

Tightly-coupled Visual-DVL-Inertial Odometry for Robot-based Ice-water Boundary Exploration

Lin Zhao[†], Mingxi Zhou[‡] and Brice Loose[‡]

Abstract—Underwater robots, like Autonomous Underwater Vehicles (AUVs) and Remotely Operated Vehicles (ROVs), are promising tools for the exploration and study of the under-ice environment and the ecosystems that thrive there. However, state estimation is a well-known problem for robotic systems, especially, for the ones that travel underwater. In this paper, we present a tightly-coupled multi-sensors fusion framework to increase localization accuracy that is robust to sensor failure. Visual images, Doppler Velocity Log (DVL), Inertial Measurement Unit (IMU) and Pressure sensor are integrated using a Multi-State Constraint Kalman Filter (MSCKF) for state estimation. Besides, a modified keyframe-based clone marginalization and a new DVL-aided feature enhancement method are presented to further improve the localization performance. The proposed method is validated in the under-ice environment on Lake Michigan, USA, and the results are cross-compared with 10 other different sensor fusion setups. Overall, the integration of keyframe enabled and DVL-aided feature enhancement yielded the best performance with a Root-mean-square error of less than 2 m compared to the ground truth path over a total traveling distance of about 200 m.

I. INTRODUCTION

The ocean in polar regions plays a vital role, affecting the global biogeochemical cycle [1, 2] and the ice-water interface is an active region where transports between ice and water can set the biogeochemical and determine growth in the zones where sunlight penetrates. Despite limited resources, the under ice environment can experience large plankton blooms [3], but very little is understood about how they take place. Previously, underwater vehicles have been successfully deployed to collect physical and biogeochemical measurements under sea ice [4, 5]. Recently, under-ice AUV deployments [6]–[8] have been carried out in regions that were previously inaccessible to observation, and to produce a higher spatial coverage to fill the observation gaps left by traditional ice-anchored instruments and ice coring. AUVs and ROVs are uniquely capable of observing these environments in a manner that creates almost no disturbance, while traditional ice observations involve perforating the ice, which upsets the delicate physical structure, thereby biasing the resulting measurements. However, the localization (i.e., state estimation) is particularly challenging in underwater

environments due to the lack of GPS [9]. Under-ice AUV operation is an extreme case that AUVs could not return to the surface to obtain the GPS fixes for bounding the localization drift [10]. Therefore, the collected measurements are hard to georeference, challenging the creation of spatially coherent maps of the processes, e.g., the algae patchiness, and sea-ice surface topography.

Acoustic transducer arrays, such as Long Baseline (LBL) [11] and Ultra-short Baseline (USBL) [12], are commonly used for under-ice navigation. But, they will require installation and extrinsic calibration for transducers. In the under-ice environment, the performance of acoustic communication devices may degrade due to up-bending sound propagation and ice keel blockage [13]. To this end, self-contained underwater navigation methods are researched and often used as it requires fewer logistic operations. The most common one is the dead-reckoning navigation [14] which fuses velocity measured by a DVL and an inertial measurement sensor. However, this method suffers from unbounded position errors ranging from 8 to 22 m for different quality Inertial Navigation System (INS) [15].

For the ice-water boundary exploration, vehicles would maintain a close distance (1 to 2 m) in order to collect vital measurements (such as sea-ice roughness, light penetration and water density) to study the ice-water exchanges. The close distance makes it possible to use camera images to aid underwater localization for AUVs. In recent years, Visual-Odometry (VO)/Simultaneous Localization and Mapping (SLAM) has drawn increased attention, serving the rapid development in robot autonomy. To improve the robustness of state estimation, Visual-Inertial-Odometry (VIO) [16] has been widely used. However, underwater visual-based SLAM is still challenging because of the dynamic illumination, limited visibility, light obstruction, texture-less area and motion blur [17]. Also, Mono-VIO has a well-known issue that the metric scale is not observable if there is no acceleration excitation. In such case, additional sensors (e.g., stereo-camera [18]) are needed for reliable and accurate SLAM solutions in the underwater environment.

The algorithm development always has to consider the trade-off between accuracy and computational cost when deploying on the robotics platform. Non-linear optimization-based VIO (e.g., OKVIS [19]) allows for the reduction of error through relinearization but with a high computational cost. Filtering-based VIO (e.g., MSCKF [20]) is proven to be efficient and accurate in resources constraint applications, e.g, the state-of-the-art ARCore [21] running on mobile devices. To deal with the degraded performance that may exist

This work is supported by the National Science Foundation (NSF) under the award #1945924, and the Graduate School of Oceanography, University of Rhode Island. We also thank the field support from the Great Lake Research Center, Michigan Technological University

[†]The author is with the Department of Ocean Engineering, University of Rhode Island, Narragansett, RI 02882, USA. Email: linzhao@uri.edu

[‡]The authors are with the Graduate School of Oceanography, University of Rhode Island, Narragansett, RI 02882, USA. Emails: {mzhou, bloose}@uri.edu

when only using the one-time linearization, observability-based methodology [22] can be applied to improve the consistency. More details about the relevant works can be found in Section II.

In this paper, we present a multi-modal sensor fusion framework to address the two challenges in underwater environments: degenerate motion and challenging image conditions. Our method fuses the measurements from DVL, IMU, camera and pressure sensor in the MSCKF framework to improve the robustness of state estimation. Our main contributions¹ are:

- A modified keyframe marginalization method to increase the tracking period of features.
- A DVL point cloud enhanced feature position recovery with a new data association and estimation approach, presented in Section IV.
- Algorithm validation and comparison with different sensor enable/disable settings using a real-world under-ice ROV data set, presented in Section V

The remaining paper is organized as follows: the next section reviews the related work on multi-sensor fused state estimation with an emphasis on underwater environments. Section III introduces the filter implementation with multi-sensor setup and the implementation of the keyframe method. Section IV presents our DVL-aided feature enhancement approach. Section V presents the experiment results, and we will conclude the paper and discuss our future plans in Section VI.

II. RELATED WORK

When an underwater vehicle is operated close to a target (e.g., seafloor or sea-ice), the measurements from DVL and INS are commonly fused for dead-reckoning (DR). To increase localization accuracy, DR integrated with a pressure sensor in a tightly-coupled EKF for under-ice navigation was presented in [23]. In addition, acoustic beacon aided DR solutions using LBL [24] or USBL [25] have existed for years to bound the odometry drift. When other acoustic sensors, such as multibeam echosounder (MBES) [26] and multibeam forward-looking sonar (MFLS) [27], are available, geophysical observations made by the sonars could further improve localization robustness. Besides velocity measurements, DVL also generates sparse point clouds to provide additional measurements for localization. For example, in [28], the author presented the factor-graph SLAM using parameterized planar features from sparse point clouds.

