2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Mapping Pamir: Multi-Session Visual-Inertial SLAM and 3D Reconstruction of an Underwater Shipwreck

Michalis Chatzispyrou[a*], Luke Horgan[b*], Hyunkil Hwang[a*], Harish Sathishchandra[b*], Chinmay Burgul[a], Monika Roznere[c], Alberto Quattrini Li[d], Philippos Mordohai[b], Ioannis Rekleitis[a]

***Abstract*— This paper presents a framework for multi-session mapping of underwater environments utilizing an affordable action camera. The Visual-Inertial data are augmented by water depth recordings from a dive computer. SVIn2, an open-source VI-SLAM framework is utilized to generate a trajectory and a sparse reconstruction for each session. Utilizing the keyframes extracted from SVIn2, and the estimated camera poses, a Structure-from-Motion (SfM) framework – COLMAP – is employed for global optimization and produce a dense reconstruction of the target environment. The presence of calibration targets at fixed locations, when available, is used to estimate the coordinate transformation between different data collection sessions, thus transforming the different sessions into the same coordinate frame. The proposed pipeline is employed for the mapping of a shipwreck off the coast of Barbados. For the first time, both the exterior and the accessible interior parts of the wreck were mapped in two sessions, while a third session employed two cameras with different fields of view.**

I. Introduction

Accurate mapping of underwater structures is crucial for several domains, including underwater archaeology [1], [2], [3], [4], off-shore energy platform inspection [5], and environmental monitoring [6]. However, underwater vision has proven to be extremely challenging [7], [8], [9]. Additionally, the deployment of an autonomous underwater vehicle (AUV) is expensive and time consuming. Early work proposed the use of an inexpensive action camera [10] to deploy SVIn2 [11], a robust Visual-Inertial (VI) SLAM framework based on OKVIS [12]. The advantage of the specific camera (GoPro Hero9-Hero13) is that it encodes video at 30 fps (or higher) and Inertial Measurement Unit (IMU) data at 100 Hz in a single video file, thereby allowing VIO and VI-SLAM algorithms to be run on it. Please refer to the work of Joshi *et al*. [10] for more details and a comparison of different open-source VIO/VI-SLAM packages. From the formulation of the VI-SLAM problem, roll and pitch are observable, but the position and yaw orientation are not [13], [14]. Although in SVIn2 [11] a water-depth sensor was used, the GoPro action camera does not have one. However, since most divers carry a dive computer that records the depth in the water during the dive, this information is available for each deployment.

Fig. 1: GoPro setup deployed over the Pamir shipwreck, Barbados.

The challenging nature of underwater environments makes extensive deployments quite difficult; nitrogen loading and the danger of decompression sickness in conjunction with the amount of breathing gas a diver can carry limit the available time underwater. This requires breaking the data collection over multiple dives. Most VI-SLAM approaches generate a sparse reconstruction of the environment, while dense reconstructions often require a synchronized stereo camera setup [15]. On the other hand, Structure-from-Motion (SfM) frameworks that perform global optimization, such as COLMAP [16], are extremely slow for large numbers of images. Utilizing all the images of a video sequence, at over 100k frames per hour, is prohibitively expensive with COLMAP, while uniform sampling of the video results in tracking difficulties due to fast motions etc. In this work, we propose to utilize the *keyframes* generated by SVIn2 as input for bundle adjustment in COLMAP. Nearby keyframes have adequate overlap between them while maintaining sufficient spatial separation to provide large baselines for triangulation. Furthermore, the trajectory produced by SVIn2 is used as a pose prior in COLMAP, thus providing correct scale for the resulting reconstruction and injecting the IMU data into COLMAP. The Z-axis (depth) estimates from SVIn2 are corrected using the water depth measurements from a dive computer, in order to have access to the absolute values of the z-axis. In the case of multiple sessions where common fiducial markers (targets) remain in fixed locations, the coordinate transformations from the camera to the targets are utilized to bring the different trajectories into a common frame of reference.

We apply the proposed framework to mapping the Pamir shipwreck off the coast of Barbados. From more than two hours (167 minutes) of GoPro video (300,000+ frames) captured over three separate dives, our results demonstrate a successful mapping of the wreck's exterior and interior parts.

*The first four authors have contributed equally to this work and are listed in alphabetical order. [a]University of Delaware, Newark, DE, USA, {michalis,hkhwang,cmburgul,yiannisr}@udel.edu. [b]Stevens Institute of Technology, Hoboken, NJ, USA, {lhorgan,hsathish,pmordoha}@stevens.edu [c]Binghamton University, Binghamton, NY, USA mrozner1@binghamton.edu [d]Dartmouth College, Hanover, NH, USA alberto.quattrini.li@dartmouth.edu

This research has been supported in part by the National Science Foundation under grants 1943205, 2024541, 2024653, and 2024741. The authors are also grateful for equipment support by Halcyon Dive Systems, Teledyne FLIR LLC, and KELDAN GmbH lights.

This paper makes the following contributions: a) Introducing accurate z-axis absolute measurements from a dive computer into the VIO/VI-SLAM process. b) Introducing a new methodology for frame selection to be used by COLMAP [16], eliminating discontinuities (multiple submodels) produced by rapid field of view changes. c) Introducing correct scale to monocular bundle adjustment by utilizing the SVIn2 produced camera poses. Additionally, four VIO datasets (including water depth) from a shipwreck are released to the community.

II. Related Work

In this section, we review prior research in Structure-from-Motion (SfM), Simultaneous Localization and Mapping (SLAM), dense 3D reconstruction, and multi-session mapping in underwater environments. We distinguish between **sparse** and **dense** approaches. The former estimate camera parameters (if a calibration process has not been previously carried out), 6D camera poses, and a 3D point cloud (by reconstructing image features tracked/detected across multiple images). Dense methods receive camera poses, and sometimes the sparse point cloud, and compute depth for almost every pixel in the images. We also review pipelines covering both the sparse and dense aspects of 3D reconstruction.

