Hybrid-VINS: Underwater Tightly Coupled Hybrid Visual Inertial Dense SLAM for AUV

Yaming Ou, Junfeng Fan, *Member, IEEE*, Chao Zhou, Pengju Zhang, and Zeng-Guang Hou, *Fellow, IEEE*

***Abstract*—Traditional visual simultaneous localization and mapping (SLAM) methods primarily rely on passive vision, such as monocular cameras, which often exhibit reduced accuracy in localization and struggle to create dense maps in low-light underwater conditions. In this article, a tightly coupled hybrid visual inertial navigation system (VINS), named Hybrid-VINS, is proposed by fuzing active and passive vision, which enables robust localization and dense mapping underwater. Specifically, a self-designed active vision device called underwater binocular structured light (UBSL) is integrated to provide more accurate depth estimation of the scene, enhancing the initialization and visual feature tracking process of monocular VINS. As well as it addresses the deficiency of VINS in dense mapping. In addition, a more robust hybrid vision-aided loop closure detection algorithm is proposed to mitigate issues stemming from pure passive vision misjudgments and mismatches. Furthermore, an underwater autonomous hybrid vision system is developed in both simulated and real-world underwater environments to collect various datasets for verifying the performance of Hybrid-VINS. Experimental results demonstrate that, compared with passive VINS, Hybrid-VINS holds greater promise for underwater high-precision localization and dense mapping.**

***Index Terms*—Hybrid vision, sensor fusion, state estimation, underwater simultaneous localization and mapping (SLAM), underwater vehicles.**

I. Introduction

AUTONOMOUS underwater vehicle (AUV) [1], [2], [3] plays an increasingly important role in human ocean engineering such as marine resource exploration and mapping of undersea structures [4]. To further improve its autonomy, simultaneous localization and mapping (SLAM) technology is indispensable [5]. Specifically, high-precision localization serves as the foundation for autonomous robot navigation [6], [7]. Dense mapping holds greater significance for robots to explore the unknown sea than sparse mapping. Compared with pure visual odometry, visual inertial navigation system (VINS) provides superior dynamic tracking performance [8] and is more suitable for AUV with irregular movements. However, a complete VINS still relies heavily on the vision module for feature extraction and tracking, which poses challenges underwater due to the sparseness of underwater features, absorption of light by water, scattering by particles and refraction effects during imaging. Consequently, the existing VINS has limited effectiveness for direct use underwater, necessitating improvements in localization accuracy and mapping performance.

In order to improve the effectiveness of vision SLAM systems in underwater environments, the current mainstream approach involves introducing additional vehicle information. For instance, Joshi et al. [9], [10] combined the kinematic characteristics of underwater robots to propagate attitude when the visual inertial odometry (VIO) fails. Introducing absolute position information from a long baseline (LBL) system, Song et al. [11] proposed an acoustic-visual inertial navigation system to address the global position agnostic issue of underwater VINS. In addition, Vargas et al. [12] utilized doppler velocity log (DVL) and inertial measurement unit (IMU) for dead reckoning (DR) to achieve acoustic absolute odometry to enhance the stability of the visual SLAM system in poor visibility and insufficient texture scene. Similarly, Xu et al. [13] fused DVL and IMU to implement acoustic relative odometry, which was integrated into the ORB-SLAM3 [14] system to improve its robustness. Considering the large cumulative error of DR, Huang et al. [15] used only the velocity measurements from DVL and integrated them into the visual odometry. Although the above systems utilized additional sensor information to enhance localization accuracy in SLAM, the mapping results often exhibit limited improvement.

Considering that additional high-precision 3-D information not only improves the localization accuracy of the SLAM

Manuscript received 8 April 2024; revised 27 June 2024 and 19 July 2024; accepted 26 July 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62373354, in part by the Beijing Natural Science Foundation under Grant 4232057, in part by the Youth Innovation Promotion Association, Chinese Academy of Sciences under Grant 2022130, and in part by the Fund of Key Laboratory of Artificial Intelligence, Ministry of Education under Grant AI2022010. *(Corresponding author: Junfeng Fan.)*

Yaming Ou, Junfeng Fan, and Chao Zhou are with the Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: ouyaming2021@ia.ac.cn; junfeng.fan@ia.ac.cn; chao.zhou@ia.ac.cn).

Pengju Zhang and Zeng-Guang Hou are with the State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: pengju.zhang@ia.ac.cn; zengguang.hou@ia.ac.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TIE.2024.3440515.

Digital Object Identifier 10.1109/TIE.2024.3440515

system, but also improves the map building effect, researchers have begun integrating point cloud data from other sensors. Using a BlueRobotics Ping Sonar to measure the distance between the vehicle and the seabed, Xu et al. [16] obtained the depth information of partially tracked feature points in the visual odometry, which were then directly used for state estimation. Rahman et al. [17] proposed a multisensor fusion-based underwater SLAM system, named SVIn2. In which, scanning profile sonar range information was used for improved mapping and localization. Similarly, Zhao et al. [18] proposed a DVL point cloud enhanced feature position recovery algorithm by utilizing the four specific 3-D points from DVL to improve the localization performance. However, the aforementioned method retrieved only a few 3-D points each time, thus limiting its effectiveness in improving SLAM performance. Utilizing multibeam forward-looking sonars with richer measurement data, Cardaillac and Ludvigsen [19] proposed a camera-sonar combination algorithm, enhancing scale estimation and mapping of ORB-SLAM3. However, acoustic-based distance measurement suffers from elevation ambiguity, leading to uncertainty in associating with image feature points. What is more, the loop closure detection module of the above system is still mainly based on a purely passive vision scheme, which tends to misjudge and mismatch in underwater environments, resulting in poor system robustness [20].

In summary, the existing VINS still exhibits the following deficiencies in practical applications for AUV.

1) *Localization*: Inaccurate feature depth recovery reduces localization accuracy.
2) *Mapping*: Insufficient information density hampers detailed observation of unknown underwater environments.
3) *Loop Detection*: Solutions relying solely on passive vision tend to suffer from misjudgments and mismatches.

Motivated by the above works, accurate distance information would greatly improve the performance of visual SLAM systems. As demonstrated in our previous research work [21], [22], an active vision-based underwater binocular structured light (UBSL) is highly advantageous for underwater dense mapping [23]. Therefore, in this article, we aim to integrate the active vision device UBSL into the VINS to achieve a more robust dense SLAM. Some innovative contributions are made as follows.

1) An underwater tightly coupled hybrid visual inertial dense SLAM framework, named Hybrid-VINS, is proposed, which is more suitable for underwater scenarios. To the best of our knowledge, this is the first underwater SLAM system to utilize active vision information to assist passive vision.
2) The self-designed structured light system is used to correct the depth measurement of some features during passive vision initialization and tracking, which improves the localization accuracy. In addition, the introduction of the structured light system information realizes the VINS dense mapping, which is very rare underwater.
3) A more robust hybrid vision-aided loop closure detection algorithm is proposed to overcome the inaccuracy of purely passive vision loop factor.

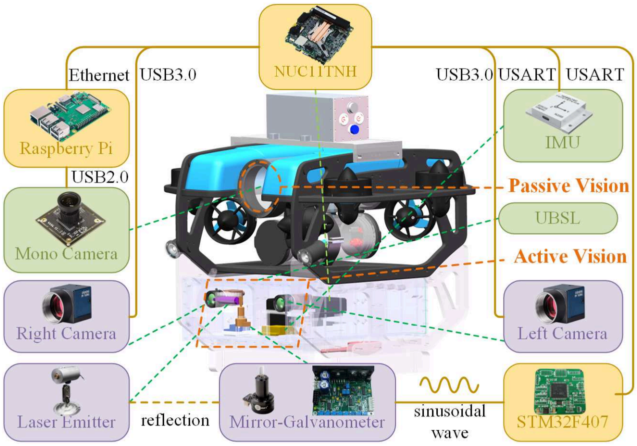

Fig. 1. Design and sensors of hybrid vision system.

4) The underwater autonomous hybrid vision system is developed in an underwater simulation environment and the real underwater world, respectively, to collect multiple datasets to validate the performance of Hybrid-VINS.