Compared to acoustic sensors, visual data contain more information and could be leveraged for localization. Recently, visual-based SLAM [29, 30] has been significantly researched in the robotic community across various domains and applications. Even though there is significant SLAM research done in the marine robotics community, significant technological hurdles remain, such as poor image quality in a low light environment, featureless ice terrain, and limited onboard processing capability.

Offline methods, such as the Structure From Motion (SfM), have been applied to recover large-scale underwater 3D scenes and camera poses [31, 32]. Yet, these methods are computationally expensive and not realistic to run online on an AUV. On the other hand, VO is capable of processing images at high frame rates (e.g., 10-20Hz), and the uses of monocular camera and stereo-camera have been investigated for underwater scenarios [33]–[36]. As mentioned in the survey [17], visual measurements are usually combined with other sensors (e.g., IMU) for improved performance. For example, the pressure sensor can be used to aid the VIO algorithms using filter-based and optimization-based methods [37] and [38]. The SVIn2 [18] took another step forward, which fuses the measurements from stereo cameras, IMU, depth sensor and profiling sonar in a keyframe-based nonlinear optimization for underwater localization. Similar to our method presented in this paper, DVL velocity fused VIO are developed in [39]–[41] for hull inspection, harbor exploration and deep-sea operation. However, our work herein furthers the field by using the sparse point cloud from the DVL to enhance the feature estimation for vehicle pose updates.

Dense point clouds fusing with images have been widely used to enhance feature 3D position estimation in computer vision. In [42], the authors presented a LiDAR-enhanced SfM pipeline that fuses the dense LiDAR point clouds with the matched visual features in a joint optimization to solve camera motion and feature position. In [43, 44], LiDAR point cloud is used to interpolate the depth for the detected camera features. However, such dense point clouds are typically not available for underwater robotics, or a power-hungry multibeam sonar is needed. To our best knowledge, there is limited research on using sparse point clouds for Visual SLAM. For example, the method presented in [45, 46] used the sparse range measurements from a single-beam echosounder to recover the depth information for a monocular SLAM system. To further advance the visual SLAM using sparse point clouds, our method in this paper will employ non-uniform sparse point clouds from a DVL sensor to aid feature 3D position estimation.

III. FILTER DESCRIPTION

In this section, we will present the tightly-coupled multi-sensors fusion based on the state-of-art MSCKF [20] framework. For underwater robots, we further present the measurement update equations for the DVL and pressure sensor, followed by our keyframe selection strategy.

A. State Vector

We follow the notation in [29] and define the system state to be \mathbf{x}_k at the time step, k . As shown in Eq. 1 to 3, the system state consists of the current IMU state, \mathbf{x}_{IMU} and the clone state, \mathbf{x}_C , which contains n past IMU poses. In Eq. 2 and 3, ${}^I_k\bar{q}$ [47] is the unit quaternion representing the rotation from the global frame $\{G\}$ to the IMU frame $\{I_k\}$ at time k , ${}^G\mathbf{p}_{I_k}$ and ${}^G\mathbf{v}_{I_k}$ are the IMU position and velocity with respect to $\{G\}$, \mathbf{b}_g and \mathbf{b}_a describe the biases of the angular velocity and linear acceleration measured by the gyro and

¹code: https://github.com/GSO-soslab/msckf_dv10

accelerometer in an IMU. The cloned IMU poses are denoted by $\{^i_G \bar{q}$ and $^G \mathbf{p}_{I_i}\}, i \in [1, n]$.

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{x}_{IMU}^\top & \mathbf{x}_{Clone}^\top \end{bmatrix}^\top \quad (1)$$

$$\mathbf{x}_{IMU} = \begin{bmatrix} I_k \bar{q}^\top & G \mathbf{p}_{I_k}^\top & G \mathbf{v}_{I_k}^\top & \mathbf{b}_g^\top & \mathbf{b}_a^\top \end{bmatrix}^\top \quad (2)$$

$$\mathbf{x}_{Clone} = \begin{bmatrix} I_G \bar{q}^\top & G \mathbf{p}_{I_1}^\top & \dots & I_n \bar{q}^\top & G \mathbf{p}_{I_n}^\top \end{bmatrix}^\top \quad (3)$$

In this paper, we define $\mathbf{x} = \hat{\mathbf{x}} \boxplus \tilde{\mathbf{x}}$, where \mathbf{x} is the true state, $\hat{\mathbf{x}}$ is estimation, $\tilde{\mathbf{x}}$ is the error state, and the \boxplus operation maps the vector to a given manifold [48]. For quaternions, we define the quaternion boxplus operation using the left quaternion error in Eq. 4 where $\boldsymbol{\theta}$ is the Euler angles.

$$\bar{q} \boxplus \delta \boldsymbol{\theta} \triangleq \begin{bmatrix} 1 & \delta \boldsymbol{\theta} \\ 2 & \\ & 1 \end{bmatrix} \otimes \bar{q} \quad (4)$$

B. IMU Propagation

The state is propagated from $k-1$ to k time step using the generic nonlinear IMU kinematics model [49] with IMU measurements, including linear accelerations ($^I \mathbf{a}_m$) and angular velocities ($^I \boldsymbol{\omega}_m$).

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, ^I \mathbf{a}_m, ^I \boldsymbol{\omega}_m, \mathbf{n}_I) \quad (5)$$

where $\mathbf{n}_I = [\mathbf{n}_g^\top \quad \mathbf{n}_a^\top \quad \mathbf{n}_{\omega_g}^\top \quad \mathbf{n}_{\omega_a}^\top]^\top$, including the zero-mean Gaussian noise (\mathbf{n}_g and \mathbf{n}_a) and the random walk bias noise (\mathbf{n}_{ω_g} and \mathbf{n}_{ω_a}) for the gyroscope and accelerometer. The estimated state and propagated covariance are:

$$\hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}, ^I \mathbf{a}_m, ^I \boldsymbol{\omega}_m, \mathbf{0}) \quad (6)$$

$$\mathbf{P}_{k|k-1} = \Phi_{k|k-1} \mathbf{P}_{k-1|k-1} \Phi_{k|k-1}^\top + \mathbf{G}_{k-1} \mathbf{Q} \mathbf{G}_{k-1}^\top \quad (7)$$

where $\hat{\mathbf{x}}_{k|k-1}$ denotes the estimated state at time k given the measurements at time $k-1$, $\Phi_{k|k-1}$ and \mathbf{G}_{k-1} are system Jacobian and noise Jacobian of the nonlinear system [20], and \mathbf{Q} is a discrete-time covariance matrix of IMU noise \mathbf{n}_I .

