IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 22, 2021, accepted June 22, 2021, date of publication June 24, 2021, date of current version July 7, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3092180

Extrinsic Visual–Inertial Calibration for Motion Distortion Correction of Underwater 3D Scans

MIGUEL CASTILLÓN, ROGER PI, NARCÍS PALOMERAS, AND PERE RIDAO, (Member, IEEE)
Computer Vision and Robotics Research Institute (VICOROB), University of Girona, 17003 Girona, Spain

Corresponding author: Miguel Castillón (miguel.castillon@udg.edu)

This work was supported in part by the GIRONA1000 project under Grant DPI2017-86372-C3-2-R, in part by the Spanish Ministry of Science, in part by the EUMarineRobots (EUMR) project under Grant H2020-INFRAIA-2017-1-twostage-731103, in part by the European Commission, in part by the University of Girona under Doctoral Grant IFUdG2019, and in part by the Secretaria d'Universitats i Recerca del Departament d'Economia i Coneixement de la Generalitat de Catalunya under Grant 2019FI_B_00812.

ABSTRACT Underwater 3D laser scanners are an essential type of sensor used by unmanned underwater vehicles (UUVs) for operations such as navigation, inspection, and object recognition and manipulation. Scanners that acquire 3D data by sweeping a laser plane across the scene can provide very high lateral resolution. However, their data may suffer from rolling shutter effect if the change of pose of the robot with respect to the scanned target during the sweep is not negligible. In order to compensate for motion-related distortions without the need for point cloud postprocessing, the 6-DoF pose at which the scanner acquires each line needs to be accurately known. In the underwater domain, autonomous vehicles are often equipped with a high-end inertial navigation system (INS) that provides reliable navigation data. Nonetheless, the relative pose of the 3D scanner with respect to the inertial reference frame of the robot is not usually known a priori. Therefore, this paper uses an ego-motion-based calibration algorithm to calibrate the extrinsic parameters of the visual-inertial sensor pair. Simulations are performed to quantify how miscalibration affects motion-related distortion. The method is also evaluated experimentally in laboratory conditions.

INDEX TERMS 3D sensing, underwater robotics, visual-inertial calibration, odometry-based mapping.

I. INTRODUCTION

Unmanned underwater vehicles (UUVs) are being increasingly used in industry out of safety and cost reasons. In particular, autonomous underwater vehicles (AUVs) are already performing tasks like inspection [1], object recognition [2], manipulation [3], or navigation [4]. Sensing their surroundings is essential for them to successfully carry out their tasks. Therefore, they are usually equipped with some type of 3D sensor, which are mainly based either on acoustic (SONAR) or light signals. Optical sensors are further divided into passive (stereo vision, structure from motion (SfM)) or active (LiDAR). The main advantage of active optical sensors is that their lateral resolution and refresh rate are much higher than acoustic [5]. Actively projecting structured light makes them suitable to work in featureless environments. Their relatively short range is usually enough for intervention tasks, since the robot needs to get close to the target.

Some underwater 3D scanners illuminate the whole scene at once with a certain spatial pattern [6]–[9] and retrieve 3D information of the whole field of view (FoV) at the same time. They can be considered global shutter sensors because their acquisition time for the whole scene is extremely short and are therefore suitable to scan scenes in which high dynamics are present. However, they can only provide limited resolution [9].

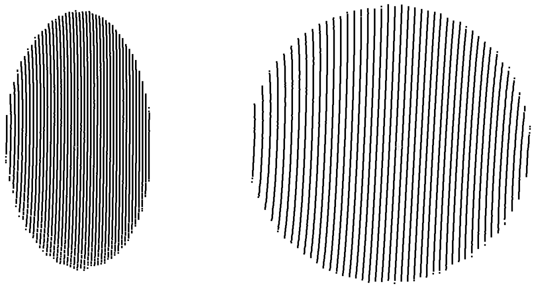


FIGURE 1. Relative motion between the scanner and the object produces distortions in the resulting point cloud. An example scanning a spherical object of radius 100 mm is shown in the left image. This distortion can be compensated by compounding each scanned line with the corresponding robot pose. The result is shown in the right image.

Another popular approach are laser line scanners (LLSs) [10]–[15]. These 3D scanners acquire the scene by sweep-

The associate editor coordinating the review of this manuscript and approving it for publication was Jiajia Jiang.