State estimation underwater is challenging due to color saturation, floating particulates, and limited visibility [8]. Beall *et al*. [17] demonstrated accurate sparse 3D reconstruction of underwater structures, such as corals, from synchronized high definition videos collected using a wide baseline stereo rig. Vargas *et al*. [18] proposed robust visual SLAM underwater, leveraging acoustic, inertial, and altimeter/depth sensors in addition to cameras. Tightly coupled fusion of visual, inertial, and pressure sensors using forward and backward IMU preintegration is discussed in [19]. We use the approach of Rahman *et al*. [11] to obtain robust camera pose estimates by fusing visual and inertial information in real time. Joshi *et al*. [10] augmented a visual SLAM algorithm so that, after loop closures, the map is deformed to preserve the relative pose between each point and its attached keyframes.

Among the few authors who tackle dense underwater stereo, Queiroz-Neto *et al*. [20] modeled light propagation to overcome poor contrast and illumination. Wang *et al*. [15] presented a pipeline for dense 3D reconstruction based on SVIn2 for visual odometry, real-time stereo matching across frames from a stereo rig and depth map fusion. The synchronized stereo rig is crucial for obtaining consistent and metric depth estimates for each frame.

Recently, 3D modeling, especially for view synthesis, has been addressed by radiance fields, in implicit (Neural Radiance Field - NeRF) [21], [22] or explicit (Gaussian Splatting - GS) [23], [24] form. Their overall success has led to adoption of these techniques in underwater environments. It should be noted that camera poses are externally provided, in general. Mechanisms to model the water as the medium through which light travels in volumetric rendering have been integrated in NeRF [25], [26], [27], [28] or GS [29], [30], [31]. For instance, SeaThru-NeRF [25] develops a rendering model for scattering media that is able to learn the parameters of the medium along with those of the radiance. SeaSplat [30] extends Gaussian Splatting with the ability to render through a medium, and learns the parameters of the medium, backscatter, and attenuation. These methods have achieved impressive novel view synthesis, but geometrically their results are less accurate than conventional Multiple-View Stereo (MVS). To the best of our knowledge, all mentioned underwater radiance fields have been applied to small datasets of dozens of images.

Several complete systems addressing both sparse and dense 3D reconstruction have been deployed. Kalacska *et al*. [32] applied SfM and MVS on aerial and underwater imagery to study freshwater ecosystems. Efforts on coral reef monitoring have employed photogrammetric methods underwater [33], [34], [35]. In a recent publication, Zhong *et al*. [36] reviewed and compared state-of-the-art components for all stages of the photogrammetric pipeline on a few datasets, each containing a few hundred high-resolution images of coral reefs. In a previous research effort that shares many of our objectives, Mahon *et al*. [2] presented the design and deployment of a vision-based underwater mapping system to conduct an archaeological survey of the submerged ancient town of Pavlopetri. SLAM followed by dense multi-view reconstruction were used to generate photorealistic 3D models of a large site.

Given the difficulty of mapping an underwater area more than once, it is not surprising that prior underwater multi-session SLAM approaches required strong assumptions. Williams *et al*. [37] presented work on combining two multibeam surveys of a shipwreck. For both surveys, GPS and water depth measurements were available. Therefore their multi-session SLAM tool focused on fine-tuning the registration between the two dives with the help of SIFT feature matching, for global loop closure. Burguera and Bonin-Font [38] proposed a multi-session mapping framework with visual odometry, also based on SIFT feature matching. However, they initiated each dive at the same spot (at one calibration board) and stayed at a constant altitude. There are also underwater multi-session SLAM approaches for different sensor modalities [39] or for exploiting the geometrical characteristics of the target (e.g., ship hull) [40].

III. Proposed Framework

A. Experimental setup

Most underwater mapping operations are extremely complex and incur a high time and financial cost. In this work, we introduce a simplified approach that utilizes items commonly carried by scuba divers, such as action cameras and dive computers. The proposed framework provides an easy-to-use, inexpensive approach to 3D reconstruction, comprised of only off-the-shelf action cameras (GoPro™ Hero9 black, or later), a dive computer that can upload the depth profile (Shearwater™ Perdix2 AI is used), and, if the natural illumination is inadequate, additional underwater video lights (two Keldan Video 8XR Ambient 18000lm were deployed; see Fig. 1 for the two lights mounted on the frame with the

two GoPro cameras). In the experiments presented in this paper, two GoPro™ Hero9 black action cameras are rigidly mounted on an aluminum frame with handles, so that their fields of view overlap; see Fig. 1. Some experiments reported in Section IV were performed using unsynchronized videos from both cameras, while other experiments were performed using video from one camera traversing the scene twice.

The videos from the camera are converted into a ROS bagfile utilizing the framework proposed by Joshi *et al*. [10], [41]. This framework encodes the video (at 30 fps) and the IMU (100 Hz) datastream as ROS messages with synchronized timestamps. The resulting bagfile can be played back and used as input to a number of VIO/VI-SLAM packages; in our case SVIn2 [11] is used.

Datasets corresponding to three different sessions were recorded in Barbados, at the Pamir shipwreck. With an approximate length of 50m, Pamir rests at a depth of 8m at the bow, and 17m at the seafloor. The first session (data collected in 02/2024) involved two GoPro cameras mounted on a sensor rig with different fields of view; we refer to them as *LeftOnRig* and *RightOnRig*. The second and third session (01/2025) were recorded on the same day, but in two different dives. We refer to these sessions as *Pamir1*, and *Pamir2*. More details are provided in the experiments section.

B. Camera Calibration

While camera calibration above water is a simple matter [42], [43], [44], underwater settings present many challenges as the light passes through more media (water, acrylic enclosure, air, lens, etc.) [45], [46]. We enclose the GoPros in air-tight enclosures with flat-pane windows. It is known that flat-pane windows incur additional light refractions, making cameras incompatible with the traditional perspective projection model. While the Pinax model [47] accurately represents cameras behind flat-pane windows, it requires prior knowledge of the enclosure and water refraction index [48]. The pinhole camera model with radial-tangential distortion, which is more accessible to implement, can approximate these unique distortions – as long as the camera is placed as close to the window as possible. This is true in our case, and thus we calibrate the intrinsic parameters based on this model with the help of the calibration boards in Fig. 4.

C. VI-SLAM: SVIn2

To produce the keyframes and corresponding trajectory with correct scale, we utilize SVIn2 [11], a tightly-coupled keyframe-based SLAM system with loop closure, which is able to fuse data from IMU and single or multiple cameras and has been shown to perform well underwater [8].