C. DVL Velocity Measurement Update

The DVL velocity measurement is defined in Eq. 8 which is a function of the linear and angular velocity in the IMU frame, the relative transformation ($^I_D \mathbf{R}$ and $^I \mathbf{p}_D$) between the IMU frame and the DVL frame, and the rotation ($^I_G \mathbf{R}$) from the global frame to the IMU frame. We also have the measurement noise $\mathbf{n}_D \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_D)$, and the skew-symmetric matrix of the IMU's angular velocity denoted by $[^I \boldsymbol{\omega}]_\times$. For EKF update, the Jacobian matrix $\mathbf{H}_{D,k}$ with respect to the state, \mathbf{x}_k) can be found in [23].

$$\mathbf{z}_{D,k} = h_D(\mathbf{x}_k) + \mathbf{n}_D = ^I_D \mathbf{R}^\top ({}^I_G \mathbf{R}^G \mathbf{v}_{I_k} + [^I \boldsymbol{\omega}]_\times ^I \mathbf{p}_D) + \mathbf{n}_D \quad (8)$$

D. Pressure Measurement Update

The pressure measurement can be written in Eq. 9 where $\mathbf{s} = [0 \quad 0 \quad 1]^\top$ used for selecting the third dimension, ${}^P \mathbf{P}_{in} = [0 \quad 0 \quad {}^P p_{in}]^\top$ and ${}^P p_{in}$ are the pressure measurement at the initial position, ${}^P \mathbf{P}_k = [0 \quad 0 \quad {}^P p_k]^\top$ and ${}^P p_k$ is the pressure measurement at timestamp k , the three rotation matrices (${}^D_P \mathbf{R}$, ${}^D_I \mathbf{R}$, and ${}^I_G \mathbf{R}^\top$) are used to transform the pressure measurement into the global frame, and n_{p_z} is a zero-mean white Gaussian noise. For EKF update, the

Jacobian matrix \mathbf{H}_{p_z} with respect to the state, \mathbf{x}_k), can be found in [23].

$$z_{p_z,k} = h_{p_z}(\mathbf{x}_k) + n_{p_z} = \mathbf{s}_G^I {}^I_G \mathbf{R}^\top {}^D_I \mathbf{R} {}^D_P \mathbf{R} ({}^P \mathbf{P}_{in} - {}^P \mathbf{P}_k) + n_{p_z} \quad (9)$$

E. Visual Measurement Update

We perform point feature tracking on a selected image and use the feature tracking results in multiple sequential images (or keyframe images) to update the system state and covariance. For a well-calibrated camera, the measurement of a feature in the camera frame $\{C_i\}$ is the perspective projection of its 3D position ${}^{C_i} \mathbf{p}_f = [{}^{C_i} x, {}^{C_i} y, {}^{C_i} z]^\top$ onto the normalized plane, which is given by Eq. 10 where ${}^{C_i} \mathbf{p}_f$ is given by Eq. 11. In Eq. 10 and 11, $\pi(\cdot)$ is the perspective projection function, function $\tau(\cdot)$ transforms a point based on the given transformation matrix, \mathbf{n}_C is the zero-mean white Gaussian noise for camera measurement, $\{{}^C_I \mathbf{R}, {}^C \mathbf{p}_I\}$ is the extrinsic calibration between IMU and camera, $\{{}^I_G \mathbf{R}, {}^G \mathbf{p}_{I_k}\}$ is the cloned IMU pose at image time k , ${}^G \mathbf{p}_f$ is the feature's 3D position in the global frame.

$$\mathbf{z}_{C,i} = \pi({}^{C_i} \mathbf{p}_f) + \mathbf{n}_C = \begin{bmatrix} C_{i,x}/C_{i,z} \\ C_{i,y}/C_{i,z} \end{bmatrix} + \mathbf{n}_C \quad (10)$$

$${}^{C_i} \mathbf{p}_f = \tau({}^G \mathbf{p}_f, {}^C_I \mathbf{T}_G^I) = {}^C_I {}^I_G \mathbf{R} ({}^G \mathbf{p}_f - {}^G \mathbf{p}_{I_k}) + {}^C \mathbf{p}_I \quad (11)$$

For each feature, we compute the residual using the Jacobian matrices of the measurement function (Eq. 10) with respect to the state and feature (\mathbf{H}_x and \mathbf{H}_f)

$$\tilde{\mathbf{z}}_{C_k} = \mathbf{H}_x \tilde{\mathbf{x}}_k + \mathbf{H}_f {}^G \tilde{\mathbf{p}}_f + \mathbf{n}_C \quad (12)$$

Next, we apply the left nullspace [20] of the feature Jacobian to convert Eq. 12 to 13, and used it for EKF update.

$$\mathbf{N}^\top \tilde{\mathbf{z}}_{C_k} = \mathbf{N}^\top \mathbf{H}_x \tilde{\mathbf{x}}_k + \mathbf{N}^\top \mathbf{n}_C \quad (13)$$

F. Keyframe Marginalization

During underwater exploration, the vehicle may hover or move closer for detailed visual investigation. While hovering will cause a small translation between two consecutive image frames, the vertical movements will cause a small feature disparity within two successful tracking steps. Both cases will lead to degraded triangulation results, which brings in bad visual updates for state estimation. In [50], the authors propose a marginalization strategy that updates features in a consistent way for the hovering case. However, in a more general way, we intend to use keyframe marginalization to handle more degenerate motions. Inspired by the keyframe-based visual SLAM [30], we have implemented a strategy to insert the IMU clones based on three criteria, feature numbers, motion constraint, and scene constraint. If a new image has more than 50 features detected, its translation (estimated from DVL-IMU fused odometry) has exceeded 0.1m, and more than 10% of the features from the previous frame have disappeared, this new image will be treated as a keyframe.