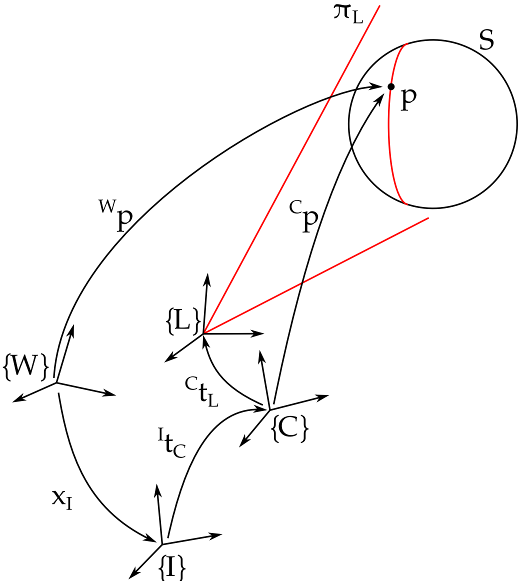


FIGURE 2. Geometrical scheme of the approach used for motion distortion compensation. The goal of the approach is referencing each scanned point p with respect to a fixed reference frame $\{W\}$. Please note that the INS pose ${}^{W}x_I$ is time-dependant (it changes as the robot moves) but the transformations between INS, camera and laser (${}^{I}t_C$ and ${}^{C}t_L$, respectively) are fixed. p represents an arbitrary point at the intersection between the laser plane π_L and the target surface S.

ing a laser plane. One of their main advantages is their high point cloud density. They typically make use of collimated laser light sources with a wavelength ranging between the green and blue spectrum. This choice of wavelengths responds to the high attenuation of light sources at other frequencies when transmitted in water [16]. For example, IR-based sensors are only capable of scanning at very short ranges [17]–[19]. A challenge for LLSs is that it takes them a certain amount of time to sweep the laser plane across their entire FoV. During this time, the robot's pose may have changed significantly, which would introduce motion-related distortions in the outcoming point cloud (rolling shutter effect [20], see figure 1). This problem is especially present in inspection and intervention missions where high resolution is desired. In these cases, the sensor needs to scan a higher number of lines, resulting in a longer scanning time. If no compensation for the motion distortion is performed, the accuracy of the resulting point cloud is severely diminished even for low speeds of the robot. Usually, these tasks only allow distortion levels in the order of mm or a few cm.

The approach to compensate for motion distortion followed in this paper is based on referencing each line scanned by the LLS to its corresponding temporally unique frame. This approach is shown in figure 2. This way, rather than considering only one frame for all the lines in one scan, there are as many frames as scanned lines. In order to achieve a highly accurate point cloud, the 6-DoF pose of the scanner's camera $\{C\}$ at the time of acquiring each line needs to be precisely known. This is done in this paper by composing the corresponding pose of the INS ${}^{W}\boldsymbol{x}_I$ with the relative transformation between the INS and the laser scanner ${}^{I}\boldsymbol{t}_C$.

The LLS triangulates the position of the points that lie at the intersection between the laser plane π_L and the target surface S. Each of these points $\boldsymbol{p}$ complies with:

$$\boldsymbol{p} \in \pi_L \cap S \tag{1}$$

The LLS returns the 3D pose of each of these points with respect to the camera frame ${}^{C}\boldsymbol{p}$. They can be then expressed in the world reference frame using the following composition (see appendix):

$${}^{W}\boldsymbol{p} = {}^{W}\boldsymbol{x}_I \oplus {}^{I}\boldsymbol{t}_C \oplus {}^{C}\boldsymbol{p} \tag{2}$$

The transformation ${}^{I}\boldsymbol{t}_C$ is not usually known a priori, and therefore an ego-motion-based extrinsic calibration is previously performed. It can be deduced that a fine calibration is paramount to achieving an accurate point cloud.

It should be pointed out that the proposed approach makes a number of assumptions: (i) the accumulated drift of the INS data in each scan is negligible, (ii) there is a good synchronization between INS and laser data, and (iii) there is a known marker in the scene such as a checkerboard or an ArUco [21], [22] for calibration. These assumptions are realistic because of the high-end INS and LLS available at the lab, as will be explained in section III, and because the calibration will be performed in controlled laboratory conditions.

The goal of this paper is two-fold: First, a robust calibration algorithm is developed and fed it with sufficient data to achieve an accurate result. Second, this result is applied to compensate for the motion distortion of each scan.

The remaining of the paper is structured as follows. First, the relevant state of the art is reviewed in section II. Later, the experimental set-up is described in section III. Then, the ego-motion-based calibration algorithm used in this paper is explained in section IV. The simulated and experimental results are presented in sections V and VI, respectively. Finally, the drawn conclusions are summarized in section VII.

