2019 International Conference on Robotics and Automation (ICRA)
Palais des congres de Montreal, Montreal, Canada, May 20-24, 2019

Underwater Terrain Reconstruction from Forward-Looking Sonar Imagery

Jinkun Wang, Tixiao Shan and Brendan Englot

***Abstract*— In this paper, we propose a novel approach for underwater simultaneous localization and mapping using a multibeam imaging sonar for 3D terrain mapping tasks. The high levels of noise and the absence of elevation angle information in sonar images present major challenges for data association and accurate 3D mapping. Instead of repeatedly projecting extracted features into Euclidean space, we apply optical flow within bearing-range images for tracking extracted features. To deal with degenerate cases, such as when tracking is interrupted by noise, we model the subsea terrain as a Gaussian Process random field on a Chow–Liu tree. Terrain factors are incorporated into the factor graph, aimed at smoothing the terrain elevation estimate. We demonstrate the performance of our proposed algorithm in a simulated environment, which shows that terrain factors effectively reduce estimation error. We also show ROV experiments performed in a variable-elevation tank environment, where we are able to construct a descriptive and smooth height estimate of the tank bottom.**

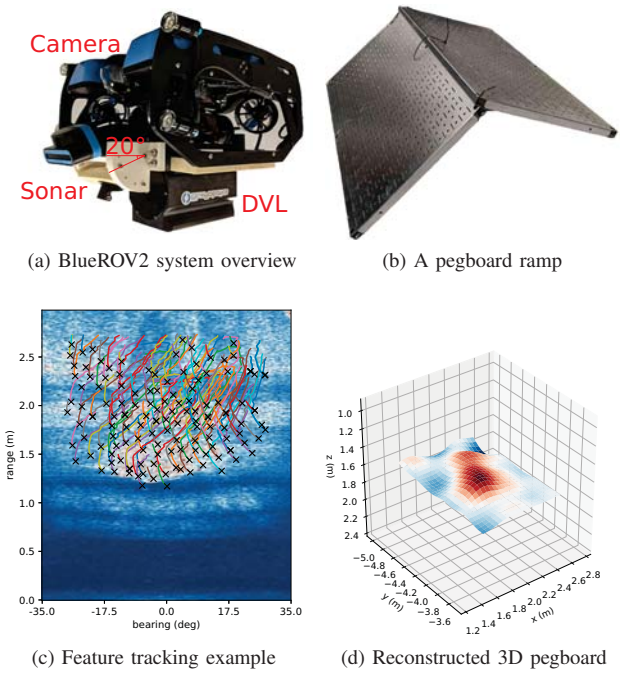
I. Introduction

Over the last two decades, the application of autonomous underwater vehicles (AUVs) has proliferated across challenging perceptual tasks such as pipeline and ship hull inspection, bathymetric survey, and structure mapping. Many efforts have been devoted to achieving the autonomy of underwater vehicles for such tasks, and an accurate representation of the environment is an essential prerequisite for such autonomy. To acquire such a representation of the environment, optical sensors (cameras) or acoustic sensors (sonars) are typically utilized. Although a camera can capture fine details of the underwater environment, its capability is often limited by the turbidity of the water. Additionally, illumination changes may make captured data unreliable. On the other hand, sonar will function in water that has high turbidity, offering long-range visibility and a wide aperture.

Among much of the underwater simultaneous localization and mapping (SLAM) literature, mapping is limited to 2D representations of the environment [1], [2]. However, constructing a 3D representation of the underwater environment with sonar is challenging due to its physical limitations. Sonar is plagued by high levels of noise and low resolution, and the lack of an elevation angle in a sonar measurement is another major obstacle to achieving accurate 3D mapping.

In [3], two sonars mounted orthogonally on a torpedo-shaped AUV are used to support scan-matching within a SLAM framework, and the vertical sonar with a narrow beam is used for 3D mapping. In [4], point features are extracted and registered to those from other sonar images, and the pairwise transformation between images is used to constrain the vehicle. However, this work assumes that an imaging sonar is aligned with the terrain surface, otherwise the transformation is erroneous. The detailed model of a ship hull can be generated by configuring a multibeam sonar in profiling mode with narrower vertical beam-width [5]. Elevation recovery from acoustic shadows is proposed in [6], but the application is limited when shadows do not exist.

Acoustic structure from motion (ASFM) [7] is proposed to address the ambiguity of the elevation angles associated with an imaging sonar's returns from the surrounding environment. The influence of different robot motion primitives on this degeneracy is discussed in [7], [8]. Data association in ASFM based on reprojection error is proposed in [9]. Non-parametric and semi-parametric factors are introduced to handle under-constrained landmarks in [10]. Degeneracy is determined by examining the eigenvalues of the Jacobian matrix of a specific landmark. However, under-constrained landmarks are beneficial only for localization, and elevation angles can't be accurately estimated from graph optimization.