When we reach the maximum number of keyframes, the algorithm will perform a marginalization similar to the one in S-MSCKF [51]. While the standard MSCKF [20] uses

the feature measurements in several poses (one-third of slide window evenly spaced), we will only use the feature measurements in two keyframes (the oldest one and the second-latest shown in Fig. 1) for standard MSCKF-feature update to reduce the computational cost. We also implemented another change during marginalization. Instead of removing two poses that were used in marginalization from the state matrix, our method will keep the second-latest pose even though it was used for marginalization. The second-latest pose is saved because of the following reasons. Any non-max-tracked features (i.e., f_{j+1} in Fig. 1) detected in second-latest pose will have minimum impact on the update because these features only have one measurement when doing the marginalization. However, keeping these features and the pose will allow us to use them in the future for sufficient MSCKF-feature updates.

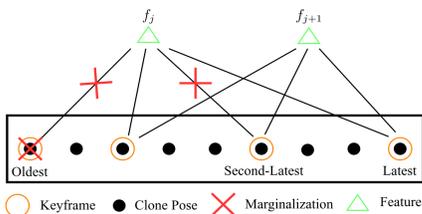


Fig. 1: The feature marginalization when clones at maximum.

IV. DVL-AIDED FEATURE ENHANCEMENT

During operation, the vehicle will keep a close distance (1-2 meters) to the ice such that the up-looking camera only has a small coverage, which causes a relatively short feature tracking distance. In other words, the relative pose changes between keyframes will be small which may lead to bad feature triangulation. To deal with this problem, we designed a DVL-aided feature enhancement strategy that uses the DVL point clouds obtained from 4 beams to constrain the feature depth. The detailed approach is discussed in the remaining content in this section.

A. Data Association

Typically, 3D-2D data association between point cloud and image feature require one-by-one matching [42]. However, DVL only provides sparse point clouds, making exact point matching challenging. Therefore, we attempt to find the nearest DVL point cloud around each feature, then apply bilinear interpolation to obtain the possible depth at the feature's location. Our method has three steps, identifying the anchor frame of the feature, identifying relevant DVL point clouds, and finding the DVL point cloud associated with the camera features. The anchor frame is selected based on the feature's location in the normalized plane. Among all keyframes, we select the one to be the anchor frame when the feature has a minimum offset from the camera origin because this feature will be highly likely to be bounded by the DVL points nearby in the anchor frame. The anchor frame will be treated as the reference frame for feature triangulation. To identify relevant DVL point clouds, we sort the buffered

DVL point cloud based on the timestamp difference between the DVL point clouds and the anchor frame. The m number of DVL point cloud with the smallest time differences will be selected for feature enhancement. To match the selected DVL point cloud with the feature, we design the following three-step procedure with pseudo-code displayed in Algorithm 1.

Algorithm 1: Feature and Cloud Match

input : A list of DVL timestamps with clouds
 $\{^{D_t}, D_t \mathcal{P}\}, i \in (1, m)$
 A list of IMU timestamps with clones
 $\{^{I_t}, I_t T \in SE(3)\}, j \in (1, n)$
 The threshold of standard deviation σ_z to filter outlier
 The normalized feature measurement (u, v) and the pose of the anchor frame ${}^C T$

output : cloud at camera frame ${}^{C_i} \mathcal{P}$

```

1  ${}^{C_i} \mathcal{P} \leftarrow \text{InitializeToEmpty}();$ 
2 foreach  $i \in (1, m), j \in (1, n)$  do
3   /* Step 1: Interpolate DVL Pose */
4   if  $I_t \leq D_t \leq I_{j+1}$  then
5      ${}^G T_{D_t} \leftarrow \text{Interpolation}({}^G T_{I_j}, {}^G T_{I_{j+1}}, I_t, I_{j+1})$ 
6   else
7     continue
8   end
9   /* Step 2: Outlier Rejection */
10   ${}^{G_i} \mathcal{P} \leftarrow \tau({}^{D_t} \mathcal{P}, {}^G T_{D_t} {}^G T)$ 
11  if  $\neg \text{Filter}({}^{G_i} \mathcal{P}, \sigma_z)$  then
12    continue
13  end
14  /* Step 3: Check Coverage */
15   ${}^{C_i} \mathcal{P} \leftarrow \tau({}^{G_i} \mathcal{P}, {}^C T)$ 
16  if  $\neg \text{PointInPolygon}(\pi({}^{C_i} \mathcal{P}), (u, v))$  then
17    continue
18  end
19 end
20 return  ${}^{C_i} \mathcal{P}$ 

```

First, we will interpolate the IMU pose at DVL timestamp based on IMU clone poses. For this step, We adopt the linear interpolation [52] as described in Eq. 14, where $\exp(\cdot)$ and $\log(\cdot)$ are the $SO(3)$ matrix exponential and logarithmic functions [53], D_t is the DVL timestamp, I_{a_t} and I_{b_t} are the beginning and end of the IMU clone interval timestamps.

$$\lambda = (D_t - I_{a_t}) / (I_{b_t} - I_{a_t}) \quad (14a)$$

$${}^I_G \mathbf{R} = \exp(\lambda \log({}^{I_b} \mathbf{R} {}^{I_a} \mathbf{R}^\top)) {}^{I_a} \mathbf{R} \quad (14b)$$

$${}^I_G \mathbf{p} = (1 - \lambda) {}^G \mathbf{p}_{I_a} + \lambda {}^G \mathbf{p}_{I_b} \quad (14c)$$

Second, we remove the outliers in the selected DVL point clouds. The DVL point cloud ${}^{D_t} \mathcal{P}$ will be transformed to the global frame based on the known transform between IMU and DVL. Since the vehicle will be operated close to the ice (1-2 meters), the 4 points obtained in a single DVL measurement will be close to each other, therefore, the terrain will not change significantly. However, during field trials, we may encounter ice-openings (either manually drilled or naturally formed), which causes outliers in the DVL point cloud. Therefore, we remove the selected DVL point cloud with depth outside the standard deviation, σ_z .

Third, we will check each selected DVL point cloud (4 points from each beam) if it bounds the feature. The DVL point cloud will be projected to the normalized plane of the anchor frame. If the feature is located inside the area bounded by 4 DVL points in the normalized plane, we will then use the DVL point cloud for feature enhancement.