II. Mechatronic Design and Sensors

To evaluate the performance of Hybrid-VINS in a real underwater environment, an underwater autonomous hybrid vision system is designed, as shown in Fig. 1. The sensor component of the system comprises an active vision module, a passive vision module, and an inertial measurement module. Among them, the active vision module includes a binocular camera, a laser transmitter, and a galvanometer. The binocular camera has a resolution of 640 × 512 pixels, operating at 100 Hz. The laser transmitter emits a blue line laser, which is directed toward the galvanometer and then reflected onto the external object. Then, the galvanometer is controlled by a microcontroller unit to achieve left and right cyclic oscillation, with the oscillation angle reaching 120°. This setup enables the laser to scan the scene, providing 320 high-precision 3-D points per measurement with a vertical sector angle of 45°. The measurement frequency can reach 70 Hz. Moreover, the passive vision module consists of a monocular with an image resolution of 374 × 260 and a frequency of 20 Hz. The inertial measurement module comprises an IMU that can provide information about the linear acceleration and angular velocity of the AUV at a frequency of 400 Hz.

III. The Framework of Hybrid-VINS

Inspired by [24], [25], and [26], the overall framework of Hybrid-VINS is designed as illustrated in Fig. 2. It mainly consists of four modules: preprocessing, hybrid vision fusion, factor graph optimization, and hybrid vision-aided loop closure detection.

A. Preprocessing

Preprocessing module is mainly responsible for the preliminary information processing of VINS, including visual feature extraction, IMU preintegration, and vision-inertia initialization.

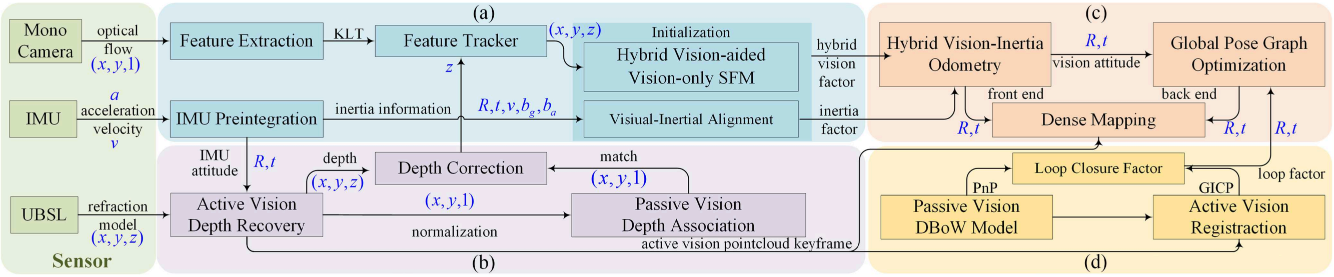

Fig. 2. Underwater tightly coupled hybrid visual inertial dense SLAM framework, named Hybrid-VINS. (a) Preprocessing. (b) Hybrid vision fusion. (c) Factor graph optimization. (d) Hybrid vision-aided loop closure detection.

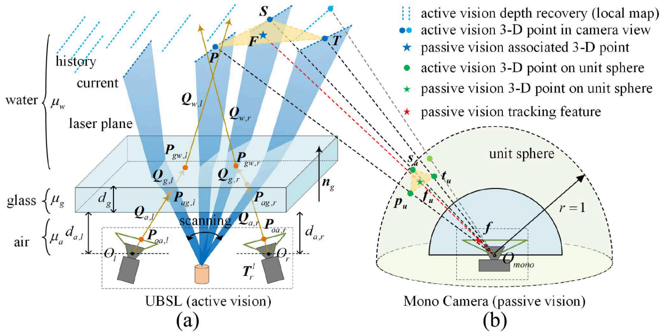

Fig. 3. Underwater hybrid vision fusion model. (a) Active visual depth recovery. (b) Passive vision depth association.

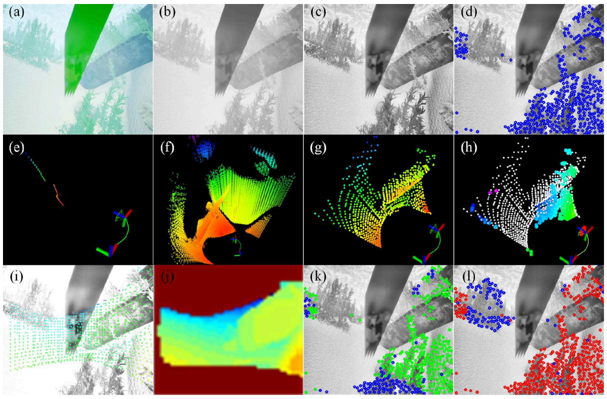

Fig. 4. Underwater hybrid vision fusion process. (a)–(c) Raw, gray, enhanced images. (d), (k), and (l) Feature images. Where blue points are passive visual features, green points are hybrid visual features, and red points are tracking features. (e)–(g) Active vision depth recovery. (h) Passive vision depth association. (i) Local map projection. (j) Rasterized map.

1) Vision Feature Extraction: Affected by diminished illumination and sparse texture features underwater, extracting optical flow features is comparatively easier than extracting feature points. Therefore, in Hybrid-VINS, optical flow-based feature tracking is adopted. Moreover, underwater imagery typically exhibits a bluish-green coloration, as depicted in Fig. 4(a). We first convert the image to grayscale and subsequently apply the underwater image enhancement algorithm CLAHE to accentuate scene details, as shown in Fig. 4(b) and 4(c). Finally, the features in the image are maintained at a certain number and the extraction results are shown in Fig. 4(d).

2) IMU Preintegration: In Hybrid-VINS, the high-frequency IMU information serves multiple purposes: first, it aids in scale initialization of the monocular SLAM module and contributes inertial constraints for back-end optimization. Moreover, the high-frequency IMU data enables robust state estimation of the robot at high frequencies, thereby compensating for motion aberrations in active vision. The IMU preintegration formulation is provided in [27].

3) Visual Inertial Initialization: In Hybrid-VINS, the initialization process starts with the implementation of the visual structure from motion (SFM), and then the results are aligned with IMU data to calibrate the gravity vector, scale, and inertial bias. Additionally, we use the active vision information to correct the depth of some feature in the SFM process, as described in Section III-B, so as to improve the accuracy of the visual initialization.

B. Hybrid Vision Fusion

1) Problem Formulation: In general, monocular SLAM relies on matching features between consecutive frames to generate 3-D map points using the triangulation model. Accurate 3-D map points are crucial information for the subsequent optimization process. But the accuracy of 3-D map points is constrained by factors such as the imprecision of underwater feature matching, the influence of refraction effects, and the presence of scale errors. In contrast, active vision, relying on the principle of refraction measurement, can provide high-precision 3-D point information. However, active vision 3-D points and passive vision feature points do not exhibit one-to-one correspondence. Therefore, the hybrid vision fusion module primarily addresses the challenge of determining the 3-D points corresponding to certain passive vision features based on the 3-D point sets provided by active vision.

2) Active Vision Depth Recovery: Given that the camera is typically fixed within a waterproof chamber, objects in the water are observed through transparent glass, leading to the occurrence of refraction effects. In pursuit of high-precision 3-D point cloud information, the investigation of a refraction-based measurement model is imperative. Building upon existing research, our measurement model is depicted in Fig. 3(a). Based on the camera's pinhole imaging model

$$\boldsymbol{P}_{oa,j} = \boldsymbol{K}_j^{-1}\boldsymbol{p}_j \tag{1}$$

where the second subscript j corresponds to l or r in the figure, denoting the left and right cameras, respectively. $\boldsymbol{K}_j$ is the intrinsic matrix of the camera j. $p_k = [u_k, v_k, 1]^T$ denotes the chi-square coordinate of a point on the laser centerline in the imaging plane. $\boldsymbol{P}_{*,j}$ denotes the intersection of light rays with different media. Since $\boldsymbol{O}_j$, $\boldsymbol{P}_{oa,j}$, and $\boldsymbol{P}_{ag,j}$ are covariant

$$\boldsymbol{Q}_{a,j} = \frac{\overrightarrow{P_{oa,j}P_{ag,j}}}{\|\overrightarrow{P_{oa,j}P_{ag,j}}\|} = \frac{\overrightarrow{O_jP_{ag,j}}}{\|\overrightarrow{O_jP_{ag,j}}\|} = \frac{\overrightarrow{O_jP_{oa,j}}}{\|\overrightarrow{O_jP_{oa,j}}\|} \tag{2}$$

where $\boldsymbol{Q}_{*,j}$ denotes the normal vector of light rays in different media. $\boldsymbol{P}_{oa,j}$ can be obtained from (1), substituting it gives

$$\boldsymbol{P}_{ag,j} = \frac{\boldsymbol{Q}_{a,j}}{\boldsymbol{Q}_{a,j}\cdot\boldsymbol{n}_g}d_{a,j}. \tag{3}$$