(a) BlueROV2 system overview (b) A pegboard ramp

(c) Feature tracking example (d) Reconstructed 3D pegboard

Fig. 1: Overview of this work. A multibeam sonar mounted at 20 degrees downward is used to observe objects on the floor, including a pegboard ramp. Extracted features are tracked for data association, and smooth terrain constraints are incorporated in a factor graph. Finally, a GP terrain map can be built using estimated 3D features.

J. Wang, T. Shan and B. Englot are with the Department of Mechanical Engineering, Stevens Institute of Technology, Castle Point on Hudson, Hoboken NJ 07030 USA, {jwang92, tshan3, benglot}@stevens.edu.

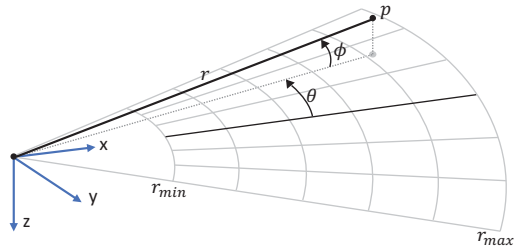


Fig. 2: Imaging sonar model. A feature p can be represented as $[r, \theta, \phi]^T$ in a spherical coordinate frame. Note that the range r and the azimuth angle θ of p can be directly derived from measurements, while the elevation angle ϕ is lost in the 2D sonar image.

In this work, we propose a feature-based SLAM formulation for underwater vehicles performing terrain mapping with automatic feature extraction and data association. Our work is similar to the ASFM framework seeking to solve elevation ambiguity, but includes the following novel contributions. We propose using an optical flow method for tracking features directly within raw sonar images. To cope with under-constrained features resulting from tracking failures, we introduce correlation between feature points, assuming they are sampled from a Gaussian Process terrain map. The model is approximated by a Gaussian Process random field on a tree structure, which can be easily integrated into a factor graph. We present this feature-based SLAM framework with tracking-based data association and terrain factors in the next two sections, followed by simulation and real experiments with a remotely operated vehicle (ROV).

II. Feature-based SLAM with Imaging Sonar

In this section, we give a brief introduction to the geometry of imaging sonar measurements and feature-based SLAM, then data association is solved by feature tracking using an optical flow method.

A. Imaging Sonar Model

Given a feature $\mathbf{l}^s = [x, y, z]^T$ in the sensor frame represented in Cartesian coordinates, we can describe it as $\mathbf{s} = [r, \theta, \phi]^T$ in spherical coordinates, where r is the range to the sensor origin, θ is the azimuth and ϕ is the elevation angle (see Fig. 2). The conversion between these two forms can be expressed with the following equations:

$$\mathbf{l}^s = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r\cos\phi\cos\theta \\ r\cos\phi\sin\theta \\ r\sin\phi \end{bmatrix} \tag{1}$$

$$\mathbf{s} = \begin{bmatrix} r \\ \theta \\ \phi \end{bmatrix} = h(\mathbf{l}^s) = \begin{bmatrix} \sqrt{x^2+y^2+z^2} \\ \arctan 2(y, x) \\ \arctan 2(z, \sqrt{x^2+y^2}) \end{bmatrix}. \tag{2}$$

Though the range r and the azimuth angle θ of feature $\mathbf{l}^s$ can be directly derived from the 2D sonar image, the elevation angle $\phi = h_\phi(\mathbf{l}^s)$ is lost due to the wide vertical aperture. In other words, the mapping of a 3D world into a 2D sonar image eliminates the elevation information, which results in the ambiguity of features appearing along any $|\phi| \leq \phi_{\max}$ arc. Although the vertical aperture is significantly reduced by leveraging a lens or using a profiling sonar, a large field of view in elevation angle is often beneficial for a robot's situational awareness.

B. Feature-based SLAM

We formulate feature-based SLAM as a least-squares problem using the same notation as [11]. We assume a Gaussian measurement model between robot state $\mathbf{x}_i \in \mathcal{X}$ and feature position $\mathbf{l}_j \in \mathcal{L}$, and we assume that the process model given an input $\mathbf{u}_i \in \mathcal{U}$ is as follows:

$$\mathbf{x}_i = f_i(\mathbf{x}_{i-1}, \mathbf{u}_i) + \mathbf{w}_i, \;\; \mathbf{w}_i \sim \mathcal{N}(\mathbf{0}, \Lambda_i), \tag{3}$$

$$\mathbf{z}_k = h_k(\mathbf{x}_{i_k}, \mathbf{l}_{j_k}) + \mathbf{v}_k, \;\; \mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \Gamma_k), \tag{4}$$

where i_k, j_k denote the associated state and feature corresponding to the k-th measurement (see Sec. II.C). The evolution of state is modeled by f using measurements from navigation sensors, including an inertial measurement unit (IMU) and Doppler velocity log (DVL). The observation is predicted in h by transforming features from the global frame to the sonar frame parameterized in spherical coordinates, or formally, $h(\mathbf{x}, \mathbf{l}) = h(\mathbf{l}^s)$ in Eq. 2. The estimate is obtained by solving the nonlinear least-squares equation,

$$\mathcal{X}^*, \mathcal{L}^* = \arg\min_{\mathcal{X},\mathcal{L}} \sum_i ||\mathbf{x}_i - f_i(\mathbf{x}_{i-1}, \mathbf{u}_i)||^2_{\Lambda_i} + \sum_k ||\mathbf{z}_k - h_k(\mathbf{x}_{i_k}, \mathbf{l}_{j_k})||^2_{\Gamma_k}. \tag{5}$$