B. Feature Enhancement

In [54], bilinear interpolation is applied to obtain the specific terrain depth from a digital elevation model (DEM). However, bilinear interpolation requires data located inside a rectilinear grid, which is not our case since the projected DVL point cloud could be an arbitrary quadrilateral. To transform an arbitrary quadrilateral to a unit square, we adopt a bilinear mapping function [55]:

$$u(\xi, \eta) = \alpha_0 + \alpha_1 \xi + \alpha_2 \eta + \alpha_3 \xi \eta \quad (15a)$$

$$v(\xi, \eta) = \beta_0 + \beta_1 \xi + \beta_2 \eta + \beta_3 \xi \eta \quad (15b)$$

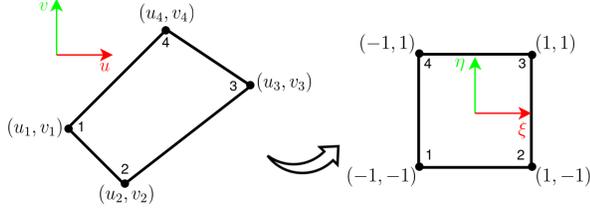


Fig. 2: The mapping process from an arbitrary quadrilateral to unit square

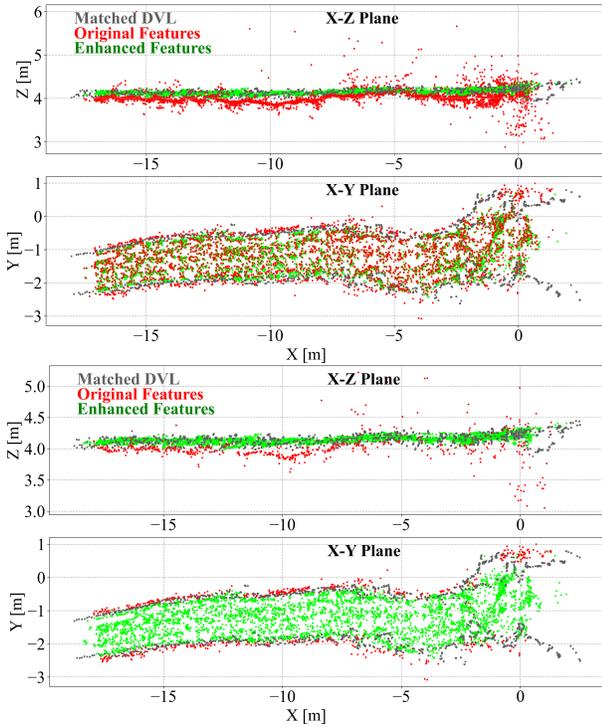


Fig. 3: Top: Comparison between triangulation and enhanced feature position, Bottom: Comparison between matched and not matched features. Matched DVL point cloud (only placed on edges) is gray, normal triangulated features are red, and enhanced features are green.

The coefficients α, β can be solved after substituting the normalized cloud point coordinates $\{\mathbf{u}_D, \mathbf{v}_D\}$ and square vertex coordinates $\{\xi, \eta\}$ shown in Figure 2. After that, the normalized feature coordinate can be mapped into the unit square and the bilinear interpolation can be applied to estimate the feature depth.

Feature position recovery has two steps, normal triangulation and position enhancement. First, for each feature, Direct linear transformation (DLT) [56] triangulation is applied if more than two observations in the keyframes. After that, inverse-depth parameterization [20] based nonlinear optimization is used for refining the feature position. Next, the feature position obtained from the two previous steps is corrected by multiplying the scaling ratio (z_a/z_b) where z_b is the feature depth computed from the two previous steps and z_a is the feature depth interpolated from the DVL point cloud. This change will then be incorporated into the visual measurement update discussed in Section III.E.

As shown in Figure 3, the first and second figures visualize the normal triangulation result (red) and enhanced feature position (green). The feature depth is well recovered since they almost align with the DVL point clouds (gray). The third and fourth figures visualize the enhanced (green) and un-enhanced (red) features. Most of the enhanced features are located inside DVL point clouds. Because of the keyframe strategy, there is no sufficient measurement update for each image frame. We intend to keep both enhanced and not enhanced features for measurement update.

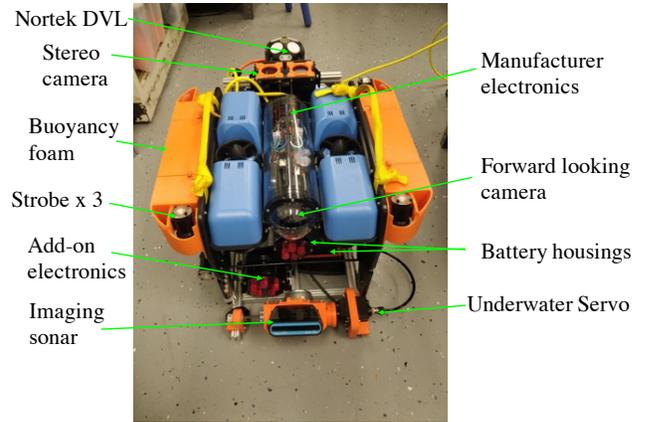


Fig. 4: The modified BlueROV-2 used in the experiment.

V. EXPERIMENT RESULTS

A. Experiment data set

In March 2021, we have conducted an under-ice experiment under the frozen Keweenaw Waterway in Michigan using a modified BlueROV2 [57] with a suite of sensors shown in Fig. 4. The ice thickness is about 30 cm. An ice hole (about 1m by 1m) was cut for deploying the ROV while several small ice holes (shown in Fig. 5) were also drilled along the transect. They are spaced at 10 meters except the second one. During the experiment, the ROV is remotely controlled by the pilot to drive along a straight line multiple times (roughly 40 meters each way) from a position at 4 meters deep, resulting in a total traveling distance of about 200 m and a total duration of about 20 minutes.

We used the experimental data set to validate our proposed sensor fusion framework. In the data set, the up-looking stereo-camera is running at 15 Hz with a raw image size



Fig. 5: The Metashape reconstructed result. The largest ice-hole on the left side is the starting point for the vehicle, the total length of this reconstructed result is roughly 40 meters

of 1616 by 1240 pixels. Only one camera was used for this experiment. The upward-looking DVL is pinging at 4 Hz, the IMU is running at 100 Hz, and the pressure sensor on the DVL is sampling at 2 Hz. The standard deviation (SD) of the DVL single ping at 3 m/s is about 0.005 m/s from Nortek technical specification. During the data collection, the vehicle moves at about 0.4 m/s and the transformation between DVL and IMU is roughly measured. Therefore, we set the velocity SD to between 0.0375 - 0.1 m/s for the experiment. The original result shown in [57] used the robot localization without correcting the time delays (about 10 seconds) between the IMU and DVL due to the DVL driver issue. Even though the localization in [57] shows a low drift, it may be a coincidence. In this data set, we have corrected the delays during the validation process. One unique feature in this data set is that, occasionally, the ROV is controlled to hover in place. Such maneuvers will challenge the visual SLAM performance since during the hovering no significant translation is available for feature triangulation. In application, hovering may be needed in several key locations during an under-ice exploration to collect more measurements on abnormal biogeochemical processes, e.g., a salt brine injection and algae bloom.

