal environment with a magnetic field. It further shows that our method is effective.

In the second experimental scenario of the Yellow Sea area, we did not test the proposed system near ports in the shallow sea, but far from the coast. The scenario we set up is shown in Fig. 12 (a). In this complex dynamic environment, it was quite hard for us to get the ground truth. We used the

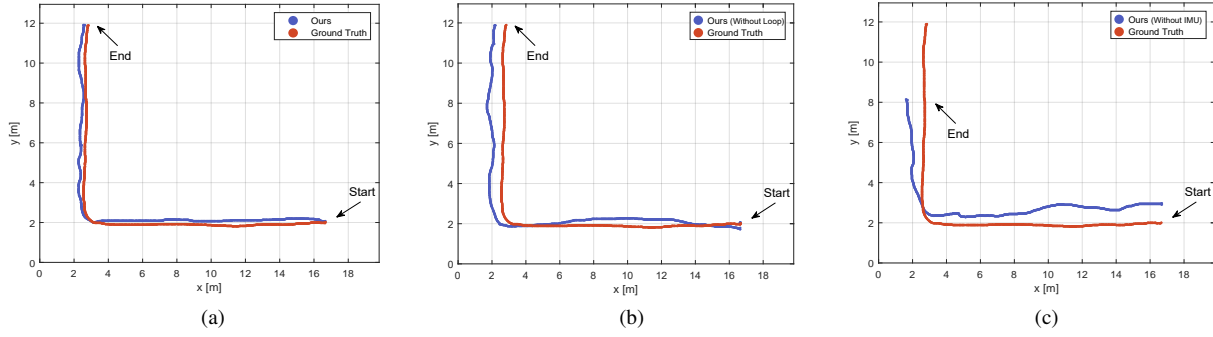


Fig. 9. Comparison of trajectory (blue) and ground truth (red). (a) Our method (Stereo camera+IMU+Loop). (b) Our method without loop (Stereo camera+IMU). (c) Our method without IMU (Stereo camera+Loop). Best viewed in color.

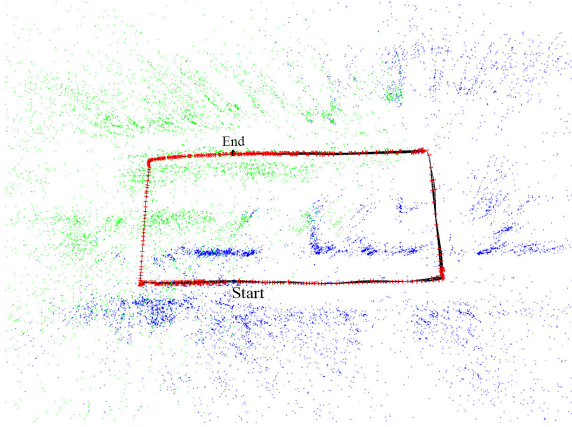


Fig. 10. Experimental results of the proposed system in underwater loop closure detection. Both blue and green represent map points, where green represents current local map points and red represents keyframes.

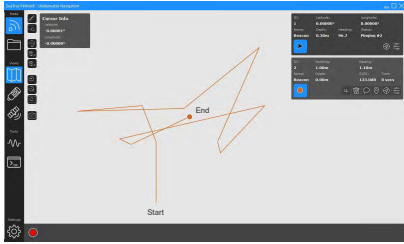


Fig. 11. Drift trajectories are generated in the SeaTrac PinPoint-Underwater-Navigation Software.

custom-made sensor suite to evaluate the proposed method by navigating around the visible rocks. Based on our experimental equipment, we can only evaluate it visually without the ground truth. Fig. 12 (b), (c) and (d) show aerial views of trajectories generated by the visualization tool in Pangolin, respectively. It also shows the position of visual features. It is obvious that the proposed system draws the trajectory in real-time as the AUV gradually moves forward, up and down. Therefore, the results show that our method also works in marine space far from the coast.