In previous work on 3D SLAM using imaging sonar, no constraints are imposed on the elevation angle, and its ambiguity is clarified by solving the least-squares equations, which may contain measurements of the same feature from a variety of perspectives. However, in practice, the nonlinear system regularly becomes ill-posed without a proper initial estimate and sufficient sensor motion [7], [8]. Advanced multi-beam sonars feature a narrow vertical aperture (12° in our experiments), and given the mounting configuration shown in Fig. 1a, elevation angles of observed features are distributed symmetrically around zero. Therefore in this work, an imaginary elevation angle $\tilde{\phi}$ is incorporated into the measurement vector $\mathbf{z} = [r, \theta, \tilde{\phi}]^T$, which is varied based on the predicted elevation $h_\phi(\mathbf{l}^s)$,

$$\tilde{\phi} = \begin{cases} \hat{\phi}, & : |h_\phi(\mathbf{l}^s)| \leq \phi_{\max} \\ 0, & : |h_\phi(\mathbf{l}^s)| > \phi_{\max}. \end{cases} \tag{6}$$

The measurement noise covariance matrix is simplified to $\Gamma = \text{diag}([\sigma_r^2, \sigma_\theta^2, \sigma_{\tilde{\phi}}^2])$, with the standard deviation of elevation noise set to half the vertical aperture, and $\phi_{\max} = 3\sigma_{\tilde{\phi}}$.

C. Feature Tracking

As a variety of noise exists in sonar images, such as Gaussian, impulse and speckle noise, we rely on the A-KAZE feature detector [12] as in previous work on this subject [10], which is designed to describe features at different smoothing scales while retaining details. An example of extracted A-KAZE features is shown in Fig. 1(c) (denoted by black $\times$).

The correspondence (i_k, j_k) between features detected in different sonar frames has been solved through data association techniques [9], [13] and point cloud registration [4]. Data association requires the computation of feature uncertainty in 3D space, which is computationally expensive in the presence of dense features. The ambiguity of elevation angles also presents a challenge.

In this work, our ROV moves at a relatively slow speed, and we seek to match features by tracking based on optical flow. Optical flow, specifically the Lucas-Kanade method [14], is developed on two assumptions: (1) feature intensities do not change between consecutive frames, and (2) neighboring pixels have similar motion. Although acoustic returns from objects at different elevation angles have different intensities, the assumptions hold well in practice with sonar images acquired at high frequency and with slow motion.

The feature tracking works as follows. We extract A-KAZE features that are used for tracking in the initial frame, and new features are introduced if they are not in close proximity to current features. Tracking of a feature stops when the minimum eigenvalue criterion isn't satisfied in the Lucas-Kanade method, or the distance between descriptors computed at previous and current feature locations is larger than a designated threshold. The tracking history of A-KAZE features is visualized in Fig. 1(c). Feature motions are consistent with the trajectory, but tracking is error-prone when the range to a feature is larger than 2.0 meters. One reason is that sonar has limited angular resolution, causing sonar imagery further away from the origin to be blurrier along the bearing-axis. There are also a few short feature tracks that don't start from the top of an image due to tracking failure. To some extent, estimation error is attributed to the insufficient measurement constraints from short tracks.

III. Terrain Model with Gaussian Process Random Fields

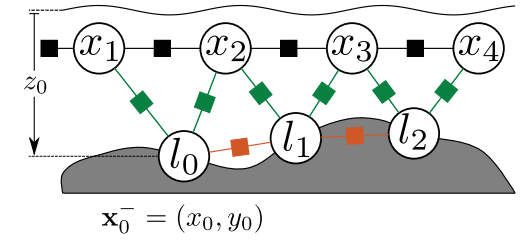
In this section, we discuss the Gaussian Process terrain model and the approximation methods required for it to be integrated into a factor graph.

A. Gaussian Process Terrain Models

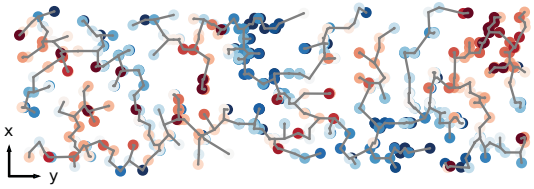
We frame the mapping part of SLAM as a terrain modeling problem. Let $\mathbf{x}^- = [x_i, y_i]^T$ be the $x-y$ location of feature $\mathbf{l}_i$, let $m_i = z_i$ be the height of the feature, which is defined as the vertical distance from the water surface to the feature in Fig. 3a, and let $X^-_{M\times 2}$ and $\mathbf{m}_{M\times 1}$ be the 2D location matrix and height vector for feature set $\mathcal{L}$. The terrain model maps 2D location to height $m = g(\mathbf{x}^-)$.