Here we highlight the elements regarding keyframes used in our proposed approach. SVIn2 has a frontend that processes each incoming frame at the camera frame rate, performs local optimization for visual-inertial odometry, maintains a bounded window of *keyframes*, and marginalizes states and features which are never used again once they are out of the window, to limit the computation required by the optimization. Keyframes are selected from current frames and old keyframes in the window when the ratio of matched and new keypoints is small. The small ratio corresponds to a significant change in the current scene with some overlap with past scenes. The SVIn2 backend performs loop closure and generates a globally optimized trajectory that will be used in our approach: the selected keyframes are added in a pose graph, where each pose gets corrected based on the geometric constraints introduced by the matched descriptors among keyframes.

D. Absolute Water Depth Correction

The resulting trajectory from SVIn2 starts at an arbitrary location $[0, 0, 0]$. As such, while relative motions are correctly recorded, the absolute values are unobservable [13], [14]. On the other hand, a dive computer records accurate water depth measurements for the entirety of the dive; see Fig. 2(a) for the two unsynchronized signals from the 2024 LeftOnRig dataset. Though both signals have approximately the correct time and date, the two clocks often drift, resulting in time differences of several seconds. Furthermore, the dive computer records data at a much slower frequency than SVIn2; in our case Perdix2 AI records every 10 seconds (0.1 Hz) while SVIn2 produces data at different frequencies; 1.57 Hz and 1.89 Hz, for 2024 and 2.3 Hz, and 3.4 Hz for the 2025 datasets, depending on the selected keyframes.

For each dataset, a common time period is selected, and both time series are interpolated to 100 Hz, such that the two signals have the same frequency. The cross-correlation between the two signals (SVIn2 and Perdix) was used to estimate the time shift; see Fig. 2(b) where the two signals from the 2024 LeftOnRig dataset are converted to Perdix time, while still in different scales (Perdix blue, and SVIn2 red). The next step is linear regression to calculate the shift in values; see Fig. 2(c) for the scatter plot of all the data and the results of linear regression. Finally, the SVIn2 time series of the z-coordinate is transformed to the real depth; see Fig. 2(d) for the two signals converted in the same time and depth for the 2024 LeftOnRig dataset. The 2024 RightOnRig dataset shares the same depth profile and the plots are omitted. The same process is applied to the Pamir1 (see Fig. 3 left) and Pamir2 (see Fig. 3 right) datasets. For the three sessions, the GoPro time series was shifted 289.8 sec, 25.94 sec, and 29.91 sec respectively. The depth offsets were 7.74, 8.11, and 10.95 meters, respectively. These results are consistent with the experiments, as the GoPro started for the two first datasets (2024 Left and Right on Rig and Pamir1) at the bow of the wreck, which is shallower, while the third dataset (Pamir2) was started at the middle of the ship, which rests in deeper water.

E. Trajectory Correlation

To transform the two trajectories into a unified coordinate frame, we utilize a calibration target placed at the bow, where deployment starts. Fig. 4 shows the two targets on the Pamir wreck. The *aprilgrid*[1] package is employed to detect the

[1]https://github.com/powei-lin/aprilgrid

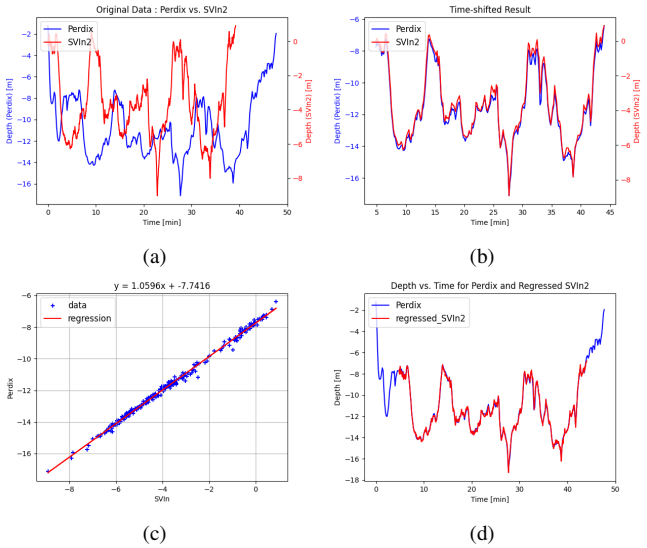


Fig. 2: (a) The SVIn2 z-coordinates (red) and the dive computer depth (blue) before synchronization for the 2024 dataset (LeftOnRig). (b) The two time series shifted to a common time but at different scales. (c) Linear regression between the SVIn2 and the Perdix data. (d) The depth estimates time-shifted and with adjusted depth; in red, the trajectory produced by SVIn2; in blue, the data from the dive computer, for the 2024 LeftOnRig dataset.

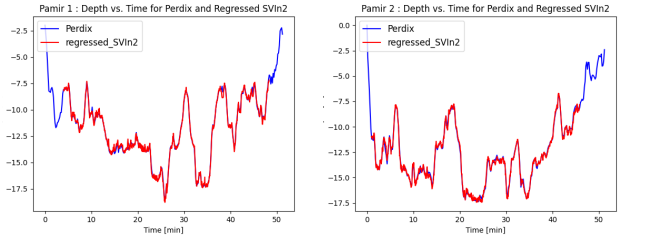


Fig. 3: The depth estimates for the other two datasets (Pamir1 and Pamir2), time-shifted and depth aligned, in red is the trajectory produced by SVIn2, in blue are the data from the dive computer.

calibration targets. For each observation, we compute the pose of the target in the global (world) frame. Let $T_{C_i}^{W}$ be the pose of camera i in the world frame W, and $T_{m}^{C_i}$ be the relative pose of the marker m with respect to camera i. The pose of the marker in the world frame, $T_{m,i}^{W}$, is derived as:

$$T_{m,i}^{W} = T_{C_i}^{W} T_{m}^{C_i} \tag{1}$$

First, we estimate the average pose of the calibration target. Because the calibration target consists of multiple individual tags, we perform pose estimation for each constituent tag and average their translations and orientations. Let $t_k \in \mathbb{R}^3$ be the translation of the k-th tag, and $\Theta_k = [\phi_k, \theta_k, \psi_k]^T$ be its orientation expressed as roll, pitch, and yaw angles. The average position $\bar{t}$ and orientation $\bar{\Theta}$ over N tags are calculated as:

$$\bar{t} = \frac{1}{N}\sum_{k=1}^{N} t_k, \quad \bar{\Theta} = \frac{1}{N}\sum_{k=1}^{N} \Theta_k \tag{2}$$

Despite the theoretical risks of gimbal lock associated with Euler angle averaging, the close proximity of our candidate poses ensured a stable global transformation. In cases of higher rotational uncertainty, mapping the rotations to a quaternion space would provide a more generalized solution.