To validate the performance of the proposed localization system, we calculated the average processing time of three threads: visual odometry, local mapping, and loop closing, and analyzed the Root Mean Square Error (RMSE) by us-

ing NVIDIA Jetson TX2 in experimental scenarios. Detailed statistics are shown in Table II. Please note that while the visual odometry is continuously processing new frames, local mapping and loop closing run in parallel threads. In particular, the loop closure detection does not need to process each frame. The visual odometry thread processing time is 98 ms, which shows that the localization system can run in real-time. In the stereo-inertial configuration, the RMSE of our system is 0.39 m. Subsequently, we tested the proposed system without any loop closure detection, and the localization accuracy decreased significantly, increasing to 0.66 m. Finally, we tested the proposed system without IMU. Unfortunately, the system performed poorly or even fails. Especially when the AUV sailed to a corner in low-textured underwater environments, the system was prone to collapse after the tracking was lost. In this case, we cannot obtain the complete underwater trajectory. We used the partial trajectory to calculate the error, which increases to 3.07 m.

TABLE II  
STATISTICS FOR ALGORITHM PERFORMANCE

Average Runtime (ms)			Localization Errors (m)		
Visual Odometry	Local Mapping	Loop Closing	Stereo+IMU With Loop	Stereo+IMU Without Loop	Stereo+Loop Without IMU
98	251	39	0.39	0.66	3.07

We also analyzed the Mean Absolute Error (MAE) for the localization while using the stereo-inertial configuration. The MAE value is 0.44 m. Fig. 13 illustrates the cumulative distribution functions (CDFs) of the localization errors. Meanwhile, we compare our method with two related underwater localization methods proposed by Jung et al. [49] and Lee et al. [50], both of which are vision-based localization for AUVs. The comparison of localization accuracy is shown in Table III. As we can see from the results that our method has a smaller RMES than the other works, and the self-localization system has advantages in coastal environments with a magnetic field.

### C. Influence of different times on results

Real-time AUV self-localization in an unstable marine environment is a challenging task. Water flow or tide, particles and plankton all could cause various imaging backgrounds,



