Multi-Robot Ground Texture SLAM with Communication-Efficient Data Sharing

Kyle M. Hart[1,2], Yewei Huang[3], Brendan Englot[2]

***Abstract*— When operating in challenging sparse environments such as a ship's deck or an empty road, traditional visual simultaneous localization and mapping (SLAM) algorithms may struggle with the lack of outward facing salient features. We propose a distributed multi-robot SLAM system that exploits the ground texture beneath each robot as a source of features for navigation, and selectively shares this information with other robots to improve SLAM results without overwhelming communication channels. We show with initial experiments that this approach yields more accurate pose estimation than relying on single-robot ground texture SLAM for each robot.**

I. Introduction

Harsh environments often pose unique challenges for robotic systems conducting simultaneous localization and mapping (SLAM). In particular, robotic systems with visual sensors operating on the decks of aircraft carriers or large empty warehouses must handle a lack of salient outward features to use for navigation due to the flat, empty nature of the environment. This precludes the use of common visual SLAM systems. Additionally, ships at sea are moving reference frames, both from the ship traversing and from sway due to ocean movement. This violates the inertial reference frame assumptions made for most ground robot systems that use inertial measurement units. An inability to rely on these sensors makes navigation by shipboard robots particularly challenging. Fortunately, regardless of the lack of outward features or reference frame motion, the ground texture underneath the robot provides a useful source of features for navigation.

Our prior work [1] has established a single-robot visual SLAM system capable of using salient features found in images of the ground texture to reliably map and navigate. Given the system uses a downward-facing camera for perception, a single robot may struggle to map large-scale environments in a time-efficient manner, due to the limits of the field of view of the downward-facing camera.

To address this, we introduce a distributed multi-robot ground texture SLAM system. This system features a SLAM algorithm that processes images of the ground for salient features and shares this information with other robots to assist in their SLAM routines. These ground images are the only source of information for the system. Importantly, this information sharing is adapted to reduce bandwidth usage by first sharing compact image descriptor information and only subsequently sharing full information when requested. In this work, we describe this process and demonstrate its effectiveness on a collected representative dataset.

[1]Naval Air Warfare Center Aircraft Division (NAWCAD), Lakehurst, NJ, 08733, USA, kyle.m.hart2.civ@us.navy.mil

[2]Department of Mechanical Engineering, Stevens Institute of Technology, Hoboken, NJ, 07030, USA, benglot@stevens.edu

[3]Department of Computer Science, Dartmouth College, Hanover, NH, 03755, USA, yewei.huang@dartmouth.edu

II. Problem Description

The operating environment considered here consists of an approximately planar surface. Within it, K ground robots move through arbitrary trajectories, represented by a series of poses, $\mathbf{x}_{k,t}$, where k denotes which robot and t is the timestep of the pose. Since the robots drive on a flat surface, poses are defined in $SE(2)$. Robots have no *a priori* knowledge of each other's poses, but do know how many robots there are and are able to communicate with each other.

Each robot is equipped with a downward-facing monocular color camera. At each pose, the robot receives observation $\mathbf{z}_{k,t}$, a color image of the ground. The cameras are calibrated such that the intrinsic matrix, $\mathbf{K} \in \mathbb{R}^{3\times3}$, and distortion coefficients, $\mathbf{d}$ are known. Additionally, the 3D pose of the camera relative to the robot, $\mathbf{T}_{RC} \in \mathbb{R}^{4\times4}$, is known.

The goal here is to develop an approach that accurately estimates $\mathbf{x}_{k,t}$, for all robots, k, and all timesteps, t, using only the camera calibration information, $\mathbf{K}$ and $\mathbf{d}$; camera pose information, $\mathbf{T}_{RC}$; and relative pose measurements $\mathbf{z}_{k,t}$ between a current robot pose and historical robot poses ($t < t_{\text{now}}$), estimated from feature correspondences and geometric verification.

III. Proposed Approach

To address the challenges in the preceding section, we present a multi-robot ground texture SLAM system. It combines the ground texture loop closure pipeline developed in our previous ground texture work [1] with a modified version of the information sharing approach introduced in our previous works on underwater multi-robot SLAM [2] [3].

When a robot receives an image of the ground texture from a downward-facing camera, it will look for intra-robot loop closures as is typical in a single-robot SLAM system. Additionally, it also looks for loop closures against observation information previously provided by other robots. This is a two step process that first compares compact image-level descriptors, and only requests more information if a certain criterion is met. At the end of the process, this robot sends out its own image-level descriptor to enable other robots to do the same. A similar process occurs when receiving this information from other robots. All pose estimates and loop closures are handled using factor graphs. Fig. 1 shows the information flow, and each part is described in more detail below.