Next, we filter outlier observations for each target across all camera frames. Let $t_{m,i}$ be the translation of target m estimated from camera i, and $\bar{t}_m$ be the mean translation of that target across all observations. The Euclidean distance D_i for each observation is calculated as:

$$D_i = \|t_{m,i} - \bar{t}_m\|_2 \tag{3}$$

We compute the mean μ_D and standard deviation σ_D of these distances. Any target estimate with a distance D_i falling outside one standard deviation from the mean is considered an outlier and is removed. Subsequently, we filter the remaining inliers based on their orientation estimates, which are typically more susceptible to noise. From the rotation matrix of each pose, we extract the principal axes angles $[\phi_i, \theta_i, \psi_i]^T$. We compute the mean and standard deviation for each angle component. An estimate is removed if any of its roll, pitch, or yaw angles fall outside one standard deviation from their corresponding mean. An updated, refined average pose $\bar{T}_m^W$ is then recalculated using only this strict inlier set.

To align two distinct trajectories (e.g., Trajectory A and Trajectory B), we leverage the refined poses of the same physical target observed in both coordinate systems. Let T_m^A and T_m^B be the estimated poses of the target in World Frame A and World Frame B, respectively. The transformation T_B^A that maps coordinates from Frame B into Frame A is computed as:

$$T_B^A = T_m^A (T_m^B)^{-1} \tag{4}$$

This calculation is performed independently for each of the two calibration targets, yielding a set of candidate transformations $\{T_{B,j}^A\}_{j=1}^M$ between the two trajectories. Finally, to establish a single global transformation to apply to the entirety of Trajectory B, we average these candidate transformations. We decouple the translation and rotation components of each candidate transformation $T_{B,j}^A$. The rotation matrices are converted into roll, pitch, and yaw angles, denoted as $\Theta_j = [\phi_j, \theta_j, \psi_j]^T$, and the translation components are extracted as t_j. The final global translation $\bar{t}_{\text{global}}$ and global orientation $\bar{\Theta}_{\text{global}}$ are then calculated as the arithmetic means of these components:

$$\bar{t}_{\text{global}} = \frac{1}{M}\sum_{j=1}^{M} t_j, \quad \bar{\Theta}_{\text{global}} = \frac{1}{M}\sum_{j=1}^{M} \Theta_j \tag{5}$$

These averaged translation and orientation values are recombined into a single global transformation matrix, which is applied to move the entirety of Trajectory B into the coordinate frame of Trajectory A.

F. Global Bundle Adjustment

To improve the geometric consistency of the camera poses and to prepare the data for dense reconstruction, we use the sparse COLMAP pipeline [16] on the *keyframes* of SVIn2.

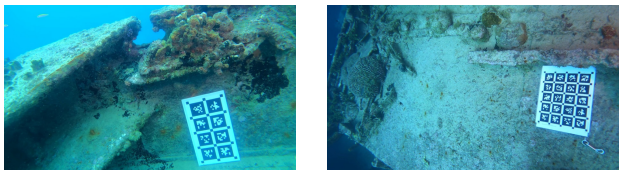


Fig. 4: The 4-by-2 and the 5-by-4 targets at Pamir wreck, which stayed at a fixed location for the Pamir1 and Pamir2 sessions.

Unlike SVIn2, COLMAP is a Structure-from-Motion system designed for collections of images, which have not necessarily been acquired sequentially or even by the same camera. This makes it possible to link images across trajectories by establishing feature correspondences between them, even if these trajectories do not contain common fiducial markers such as those in Section III-E above. Natural features observed in the images are sufficient.

COLMAP uses SIFT features [49] to detect pairwise pose constraints among the images, consolidates these constraints, and extracts the final camera poses by refining the poses generated by SVIn2. Specifically, we use the sparse modules of COLMAP as follows:

- Extract SIFT features from the images of two or more trajectories;
- Detect corresponding features across all image pairs;
- Initialize the poses of the cameras using poses from the aligned SVIn2 trajectories;
- Iteratively reduce reprojection error via bundle adjustment.

This approach benefits from the strengths of both the VIO and the SfM systems: it operates in metric space with the scale provided by visual-inertial odometry and the depth sensor; it avoids processing all frames with COLMAP, which would be prohibitively costly in compute and memory usage. Moreover, COLMAP receives frames that were selected as keyframes instead of a decimated sequence; it improves the overall precision of camera poses and reconstructed features via bundle adjustment. Empirically, we have observed that running COLMAP on the keyframes yielded breaks in the trajectory due to tracking difficulties. On the other hand, using the poses from SVIn2 is more robust due to the availability of the IMU signals. (Note that we are not able to use GLOMAP [50], a recent extension of COLMAP, because it does not support refinement of externally provided initial poses.)

G. Dense Reconstruction

COLMAP's dense reconstruction pipeline [51] consists of three steps: depth map estimation using patch-match stereo with per-pixel view selection, depth map fusion, and mesh generation. The first step estimates a depth per pixel using each input image with a camera pose as reference, and selecting target images for each pixel according to geometric criteria. The output is a set of depth maps, which are further improved by the subsequent fusion step. Finally, a triangular mesh can be obtained either by solving a Poisson problem [52], which computes an indicator function signifying which parts of space are inside or outside the surface.

IV. Experimental Results

A. Datasets

Three dives were performed at the Pamir shipwreck with durations of 55 (2024), 51 and 51 (2025) minutes respectively, while data were recorded for 39, 44, and 45 minutes respectively. In the first dive, the cameras were attached to a sensor rig; see Fig. 7 for a photo of the setup. For the duration of the second and third dives, two calibration targets (4-by-2 and 5-by-4) were fixed in separate locations. The application of SVIn2 to the three sessions resulted in $3{,}699$ and 4.464 keyframes for the left and right cameras of the 2024 session and $5{,}402$, and $8{,}285$ keyframes for the two 2025 sessions; a selection of $21{,}850$ keyframes out of a total of approximately $300{,}600$ frames. The datasets are available online[2].