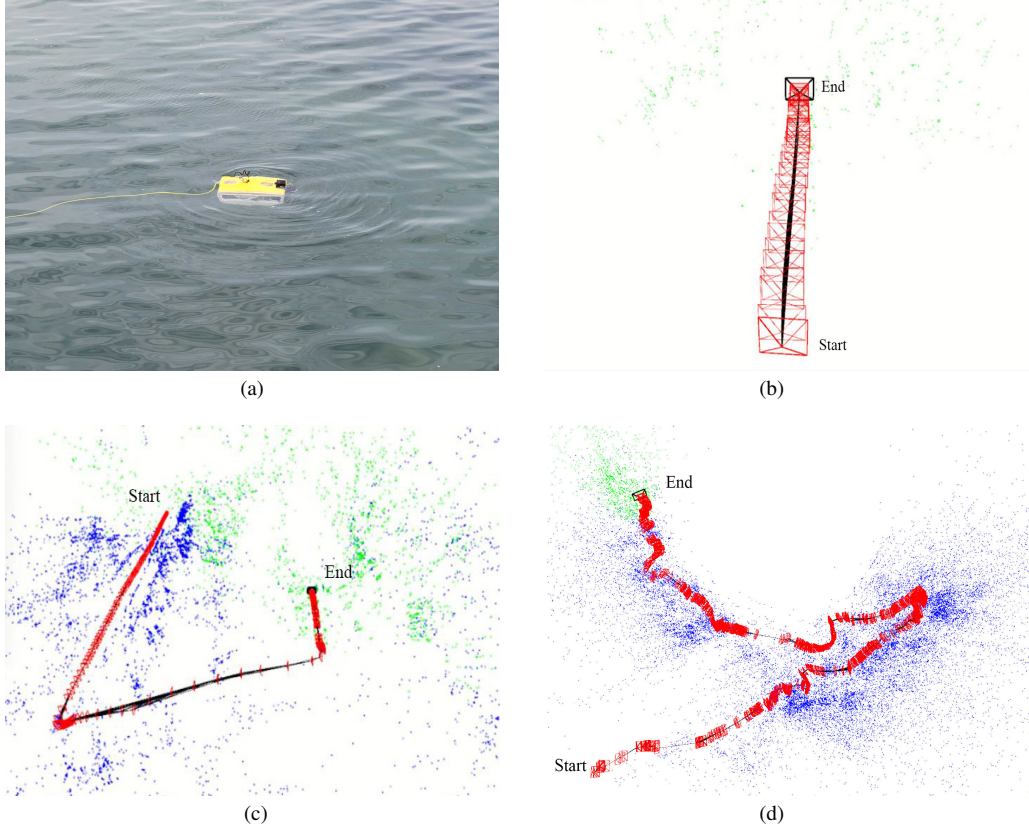


Fig. 12. Experimental results of the proposed method for AUVs in marine space far from the coast. In (a), we chose an area located far from the coast. Aerial views of the different trajectories are shown in (b), (c) and (d). It can be clearly shown that the proposed system is also suitable for similar selected complex areas. (a) The experimental scenario is far from the coast. (b), (c) and (d) Aerial views of the different trajectories shown in Pangolin. Both blue and green represent map points, where green represents current local map points and red represents keyframes.

TABLE III  
COMPARISON OF LOCALIZATION ACCURACY

Method	Jung et al. [49]	Lee et al. [50]	Ours
RMSE (m)	0.67	0.52	0.39

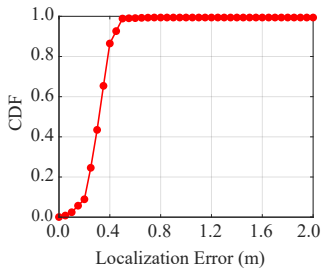


Fig. 13. CDF of localization errors.

which could bring the noise to the system. In order to compare the influence of dynamic water current, low illuminations and colour shift on the experimental results, we tested the proposed system at different times of the day. At 7:00 am, at low tide, the sea was quiet and the underwater light was sufficient. At this moment, the RMSE was 0.36 m. At 11:00 am, the AUV floated slightly with the water during the rising tide of the

sea. The RMSE now was increased to 0.39 m. At 2:00 pm, the sea was at a high tide, and the AUV swung greatly with the water. The RMSE was 0.44 m, and sometimes the system was prone to collapse. At 6:00 pm, the light intensity in the Yellow Sea was weak and the colour of the water was dark. At this time, it was easy to lose features, resulting in localization failure. We leveraged auxiliary LED lights to cope with low light conditions, and the RMSE was 0.52 m.

## VI. CONCLUSION

This paper proposed a robust underwater self-localization scheme based on stereo vision-inertia with low-cost sensors and low computational processors. Specifically, this method merges point and diagonal features, as well as IMU measurements to overcome the challenges caused by complex unstable marine scenarios. Our underwater loop detection algorithm is based on the combination of points and diagonal segments, which can extract effective binary descriptors. In addition, we develop an AUV self-localization system based on a real-time, portable, low-cost, and small volume sensor suite, including a stereo camera, and a low-priced IMU. Our proposed method was evaluated by our custom-made underwater sensor suite on real-world maritime space. Experimental results show that the proposed approach can implement the localization task with an accuracy of 0.39 m. In particular, our scheme is more suitable than the SeaTrac X150 USBL for accurate AUV

self-localization in coastal environments with magnetic fields. Therefore, the proposed underwater self-localization approach provides an effective navigation solution for AUVs in the complex marine environment.

It should be noted that there is still a great many of work to be done for the AUV in order to further improve the localization and mapping accuracy. For example, how to use the strong feature learning ability of deep neural networks to extract potential features? How to improve the accuracy of localization in challenging unstable marine environments? How to apply decentralized visual SLAM to AUVs in extreme circumstances? We will try to address some of these challenges in our future research.