Gaussian Processes (GPs) are a non-parametric approach to learn a latent function enforcing correlation among input data. We use Gaussian Process regression to model the terrain data [15], and thus feature heights are spatially dependent in spite of the fact that they are independently observed and tracked. Mathematically,

$$g(\mathbf{x}^-) \sim \mathcal{GP}(0, k(\mathbf{x}^-, \mathbf{x}^{-\prime})), \tag{7}$$



(a) Factor graph with terrain factors



(b) Terrain factor construction using CLT

Fig. 3: Terrain factors (orange) connecting two landmark nodes are constructed on the edges of a Chow–Liu tree. The tree is built with projected 2D features $\{\mathbf{x}_i\}$, and nodes are colored by optimized terrain height z_i for visualization.

in which the mean function is zero, and $k(\cdot,\cdot)$ is the covariance function that defines the similarity between a pair of height variables. Under the GP assumption, the aggregated height vector is distributed as a joint Gaussian distribution with dense covariance matrix,

$$\mathbf{m} \sim \mathcal{N}(\mathbf{0}, K(X^-, X^-)). \tag{8}$$

The predicted mean at any location given existing observations is expressed as

$$\mathbb{E}[\mathbf{m}_*] = K(X^-_*, X^-)K^{-1}(X^-, X^-)\mathbf{m}. \tag{9}$$

GPs bring several benefits to the terrain reconstruction problem. Under-constrained landmarks can be handled more effectively, whether this stems from the fact that feature detection is less accurate on sonar images with low signal-to-noise ratio, or from an overly simplistic motion a robot has executed. The addition of a GP model imposes constraints on the shape of the terrain formed by the observed features. The impact of a GP terrain model is depicted in Fig. 4, where 20 samples are drawn from the distributions $\prod \mathcal{N}(\phi_i|\tilde{\phi}_i, \sigma^2_\phi)$ and $\prod \mathcal{N}(\phi_i|\tilde{\phi}_i, \sigma^2_\phi)\mathcal{N}(\mathbf{m}|\mathbf{0}, K(X^-, X^-))$. Here, we only consider range and elevation angle, and we use a feature's true elevation as a mean value (with the parameters mentioned in Sec. IV-A). In addition to the above advantages, features are sparse in underwater environments, thus leaving gaps between 3D points. GP regression enables rich and reasonable inference in regions without measurements.

B. Terrain Factors

The GP model is essentially a giant factor involving all landmark nodes, and the optimization cost is prohibitive. In order to implement correlated terrain factors while maintaining sparsity of the factor graph, we approximate the full GP model with a Gaussian Process random field [16] defined on a tree structure. A product of conditional distributions is

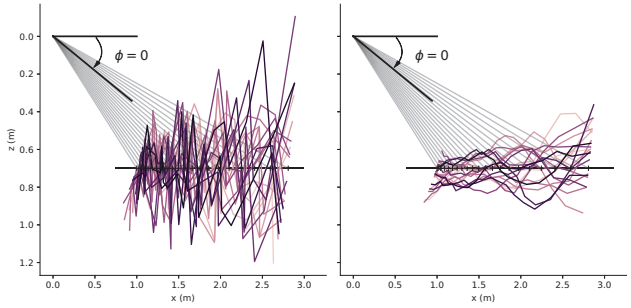


Fig. 4: Terrain height estimates resulting from 20 noise-corrupted measurements of the 20 features in the sonar's field of view, both without (left) and with (right) a GP terrain model. The actual (flat) ground is marked in black while terrain estimates are colored. The zero-elevation plane of the sonar is represented as a black line.

implied from a tree structure,

$$p(\mathbf{m}) \approx q_{\text{tree}}(\mathbf{m}) = p(m_{\text{root}}) \prod_{i \neq \text{root}} p(m_i | m_{\pi_i}), \tag{10}$$

where π_i is the parent of node i.

Given the assumption that $[m_i, m_{\pi_i}]^T$ are distributed as a Gaussian process, we can formulate the joint distribution and conditional distribution as

$$\begin{bmatrix} m_i \\ m_{\pi_i} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} k_{ii} & k_{i\pi_i} \\ k_{\pi_i i} & k_{\pi_i \pi_i} \end{bmatrix} \right), \tag{11}$$

$$\begin{aligned} m_i | m_{\pi_i} &\sim \mathcal{N}(\mu_{i|\pi_i}, \Sigma_{i|\pi_i}) \\ &= \mathcal{N}(k_{i\pi_i} k_{\pi_i \pi_i}^{-1} m_{\pi_i}, k_{ii} - k_{i\pi_i}^2 k_{\pi_i \pi_i}^{-1}), \end{aligned} \tag{12}$$

where k_{ij} is defined as $k(\mathbf{x}_i^-, \mathbf{x}_j^-)$. Accordingly, the terrain constraint is expressed as

$$m_i = \mu_{i|\pi_i} + \epsilon_i, \epsilon_i \sim \mathcal{N}(0, \Sigma_{i|\pi_i}). \tag{13}$$