According to Snell's refraction principle, the direction vector of the refracted light can be determined by a linear combination of the normal vector of the refracting plane and the direction vector of the incident light, expressed as

$$\boldsymbol{Q}_{g,j} = \alpha_{ag}\boldsymbol{Q}_{a,j} + \beta_{ag}\boldsymbol{n}_g \tag{4}$$

where α_{ag} and β_{ag} can be calculated from the refractive index

$$\begin{cases} \alpha_{ag} = \frac{\mu_a}{\mu_g} \\ \beta_{ag} = \sqrt{1-\alpha_{ag}^2\left[1-(\boldsymbol{Q}_{a,j}\cdot\boldsymbol{n}_g)^2\right]} - \alpha_{ag}\boldsymbol{Q}_{a,j}\cdot\boldsymbol{n}_g \end{cases} \tag{5}$$

where μ_a, μ_g denote the refractive index of air and glass, respectively. Then, the ray intersection $\boldsymbol{P}_{gw,j}$ can be computed

$$\boldsymbol{P}_{gw,j} = \boldsymbol{P}_{ag,j} + \frac{\boldsymbol{Q}_{g,j}}{\boldsymbol{Q}_{g,j}\cdot\boldsymbol{n}_g}d_g. \tag{6}$$

Similar to (4) and (5)

$$\boldsymbol{Q}_{w,j} = \alpha_{gw}\boldsymbol{Q}_{g,j} + \beta_{gw}\boldsymbol{n}_g. \tag{7}$$

At this point, $\boldsymbol{P}_{gw,j}$ and $\boldsymbol{Q}_{w,j}$ are known. Due to the existence of systematic errors, the two lines may be heterogeneous. Assume that the target point $\boldsymbol{P}$ is the midpoint of the common perpendicular of the two anisotropic straight lines, so its coordinates can be calculated by $\boldsymbol{P}_{gw,l}$, $\boldsymbol{P}_{gw,r}$, $\boldsymbol{Q}_{w,l}$, and $\boldsymbol{Q}_{w,r}$. Up to this point, scene depth can be recovered based on active vision.

3) Passive Vision Depth Association: According to the active vision depth recovery model depicted in Fig. 3(a), a single computation of UBSL can obtain a single-line point cloud, referred to as a Scan, as shown in Fig. 4(e). Due to the limited information contained in a single Scan, Hybrid-VINS employs a sliding window approach to sustain a cloud of Scan points within a window, referred to as a local map. Specifically, by leveraging IMU preintegration, Hybrid-VINS can derive IMU-rate odometry information. Utilizing this high-frequency odometry, the Scan within the window can be converted from the UBSL coordinate system to a unified world coordinate system and fused into a denser local point cloud map, as shown in Fig. 4(f). The local map is then transformed to the monocular camera coordinate system. Due to the camera's limited field of view, we segment and eliminate the portions of the local map that fall outside the field of view. Additionally, voxel filtering is applied to the point cloud to decrease subsequent computational load, as demonstrated in Fig. 4(g). Subsequently, employing the camera imaging model, we project the point cloud acquired from active vision onto the passive vision imaging plane, as depicted in Fig. 4(i), thereby validating the accuracy of the point cloud segmentation. To enhance efficiency further, Fig. 4(i) undergoes rasterization. Within each raster, only the point with the smallest depth value is retained, while other points are discarded. Next, pseudocolor mapping is performed using the depth values of the preserved points within the raster. This results in Fig. 4(j), which can be interpreted as a local depth map corresponding to Fig. 4(c). Subsequently, all the retained points are scaled, projected onto the unit sphere of the monocular camera, and then stored into the K-dimensional (KD) tree. Likewise, for visual features extracted by passive vision, they are projected from the pixel plane onto the unit sphere in a similar manner. For each optical flow feature $\boldsymbol{f}$, assuming its coordinates on the unit sphere are $\boldsymbol{f}_u(x_{f_u}, y_{f_u}, 1)$, locate the three nearest active visual points $\boldsymbol{p}_u, \boldsymbol{s}_u, \boldsymbol{t}_u$, corresponding to $\boldsymbol{P}, \boldsymbol{S}, \boldsymbol{T}$ in 3-D space, as shown in Fig. 3(b). To acquire the coordinates of the 3-D points corresponding to the optical flow feature $\boldsymbol{f}$, the intersection point $\boldsymbol{F}$ of the ray $\boldsymbol{l}(\boldsymbol{O}_{mono}\boldsymbol{f}_u)$ with the plane $\boldsymbol{a}$ formed by $\boldsymbol{P}, \boldsymbol{S}, \boldsymbol{T}$ is considered to be the corresponding associated point. The calculations are as follows:

$$\begin{cases} \boldsymbol{n}_1 = \boldsymbol{P} - \boldsymbol{S} \\ \boldsymbol{n}_2 = \boldsymbol{S} - \boldsymbol{T} \end{cases} \tag{8}$$

where $\boldsymbol{n}_1, \boldsymbol{n}_2$ denote direction vectors, respectively. Then, the plane $\boldsymbol{a}$ can be expressed as: $\boldsymbol{a} = \boldsymbol{P} + \lambda\boldsymbol{n}_1 + \mu\boldsymbol{n}_2$, where λ and μ are arbitrary real numbers. In the monocular camera coordinate system, the point $\boldsymbol{O}_{mono}$ has the coordinates $(0, 0, 0)$, and the ray $\boldsymbol{l}$ can be expressed as $\boldsymbol{l} = \boldsymbol{f}_u t$, where t is a positive real number. Assuming the coordinates of $\boldsymbol{P}$ are $(x_{\boldsymbol{P}}, y_{\boldsymbol{P}}, z_{\boldsymbol{P}})$, λ, μ, and t can be solved by the following equation:

$$\begin{bmatrix} n_{1,x} & n_{2,x} & -x_{f_u} \\ n_{1,y} & n_{2,y} & -y_{f_u} \\ n_{1,z} & n_{2,z} & -1 \end{bmatrix} \begin{pmatrix} \lambda \\ \mu \\ t \end{pmatrix} = \begin{pmatrix} -x_{\boldsymbol{P}} \\ -y_{\boldsymbol{P}} \\ -z_{\boldsymbol{P}} \end{pmatrix}. \tag{9}$$

Finally, the coordinates of the associated point $\boldsymbol{F}$ can be obtained by substituting t into ray $\boldsymbol{l}$. The association and tracking results are shown in Fig. 4(h), 4(k), and 4(l).

C. Factor Graph Optimization

In Hybrid-VINS, a tightly coupled factor graph optimization framework is proposed. The state variables in per frame are defined as follows:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{R}^T & \boldsymbol{p}^T & \boldsymbol{v}^T & \boldsymbol{b}_a^T & \boldsymbol{b}_g^T & \boldsymbol{\lambda}_1 & ,\ldots, & \boldsymbol{\lambda}_m \end{bmatrix} \tag{10}$$

where $\boldsymbol{R}$, $\boldsymbol{p}$, and $\boldsymbol{v}$ denote the rotation, position, and velocity of the state in the world frame. $\boldsymbol{b}_a$ and $\boldsymbol{b}_g$ represent the accelerometer bias and gyroscope bias. The $\boldsymbol{\lambda}_m$ denotes the inverse depth corresponding to the mth passive visual feature on the current frame. In Hybrid-VINS, sliding window optimization using multiframe state variables together is presented as follows:

$$\mathcal{X} = \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \boldsymbol{x}_3 & ,\ldots, & \boldsymbol{x}_n \end{bmatrix} \tag{11}$$

where n indicates the window size. Various constraint factors are used to optimize these state variables.