A. Image Processing

When an image is first received from the camera, the system processes it to identify any features of interest. This involves first rectifying the image based on the calibration parameters, $\mathbf{K}$ and $\mathbf{d}$. Then, the system detects and describes key features in the image using ORB [4]. Since the height of the camera is known, the features are also projected from pixel space into the ground plane, as measured from the robot's frame of reference. This is the same process outlined in our prior work [1]. Lastly, the system computes an image-wide descriptor using the visual bag of words approach described by Galvez-López and Tardos [5]. This descriptor represents the prevalence of each individual vocabulary "word" among all descriptors for all key features within the image. As this is a single descriptor for the whole image, it is a more compact piece of information that can be subsequently transmitted and used for inter-robot loop closure searches.

B. Single-Robot SLAM

After extracting features and descriptors from the current image, the system starts to estimate its pose using prior observations. First, it performs visual odometry using the previous observation. This process involves first matching features between the two images using the standard Lowe's Ratio Test [6] to compare distances in descriptor space. Then, matched features are used to estimate the transform between the two poses using robust M-Estimators. As features are already projected into the ground plane, the result is the transform experienced by the robot moving from the pose associated with the first observation to the pose associated with the second observation. The system then adds this sequential visual odometry factor into the factor graph.

The robot then searches its own prior observations for intra-robot non-sequential loop closures. It uses a similar process as conducted for visual odometry, however three threshold criteria must be met to be considered a loop closure, as in our prior work. These criteria are visual bag of words similarity scores, the number of feature matches between images, and a confidence score based on the covariance of the estimated transform. If all three are met, the system adds this loop closure factor to the factor graph.

C. Information Sharing

To aid other robots in the group, each robot sends relevant observation information to the others. However, to limit bandwidth impact, the information is broken down into parts with the larger information only sent when requested. At the end of each intra-robot loop closure search described in Sec. III-B, the robot sends out its updated pose information and the visual bag of words descriptor for each observation.

This is significantly less information than if the full details were sent. For comparison, the set of features for a given observation are $2 \times N$ floating point numbers, where N is the number of features in the image. N is tunable, but is often on the order of 1000. The associated set of descriptors consists of $N \times M$ 8-bit integers, where M is the size of the descriptor. For ORB, that is 32. In contrast, the visual bag of words descriptor is a set of pairs of one integer and one floating point number representing the ID of the vocabulary word and its associated weight, respectively. The exact size varies based on how many vocabulary words are in the descriptor, but is around 1000 in the experiments described in Sec. IV.

The system sends this information out for every observation. Then, it will only send out the full set of features and descriptors for an observation when requested by another robot. So if an observation is not a potential loop closure, the full information will never be sent.

D. Multi-Robot SLAM

In addition to searching for loop closures among its own observations, each robot also searches for loop closures among the observations received from other robots. The process is similar to the one used for intra-robot searches described in Sec. III-B. However, as highlighted in Sec. III-C, only the visual bag of words vector is initially available. Using this vector, the system calculates the visual bag of words similarity score, which is the first of the three loop closure thresholds. If the candidate meets the threshold, then the robot requests the full feature and descriptor information from the other robot. The rest of the loop closure process then proceeds as usual with successful candidates added to the factor graph.

When an inter-robot loop closure to a particular robot is identified for the first time, the system also merges the pose estimates from the other robot into its factor graph. As the robots have no prior knowledge of each other's relative poses, pose estimates are sent in the other robot's frame of reference. The robot uses its own factor graph to estimate the transform that projects these pose estimates into this robot's frame of reference. Then a series of partner robot trajectory factors are created using the transforms between sequential poses and added to the map. As new pose estimates are sent from the other robots, the system updates these factors and estimates. Once this merging occurs, a complete multi-robot factor graph emerges, as illustrated in Fig. 2.

IV. Experiment and Results

To test the proposed framework, we first collected a representative dataset, then emulated a multi-robot setup using this data and ROS [7].