B. Results

The ground truth vehicle path is generated using Agisoft Metashape based on SfM technique, a rendering of the ice surface is shown in Fig. 5. For comparison, we use the evo [58] toolbox to align (recovery orientation and scale) Metashape ground truth and estimated odometry created from different sensor fusion methods. We only selected a short amount of time (90 seconds about 10 meters) at the beginning for alignment. Herein, we compare the localization results from 10 settings, as shown in Table I against the ground truth path.

TABLE I: Setup with different sensor suites and features. "Y" means used and "N" means not used.

Case #	1	2	3	4	5	6	7	8	9	10
Visual	Y	Y	Y	Y	Y	N	Y	Y	Y	Y
DVL	Y	Y	Y	Y	Y	Y	N	N	N	N
IMU	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Pressure	Y	Y	Y	Y	N	Y	N	N	Y	Y
Enhancement	Y	N	Y	N	Y	N	N	N	N	N
Keyframe	Y	Y	N	N	Y	N	Y	N	Y	N

For all tracks, we used identical parameters in the MSCKF and system initialization is conducted using the method from [23]. We used CLAHE [59] with KLT [60] method for

the front-end feature tracking because the descriptor based methods, such as the ORB and KAZE, didn't provide us with a consistent tracking result. We found that the descriptor-based method can be confused by the air bubbles in the ice which appear in similar shapes and sizes. We present all the resulting vehicle paths estimated from case 1 ~ 6 in Fig. 6(a) with different colors. Noted that VIO options (case 7 ~ 10) are not visualized since those runs failed quickly at the beginning because of the hovering maneuvers. From Fig. 6(a), we can easily observe that the odometry generated without the visual assist (case 6) is drifting away from the ground truth. In contrast, the paths generated with visual assistant stay closer to the ground truth path, especially, during the first and the second transects. We believe that the angle offset between tracks and the ground truth during the third and fourth segments may due to the hovering maneuverings (2-3 minutes) near the ROV deployment hole at the end of the second segment.

To further compare the performance in different cases, we computed the Absolute Trajectory Error (ATE) [61] in X and Y and the X-Y plane between the ground truth path and aligned each odometry. The statistical values are listed in Table II and the X-Y errors are presented in Fig. 6(b). Based on that, we could see that the integration of visual measurement into the MSCKF will help with reducing errors. Overall, the drift in the Y direction (transversal to the vehicle transects) is higher than in the x direction. This may be the fact that the vehicle's sway velocity is slightly small than its surge speed. Therefore, a lower SNR may be expected in the transversal direction, causing the increased drift.

TABLE II: ATE with RMSE metric for different cases.

Case #	1	2	3	4	5	6
RMSE(X)	0.39	0.42	1.34	1.23	1.45	3.54
RMSE(Y)	1.82	1.86	2.09	2.29	1.45	2.87
RMSE(X-Y)	1.11	1.14	1.71	1.76	1.45	3.21

When comparing the statistical values in Table II, we have several findings. First, visual-fused solutions are better than case 6 which only used the DVL, IMU, and pressure measurements. Second, case 1 and 2 are better than case 3 and 4. This comparison allows us to highlight the benefit of having keyframe selection mechanism which allows a longer translation for a better result in feature triangulation, ultimately affecting the localization. Third, case 1 is better than case 5 means pressure update actually helped the 2D pose estimation. Fourth, our method (case 1 with DVL-aided feature enhancement and keyframe selection enabled) produced the lowest RMSE. However, case 2 (with DVL-

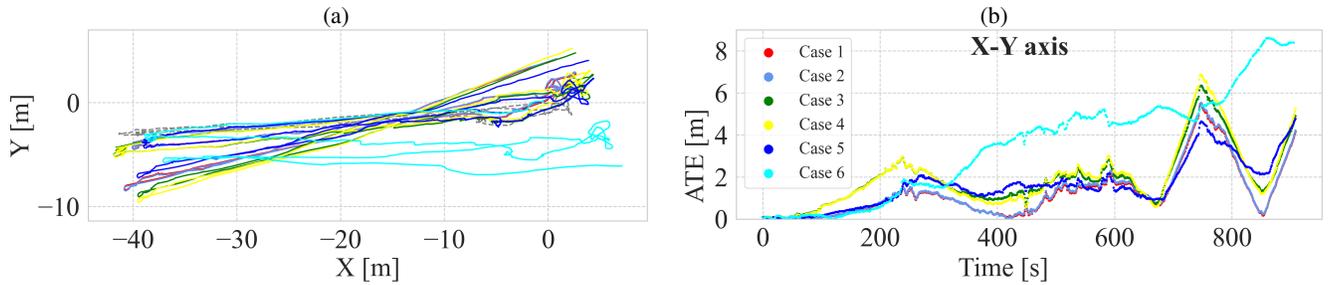


Fig. 6: The evaluation result. (a) The aligned trajectories for case 1 to case 6. (b) The ATE translation RMSE for X-Y axis.

aided feature enhancement disabled but keyframe selection enabled) is only slightly worse than case 1. This small improvement may be mainly caused by two following reasons. First, the detected visual features are too close to the vehicle (roughly between 1-2 meters). Therefore, the feature's position in z corrected by the DVL measurements is relatively small (even though the improvement is visible in Fig. 3), resulting in a small impact on the state estimation. Second, the feature measurements noise is set to 0.09 pixel which is relatively high compared to 0.0035 we set when testing the VIO on simulated data from OpenVINS. We also tried 0.01 for our data, the localization error was larger than the shown result. Therefore, we think there are still room for improvement, especially, in the front-end feature tracking.

VI. CONCLUSIONS AND FUTURE PLAN

In this paper, we presented a tightly-couple Visual-DVL-Inertial odometry for underwater robots. A modified keyframe selection and marginalization method was introduced, and a DVL-aided feature enhancement approach is realized to further improve the localization performance. With those key contributions, we have validated the complete framework with a challenging under-ice data set. Based on the statistical values, we found our method has the lowest error with an RMSE of 1.11 meters for X-Y plane translation.