B. Sparse reconstruction and Camera Trajectory Generation

SVIn2 produces a consistent sparse representation for each of the three sessions. As can be seen in Fig. 6, the wreck is reconstructed where the camera passed. However, coverage remains incomplete. Mapping a large, structure such as a wreck, requires several dive sessions.

a) Single Session (Pamir1): The camera positions from SVIn2 were used as "GPS" initialization points for COLMAP. As can be seen in Fig. 8 the camera positions from SVIn2 (in red) have been refined to the COLMAP final camera positions (in yellow) to satisfy the global optimization constraints. The resulting reconstruction is also presented in the same image.

b) Multi-Session Reconstruction: As can be seen in Fig. 9, utilizing the two sessions of 2025 resulted in more area covered, including several parts of the ship's interior. In Fig. 10 details of the reconstruction can be seen. Figures 10(a-b) are from the 2024 deployment and (c-e) from 2025. The bow and stern of the wreck are presented on the top row using the 2024 datasets. The second row presents data from the 2025 deployment, including an interior view of the engine room (Fig. 10(e)). In Fig. 10(d), the target is visible at the bow of the ship.

C. Dense Reconstruction

We applied screened Poisson surface reconstruction [52], implemented by COLMAP, on the 2025 data. Views of the resulting mesh can be seen in Fig. 11. Zoomed in details can be seen in Fig. 12. In particular, Fig. 12(a) presents the Poisson reconstruction of the area shown in Fig. 10(a) and (d); the target is clearly visible at the bow. In Fig. 12(b) the partial reconstruction of the interior of the engine room can be seen. Finally, Fig. 12(c) presents a reconstruction of the rudder and propeller of the ship.

[2]https://github.com/AutonomousFieldRoboticsLab/Pamir_Visual_Inertial_Dataset

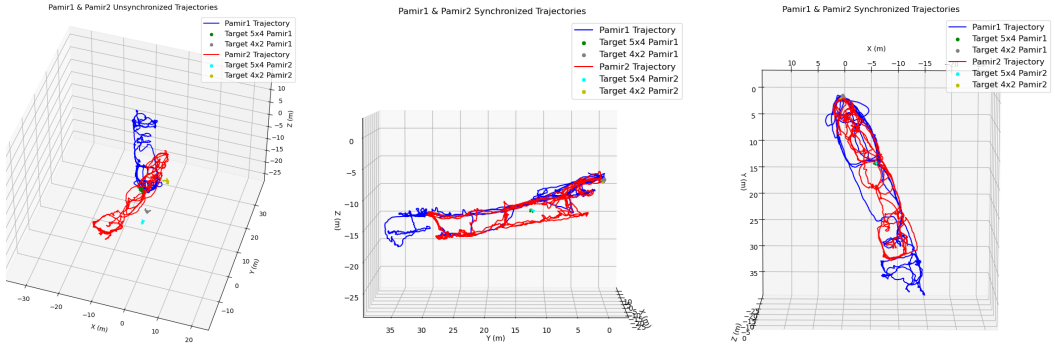


Fig. 5: The left figure presents the second and third trajectory as they were produced by SVIn2; please note that each trajectory starts at [0,0,0]. Both targets detected are also plotted for each session relative to their corresponding trajectories. The middle and right figures present the two trajectories brought to the same reference frame and the depth (z-axis) is adjusted based on the dive computer data. Both targets appear in the same place and the outline of the wreck is clearly visible.

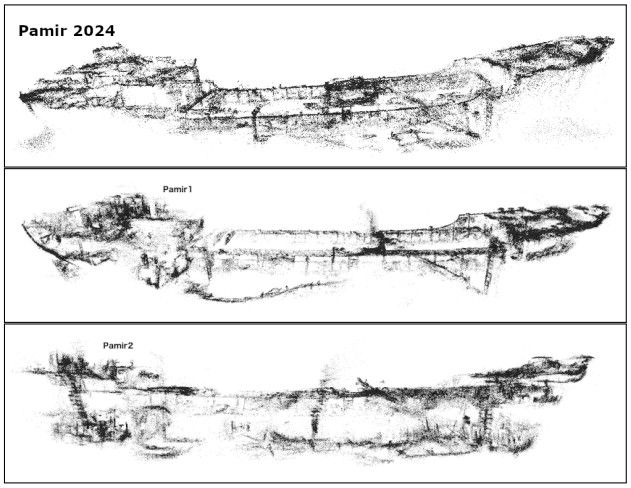


Fig. 6: The sparse reconstruction from SVIn2 for the three sessions. Different parts of the wreck were mapped, including the interior sections in the bow and the stern of the wreck.

V. Conclusions and Future Work

The proposed framework produced accurate dense reconstructions of a shipwreck at the correct water depth. Leveraging commonly available tools enables the reconstruction out of more than 300,000 frames by utilizing the keyframes produced by SVIn2. Even with the reduced number of images, global optimization (COLMAP) had to operate for several days in order to produce dense reconstruction. The proposed pipeline (converting video to ROS bagfile, running SVIn2, incorporating depth measurements from a dive computer, and finally feeding the produced keyframes and camera poses into COLMAP) is combined into a docker package that can be run on any platform; the code [3] is available as open source.

Future work will extend the multi-sensor fusion approach [53], [54], [55] to incorporate a number of asynchronous camera/IMU data streams in a common framework. As the cameras are not synchronized, the accelerometer data will be utilized to estimate the time-shift between the different cameras, while views of a calibration target will be used to estimate the extrinsics between different cameras. The possibility for a group of scientists equipped with action cameras and dive computers to accurately map a site of interest and produce models with correct scale will enable new scientific discoveries and advance the quality of environmental and infrastructure monitoring.