To collect ground texture data, we configured a robot with a downward facing camera, as shown in Fig. 3. The camera is an Intel RealSense D435i that has been calibrated to obtain its pose relative to the robot, $\mathbf{T}_{RC}$, its intrinsic matrix, $\mathbf{K}$, and the distortion coefficients for image rectification, $\mathbf{d}$. We also placed markers on the robot so that we could record

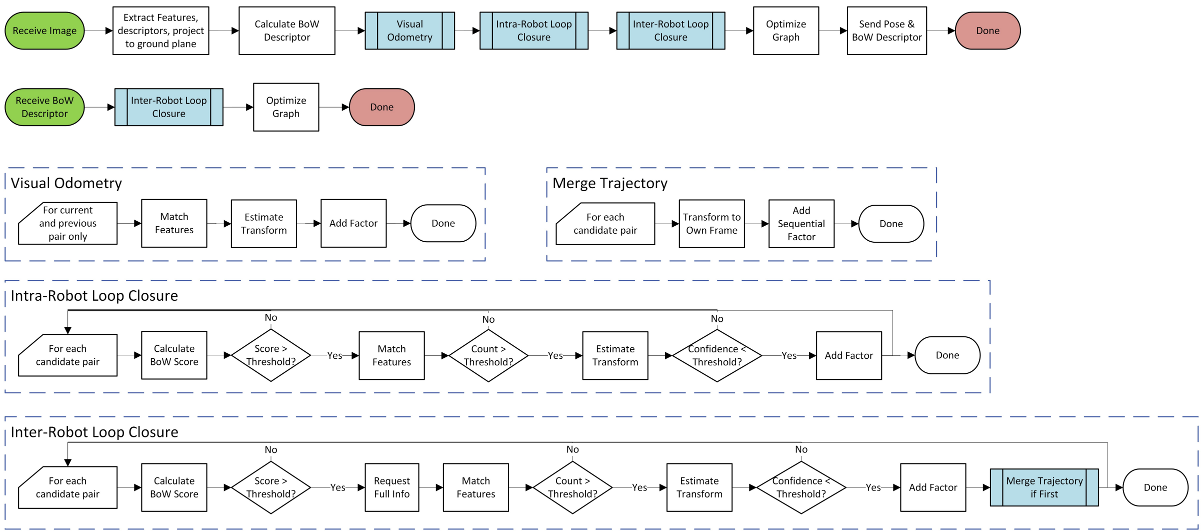

Fig. 1. An overview of the different processes within the proposed multi-robot SLAM system. *Top two lines:* processes start when the system receives either images from the robot's downward facing camera or compact visual bag of words (BoW) frame descriptors from other robots. The system then conducts various steps related to loop closures, odometry and trajectory merging depending on which process is followed. *Bottom three lines:* The steps for each sub-process that adds to the robot's factor graph and thus its pose estimates.

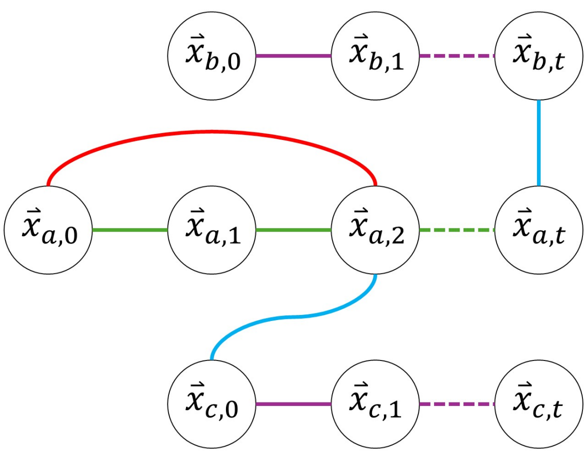

Fig. 2. An example multi-robot factor graph. Each circle, $\vec{x}_{k,t}$ represents the pose of one robot, k, at one observation, t. Lines represent factors. Green lines are sequential visual odometry factors. Red lines are non-sequential intra-robot loop closures. Blue lines are inter-robot loop closures. Purple lines are partner robot trajectory factors created based on pose estimates from other robots.

Fig. 3. Robot used for data collection, which features a downward facing camera and markers for recording pose using a motion capture system.

its ground truth pose during the experiment with a motion capture system placed around the experimental area. As we only have one robot, we used it to record each vehicle's session sequentially.

The experimental area is a carpeted area within the lab. For each session, the robot starts at an arbitrary point and the user manually drives it through the environment. The convention within the ground texture community is that datasets are comprised of a series of static images taken at various points along the trajectory, as illustrated in datasets introduced by Zhang et al. [8] and Schmid et al. [9]. Therefore, the user periodically stops the robot, collects an image of the ground with the downward facing camera, and records the robot's ground truth pose using the motion capture system. Fig. 4 shows an example image from the camera. This process repeats for each observation in the session. The robot is then placed at a new location and the process starts again to record data for the second robot's trajectory.