1) Hybrid Vision Factor: For the image frame c_i in Hybrid-VINS, certain features in all successful matches with c_j acquire high-precision map points $\hat{\mathcal{A}}_l^{c_i}(l = 1,\ldots,m)$ through active vision in Section III-B, as indicated by the green point in Fig. 4(k). The remaining features $[\hat{u}_l^{c_i}, \hat{v}_l^{c_i}](l = 1,\ldots,n)$ corresponding to the map points obtained by passive visual triangulation with inverse depth $\lambda_l^{c_i}$ are denoted as $\hat{\mathcal{P}}_l^{c^i}([\hat{u}_l^{c_i}, \hat{v}_l^{c_i}, \lambda_l^{c_i}])$. Thus, the hybrid vision factor consists of two components, the first of which is the reprojection constraint factor

$$\boldsymbol{r}_{\mathcal{H}_1}(\mathcal{X}) = \sum_{l=1}^{m}\left(\hat{\mathcal{Q}}_l^{c_j} - \frac{f(\hat{\mathcal{A}}_l^{c_i})}{\|f(\hat{\mathcal{A}}_l^{c_i})\|}\right) + \sum_{l=1}^{n}\left(\hat{\mathcal{Q}}_l^{c_j} - \frac{f(\hat{\mathcal{P}}_l^{c_i})}{\|f(\hat{\mathcal{P}}_l^{c_i})\|}\right) \tag{12}$$

where

$$\begin{cases} \hat{\mathcal{Q}}_l^{c_j} = {\pi_e}^{-1}\left(\left[\hat{u}_l^{c_j}, \hat{v}_l^{c_j}\right]^T\right) \\ f(\hat{\mathcal{P}}_l^{c_i}) = \boldsymbol{R}_b^c(\boldsymbol{R}_w^{b_j}(\boldsymbol{R}_{b_i}^w(\boldsymbol{R}_c^b {\lambda_l^{c_i}}^{-1}\pi_c^{-1}([\hat{u}_l^{c_i}, \hat{v}_l^{c_i}]^T) \\ \qquad\qquad +\boldsymbol{p}_c^b) + \boldsymbol{p}_{b_i}^w - \boldsymbol{p}_{b_j}^w) - \boldsymbol{p}_c^b) \\ f(\hat{\mathcal{A}}_l^{c_i}) = \boldsymbol{R}_b^c(\boldsymbol{R}_w^{b_j}(\boldsymbol{R}_{b_i}^w(\boldsymbol{R}_c^b \cdot \hat{\mathcal{A}}_l^{c_i} + \boldsymbol{p}_c^b) + \boldsymbol{p}_{b_i}^w - \boldsymbol{p}_{b_j}^w) - \boldsymbol{p}_c^b). \end{cases} \tag{13}$$

Among them, ${\pi_e}^{-1}(\cdot)$ is the back-projection function. In addition, the second part of the hybrid vision factor is

$$\boldsymbol{r}_{\mathcal{H}_2}(\mathcal{X}) = \sum_{l=1}^{m}(g(\hat{\mathcal{A}}_l^{c_i}) - \lambda_l^{c_i}) \tag{14}$$

where $g(\cdot)$ denotes the calculation of the inverse depth.

2) Inertia Factor: The preintegrated inertia factor for two consecutive frames of IMU measurements b_i and b_j is

$$\boldsymbol{r}_{\mathcal{I}}(\mathcal{X}) = \left[\delta\boldsymbol{\alpha}_{b_{i+1}}^{b_i}, \delta\boldsymbol{\beta}_{b_{i+1}}^{b_i}, \delta\boldsymbol{\theta}_{b_{i+1}}^{b_i}, \delta\mathbf{b}_a, \delta\mathbf{b}_g\right] \tag{15}$$

where $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\theta}$ are preintegrated variables.

3) Loop Factor: Suppose that the current keyframe k_j is identified as being looped back from keyframe k_i by the loop closure detection algorithm (see Section III-D), and the loop transformation matrix is $\hat{\boldsymbol{T}}_{k_j}^{k_i}$.

Similarly, assuming that the odometry attitude matrices corresponding to frames k_i and k_j are $\boldsymbol{T}_{k_i}^w$ and $\boldsymbol{T}_{k_j}^w$, respectively, the loop factor is given by

$$\boldsymbol{r}_{\mathcal{L}}(\mathcal{X}) = \log(\hat{\boldsymbol{T}}_{k_j}^{k_i} \cdot (\boldsymbol{T}_{k_j}^w)^{-1} \cdot \boldsymbol{T}_{k_i}^w)^{\vee} \tag{16}$$

where $\log(\cdot)^{\vee}$ denotes the map from the 3-D rotation group SO(3) to the Lie-Algebra $\mathfrak{so}(3)$.

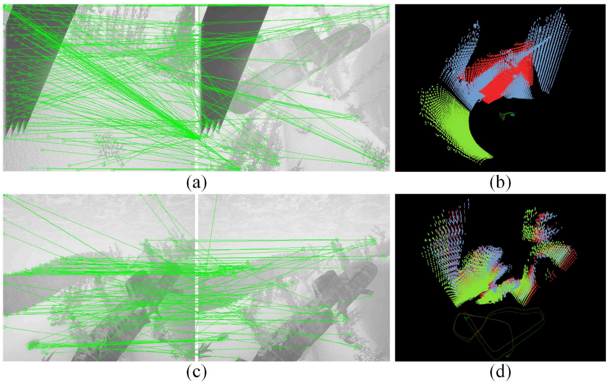

Fig. 5. Hybrid vision-aided loop closure detection case. (a) and (b) Misdetection occurs in passive vision, while active vision involves screening. (c) and (d) Passive vision experiences mismatches, while active vision enhances accuracy.

D. Hybrid Vision-Aided Loop Closure Detection

In underwater environments, due to the generally poor imaging quality, loop closure detection methods based solely on pure passive vision often encounter false detection, as illustrated in Fig. 5(a). Furthermore, due to mismatching issues, as depicted in Fig. 5(c), loop factors calculated solely from passive vision are prone to inaccuracies. Considering the point cloud matching obtained by active vision will fail when passive vision misdetects, as shown in Fig. 5(b). And when successful loop closure detection occurs, registration based on the point cloud obtained from active vision can improve the map consistency, as shown in Fig. 5(d). Therefore, this article introduces active vision and proposes a more robust hybrid vision-aided loop closure detection algorithm.

The proposed algorithm comprises two main stages. The first stage is the passive vision loop detection part, where the common DBoW2 is utilized. Differently, after the successful detection of passive visual loop closure, the active visual loop closure detection is conducted in second stage to prevent misjudgments and enhance the accuracy of loop factor. Based on the timestamps of the matching passive vision keyframes K_{pc} and K_{ph}, the active vision keyframes K_{ac} and K_{ah} with the closest time are indexed. Then, K_{ac} and K_{ah} are expanded by including all the keyframes within a window, forming the local maps K'_{ac} and K'_{ah}. Then, the point cloud matching algorithm generalized iterative closest point (GICP) [22] is utilized to calculate the transformation matrix between K'_{ac} and K'_{ah}, in which the initial value uses the result of the first stage. Then active visual loop closure determination is executed. The overall detection process see Algorithm 1.