The terrain factors are error functions between two elevation variables, which in turn are incorporated into the least-squares minimization in Eq. 5,

$$||m_{i=\text{root}}||_{k_{ii}}^2 + \sum_{i \neq \text{root}} ||m_i - \mu_{i|\pi_i}||_{\Sigma_{i|\pi_i}}^2. \tag{14}$$

C. Implementation

The tree structure relating a set of features is determined using the Chow-Liu algorithm [17], [18], which constructs the Chow–Liu tree (CLT) by minimizing the Kullback–Leibler divergence between the joint Gaussian distribution and the approximated distribution. In essence, the problem is to find the maximum mutual information spanning tree on a graph, which contains edges connecting any two 2D points with weights defined by the mutual information between two random elevation variables,

$$I(m_i, m_j) = -\frac{1}{2} \log(1 - \rho^2), \tag{15}$$

where $\rho = \frac{k_{ij}}{\sqrt{k_{ii} k_{jj}}}$ is the correlation coefficient. The mutual information is monotonically decreasing with respect to the Euclidean distance between two 2D points if we use a stationary covariance function. As a consequence, the maximum mutual information spanning tree is equivalent to the Euclidean minimum spanning tree, and searching can be limited to edges in a Delaunay triangulation of 2D points.

The terrain factor requires the 2D location $\mathbf{x}^-$ of a feature to compute the conditional distribution in Eq. 12. Since our vehicle's motion is mostly along the x-axis (forward), ambiguity in feature position is likely to appear along the z-axis [7]. Take the sensor configuration shown in Fig. 1a as an example. The spread due to elevation change has less impact on the projected $x - y$ plane than it does in z:

$$\frac{\Delta z}{\Delta xy} = \frac{r \sin(20° + 6°) - r \sin(20° - 6°)}{r \sin(20° - 6°) - r \cos(20° + 6°)} \approx 2.75. \tag{16}$$

Therefore, we optimize the trajectory and feature positions without introducing terrain constraints, and then terrain factors are constructed upon estimating the 2D feature locations X^-, followed by further optimization of the whole system.

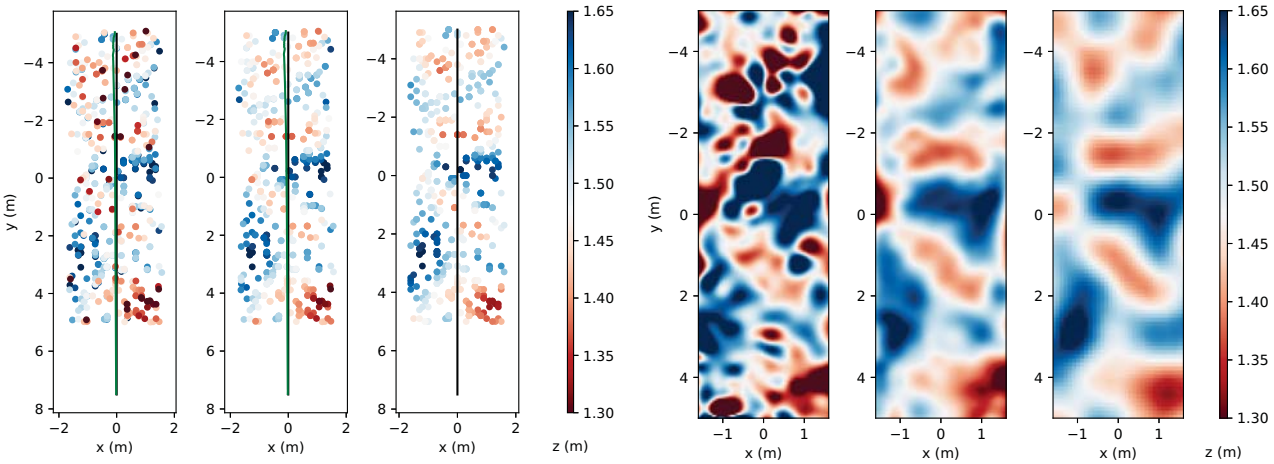
IV. Experiments and Results

To validate the proposed methodology, simulated and real experiments are carried out. In both cases, an underwater vehicle is commanded to move forward at low speed (0.2 m/s) while holding a fixed depth. The sonar is mounted at 20 degrees downward from horizontal for better coverage of the terrain, as depicted in Fig. 1a. We implement our algorithm upon GTSAM [19], and in particular, DogLeg optimization is used to perform factor graph optimization. Functions in OpenCV are utilized for feature extraction (A-KAZE) and tracking (iterative Lucas-Kanade method with pyramids). We employ Gaussian Process regression using the scikit-learn library [20], and the Matern kernel ($\nu = 5/2$) remains fixed without optimization.