We are planning our future research in two directions, upgrading the existing frame and integrating more perception sensors. Currently, our visual measurement noise is set relatively high. But, we expect the improved front-end feature tracking could allow us to lower the measurement noise for better localization results. Previous research [62] has shown that a well-calibrated transform between DVL and IMU will improve navigation accuracy. We are also interested in integrating extra perception sensors such as a forward-looking sonar that could provide feature measurements at a further distance with scales. However, imaging sonar normally has a wide elevation angle that could not be directly resolved from the image, which will pose challenges in feature detection and tracking. In the future, we are also interested in evaluating our algorithm on other underwater datasets and comparing it with the state-of-art SLAM such as SVIn2 [18] and OpenVINS [29].

REFERENCES

- [1] M. Vancoppenolle, K. M. Meiners, C. Michel, L. Bopp, F. Brabant, *et al.*, "Role of sea ice in global biogeochemical cycles: emerging views and challenges," *Quaternary Science Reviews*, vol. 79, pp. 207–230, 2013.
- [2] B. Loose, A. C. Naveira Garabato, P. Schlosser, W. J. Jenkins, *et al.*, "Evidence of an active volcanic heat source beneath the pine island glacier," *Nature Communications*, vol. 9, no. 2431, 2018.
- [3] K. R. Arrigo, D. K. Perovich, R. S. Pickart, Z. W. Brown, V. Dijken, *et al.*, "Massive phytoplankton blooms under arctic sea ice," *Science*, vol. 336, no. 6087, pp. 1408–1408, 2012.
- [4] P. Wadhams, "Arctic ice cover, ice thickness and tipping points," *Ambio*, vol. 41(1), pp. 23–33, 2012.
- [5] B. H. Robison, M. Vernet, and K. L. Smith, "Algal communities attached to free-drifting, antarctic icebergs," *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 58, no. 11, pp. 1451–1456, 2011.
- [6] A. Spears, M. West, M. Meister, J. Buffo, C. Walker, *et al.*, "Under ice in antarctica: The icefin unmanned underwater vehicle development and deployment," *IEEE Robotics Automation Magazine*, vol. 23, no. 4, pp. 30–41, 2016. 1
- [7] A. B. Phillips, M. Kingsland, N. Linton, W. Baker, L. Bowring, *et al.*, "Autosub 2000 under ice: Design of a new work class auv for under ice exploration," in *2020 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, 2020, pp. 1–8. 1
- [8] G. Williams, T. Maksym, J. Wilkinson, C. Kunz, C. Murphy, *et al.*, "Thick and deformed antarctic sea ice mapped with autonomous underwater vehicles," *Nature Geoscience*, vol. 8, pp. 61–67, 2014. 1
- [9] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 131–149, 2014.
- [10] L. D. L. Barker, M. V. Jakuba, A. D. Bowen, C. R. German, T. Maksym, *et al.*, "Scientific challenges and present capabilities in underwater robotic vehicle design and navigation for oceanographic exploration under-ice," *Remote Sensing*, vol. 12, no. 16, 2020.
- [11] M. V. Jakuba, C. N. Roman, H. Singh, C. Murphy, C. Kunz, *et al.*, "Long-baseline acoustic navigation for under-ice autonomous underwater vehicle operations," *Journal of Field Robotics*, vol. 25, no. 11–12, pp. 861–879, 2008.
- [12] A. Kukulya, A. Plueddemann, T. Austin, R. Stokey, M. Purcell, *et al.*, "Under-ice operations with a REMUS-100 AUV in the arctic," in *2010 IEEE/OES Autonomous Underwater Vehicles*, Monterey, CA, USA, Sep. 2010, pp. 1–8.
- [13] L. Freitag, P. Koski, S. Singh, T. Maksym, and H. Singh, "Acoustic communications under shallow shore-fast arctic ice," in *OCEANS 2017 - Anchorage*, 2017, pp. 1–5.
- [14] L. Whitcomb, D. Yoerger, and H. Singh, "Advances in doppler-based navigation of underwater robotic vehicles," in *Proceedings 1999 IEEE International Conference on Robotics and Automation*, vol. 1, 1999, pp. 399–406 vol.1.
- [15] S. A. T. Randeni P., N. R. Rypkema, E. M. Fischell, A. L. Forrest, M. R. Benjamin, and H. Schmidt, "Implementation of a hydrodynamic model-based navigation system for a low-cost auv fleet," in *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, 2018, pp. 1–6.
- [16] G. Huang, "Visual-inertial navigation: A concise review," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 9572–9582.
- [17] B. Joshi, S. Rahman, M. Kalaitzakis, B. Cain, J. Johnson, *et al.*, "Experimental comparison of open source visual-inertial-based state estimation algorithms in the underwater domain," in *2019 IEEE/RSJ*