## REFERENCES

- [1] Y. Li, B. Li, W. Yu, S. Zhu, and X. Guan, "Cooperative localization based multi-AUV trajectory planning for target approaching in anchor-free environments," *IEEE Trans. Veh. Technol.*, vol. 71, no. 3, pp. 3092–3107, Mar. 2022.
- [2] Y. Wang, X. Ma, J. Wang, S. Hou, J. Dai, D. Gu, and H. Wang, "Robust AUV visual loop closure detection based on variational auto-encoder network," *IEEE Trans. Ind. Informat.*, 2022, doi:10.1109/TII.2022.3145860.
- [3] T. Kan, Y. Zhang, Z. Yan, P. P. Mercier, and C. C. Mi, "A rotation-resilient wireless charging system for lightweight autonomous underwater vehicles," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 6935–6942, Aug. 2018.
- [4] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *IEEE J. Oceanic Eng.*, vol. 39, no. 1, pp. 131–149, Jan. 2014.
- [5] W. Wang and G. Xie, "Online high-precision probabilistic localization of robotic fish using visual and inertial cues," *IEEE Trans. Ind. Electron.*, vol. 62, no. 2, pp. 1113–1124, Feb. 2015.
- [6] J. Yan, D. Guo, X. Luo, and X. Guan, "AUV-aided localization for underwater acoustic sensor networks with current field estimation," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 8855–8870, Aug. 2020.
- [7] D. Wang, X. Xu, Y. Yang, and T. Zhang, "A quasi-newton quaternions calibration method for DVL error aided GNSS," *IEEE Trans. Veh. Technol.*, vol. 70, no. 3, pp. 2465–2477, Mar. 2021.
- [8] J. Cui, G. Han, Y. Su, and X. Fu, "Non-uniform non-orthogonal multicarrier underwater communication for compressed sonar image data transmission," *IEEE Trans. Veh. Technol.*, vol. 70, no. 10, pp. 10133–10145, Oct. 2021.
- [9] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017.
- [10] D. Gálvez-López and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences," *IEEE Trans. Robot.*, vol. 28, no. 5, pp. 1188–1197, Oct. 2012.
- [11] S. Aldegheri, N. Bombieri, D. D. Bloisi, and A. Farinelli, "Data flow ORB-SLAM for real-time performance on embedded GPU boards," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2019, pp. 5370–5375.
- [12] L. Tiwari, P. Ji, Q.-H. Tran, B. Zhuang, S. Anand, and M. Chandraker, "Pseudo RGB-D for self-improving monocular SLAM and depth prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 437–455.
- [13] E. Vargas, R. Scona, J. S. Willners, T. Luczynski, Y. Cao, S. Wang, and Y. R. Petillot, "Robust underwater visual SLAM fusing acoustic sensing," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 2140–2146.
- [14] X. Gao, R. Wang, R. Demmel, and D. Cremers, "LDSO: Direct sparse odometry with loop closure," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2018, pp. 2198–2204.
- [15] R. Li, S. Wang, and D. Gu, "DeepSLAM: A robust monocular SLAM system with unsupervised deep learning," *IEEE Trans. on Ind. Electron.*, vol. 68, no. 4, pp. 3577–3587, Apr. 2021.
- [16] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.
- [17] Q. Fu, H. Yu, X. Wang, Z. Yang, Y. He, H. Zhang, and A. Mian, "Fast ORB-SLAM without keypoint descriptors," *IEEE Trans. Image Process.*, vol. 31, pp. 1433–1446, Dec. 2021.
- [18] Z. Yang and S. Shen, "Monocular visual-inertial state estimation with online initialization and camera-IMU extrinsic calibration," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 1, pp. 39–51, Jan. 2017.
- [19] X. Dong, X. Dong, J. Dong, and H. Zhou, "Monocular visual-IMU odometry: A comparative evaluation of detector-descriptor-based methods," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 6, pp. 2471–2484, Jun. 2020.
- [20] S. Xu, T. Luczynski, J. S. Willners, Z. Hong, K. Zhang, Y. R. Petillot, and S. Wang, "Underwater visual acoustic SLAM with extrinsic calibration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2021, pp. 7647–7652.
- [21] T. Zhang, J. Wang, L. Zhang, and L. Guo, "A student's T-based measurement uncertainty filter for SINS/USBL tightly integration navigation system," *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 8627–8638, Sep. 2021.
- [22] Y. Yao, X. Xu, L. Hou, K. Deng, and X. Xu, "A simple and precise correction method for DVL measurements under the dynamic environment," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 10750–10758, Oct. 2020.
- [23] P. V. Teixeira, M. Kaess, F. S. Hover, and J. J. Leonard, "Underwater inspection using sonar-based volumetric submaps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2016, pp. 4288–4295.
- [24] M. F. Fallon, J. Folkesson, H. McClelland, and J. J. Leonard, "Relocating underwater features autonomously using sonar-based SLAM," *IEEE J. Ocean. Eng.*, vol. 38, no. 3, pp. 500–513, Jul. 2013.
- [25] B. Xu, X. Wang, J. Zhang, Y. Guo, and A. A. Razzaqi, "A novel adaptive filtering for cooperative localization under compass failure and non-gaussian noise," *IEEE Trans. Veh. Technol.*, vol. 71, no. 4, pp. 3737–3749, 2022.
- [26] P. Batista, "GES long baseline navigation with unknown sound velocity and discrete-time range measurements," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 1, pp. 219–230, Jan. 2014.
- [27] Z. Jin, Y. Ma, Y. Su, S. Li, and X. Fu, "A Q-learning-based delay-aware routing algorithm to extend the lifetime of underwater sensor networks," *Sensors*, vol. 17, no. 7, pp. 1–15, Jul. 2017.
- [28] C. Lin, G. Han, M. Guizani, Y. Bi, and J. Du, "A scheme for delay-sensitive spatiotemporal routing in SDN-enabled underwater acoustic sensor networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 9280–9292, Sep. 2019.
- [29] G. Han, S. Shen, H. Wang, J. Jiang, and M. Guizani, "Prediction-based delay optimization data collection algorithm for underwater acoustic sensor networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 6926–6936, Jul. 2019.
- [30] D. Rozenberszki and A. L. Majdik, "LOL: Lidar-only odometry and localization in 3D point cloud maps," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 4379–4385.
- [31] S. Ratz, M. Dymczyk, R. Siegwart, and R. Dubé, "Oneshot global localization: Instant lidar-visual pose estimation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 5415–5421.
- [32] S. Gu, Y. Zhang, J. Tang, J. M. Alvarez, and H. Kong, "Integrating dense LiDAR-camera road detection maps by a multi-modal CRF model," *IEEE Trans. Veh. Technol.*, vol. 68, no. 12, pp. 11635–11645, Dec. 2019.
- [33] A. Couturier and M. A. Akhloufi, "A review on absolute visual localization for UAV," *Robot. Auton. Syst.*, vol. 135, Jan. 2021, Art. no. 103666.
- [34] M. Nava, A. Paolillo, J. Guzzi, L. Gambardella, and A. Giusti, "Learning visual localization of a quadrotor using its noise as self-supervision," *IEEE Robotics Autom. Lett.*, vol. 7, no. 2, pp. 2218–2225, Apr. 2022.
- [35] M. Dusmanu, O. Miksik, J. L. Schönberger, and M. Pollefeys, "Cross-descriptor visual localization and mapping," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 6058–6067.
- [36] X. Yang, L. Zhou, H. Jiang, Z. Tang, Y. Wang, H. Bao, and G. Zhang, "Mobile3DRecon: Real-time monocular 3D reconstruction on a mobile phone," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 12, pp. 3446–3456, Dec. 2020.
- [37] V. Lippiello and J. Cacace, "Robust visual localization of a UAV over a pipe-rack based on the Lie group SE (3)," *IEEE Robotics Autom. Lett.*, vol. 7, no. 1, pp. 295–302, Jan. 2022.
- [38] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2564–2571.
- [39] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015.