A. Simulation Results

The simulation environment is designed to emulate our subsequent real experiment in a towing tank, and is aimed at evaluating our algorithms quantitatively. The vehicle follows a straight-line trajectory ($z = 1.0$ m) as shown by the black line in Fig. 5(a) from $y = 7.5$ m to $y = -5$ m over 60 seconds. A random terrain is generated with average depth at $z = 1.5$ m (Fig. 5(b)). We assume a limited number of features on the terrain are trackable, 200 of which are in the field of view (Fig. 5(a)). Vehicle poses are uniformly sampled at intervals of 0.2s, and at each pose, sonar measurements are simulated. The sonar has a field of view of $r = [0\text{ m}, 3\text{ m}]$, $\theta = [-35°, 35°]$ and $\phi = [-15°, 15°]$, and Gaussian noise is added to range and bearing measurements $\sigma_r = 0.0025$ m, $\sigma_\theta = 0.01$ rad. We also introduce randomness into feature tracking across consecutive frames. Consider two observations of the same feature at step i and $i + 1$, i_p and $[i + 1]_q$, each being assigned to landmarks j_p and j_q respectively. We assume that a feature can be successfully tracked, i.e., it is assigned to the same label in the current frame as it is in the previous frame, with probability 0.95, and thus $P(j_p = j_q) = 0.95$.

The experiments are repeated for 50 independent trials to evaluate performance, and one example trial is visualized in Fig. 5. We analyze three types of error of the SLAM



(a) Top-down view of feature estimate

(b) GP terrain maps using feature points

Fig. 5: Simulation results of terrain reconstruction. From left to right: algorithm without terrain factors, with terrain factors, and ground truth. The trajectory estimate and ground truth are represented by green and black lines, respectively. Features and maps are colored according to depth (red represents higher terrain elevation).

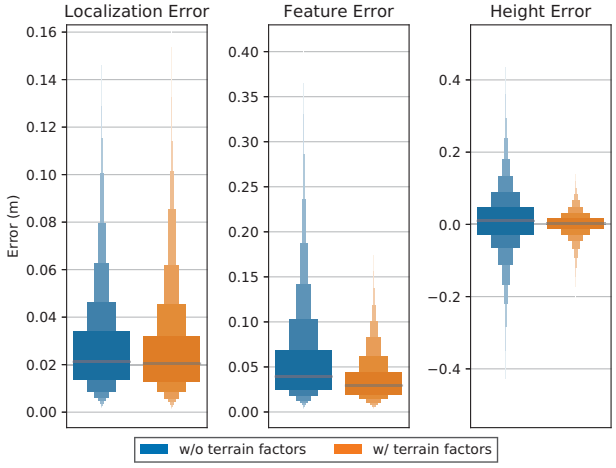
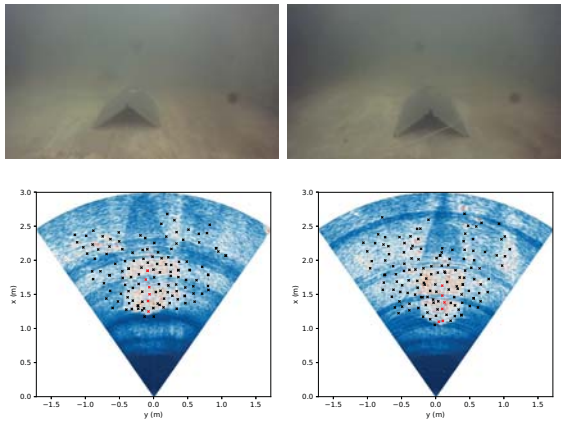


Fig. 6: Results from the simulation of Fig. 5. Box plots with quantiles are used for visualizing error distributions with outliers.



(a) Sample near location B

(b) Sample near location C

Fig. 7: Two representative camera and sonar images near location B/C (Fig. 8(a)) where one of pegboards is in the field of view. In the sonar images (colored by intensity), A-KAZE features are marked with $\times$, and red features highlight the peak heights of the pegboards, which are used in Table 1.