[3] https://github.com/AutonomousFieldRoboticsLab/Mapping_Pamir_Software

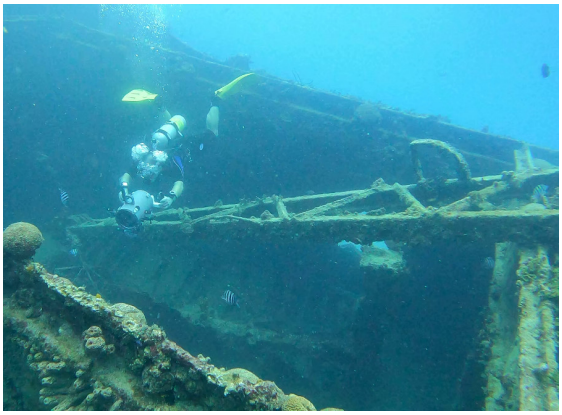


Fig. 7: 2024 deployment at the Pamir shipwreck, Barbados. The sensor rig with the two GoPros mounted to the left and right.

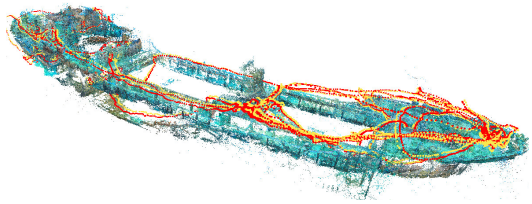


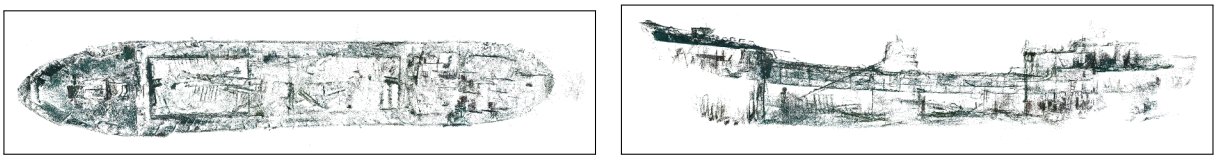
Fig. 8: The trajectory from SVIn2 (red) and the refined trajectory from COLMAP (yellow) are shown over the sparse reconstruction of the Pamir shipwreck, Barbados.

References

[1] N. Eriksson and J. Rönnby, "Mars (1564): the initial archaeological investigations of a great 16th-century swedish warship," *Int. J. Naut. Archaeol.*, 2017.

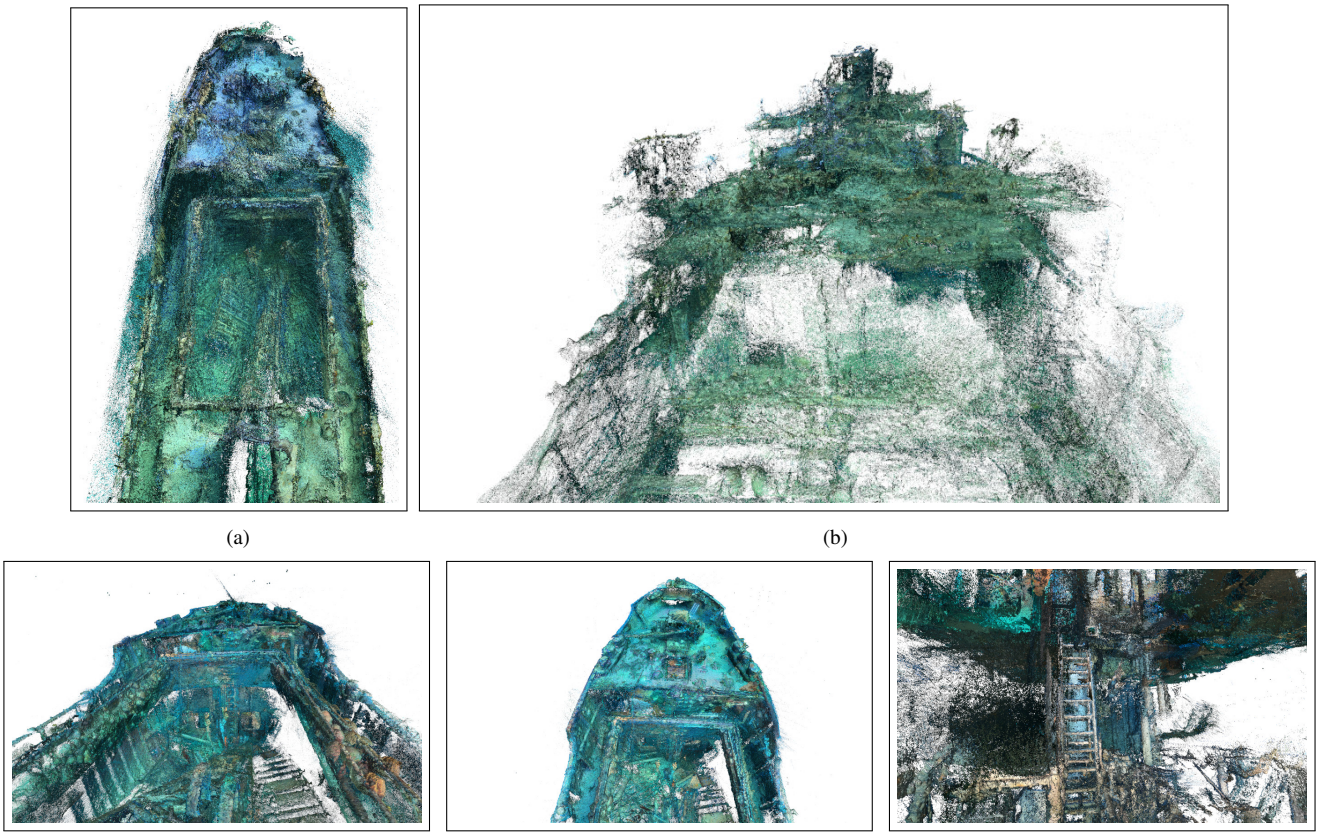
[2] I. Mahon, O. Pizarro, M. Johnson-Roberson, A. Friedman, S. B. Williams, and J. C. Henderson, "Reconstructing pavlopetri: Mapping the world's oldest submerged town using stereo-vision," in *ICRA*, 2011.

[3] S. Demesticha, D. Skarlatos, and A. Neophytou, "The 4th-century bc shipwreck at mazotos, cyprus: new techniques and methodologies in the 3d mapping of shipwreck excavations," *J. Field Archaeol.*, 2014.