IV. Experiments

A. Self-Collected Datasets

Given the limited studies on underwater hybrid vision, the absence of relevant public datasets poses challenges for experimental validation. To comprehensively validate the effectiveness of Hybrid-VINS, we develop two sets of underwater autonomous hybrid vision systems for data acquisition: one in

Algorithm 1: Hybrid Vision-Aided Loop Closure Detection

```
Input: $K_{pc}$: current passive vision keyframe; $K_{plist}$: passive vision keyframe list; $K_{alist}$: active vision keyframe list;
Output: loop closure detection flag $flag$ or loop factor $\boldsymbol{R}, \boldsymbol{t}$
Loading the bag of words library.
/* Step 1: Passive Vision Loop Closure Detection */
(Default) $K_{plist}$ stored as bag-of-words model $bow_p$.
$score_p$,$K_{ph} \longleftarrow$ Query $K_{pc}$ in $bow_p$.
if $score_p < \varepsilon_p$ (minimum passive vision score) then
    return $flag \longleftarrow$ False
end if
$\boldsymbol{R}_p, \boldsymbol{t}_p \longleftarrow$ PnP&RANSAC($K_{pc}$,$K_{ph}$)
/* Step 2: Active Vision Loop Closure Detection */
$K_{ac}$,$K_{ah} \longleftarrow$ Active keyframes by $K_{pc}$,$K_{ph}$,$K_{alist}$.
$K'_{ac}$,$K'_{ah} \longleftarrow$ Expanding $K_{ac}$, $K_{ah}$ into local maps.
$score_a$, $\boldsymbol{R}_a, \boldsymbol{t}_a \longleftarrow$ GICP($K'_{ac}$,$K'_{ah}$) with $\boldsymbol{R}_p, \boldsymbol{t}_p$.
if $score_a < \varepsilon_a$ (minimum active vision score) then
    return $flag \longleftarrow$ False
end if
return $\boldsymbol{R}, \boldsymbol{t} \longleftarrow \boldsymbol{R}_a, \boldsymbol{t}_a$
```

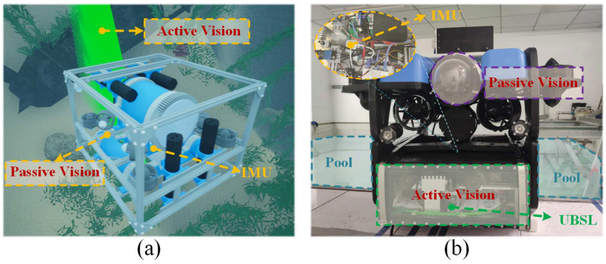

Fig. 6. Underwater autonomous hybrid vision systems. (a) Simulation prototype. (b) Physical prototype.

TABLE I
DESCRIPTION OF SELF-COLLECTED UNDERWATER HYBRID VISUAL INERTIAL DATASETS

Datasets	Path Length (m)	Time (s)	Max Speed (m/s)	Min Speed (m/s)	Ave Speed (m/s)
MarineRuin#1	96.2	275	0.504	0.267	0.378
MarineRuin#2	142.6	390	0.507	0.211	0.386
PlaneRuin	35.5	132	0.420	0.208	0.317
PierHarbor	53.2	245	0.357	0.198	0.274
RealPool#1	4.461	100	1.237	0.005	0.042
RealPool#2	5.243	60	1.836	0.005	0.060
RealPool#3	1.324	25	0.234	0.007	0.058

the simulation environment HoloOcean [28], [29], and the other in the real physical world, as illustrated in Fig. 6. Seven datasets are collected under different simulated and real scenarios, as outlined in Table I.

1) Simulation Datasets: HoloOcean is a state-of-the-art underwater multiscene and multisensor emulator, offering a variety of underwater sensors such as IMU, camera, and more. Therefore, we first build the autonomous underwater hybrid vision system in HoloOcean as shown in Fig. 6(a). Different datasets are then collected, including submarine ruins, airplane ruins, and pierharbor underwater, as shown in Fig. 7(a)–7(c). Among them, the parameter settings of all the sensors used are shown in Table II.

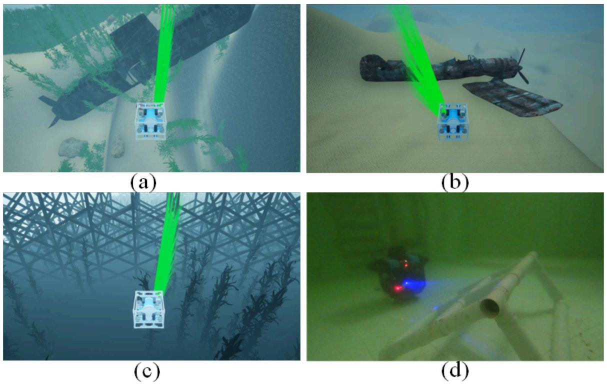

Fig. 7. Experimental scenes. (a) Submarine ruin. (b) Airplane ruin. (c) PierHarbor. (d) Real-world pool.

TABLE II
SENSORS SIMULATION PARAMETERS

Sensor	Parameters	Value
Camera	Frequency	20 Hz
Structured light	Frequency	2 Hz
IMU	Frequency	200 Hz
	Gyroscope noise	0.00123
	Accelerometer noise	0.00277
	Gyroscope walk noise	0.00388
	Accelerometer walk noise	0.00141

2) Real-World Datasets: In order to verify the capability of Hybrid-VINS in real underwater scenarios, we develop a real physical prototype as shown in Fig. 6. Then, the dataset is collected in a $4 \times 5 \times 1.5$ m^3 experimental pool with a few artificial features as shown in Fig. 7(d). A global camera mounted above the pool is used to record the robot's real trajectory [30].

B. Ablation Experiments

In Hybrid-VINS, the high-precision point cloud from active vision is first utilized to refine the 3-D map points corresponding to certain feature during passive vision initialization and tracking, thus enhancing the localization accuracy. The process is demonstrated in Fig. 4. Additionally, a hybrid vision-aided loop closure detection algorithm is proposed, significantly bolstering the robustness of pure passive vision VINS as shown in Fig. 5. To validate the effectiveness of each module, four sets of ablation experiments are deployed on the *MarineRuin#1* dataset.

1) Hybrid-VINS-PIO: Only passive visual odometry.
2) Hybrid-VINS-HIO: Only hybrid visual odometry.
3) Hybrid-VINS-HIO-PLC: Add passive visual loop closure detection module to Hybrid-VINS-HIO.
4) Hybrid-VINS-HIO-HLC: Add hybrid vision-aided loop closure detection module to Hybrid-VINS-HIO.

The results of the four sets of experimental trajectories are depicted in Fig. 8(a). It is evident that relying solely on passive

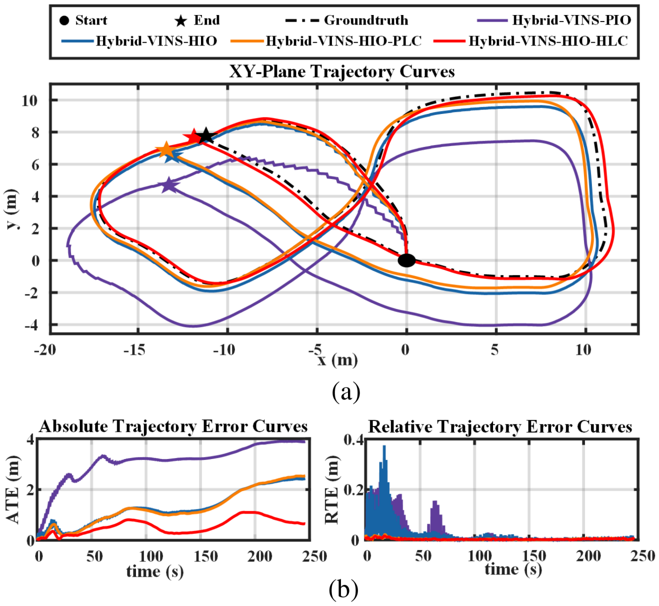

Fig. 8. Ablation experimental results in simulation dataset *MarineRuin#1*. (a) XY-plane trajectory comparison curves. (b) ATE and RTE comparison curves.

TABLE III
Ablation Experiments Error Results

Odometry		Loop Closure		ATE (m)		RTE (m)	
Passive	Active	Passive	Active	RMSE	MAE	RMSE	MAE
✓	✗	✗	✗	3.1900	3.1044	0.0299	0.0127
✓	✓	✗	✗	1.4599	1.2954	0.0268	0.0099
✓	✓	✓	✗	1.4574	1.2748	0.0049	0.0037
✓	✓	✓	✓	0.6284	0.5559	0.0042	0.0033

Note: Red text indicates the best result. Blue text indicates the 2nd best result. ✓ indicates the presence of the module, and ✗ indicates its absence.

vision leads to a severe shift. Moreover, the loop closure detection module, which depends solely on passive vision, exhibits greater deviation in certain regions due to the lower precision of the loop closure factor. However, the integration of active vision proves beneficial for both the tracking and loop closure detection modules. Overall, the trajectory stability and accuracy are greatly enhanced. To quantitatively compare the localization results, the root mean square error (RMSE) and mean absolute error (MAE) of both absolute trajectory error (ATE) and relative trajectory error (RTE) are compared separately [31]. The error curves of each group are shown in Fig. 8(b). Subsequently, data statistics about the errors are presented in Table III. The fourth group, Hybrid-VINS-HIO-HLC, exhibits the smallest error, highlighting the importance of integrating active vision into the passive vision odometry and loop closure detection module.