- International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 7227–7233.
- [18] S. Rahman, A. Q. Li, and I. Rekleitis, “SVIn2: An underwater SLAM system using sonar, visual, inertial, and depth sensor,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, Nov. 2019, pp. 1861–1868.
- [19] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, “Keyframe-based visual-inertial odometry using nonlinear optimization,” *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.
- [20] A. I. Mourikis and S. I. Roumeliotis, “A multi-state constraint kalman filter for vision-aided inertial navigation,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, Rome, Italy, May 2007, pp. 3565–3572.
- [21] Google, “Google ARCore,” <https://developers.google.com/ar/>.
- [22] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, “Analysis and improvement of the consistency of extended kalman filter based SLAM,” in *2008 IEEE International Conference on Robotics and Automation*, 2008, pp. 473–479.
- [23] L. Zhao, M. Zhou, and B. Loose, “Towards under-ice sensing using a portable ROV,” in *OCEANS 2022, Hampton Roads*, 2022, pp. 1–8.
- [24] P. A. Miller, J. A. Farrell, Y. Zhao, and V. Djapic, “Autonomous underwater vehicle navigation,” *IEEE Journal of Oceanic Engineering*, vol. 35, no. 3, pp. 663–678, 2010.
- [25] P. Li, Y. Liu, T. Yan, S. Yang, and R. Li, “A robust INS/USBL/DVL integrated navigation algorithm using graph optimization,” *Sensors*, vol. 23, no. 2, 2023.
- [26] A. Palomer, P. Ridao, and D. Ribas, “Multibeam 3D underwater SLAM with probabilistic registration,” *Sensors*, vol. 16, no. 4, p. 560, 2016.
- [27] C. Cheng, C. Wang, D. Yang, W. Liu, and F. Zhang, “Underwater localization and mapping based on multi-beam forward looking sonar,” *Frontiers in Neurobotics*, vol. 15, 2022.
- [28] P. Ozog and R. M. Eustice, “Real-time SLAM with piecewise-planar surface models and sparse 3D point clouds,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 1042–1049.
- [29] P. Geneva, K. Eickenhoff, W. Lee, Y. Yang, and G. Huang, “OpenVINS: A research platform for visual-inertial estimation,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 4666–4672.
- [30] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, “ORB-SLAM3: An accurate open-source library for visual, visual-inertial, and multimap SLAM,” *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.
- [31] H. Singh, C. Roman, O. Pizarro, R. Eustice, and A. Can, “Towards high-resolution imaging from underwater vehicles,” *The International Journal of Robotics Research*, vol. 26, no. 1, pp. 55–74, 2007.
- [32] T. Nicosevici, N. Gracias, S. Negahdaripour, and R. Garcia, “Efficient three-dimensional scene modeling and mosaicing,” *Journal of Field Robotics*, vol. 26, no. 10, pp. 759–788, 2009.
- [33] N. Gracias, S. van der Zwaan, A. Bernardino, and J. Santos-Victor, “Mosaic-based navigation for autonomous underwater vehicles,” *IEEE Journal of Oceanic Engineering*, vol. 28, no. 4, pp. 609–624, 2003. 2
- [34] M. Ferrera, J. Moras, P. Trouvé-Peloux, and V. Creuze, “Real-time monocular visual odometry for turbid and dynamic underwater environments,” *Sensors*, vol. 19, no. 3, 2019. 2
- [35] P. Corke, C. Detweiler, M. Dunbabin, M. Hamilton, D. Rus, and I. Vasilescu, “Experiments with underwater robot localization and tracking,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 4556–4561. 2
- [36] F. Bellavia, M. Fanfani, and C. Colombo, “Selective visual odometry for accurate AUV localization,” *Autonomous Robots*, vol. 41, pp. 133–143, 2017. 2
- [37] F. Shkurti, I. Rekleitis, M. Scaccia, and G. Dudek, “State estimation of an underwater robot using visual and inertial information,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 5054–5060.
- [38] C. Hu, S. Zhu, Y. Liang, and W. Song, “Tightly-coupled visual-inertial-pressure fusion using forward and backward IMU preintegration,” *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6790–6797, 2022.
- [39] A. Kim and R. M. Eustice, “Real-time visual SLAM for autonomous underwater hull inspection using visual saliency,” *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 719–733, 2013. 2
- [40] E. Vargas, R. Scona, J. S. Willners, T. Luczynski, Y. Cao, *et al.*, “Robust underwater visual SLAM fusing acoustic sensing,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 2140–2146. 2
- [41] A. G. Chavez, Q. Xu, C. A. Mueller, S. Schwertfeger, and A. Birk, “Adaptive navigation scheme for optimal deep-sea localization using multimodal perception cues,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 7211–7218. 2
- [42] W. Zhen, Y. Hu, H. Yu, and S. Scherer, “LiDAR-enhanced structure-from-motion,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 6773–6779.
- [43] J. Zhang, M. Kaess, and S. Singh, “Real-time depth enhanced monocular odometry,” in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 4973–4980.
- [44] T. Shan, B. Englot, C. Ratti, and D. Rus, “LVI-SAM: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, Xi’an, China, Jun. 2021, pp. 5692–5698.
- [45] M. Roznere and A. Q. Li, “Underwater monocular image depth estimation using single-beam echosounder,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 1785–1790.
- [46] Z. Xu, M. Haroutunian, A. J. Murphy, J. Neasham, and R. Norman, “An integrated visual odometry system for underwater vehicles,” *IEEE Journal of Oceanic Engineering*, vol. 46, no. 3, pp. 848–863, 2021.
- [47] N. Trawny and S. I. Roumeliotis, “Indirect kalman filter for 3D attitude estimation,” Department of Computer Science, University of Minnesota, Tech. Rep., 2005.
- [48] C. Hertzberg, R. Wagner, U. Frese, and L. Schröder, “Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds,” *Information Fusion*, vol. 14, no. 1, pp. 57–77, 2013.
- [49] A. B. Chatfield, *Fundamentals Of High Accuracy Inertial Navigation*. AIAA, 1997.
- [50] D. G. Kottas, K. J. Wu, and S. I. Roumeliotis, “Detecting and dealing with hovering maneuvers in vision-aided inertial navigation systems,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 3172–3179.
- [51] K. Sun, K. Mohta, B. Pfommer, M. Watterson, S. Liu, *et al.*, “Robust stereo visual inertial odometry for fast autonomous flight,” *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 965–972, 2018.
- [52] M. Li, “Visual-inertial odometry on resource-constrained systems,” Ph.D. dissertation, University of California, Riverside, 2014.
- [53] G. S. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 2*. Birkhäuser Boston, MA, 2011.
- [54] B. Claus and R. Bachmayer, “Terrain-aided navigation for an underwater glider,” *Journal of Field Robotics*, vol. 32, no. 7, pp. 935–951, 2015.
- [55] T. J. R. Hughes, *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publications, 2000.
- [56] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.
- [57] L. Zhao, M. Zhou, B. Loose, V. Cousens, and R. Turrissi, “Modifying an affordable ROV for under-ice sensing,” in *OCEANS 2021: San Diego – Porto*, 2021, pp. 1–5.
- [58] M. Grupp, “evo: Python package for the evaluation of odometry and SLAM,” <https://github.com/MichaelGrupp/evo>, 2017.
- [59] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman, and K. Zuiderveld, “Adaptive histogram equalization and its variations,” *Computer Vision, Graphics, and Image Processing*, vol. 39, no. 3, pp. 355–368, 1987.
- [60] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, ser. IJCAI’81, San Francisco, CA, USA, 1981, p. 674–679.
- [61] Z. Zhang and D. Scaramuzza, “A tutorial on quantitative trajectory evaluation for visual(-inertial) odometry,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 7244–7251.
- [62] G. Troni and L. L. Whitcomb, “Advances in in situ alignment calibration of doppler and high/low-end attitude sensors for underwater vehicle navigation: Theory and experimental evaluation,” *Journal of Field Robotics*, vol. 32, no. 5, pp. 655–674, 2015.