II. RELATED WORK

This section reviews the relevant literature on topics related to this paper. First, underwater 3D laser scanning is studied in section II-A. Then, extrinsic calibration of a visual-inertial sensor pair is analyzed in section II-B.

A. UNDERWATER 3D LASER LINE SCANNING

Underwater 3D scanners are an essential type of sensor used to acquire the geometrical shape of obstacles or objects of interest. Their different working principles were reviewed and compared in a previous work [16]. This section will focus on LLSs.

In underwater metrology tasks, the scanner is usually mounted on a static tripod with a rotational head [23]. In this case, the scanner acquires different point clouds from different viewpoints, which are then registered together with dedicated software like Leica's [24].

However, underwater LLSs are increasingly used by UUVs when performing a wide variety of dynamic tasks, including navigation [25], [26] or manipulation [27]. When mounted on a moving platform, 3D laser scanners need to account for the relative motion of the scanner with respect to the target in order to achieve a consistent point cloud. Several approaches can be found in the literature to deal with this problem.

A direct approach is embedding an inertial sensor such as an inertial measurement unit (IMU) [28], [29] or an INS [30]–[32] with the laser scanner. On the one hand, IMUs are relatively small and cheap sensors but their measurements are drift-prone. A possible solution to counteract this drift is using GPS [28]. However, since GPS signal is rapidly attenuated in water, it can only be used on surface. On the other hand, INSs can accurately measure displacements while accumulating a drift of down to 0.01% of the travelled distance in optimal conditions [33]. However, their size is typically a diameter of more than 20 cm and a weight in water of 10 kg. This is not usually a problem when the scanner is mounted (along with many other sensors) in a work-class ROV. Nonetheless, it becomes problematic if the scanning task is to be performed with a smaller AUV such as Girona500 [34]. In our approach, we try to take advantage of the high-end performance of the INS integrated in an AUV so that the size of the scanner need not be increased.

A possible approach to integrate the measurements of the scanner with the navigation system of the robot is placing the scanner in a specific position and orientation that makes it easier to measure using the CAD models of the vehicle. For example, in [35] the scanner is set up at the front of the remotely operated vehicle (ROV) and looking down. Nonetheless, due to the errors between the measured and the actual transformation from the scanner to the inertial frame of the robot, the authors in [35] are aware that they should further counteract ‘‘the short term vehicle motion that introduces errors across sequential laser images’’. Therefore, in our approach we aim at accurately calibrating this transform in order to reduce these errors for any arbitrary camera set-up.

Yet another approach to deal with motion distortion is minimizing the robot speed while making a scan. Following this idea, the robot moves slowly around the inspected structure. Because of its low speed, the displacement of the scanner during each scan can be assumed as negligible and the scans can be considered as rigid. Then, consecutive point clouds can be registered using iterative closest point (ICP) and the map is created [1]. In our approach, however, we would like to enable the robot to move at its normal operational speed while scanning and not fully rely on post-processing algorithms.

In summary, the goal of our approach when compared to the state of the art is correcting motion distortion while (i) limiting the size of the scanner, (ii) allowing to mount the scanner anywhere on the robot, (iii) allowing the robot to move at normal operational speeds, and (iv) limiting the need for post-processing software.

An essential step of our approach is achieving an accurate calibration between the camera and the inertial sensor frames. Different approaches used to calibrate visual-inertial sensor pairs inside and outside water can be found in the literature. The main ones are reviewed in section II-B.

B. EXTRINSIC CALIBRATION OF A VISUAL – INERTIAL SENSOR PAIR

Extrinsic calibration between two reference frames refers to recovering the 6-DoF transform that relates both frames. Extrinsic calibration between two or more rigidly-mounted sensors is a very relevant topic in autonomous robotics, since it allows to accurately fuse measurements coming from different sensors. In current robotic platforms above water, the sensors to be calibrated are usually cameras, navigation sensors and time of flight (ToF) LiDARs. Typical sensor pairs are camera vs. navigation or LiDAR vs. camera. The different approaches to solve this calibration are typically divided in two parts: (i) first, the front-end extracts incremental or absolute displacements from sensor readings (see section II-B1); and (ii) second, these displacements are fed to the minimization algorithm in the back-end to compute the rigid transformation (see section II-B2). This way, when the sensor type changes, it is typically enough to use a different front-end, whereas the back-end can remain the same.

1) FRONT-END

The front-end is in charge of calculating the sensor pose based on its readings. It depends strongly on the type of sensor used. In our case, we are mainly interested in cameras and inertial sensors.