C. Trajectory Evaluation Experiments

To verify the superior performance of Hybrid-VINS in underwater scenes, several common visual SLAM algorithms including ORB-SLAM3 [14], VINS-Mono [25], and underwater state-of-the-art algorithm SVIn2 [17] are employed for comparison. It is worth noting that SVIn2 only considers the vision and inertia part, denoted as SVIn2-VIO. Hybrid-VIO, as the odometry portion of Hybrid-VINS, is also compared. Subsequently, the aforementioned methods are evaluated on both the collected simulation dataset and the real dataset to assess their performance.

1) Simulation Analysis: Utilizing the previously collected datasets, comparative experiments are conducted on *MarineRuin#2*, *PlaneRuin*, and *PierHarbor*. The trajectory curves of the different methods are shown in Fig. 9. Notably, SVIn2-VIO suffers from significant drift, which may stem from inaccurate estimation of IMU parameters during its initialization process. VINS-Mono performs poorly on certain datasets such as *MarineRuin#2*, while performing well on others. This disparity is primarily attributed to its purely vision-based loop closure detection module, which may produce false detections. ORB-SLAM3 demonstrates greater stability with smaller ATE. However, its RTE curve exhibits significant variation and larger errors. In contrast, our Hybrid-VIO achieves smaller ATE and RTE and demonstrates robust performance across all datasets. By introducing the hybrid vision-aided loop closure detection algorithm, Hybrid-VINS further reduces ATE and RTE. The quantitative error results, including RMSE and MAE, are presented in Table IV. Taken together, Hybrid-VINS achieves the best performance in ATE and RTE.

2) Real-World Analysis: Compared to the simulation datasets, the motion of the underwater robot in real-world scenarios is more complex, with obvious jamming phenomena. Localization comparison experiments in real world are conducted in the *RealPool#1* and *RealPool#2* to evaluate the performance of Hybrid-VINS under such conditions.

In these datasets, effective feature point extraction is challenging due to low light and sparse texture in the scene. Both ORB-SLAM3 and SVIn2-VIO rely on scene feature points for initialization and tracking. This dependence results in the ineffectiveness of these methods. Specifically, ORB-SLAM3 fails to run because it never has enough feature points to initialize successfully. Although SVIn2-VIO can still initialize without a sufficient number of feature points, the initialization results are not accurate enough, causing the system to drift during operation and leading to localization failure. All trajectories are displayed in Figs. 10 and 11. Among them, Figs. 10(a)–10(f) and 11(a)–11(f) illustrate the process of the UBSL assisting passive visual features in the real underwater world. The trajectory curves of all methods are shown in Figs. 10(g) and 11(g). Moreover, the quantitative error analysis results are reported in Tables V and VI. Differences in the curve scales of VINS-Mono may be attributed to the influence of underwater aberrations on feature triangulation. Although not as pronounced in Hybrid-VIO, a notable cumulative error persists. By incorporating the proposed loop closure detection algorithm, Hybrid-VINS demonstrates remarkably low ATE and RTE, which are uncommon underwater.

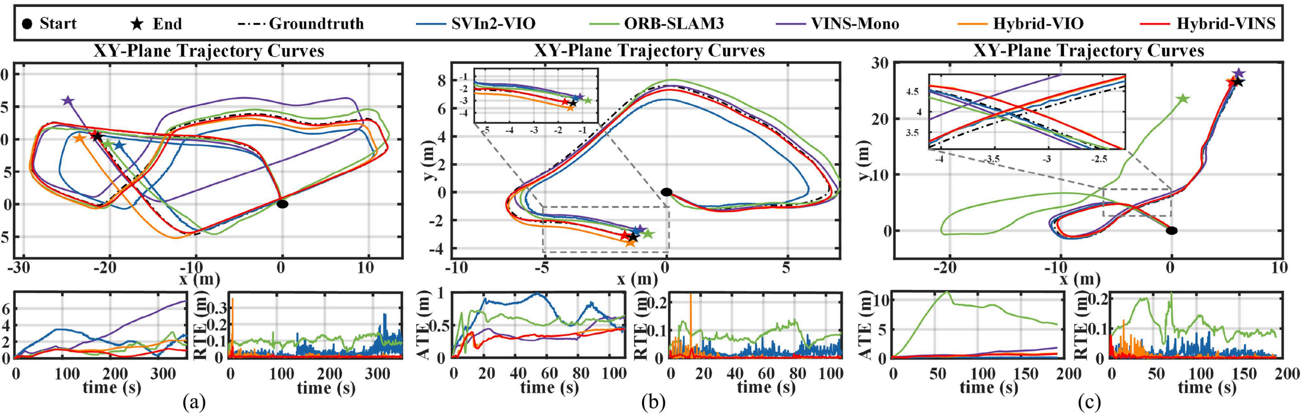

Fig. 9. Trajectory evaluation experiments in simulation datasets. (a) *MarineRuin#2*. (b) *PlaneRuin*. (c) *PierHarbor*.

TABLE IV
TRAJECTORY EVALUATION ON SIMULATION DATASETS

	MarineRuin#2				PlaneRuin				PierHarbor			
	ATE (m)		RTE (m)		ATE (m)		RTE (m)		ATE (m)		RTE (m)	
Method	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
ORB-SLAM3 [14]	1.7932	1.6787	0.0429	0.0294	0.7113	0.6829	0.0235	0.0191	0.1836	0.1804	0.0209	0.0164
SVIn2-VIO [17]	1.2706	1.0792	0.1002	0.0977	0.5720	0.5611	0.0840	0.0811	7.6834	7.2181	0.1146	0.1088
VINS-Mono [25]	3.7284	3.0709	0.0055	0.0046	0.3904	0.3669	0.0033	0.0025	1.6119	1.5296	0.0100	0.0054
Hybrid-VIO	1.3878	1.2398	0.0048	0.0034	0.3454	0.3324	0.0122	0.0034	0.5766	0.5326	0.0125	0.0066
Hybrid-VINS	0.7702	0.6808	0.0025	0.0022	0.3329	0.3208	0.0055	0.0035	0.5335	0.5100	0.0074	0.0041

Note: Red background indicates the optical flow-based method. Blue background indicates the feature point-based method. Red text indicates the best result. Blue text indicates the 2nd best result.

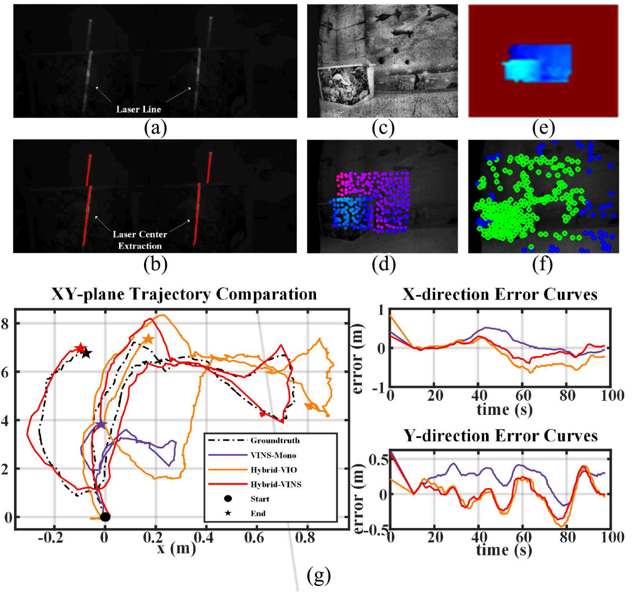

Fig. 10. Trajectory evaluation experiments in real-world *RealPool#1*. (a) UBSL image. (b) Laser centerline extraction. (c) Monocular camera image. (d) Local map projection. (e) Rasterized map. (f) Hybrid visual feature image. (g) Trajectory.