- [40] X. Zuo, X. Xie, Y. Liu, and G. Huang, "Robust visual SLAM with point and line features," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2017, pp. 1775–1782.
- [41] R. Gomez-Ojeda, F.-A. Moreno, D. Zuniga-Noël, D. Scaramuzza, and J. Gonzalez-Jimenez, "PL-SLAM: A stereo SLAM system through the combination of points and line segments," *IEEE Trans. Robot.*, vol. 35, no. 3, pp. 734–746, Jun. 2019.
- [42] Y. Wang, X. Ma, J. Wang, and H. Wang, "Pseudo-3D vision-inertia based underwater self-localization for AUVs," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7895–7907, Jul. 2020.
- [43] J. Eisele, Z. Song, K. Nelson, and K. Mohseni, "Visual-inertial guidance with a plenoptic camera for autonomous underwater vehicles," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2777–2784, Jul. 2019.
- [44] P. Zhang, Z. Wu, J. Wang, S. Kong, M. Tan, and J. Yu, "An open-source, fiducial-based, underwater stereo visual-inertial localization method with refraction correction," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2021, pp. 4331–4336.
- [45] S. Wright and J. Nocedal, "Numerical optimization," *Springer Sci.*, vol. 35, pp. 67–68, 1999.
- [46] F.-A. Moreno, J.-L. Blanco, and J. González-Jiménez, "ERODE: An efficient and robust outlier detector and its application to stereovisual odometry," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 4691–4697.
- [47] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G2O: A general framework for graph optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 3607–3613.
- [48] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: An open-source robot operating system," in *Proc. ICRA Open-Source Softw. Workshop*, 2009, pp. 5–5.
- [49] J. Jung, Y. Lee, D. Kim, D. Lee, H. Myung, and H. T. Choi, "AUV SLAM using forward/downward looking cameras and artificial landmarks," in *2017 IEEE Underwater Technology (UT)*, 2017, pp. 1–3.
- [50] D. Lee, D. Kim, S. Lee, H. Myung, and H. T. Choi, "Experiments on localization of an AUV using graph-based SLAM," in *2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 2013, pp. 526–527.