Inertial navigation sensors like INS usually fuse and integrate the information from different sensors (such as accelerometers, gyroscopes, and doppler velocity log (DVL)) and directly provide the 6-DoF pose as an output. They can also use absolute position sensors such as ultra-short baseline (USBL) or GPS (outside water) to correct the drift.

Extracting the pose from camera images can be mainly done in two fashions. If it is possible to place a known pattern in the set-up, the camera pose can be extracted from 3D-2D correspondences, using the Perspective-n-Point algorithm. In this case, each camera pose is directly referred to the pattern reference frame, so they are absolute poses and no drift is accumulated. The particularities of the underwater environment make it necessary to use an adaptation of this algorithm (see section III). If no pattern can be placed inside the FoV of the camera, SfM can be used. SfM is used to reconstruct the 6-DoF displacement between two camera frames by matching corresponding features. However, this reconstruction is up to a scale. In order to compute this scale, several approaches can be used. In the underwater domain, it is typical to use the readings of the pressure sensor or laser scalers [36]. Please note that this front-end can also be used by camera-based (triangulation) laser scanners, which is the case presented in this paper.

For ToF LiDARs, the displacement between two subsequent sensor readings can be computed by finding correspondences between their point clouds in a SfM-like manner. This is typically called LiDAR-based odometry. Examples can be found in [37], [38].

2) BACK-END

The goal of the back-end is computing the rigid 6-DoF transformation between two or more sensors using their respective egomotions. This boils down to a non-linear minimization problem that is typically tackled using either non-linear least-squares estimation or a filter-based approach. Most of the methods found in the literature use the corresponding incremental displacement of each sensor. Please note that each sensor has its own reference frame (and the transform between them is unknown). Therefore, this is a more complicated problem than bundle adjustment. In bundle adjustment, used for example in the calibration of a camera stereo pair, both cameras are referenced with respect to the same, corresponding features (for instance, a checkerboard appears in both images simultaneously).

Some approaches in the literature use least-squares estimation. In [39], [40], the authors propose a solution based on previous approaches that solve hand-eye calibration using incremental motions. In their minimization algorithm they do not only optimize the rigid transformation between the sensor pair, but also the trajectory of one of the sensors to make it more robust to noise. They also study which conditions the trajectory of the sensors need to comply with in order to make the problem fully observable. They come to the conclusion that ''so long as the axis of rotation of the incremental poses remains fixed [in 3D], any translations and any magnitude of rotation will not avoid singularity''. In [41], the authors propose a robust algorithm to calibrate multi-sensor arrays. This method also uses incremental motions but divides the problem in different steps: finding an initialization for each sensor pair, estimating first the rotational components, removing outliers, estimating the translation and finally combining the readings from all the sensors. They do not optimize the trajectory of any of the sensors.

Other authors use filter-based approaches, like [42]–[44] among others. In many cases, the filter is exploited for visual-inertial navigation. In our case, the navigation data from the robot's INS is reliable, so these methods are not further considered.

Compared to the already presented ones, other works tackle the problem using slightly different approaches. In [45], the authors include the time offset between sensors as a parameter to estimate. In [46], the trajectory is parameterized with B-splines and included in the optimization problem. In [47], only the parts of the trajectory that contribute more to the observability of the problem are considered, so that the computational complexity is reduced. Other interesting works can be found in [48]–[50].

In our work, both sensors provide absolute measurements. Consequently, the back-end approach followed in this paper considers the trajectory of both sensors as absolute poses and builds a pose graph that, using non-linear least squares, optimizes the relative sensors transform and the INS true path, as it suffers from integration drift (see section IV). This connects our work with [39], [40]. Therefore, it is important to understand the differences between both approaches.

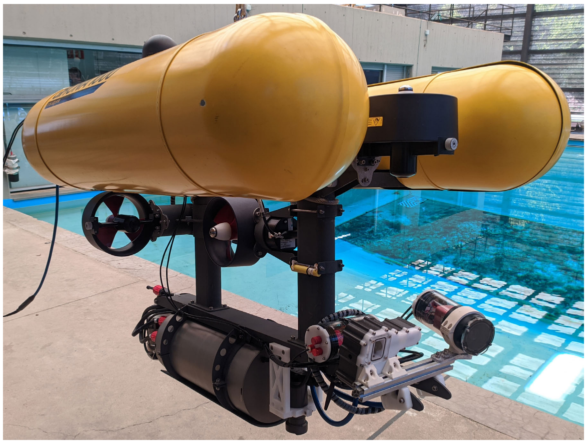


FIGURE 3. The Girona1000 equipped with the laser scanner.