Fig. 5 shows the resulting ground truth trajectories for both sessions, as measured from a world frame established using the motion capture system. Path selection can be arbitrary, but to highlight the efficacy of the proposed system, we considered two constraints. The first constraint is that one of the robots (Robot B) follows a path with very little overlap. With a trajectory like this, using only a single-robot SLAM system produces an estimated trajectory with noticeable drift due to a lack of loop closures. The second constraint is sufficient overlap between the two robot's observations to provide multiple opportunities for inter-robot loop closure.

Fig. 4. One ground texture image collected by the robot during one of its sessions. Simultaneously, the motion capture system records the robot's ground truth pose when capturing this image.

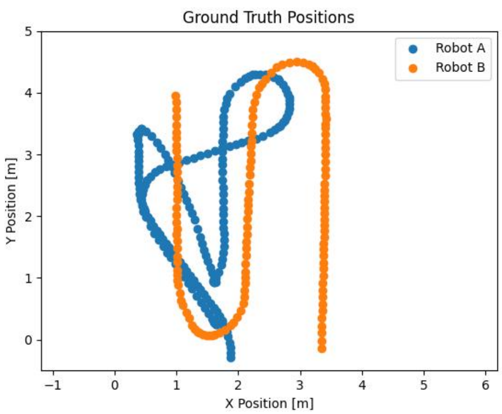

Fig. 5. The ground truth position for each session, as measured by the motion capture system. Each session represents the trajectory traveled by one robot in the multi-robot setup.

We then used this recorded data to emulate a multi-robot experiment using ROS [7]. The synthetic multi-robot experiment features two identical setups, one for each robot, running simultaneously. Each setup reads the camera images from file and sends them to an instance of the multi-robot SLAM solution described in Sec. III. Each instance then follows the described approach and outputs its best pose estimates for each observation.

Since we recorded the ground truth positions with the dataset, it is simple to calculate the errors for each robot's estimated pose at each observation point within the trajectory. For further comparison, we also ran a single-robot version of the experiment for each robot to observe how much improvement the proposed multi-robot method provides. Fig. 6 shows the pose estimates for each robot, with and without multi-robot information sharing, as well as ground truth. Additionally, Tab. I shows the root mean square error for the estimated pose of each robot, again with and without multi-robot information sharing.

As can be seen, the addition of information sharing between robots yields better performance overall. Robot B, which has no overlap in its own trajectory, benefits from additional information from Robot A. This correction is very

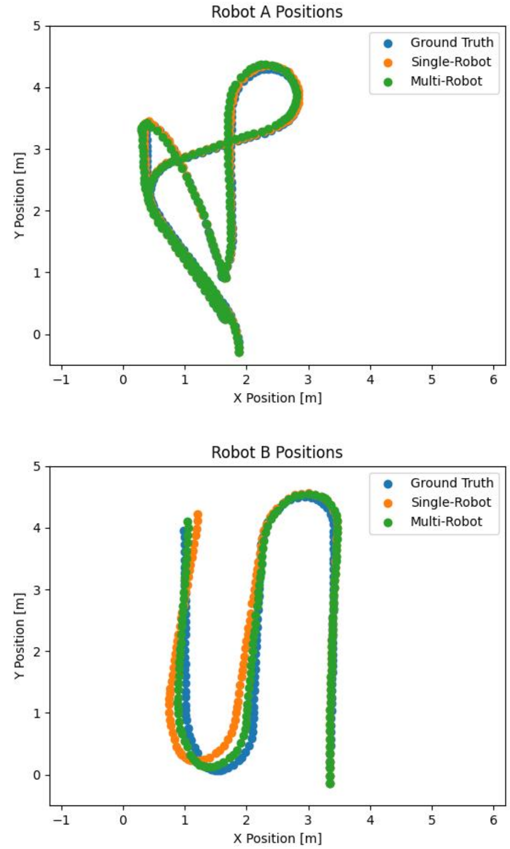

Fig. 6. Each robot's estimated position versus ground truth. *Single-Robot* refers to when the robots share no information with each other, while *Multi-Robot* refers to the methods described in this paper.