**Xiaorui Ma** (M'17) received the B.S. degree in applied mathematics from Lanzhou University, Lanzhou, China, in 2008, and Ph.D. degree in communication and information system from Dalian University of Technology, Dalian, China, in 2017. She is currently an Associate Professor at Dalian University of Technology. Her research interests include processing and analysis of remote sensing images, specially hyperspectral image classification and synthetic aperture radar image classification.



**Jie Wang** (SM'18) received his B.S. degree from the Dalian University of Technology, Dalian, China, in 2003, M.S. degree from Beihang University, Beijing, China, in 2006, and Ph.D. degree from the Dalian University of Technology, Dalian, China, in 2011, all in electronic engineering. He is currently a Full Professor at Dalian Maritime University. He used to be an Associate Professor at Dalian University of Technology from 2014 to 2017. He was a visiting researcher with the University of Florida from 2013 to 2014. His research interests include wireless localization and tracking, radio tomography, wireless sensing, machine learning, wireless sensor networks, and cognitive radio networks. He serves as an Associate Editor for IEEE Transactions on Vehicular Technology.



**Yangyang Wang** (S'22) received the B.S. and M.S. degrees in electronic information science and technology from the Xinjiang Agricultural University, Urumqi, China, in 2014, and 2017, respectively. He is currently working toward the doctoral degree from the School of Information and Communication Engineering of Dalian University of Technology (DUT), Dalian, China. Since October 2021, he has been a Visiting Fellow with the School of Computer Science and Electronic Engineering, University of Essex, Colchester, U.K. His research interests

include underwater localization, underwater simultaneous localization and mapping (SLAM), machine learning and computer vision.



**Dongbing Gu** (SM'07) received the B.Sc. and M.Sc. degrees in control engineering from the Beijing Institute of Technology, Beijing, China, in 1985 and 1988, respectively, and the Ph.D. degree in robotics from the University of Essex, Colchester, U.K., in 2004.

From October 1996 to October 1997, he was an Academic Visiting Scholar with the Department of Engineering Science, University of Oxford, Oxford, U.K. In 2000, he joined the University of Essex as a Lecturer. He is currently a Professor of Robotics with the School of Computer Science and Electronic Engineering, University of Essex. His research interests include robotics, multiagent systems, cooperative control, model predictive control, visual simultaneous localization and mapping, wireless sensor networks, and machine learning.



**Hongyu Wang** (M'98) received the B.S. degree in electronic engineering from the Jilin University of Technology, Changchun, China, in 1990, the M.S. degree in electronic engineering from the Graduate School, Chinese Academy of Sciences, Beijing, China, in 1993, and the Ph.D. degree in precision instrument and optoelectronics engineering from Tianjin University, Tianjin, China, in 1997. He is currently a Professor with the Dalian University of Technology, Dalian, China. His research interests include image processing, and underwater localization.