(a) (b)

Fig. 9: Full sparse reconstruction of the Pamir shipwreck from the two 2025 sessions. (a) Top view. (b) Side view of the wreck.



(a) (b)

(c) (d) (e)

Fig. 10: Segments of the dense reconstruction from COLMAP presenting details of the mapped structure. (a) 2024: Top view showing the cargo hold crane and target (front right). (b) 2024: Stern of the wreck. (c) 2025: Perspective of the bow and cargo hold. (d) 2025: Bow view with visible target. (e) 2025: Interior view of the engine room.

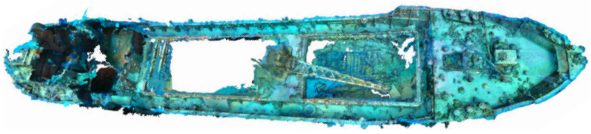


Fig. 11: Poisson mesh of selected frames from Pamir1.

[4] R. M. Eustice, H. Singh, J. J. Leonard, and M. R. Walter, "Visually mapping the RMS Titanic: Conservative covariance estimates for SLAM information filters," *Int. J. Robot. Res.*, 2006.

[5] D. Gillies, "Close range photogrammetry." *Photogram. Record*, 2015.

[6] C. G. David, N. Kohl, E. Casella, A. Rovere, P. Ballesteros, and T. Schlurmann, "Structure-from-motion on shallow reefs and beaches: potential and limitations of consumer-grade drones to reconstruct topography and bathymetry," *Coral Reefs*, 2021.

[7] A. Quattrini Li, A. Coskun, S. M. Doherty, S. Ghasemlou, A. S. Jagtap, M. Modasshir, S. Rahman, A. Singh, M. Xanthidis, J. M. O'Kane, and I. Rekleitis, "Experimental comparison of open source vision-based state estimation algorithms," in *ISER*, 2016.

[8] B. Joshi, S. Rahman, M. Kalaitzakis, B. Cain, J. Johnson, M. Xanthidis, N. Karapetyan, A. Hernandez, A. Quattrini Li, N. Vitzilaios, and I. Rekleitis, "Experimental Comparison of Open Source Visual-Inertial-Based State Estimation Algorithms in the Underwater Domain," in *IROS*, 2019.

[9] M. J. Islam, A. Quattrini Li, Y. A. Girdhar, and I. Rekleitis, "Computer vision applications in underwater robotics and oceanography," in *Computer Vision: Challenges, Trends, and Opportunities*, M. A. R. Ahad, U. Mahbub, M. Turk, and R. Hartley, Eds., 2024.

[10] B. Joshi, M. Xanthidis, S. Rahman, and I. Rekleitis, "High definition, inexpensive, underwater mapping," in *ICRA*, 2022.

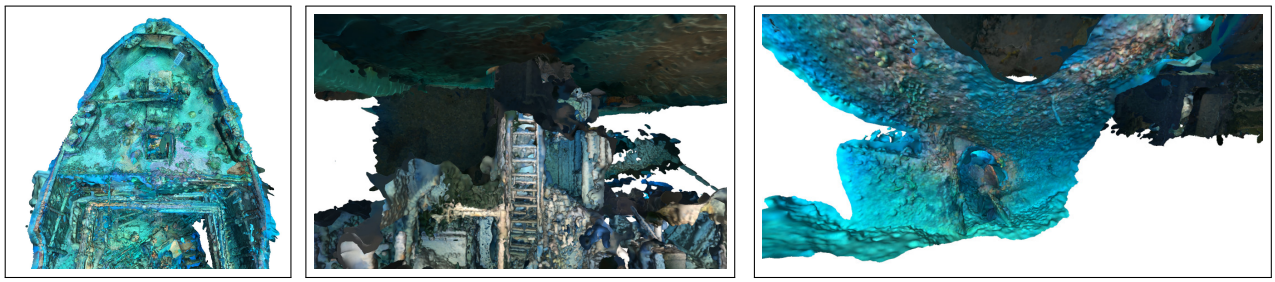
[11] S. Rahman, A. Quattrini Li, and I. Rekleitis, "SVIn2: A Multi-sensor Fusion-based Underwater SLAM System," *Int. J. Robot. Res.*, 2022.

[12] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual–inertial odometry using nonlinear optimization," *Int. J. Robot. Res.*, 2015.

[13] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Camera-imu-based localization: Observability analysis and consistency improvement," *Int. J. Robot. Res.*, 2014.

[14] Y. Yang and G. Huang, "Observability analysis of aided ins with heterogeneous features of points, lines, and planes," *IEEE Trans. Robot.*, 2019.

[15] W. Wang, B. Joshi, N. Burgdorfer, K. Batsos, A. Quattrini Li, P. Mordohai, and I. Rekleitis, "Real-Time Dense 3D Mapping of Underwater Environments," in *ICRA*, 2023.



(a) (b) (c)

Fig. 12: Details from the Poisson reconstruction. (a) Top view of the bow of the wreck, visible is the crane that has fallen into the cargo hold. (b) Inside the engine room, the ladder to the corridor above is visible. (c) The rudder and the propeller at the stern of the wreck.

[16] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *CVPR*, 2016.

[17] C. Beall, B. J. Lawrence, V. Ila, and F. Dellaert, "3D Reconstruction of Underwater Structures," in *IROS*, 2010.

[18] E. Vargas, R. Scona, J. S. Willners, T. Luczynski, Y. Cao, S. Wang, and Y. R. Petillot, "Robust underwater visual SLAM fusing acoustic sensing," in *ICRA*, 2021.

[19] C. Hu, S. Zhu, Y. Liang, and W. Song, "Tightly-Coupled Visual-Inertial-Pressure Fusion Using Forward and Backward IMU Preintegration," *IEEE Trans. Robot. Autom.*, 2022.

[20] J. Queiroz-Neto, R. Carceroni, W. Barros, and M. Campos, "Underwater Stereo," in *Brazilian Symposium on Computer Graphics and Image Processing*, 2004.

[21] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *ECCV*, 2020.

[22] J. Luo, T. Huang, W. Wang, and W. Feng, "A Review of Recent Advances in 3D Gaussian Splatting for Optimization and Reconstruction," *Image Vis. Comput.*, 2024.

[23] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering." *ACM Trans. Graph.*, 2023.