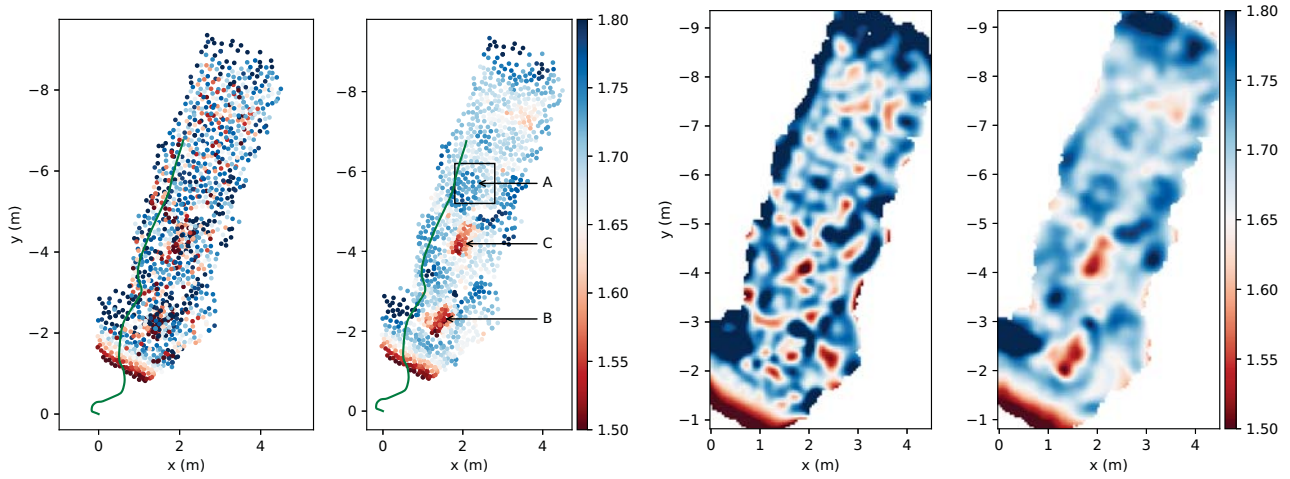
result. First, localization error is computed as the Euclidean distance between estimated vehicle position and ground truth position. Secondly, we compare the mapping error as the distance between estimated feature position and ground truth position. Thirdly, we perform Gaussian process regression using estimated feature points $(X^-, \mathbf{m})$ as training data to produce a height map as shown in Fig. 5(b); the height error is composed of the difference between prediction and ground truth at every location.

All results are presented in Fig. 6 using box plots to provide more information on the error distribution. It is clear that all types of error exhibit longer tails without adding terrain factors. GP regression in post-processing aids the terrain height estimate when feature error is small enough, however, it is not helpful when dealing with abnormalities. Overall though, smoothness constraints significantly reduce elevation error when applied in the form of terrain factors during graph optimization.

B. *Experimental Results*

Similar data was gathered using the BlueROV2, with additional sensors (Fig. 1(a)). IMU data from a VectorNav VN-100, body velocity from an RTI SeaPilot DVL (600 MHz), and depth from a Bar30 pressure sensor on the BlueROV2 were used for constructing odometry factors in GTSAM. We used the Oculus M750d multi-beam imaging sonar, operated in 1.2MHz mode with a field of view of maximum range 3 meters, horizontal aperture 70° and vertical aperture 12°. The sonar image updates at 10Hz and has range resolution of 0.005 m and an angular resolution of 0.0024 rad.

The experiment was conducted in the Stevens towing tank. We placed two pegboard "ramps" at the bottom of the tank. Illustrative representations of the pegboards are shown in Fig. 1(b). Each pegboard is 80 cm $\times$ 40 cm, and two pegboards form a ramp with height 20 cm. During the



(a) Top-down view of feature estimate

(b) GP terrain maps using feature points

Fig. 8: Experimental results in water tank. From left to right: algorithm without terrain factors, with terrain factors. See Fig. 5 for notations. The locations of two pegboards are marked by B/C, and a sample region of the tank floor is marked A. Pegboards are visible in GP terrain maps, but outliers are pervasive around pegboards without adding terrain constraints.

experiment, the vehicle is driven at a fixed depth of 1m and an approximate speed of 0.15m/s. The trajectory of the whole operation, which has a length of 8 meters over 1 minute, is shown in Fig. 8(a). The two pegboards are marked as B and C respectively in Fig. 8(a). The trajectory avoids regions directly above the pegboards to ensure successful DVL bottom tracking.

From the feature point cloud and predicted height map (Fig. 8), two protruding regions (B and C) are visible using the proposed terrain factors, whereas treating features independently produces more erroneous estimates. One reconstructed pegboard is visualized in Fig. 1d. However, terrain heights near the origin and the top boundary of the map gradually drift away from the ground truth. These regions are observed from a limited span of elevation angles, and as a consequence, those under-constrained features stay near the initial estimate with zero-elevation. Although terrain factors build pairwise connections between two feature nodes, they cannot prevent height from drifting due to degeneracy, provided that the observed terrain surface is smooth.

depth (m)	w/o terr. factors	w/ terr. factors
A	1.708 ± 0.084	1.711 ± 0.028
B	1.539 ± 0.059	1.542 ± 0.024
C	1.558 ± 0.057	1.549 ± 0.026

TABLE I: Depth with 1 standard deviation of three regions marked in Fig. 8(a) without/with terrain factors. The height of the ramp peak at B/C is calculated by the difference compared to ground A.

With regard to quantitative analysis, we only inspect the height of the ramp above the tank floor. Features that are likely to lie on the ramp peaks are manually selected in the sonar images as depicted in Fig. 7 using red $\times$ marks. The depth of the tank floor is estimated as the average height of features in region A. From the result in Table I, we can see that height estimates have lower variance when terrain factors are incorporated into the optimization, and the estimated height of the ramps is slightly closer to the actual geometry (0.2 m). However, both algorithms underestimate their true height, which may be caused by inaccurate calibration of factors including sensor displacement and speed of sound.