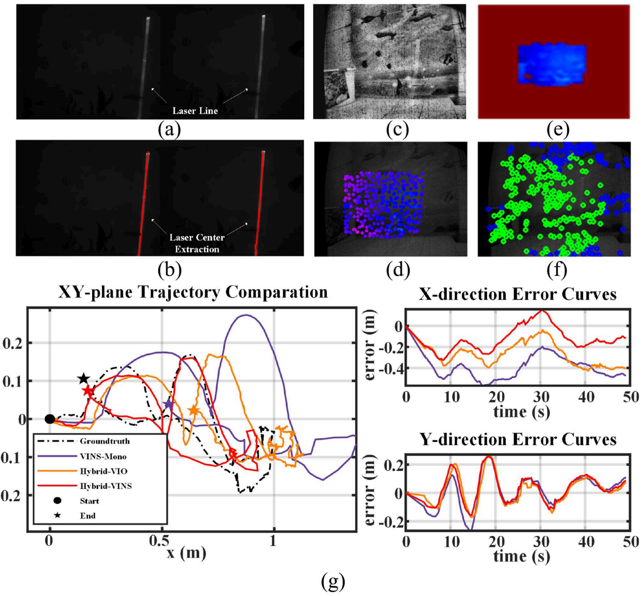

Fig. 11. Trajectory evaluation experiments in real-world *RealPool#2*. (a) UBSL image. (b) Laser centerline extraction. (c) Monocular camera image. (d) Local map projection. (e) Rasterized map. (f) Hybrid visual feature image. (g) Trajectory.

TABLE V
Trajectory Evaluation on Real-World *RealPool#1*

	X-Axis (m)		Y-Axis (m)	
Method	RMSE	MAE	RMSE	MAE
ORB-SLAM3 [14]	Fail	Fail	Fail	Fail
SVIn2-VIO [17]	Fail	Fail	Fail	Fail
VINS-Mono [25]	0.24792	0.18432	0.2555	0.23678
Hybrid-VIO	0.31151	0.25655	0.18396	0.14869
Hybrid-VINS	0.12851	0.079368	0.097053	0.072372

Note: Red text indicates the best result. Blue text indicates the 2nd best result.

TABLE VI
Trajectory Evaluation on Real-World *RealPool#2*

	X-Axis (m)		Y-Axis (m)	
Method	RMSE	MAE	RMSE	MAE
ORB-SLAM3 [14]	Fail	Fail	Fail	Fail
SVIn2-VIO [17]	Fail	Fail	Fail	Fail
VINS-Mono [25]	0.25329	0.20187	0.13352	0.087633
Hybrid-VIO	0.28208	0.22492	0.052652	0.04316
Hybrid-VINS	0.12938	0.095252	0.10227	0.074314

Note: Red text indicates the best result. Blue text indicates the 2nd best result.

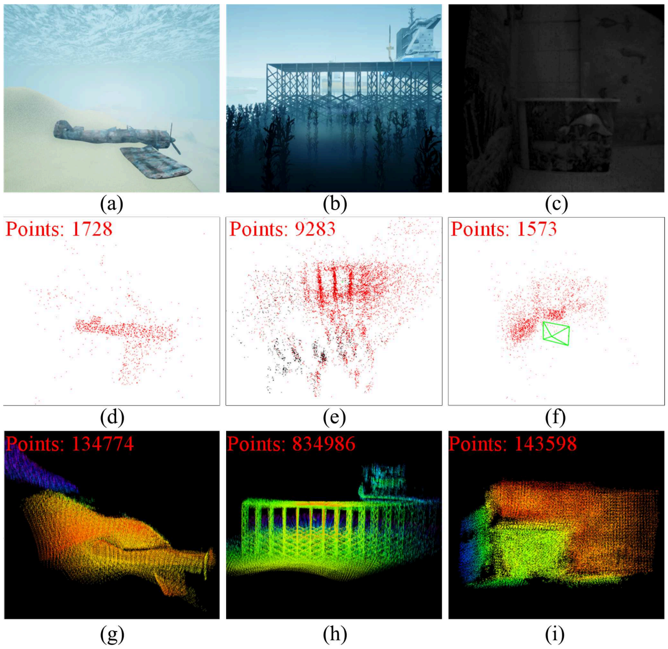

Fig. 12. Mapping experimental results. (a) *PlaneRuin* scene. (b) *PierHarbor* scene. (c) *RealPool#3* scene. (d)–(f) Maps by ORB-SLAM3 [14]. (g)–(i) Maps by our Hybrid-VINS. Note: The red numbers in the upper left corner represent the total count of 3-D points in the point cloud.

D. Dense Mapping Experiments

The acquisition of dense point cloud is crucial for in-situ underwater detailed observation. While existing SLAM systems can generally meet localization requirements, acquiring dense maps has been a significant challenge in underwater environments. Hybrid-VINS, by integrating the point cloud information provided by the self-designed active vision device UBSL, greatly improves the localization accuracy and loop closure detection robustness. In addition, by combining the dense point cloud information it obtained, the acquisition of underwater real-time dense maps is realized, which is highly valuable for underwater SLAM systems. As depicted in Fig. 12(a)–12(c), several common underwater scenes are mapped, including underwater aircraft remains, underwater artificial structures under a pierharbor, and real underwater walls. To illustrate the superiority of Hybrid-VINS in dense mapping, a comparison is made with ORB-SLAM3 [14], which is one of the few SLAM systems currently available underwater that can obtain point cloud maps. Despite setting the feature points per image frame to a high extent (reaching 8000) to acquire the densest possible point cloud map, ORB-SLAM3 still produces sparse point clouds where the shape of objects is hardly discernible, as shown in Fig. 12(d)–12(f). In contrast, Fig. 12(g)–12(i) display the dense mapping results of Hybrid-VINS, where the point cloud is several orders of magnitude denser than ORB-SLAM3. The main features of each structure are clearly observable, making it suitable for practical engineering applications underwater.

V. Conclusion

In this article, an underwater tightly coupled hybrid visual inertial dense SLAM framework, named Hybrid-VINS, is proposed to achieve more robust underwater autonomous robot localization and dense mapping. To the best of our knowledge, this is the first underwater SLAM system that integrates a structured light system to assist passive vision. Especially, some conclusion are drawn as follows.

1) An underwater hybrid vision fusion method is presented to realize depth correction of passive visual features from measurements of structured light devices.
2) A more robust hybrid vision-aided loop closure detection algorithm is proposed to overcome the inaccuracy of purely passive vision loop factor.
3) The underwater autonomous hybrid vision system is developed in an underwater simulation environment and the real underwater world, respectively, to collect multiple datasets to validate the performance of Hybrid-VINS.
4) The experimental results show that Hybrid-VINS can achieve more robust and accurate autonomous localization of underwater autonomous robots with minimal ATE and RTE compared to existing underwater SLAM systems. Meanwhile, underwater dense mapping can be realized. It has excellent prospects for underwater engineering applications.

In the future, more sensors such as sonar and DVL will be fused in order to further improve the initialization speed and robustness of Hybrid-VINS. In addition, it will be integrated with obstacle avoidance, motion planning, and other technologies for underwater vehicles to accomplish more complex underwater operations, including exploration of unknown environments, underwater archaeology, and marine resource research.

References

[1] F. Yu and Y. Chen, “Cyl-IRRT*: Homotopy optimal 3D path planning for AUVs by biasing the sampling into a cylindrical informed subset,” *IEEE Trans. Ind. Electron.*, vol. 70, no. 4, pp. 3985–3994, Apr. 2023.

[2] J. Xing, W. Jin, K. Yang, and I. Howard, "A bionic piezoelectric robotic jellyfish with a large deformation flexure hinge," *IEEE Trans. Ind. Electron.*, vol. 70, no. 12, pp. 12596–12605, Dec. 2023.

[3] C. Zhu, L. Deng, X. Wang, Z. Yin, and C. Zhou, "Design and modeling of elastic variable stiffness robotic fish tail," in *Proc. IEEE Int. Conf. Mechatron. Automat. (ICMA)*, Piscataway, NJ, USA: IEEE Press, 2022, pp. 1251–1256.

[4] Y. Ou et al., "Structured light-based underwater collision-free navigation and dense mapping system for refined exploration in unknown dark environments," *IEEE Trans. Syst., Man, Cybern.: Syst.*, early access, Mar. 18, 2024, doi: 10.1109/TSMC.2024.3370917.