[24] B. Fei, J. Xu, R. Zhang, Q. Zhou, W. Yang, and Y. He, "3D Gaussian Splatting as New Era: A Survey," *IEEE Trans. Vision Compututer Graphics*, 2024.

[25] D. Levy, A. Peleg, N. Pearl, D. Rosenbaum, D. Akkaynak, S. Korman, and T. Treibitz, "Seathru-nerf: Neural radiance fields in scattering media," in *CVPR*, 2023.

[26] A. V. Sethuraman, M. S. Ramanagopal, and K. A. Skinner, "Water-NeRF: Neural Radiance Fields for Underwater Scenes," in *OCEANS*, 2023.

[27] T. Zhang and M. Johnson-Roberson, "Beyond NeRF Underwater: Learning Neural Reflectance Fields for True Color Correction of Marine Imagery," *IEEE Trans. Robot. Autom.*, 2023.

[28] Y. Tang, C. Zhu, R. Wan, C. Xu, and B. Shi, "Neural underwater scene representation," in *CVPR*, 2024.

[29] S. Liu, J. Lu, Z. Gu, J. Li, and Y. Deng, "Aquatic-GS: A Hybrid 3D Representation for Underwater Scenes," *arXiv preprint arXiv:2411.00239*, 2024.

[30] D. Yang, J. J. Leonard, and Y. Girdhar, "Seasplat: Representing underwater scenes with 3d Gaussian splatting and a physically grounded image formation model," in *ICRA*, 2025.

[31] T. Zhang, W. Zhi, B. Meyers, N. Durrant, K. Huang, J. Mangelson, C. Barbalata, and M. Johnson-Roberson, "RecGS: Removing Water Caustic with Recurrent Gaussian Splatting," *IEEE Trans. Robot. Autom.*, 2025.

[32] M. Kalacska, O. Lucanus, L. Sousa, T. Vieira, and J. P. Arroyo-Mora, "Freshwater fish habitat complexity mapping using above and underwater structure-from-motion photogrammetry," *Remote Sens.*, 2018.

[33] T. Guo, A. Capra, M. Troyer, A. Grün, A. J. Brooks, J. L. Hench, R. J. Schmitt, S. J. Holbrook, and M. Dubbini, "Accuracy assessment of underwater photogrammetric three dimensional modelling for coral reefs," *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, 2016.

[34] I. D. Lange and C. T. Perry, "A quick, easy and non-invasive method to quantify coral growth rates using photogrammetry and 3d model comparisons," *Methods Ecol. Evol.*, 2020.

[35] J. Zhong, M. Li, H. Zhang, and J. Qin, "Fine-grained 3d modeling and semantic mapping of coral reefs using photogrammetric computer vision and machine learning," *Sensors*, 2023.

[36] J. Zhong, M. Li, A. Gruen, K. Schindler, X. Liao, and Q. Guo, "Cutting-edge 3d reconstruction solutions for underwater coral reef images: A review and comparison," *ISPRS J. Photogramm. Remote Sens.*, vol. 230, pp. 779–803, 2025.

[37] S. B. Williams, O. Pizarro, and B. Foley, "Return to antikythera: Multi-session SLAM based AUV mapping of a first century bc wreck site," in *FSR*, 2016.

[38] A. Burguera Burguera and F. Bonin-Font, "A trajectory-based approach to multi-session underwater visual slam using global image signatures," *J. Mar. Sci. Eng.*, 2019.

[39] H. Jang, S. Yoon, and A. Kim, "Multi-session underwater pose-graph slam using inter-session opti-acoustic two-view factor," in *ICRA*, 2021.

[40] P. Ozog, N. Carlevaris-Bianco, A. Kim, and R. M. Eustice, "Long-term mapping techniques for ship hull inspection and surveillance using an autonomous underwater vehicle," *J. Field Robot.*, 2016.

[41] B. Joshi. (2022) gopro_ros: Software for converting a GoPro video into a ROS 1 bagfile. [Online]. Available: https://github.com/AutonomousFieldRoboticsLab/gopro_ros

[42] T. A. Clarke and J. G. Fryer, "The development of camera calibration methods and models," *Photogramm. Rec.*, 1998.

[43] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2000.

[44] W. Qi, F. Li, and L. Zhenzhong, "Review on camera calibration," in *Chin. Control Decis. Conf.*, 2010.

[45] Y.-H. Kwpron and J. B. Casebolt, "Effects of light refraction on the accuracy of camera calibration and reconstruction in underwater motion analysis," *Sports Biomech.*, 2006.

[46] M. Roznere, A. K. Pediredla, S. E. Lensgraf, Y. Girdhar, and A. Quattrini Li, "Underwater Dome-Port Camera Calibration: Modeling of Refraction and Offset through N-Sphere Camera Model," in *ICRA*, 2024.

[47] T. Łuczyński, M. Pfingsthorn, and A. Birk, "The pinax-model for accurate and efficient refraction correction of underwater cameras in flat-pane housings," *Ocean Eng.*, 2017.

[48] M. Singh, M. Dharmadhikari, and K. Alexis, "An online self-calibrating refractive camera model with application to underwater odometry," in *ICRA*, 2024.

[49] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, 2004.

[50] L. Pan, D. Baráth, M. Pollefeys, and J. L. Schönberger, "Global Structure-from-Motion Revisited," in *ECCV*, 2024.

[51] J. L. Schönberger, E. Zheng, J.-M. Frahm, and M. Pollefeys, "Pixelwise View Selection for Unstructured Multi-View Stereo," in *ECCV*, 2016.

[52] M. Kazhdan and H. Hoppe, "Screened Poisson Surface Reconstruction," *ACM Trans. Graph.*, 2013.

[53] K. Eckenhoff, P. Geneva, and G. Huang, "MIMC-VINS: A versatile and resilient multi-imu multi-camera visual-inertial navigation system," *IEEE Trans. Robot.*, 2021.

[54] Y. Yang, P. Geneva, and G. Huang, "Multi-visual-inertial system: Analysis, calibration, and estimation," *Int. J. Robot. Res.*, 2024.

[55] M. Zhang, X. Xu, Y. Chen, and M. Li, "A lightweight and accurate localization algorithm using multiple inertial measurement units," *IEEE Trans. Robot. Autom.*, 2020.