V. Conclusions and Discussion

We offer two improvements for underwater feature-based SLAM with a forward-looking imaging sonar, for terrain mapping. First, data association is performed via optical flow tracking, which is more robust to noise and the absence of elevation angle. Second, a degenerate system is partly solved by adding terrain constraints connecting feature pairs, constraining their height values to be similar. The effectiveness of terrain factors is validated in both simulation and experiment. Outliers among the resulting feature estimates are reduced, and thus GP terrain maps are more accurate.

Looking ahead at areas for improvement, better initialization estimates can potentially be provided using the linear triangulation proposed in [8]. In addition, the real-time viability of our SLAM framework is impeded by adding terrain factors, since the construction of a CLT requires full knowledge of a feature's 2D location. This also introduces a large number of additional constraints into the factor graph, making optimization computationally costly and challenging for real-time applications on embedded platforms. Although the offline SLAM solution in Fig. 8 requires about 7 seconds of computation on a laptop equipped with an Intel Core i7-4810MQ @ 2.80 Ghz $\times$ 8, the setup of our current experiment, without loop closures, is idealistic. Future work involves exploring its application at larger scales.

Acknowledgements

This research was supported in part by the National Science Foundation, grants IIS-1652064 and IIS-1723996, and by a grant from Schlumberger Technology Corporation.

REFERENCES

[1] D. Ribas, P. Ridao, J.D. Tardos, and J. Neira, "Underwater SLAM in manmade structured environments," *Journal of Field Robotics*, 25(1112), pp. 898-921, 2008.

[2] J. Wang, S. Bai, and B. Englot, "Underwater localization and 3D mapping of submerged structures with a single-beam scanning sonar," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 4898-4905, 2017.

[3] A. Mallios, P. Ridao, D. Ribas, M. Carreras, and R. Camilli, "Toward autonomous exploration in confined underwater environments," *Journal of Field Robotics*, 33(7), pp. 994-1012, 2016.

[4] H. Johannsson, M. Kaess, B. Englot, F. Hover, and J. Leonard, "Imaging sonar-aided navigation for autonomous underwater harbor surveillance." *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4396-4403, 2010.

[5] F.S. Hover, R.M. Eustice, A. Kim, B. Englot, H. Johannsson, M. Kaess and J.J. Leonard, "Advanced perception, navigation and planning for autonomous in-water ship hull inspection," *International Journal of Robotics Research*, 31(12), pp. 1445-1464, 2012.

[6] M.D. Aykin and S. Negahdaripour, "Forward-look 2-D sonar image formation and 3-D reconstruction," *Proceedings of the IEEE/MTS OCEANS Conference*, 2013.

[7] T.A. Huang and M. Kaess, "Towards acoustic structure from motion for imaging sonar," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 758-765, 2015.

[8] Y. Yang and G. Huang, "Acoustic-inertial underwater navigation," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 4927-4933, 2017.

[9] T.A. Huang, and M. Kaess, "Incremental data association for acoustic structure from motion," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1334-1341, 2016.

[10] E. Westman, A. Hinduja, and M. Kaess,"Feature-based SLAM for imaging sonar with under-constrained landmarks," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3629-3636, 2018.

[11] M. Kaess, A Ranganathan, and F. Dellaert, "iSAM: Incremental smoothing and mapping," *IEEE Transactions on Robotics*, 24(6), pp. 1365-1378, 2008.

[12] P. F. Alcantarilla, J. Nuevo, and A. Bartoli, "Fast explicit diffusion for accelerated features in nonlinear scale spaces," *Proceedings of the British Machine Vision Conference*, 2013.

[13] L. Jie, M. Kaess, R. M. Eustice, and M. Johnson-Roberson, "Pose-graph SLAM using forward-looking sonar," *IEEE Robotics and Automation Letters*, 3(3), pp. 2330-2337, 2018.

[14] B.D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674-679, 1981.

[15] S. Vasudevan, F. Ramos, E. Nettleton, and H. DurrantWhyte, "Gaussian process modeling of largescale terrain." *Journal of Field Robotics*, 26(10), pp. 812-840, 2009.

[16] D. Moore and S. J. Russell, "Gaussian process random fields," *Advances in Neural Information Processing Systems*, pp. 3357-3365, 2015.

[17] N. Carlevaris-Bianco, M. Kaess, and R.M. Eustice, "Generic node removal for factor-graph SLAM," *IEEE Transactions on Robotics*, 30(6), pp. 1371-1385, 2014.

[18] C.K. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE transactions on Information Theory*, 14(3), pp. 462-467, 1968.

[19] Georgia Tech Borg Lab, "GTSAM," `https://bitbucket.org/gtborg/gtsam/`

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.