[5] Y. Huang, P. Li, S. Yan, M. Tan, J. Yu, and Z. Wu, "Self-localization of a biomimetic robotic shark using tightly coupled visual-acoustic fusion," *IEEE Trans. Ind. Electron.*, vol. 71, no. 10, pp. 12581–12591, Oct. 2024.

[6] D.-D. Zhao, W.-B. Mao, P. Chen, Y.-J. Dang, and R.-H. Liang, "FPGA-based real-time synchronous parallel system for underwater acoustic positioning and navigation," *IEEE Trans. Ind. Electron.*, vol. 71, no. 3, pp. 3199–3207, Mar. 2024.

[7] Z. Bi, J. Xu, and H. Fang, "Design and path planning for a worm-snake-inspired metameric (WSIM) robot," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Piscataway, NJ, USA: IEEE Press, 2022, pp. 777–782.

[8] Z. Zheng, S. Lin, and C. Yang, "RLD-SLAM: A robust lightweight VI-SLAM for dynamic environments leveraging semantics and motion information," *IEEE Trans. Ind. Electron.*, early access, Feb. 26, 2024, doi: 10.1109/TIE.2024.3363744.

[9] B. Joshi, H. Damron, S. Rahman, and I. Rekleitis, "SM/VIO: Robust underwater state estimation switching between model-based and visual inertial odometry," 2023, *arXiv:2304.01988*.

[10] B. Joshi, C. Bandara, I. Poulakakis, H. G. Tanner, and I. Rekleitis, "Hybrid visual inertial odometry for robust underwater estimation," in *Proc. OCEANS-MTS/IEEE US Gulf Coast*, Piscataway, NJ, USA: IEEE Press, 2023, pp. 1–7.

[11] J. Song, W. Li, and X. Zhu, "Acoustic-VINS: Tightly coupled acoustic-visual-inertial navigation system for autonomous underwater vehicles," *IEEE Robot. Automat. Lett.*, vol. 9, no. 2, pp. 1620–1627, Feb. 2024.

[12] E. Vargas et al., "Robust underwater visual SLAM fusing acoustic sensing," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Piscataway, NJ, USA: IEEE Press, 2021, pp. 2140–2146.

[13] S. Xu et al., "Underwater visual acoustic slam with extrinsic calibration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Piscataway, NJ, USA: IEEE Press, 2021, pp. 7647–7652.

[14] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1874–1890, Dec. 2021.

[15] Y. Huang et al., "Tightly-coupled visual-DVL fusion for accurate localization of underwater robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Piscataway, NJ, USA: IEEE Press, 2023, pp. 8090–8095.

[16] Z. Xu, M. Haroutunian, A. J. Murphy, J. Neasham, and R. Norman, "An integrated visual odometry system for underwater vehicles," *IEEE J. Ocean. Eng.*, vol. 46, no. 3, pp. 848–863, Jul. 2021.

[17] S. Rahman, A. Quattrini Li, and I. Rekleitis, "SVIn2: A multi-sensor fusion-based underwater slam system," *Int. J. Robot. Res.*, vol. 41, nos. 11–12, pp. 1022–1042, 2022.

[18] L. Zhao, M. Zhou, and B. Loose, "Tightly-coupled visual-DVL-inertial odometry for robot-based ice-water boundary exploration," 2023, *arXiv:2303.17005*.

[19] A. Cardaillac and M. Ludvigsen, "Camera-sonar combination for improved underwater localization and mapping," *IEEE Access*, vol. 11, pp. 123070–123079, 2023.

[20] H. Zhao, R. Zheng, M. Liu, and S. Zhang, "Detecting loop closure using enhanced image for underwater VINS-Mono," in *Proc. Global Oceans: Singapore*–US Gulf Coast, Piscataway, NJ, USA: IEEE Press, 2020, pp. 1–6.

[21] Y. Ou, J. Fan, C. Zhou, S. Tian, L. Cheng, and M. Tan, "Binocular structured light 3-D reconstruction system for low-light underwater environments: Design, modeling, and laser-based calibration," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–14, 2023.

[22] Y. Ou, J. Fan, C. Zhou, L. Cheng, and M. Tan, "Water-MBSL: Underwater movable binocular structured light-based high-precision dense reconstruction framework," *IEEE Trans. Ind. Inform.*, vol. 20, no. 4, pp. 6142–6154, Apr. 2024.

[23] J. Fan, Y. Ou, X. Li, C. Zhou, and Z. Hou, "Structured light vision based pipeline tracking and 3D reconstruction method for underwater vehicle," *IEEE Trans. Intell. Veh.*, vol. 9, no. 2, pp. 3372–3383, Feb. 2024.

[24] J. Zhang, M. Kaess, and S. Singh, "Real-time depth enhanced monocular odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Piscataway, NJ, USA: IEEE Press, 2014, pp. 4973–4980.

[25] 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.

[26] T. Shan, B. Englot, C. Ratti, and D. Rus, "LVI-SAM: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Piscataway, NJ, USA: IEEE Press, 2021, pp. 5692–5698.

[27] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual–inertial odometry," *IEEE Trans. Robot.*, vol. 33, no. 1, pp. 1–21, Feb. 2017.

[28] E. Potokar, S. Ashford, M. Kaess, and J. G. Mangelson, "HoloOcean: An underwater robotics simulator," in *Proc. Int. Conf. Robot. Automat. (ICRA)*, Piscataway, NJ, USA: IEEE Press, 2022, pp. 3040–3046.

[29] E. Potokar et al., "HoloOcean: Realistic sonar simulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Piscataway, NJ, USA: IEEE Press, 2022, pp. 8450–8456.

[30] B. Lu, J. Wang, Q. Zou, J. Fan, and C. Zhou, "Towards power cost analysis and optimization of a multi-flexible robotic fish," *Ocean Eng.*, vol. 294, 2024, Art. no. 116746.

[31] M. Grupp, "evo: Python package for the evaluation of odometry and slam." GitHub. Accessed: 2017. [Online]. Available: https://github.com/MichaelGrupp/evo

Yaming Ou received the B.E. degree in automation from the Southeast University, Nanjing, China, in 2021. He is currently working toward the Ph.D. degree in control theory and control engineering with the Institute of Automation, Chinese Academy of Sciences (IACAS), Beijing, China.

His research interests include underwater 3-D vision, simultaneous localization and mapping, multisensor fusion, and autonomous robot navigation.

Junfeng Fan (Member, IEEE) received the B.S. degree in mechanical engineering and automation from Beijing Institute of Technology, Beijing, China, in 2014, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences (IACAS), Beijing, in 2019.

He is currently an Associate Professor in control theory and control engineering with the Laboratory of Cognition and Decision Intelligence for Complex Systems, IACAS. His research interests include robot vision and underwater robot.

Chao Zhou received the B.S. degree in automation from Southeast University, Nanjing, China, in 2003, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences (IACAS), Beijing, China, in 2008.

From 2013 to 2014, he was a Visiting Scholar with the University of Toronto, Toronto, ON, Canada. He is currently a Professor in control theory and control engineering with the Laboratory of Cognition and Decision Intelligence for Complex Systems, IACAS. His research interests include underwater robot, bionic robot, and underwater intelligent sensing.

Pengju Zhang received the B.S. degree in automation from China University of Petroleum, Shandong, China, in 2015, the M.S. degree in instrument science and technology from Beihang University, Beijing, China, in 2018, and the Ph.D. degree in computer application technology from the University of Chinese Academy of Sciences, Beijing, in 2022.

He is currently an Assistant Professor with the State Key Laboratory of Multimodal Artificial Intelligence, Institute of Automation, Chinese Academy of Sciences, Beijing. His research interests include 3-D computer vision and deep learning.

Zeng-Guang Hou (Fellow, IEEE) received the B.E. and M.E. degrees from Yanshan University (formerly North-East Heavy Machinery Institute), Qinhuangdao, China, in 1991 and 1993, respectively, and the Ph.D. degree from the Beijing Institute of Technology, Beijing, China, in 1997, all in electrical engineering.

He is a Professor with the State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing. His research interests include neural networks, robotics, and intelligent systems.

Prof. Hou was an Associate Editor of IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE and IEEE TRANSACTIONS ON NEURAL NETWORKS. He is an Associate Editor of IEEE TRANSACTIONS ON CYBERNETICS and *ACTA Automatica Sinica* and an Editorial Board Member of Neural Networks.