TABLE I

RMSE METRICS FOR THE ESTIMATED POSES WITH AND WITHOUT MULTI-ROBOT METHODOLOGY VERSUS GROUND TRUTH.

Robot	Method	Position $[m]$	Orientation $[rad]$
A	Single	0.031	0.007
A	Multi	0.043	0.017
B	Single	0.249	0.121
B	Multi	0.114	0.056

similar to the drift corrections seen in many SLAM systems. Additionally, while Robot A's overall accuracy decreases, the effect is modest and results are still accurate to within a few centimeters.

For further illustration of the efficacy of this solution, Fig. 7 shows one example of an inter-robot loop closure. Each image is the observation taken by one of the robots, with the identified features illustrated atop each image. Additionally, this figure shows the feature matching between the two images, indicating that the system is able to successfully identify feature correspondences across images, and the transformations relating them. By estimating the transformation between images, the system can add an inter-robot factor into the factor graph and estimate the relative transform between its own frame of reference and the other robots' frames of reference. Once this is possible, all shared information can easily be transformed into the robot's frame of reference and used to increase pose estimation accuracy.

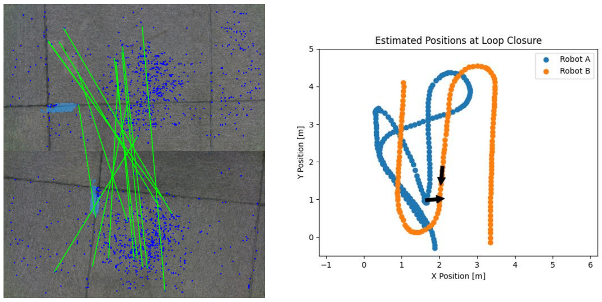

Fig. 7. *Left:* the ground texture images captured by each robot. Blue points are features identified by each robot. Green lines show those that the system matched between the two images, thus establishing an inter-robot loop closure. *Right:* The estimated poses of the robots when they make these observations.

V. Conclusion

Certain harsh environments lack outward visual features, making it challenging for robots with typical outward-looking visual SLAM systems to reliably map and navigate their environments. To address this, we propose a multi-robot SLAM system that uses the ground texture beneath the vehicles for navigation. This approach offers a consistent source of information, while enabling robots to share relevant information between each other. By using this method, ground robots can more reliably estimate their positions in the environment. Future work could consist of further reducing bandwidth usage through investigating more compact observation-level descriptors.

This work was supported in part by the Naval Air Warfare Center Aircraft Division's Naval Innovative Science and Engineering program, and in part by the Office of Naval Research (ONR), Grant N00014-24-1-2522.

References

[1] K. M. Hart, B. Englot, R. P. O'Shea, J. D. Kelly, and D. Martinez, "Monocular simultaneous localization and mapping using ground textures," in *IEEE International Conference on Robotics and Automation*, London, United Kingdom, May 2023.

[2] J. McConnell, Y. Huang, P. Szenher, I. Collado-Gonzalez, and B. Englot, "DRACo-SLAM: Distributed robust acoustic communication-efficient SLAM for imaging sonar equipped underwater robot teams," in *IEEE International Conference on Intelligent Robots and Systems*, Kyoto, Japan, Oct. 2022. [Online]. Available: http://dx.doi.org/10.1109/IROS47612.2022.9981822

[3] Y. Huang, J. McConnell, X. Lin, and B. Englot, "DRACo-SLAM2: Distributed robust acoustic communication-efficient SLAM for imaging sonar equipped underwater robot teams with object graph matching," in *IEEE International Conference on Intelligent Robots and Systems*, Hangzhou, China, Oct. 2025.

[4] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *IEEE International Conference on Computer Vision*, Barcelona, Spain, Nov. 2011.

[5] D. Galvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, Oct. 2012.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, p. 91–110, Nov. 2004.

[7] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, and W. Woodall, "Robot operating system 2: Design, architecture, and uses in the wild," *Science Robotics*, vol. 7, no. 66, p. eabm6074, May 2022.

[8] L. Zhang, A. Finkelstein, and S. Rusinkiewicz, "High-precision localization using ground texture," in *IEEE International Conference on Robotics and Automation*, Montreal, Canada, May 2019.

[9] J. F. Schmid, S. F. Simon, R. Radhakrishnan, S. Frintrop, and R. Mester, "HD ground - a database for ground texture based localization," in *IEEE International Conference on Robotics and Automation*, Philadelphia, USA, May 2022.