In [39], [40] only incremental pose measurements are used to recover sensor calibration. It is also shown how only one of those sensors true path needs to be estimated. In our work, we use sensors that provide absolute measurements, and we build the problem as a pose graph, which is more restrictive as it constrains consecutive incremental poses to share a common pose. Note that, although we could get incremental pose measurements through differentiation, the propagation of its uncertainties is not trivial and could become another source of error.

III. HARDWARE DESCRIPTION

This section briefly explains the performance characteristics of the equipment used in this work. The robotic platform used is Girona1000 (see figure 3), which is a newer version of the Girona500 [34], previously developed at the lab. The Girona1000 is a lightweight, modular intervention AUV (I-AUV) that can be easily reconfigured for different tasks by changing its payload and thruster configuration. In the current configuration, the AUV uses 5 thrusters to control the yaw, surge, sway and heave, being passively stable in pitch and roll. Its navigation system is based on an INS aided by a DVL, a fiber optic gyroscope (FoG), and a pressure sensor. The payload integrates a forward-looking 3D laser scanner [14] previously developed in the lab, which is capable of scanning 200k points/s at a scan rate of around 0.5 Hz with sub-millimetric accuracy. It uses a green laser source with a nominal wavelength of 520 nm and an output power of 50 mW. Its FoV is of approximately 40° × 40° and its lateral resolution is of up to 0.008°.

The camera model used in this paper is the one introduced in [14]. Basically, it consists on a standard in-air camera model placed behind a flat refractive surface.

The in-air model used is the standard OpenCV pin-hole camera model [51]. However, due to the refraction of light rays at the interphase between air and water, this model does not accurately represent distortions under water [52], [53]. Therefore, the double distortion process suffered by light rays in their way to the camera is explicitly modelled. For a more detailed explanation, the reader is referred to [14]. The internal parameters of the scanner are estimated in a previous calibration process.

IV. INS – CAMERA CALIBRATION

An essential step of our approach is achieving an accurate calibration between the camera and the inertial sensor frames. The back-end minimization algorithm used in this paper is explained in this section. As seen in section II-B, this method could be used to estimate the 3D pose calibration between any combination of sensors that can provide absolute pose observations. In our case, we will assume that the desired calibration ${}^{I}\boldsymbol{t}_C$ is the one from the reference frame of the INS $\{I\}$ to the reference frame of the scanner's camera $\{C\}$.

The goal of the task is to find the camera pose with respect to the INS frame, namely ${}^{I}\boldsymbol{t}_C \in SE(3)$, given a set of n observed, noisy INS poses $z_I = [z_{I1}, \ldots, z_{Ii}, \ldots, z_{In}]$ with respect to the north-east-down (NED) frame $\{W_I\}$, and a corresponding set of n observed, noisy camera poses $z_C = [z_{C1}, \ldots, z_{Ci}, \ldots, z_{Cn}]$, with respect to the reference frame of the observed feature or pattern $\{P\}$.

Figure 4 shows a scheme relating the variables of interest. For the sake of readability, the reference frame to which some variables are referred is not explicitly written, but they can be directly inferred from the figure. The actual and observed INS poses $\boldsymbol{x}_I$ and z_I are always referred to $\{W_I\}$, whereas the observed camera poses z_C are always referred to the pattern $\{P\}$.

All the observations of both the camera and the INS are assumed to have zero-mean Gaussian noise. Since the actual trajectory of the robot $\boldsymbol{x}_I$ and the transform ${}^{W}\boldsymbol{t}_P$ are not known in reality, they are optimized along with ${}^{I}\boldsymbol{t}_C$. The actual, unobserved pose of the inertial reference frame of the robot at time i is named $\boldsymbol{x}_{Ii}$. Now, this problem is modelled as the pose graph in figure 5. Note that the observations from all the robot poses are connected through a common feature: the calibration pattern. The goal of the algorithm is to find the optimal trajectory $\boldsymbol{x}_I^*$ and the optimal transformations ${}^{W}\boldsymbol{t}_P$ and ${}^{I}\boldsymbol{t}_C$ given the observations from the INS and the camera. This can be formulated as maximizing the full posterior probability of the optimization variables $(\boldsymbol{x}_I, {}^{W}\boldsymbol{t}_P, {}^{I}\boldsymbol{t}_C)$ given the set of measurements (z_I, z_C) [54]:

$$\boldsymbol{x}_I^*,\ {}^{W}\boldsymbol{t}_P^*,\ {}^{I}\boldsymbol{t}_C^* = \underset{\boldsymbol{x}_I,\,{}^{W}\boldsymbol{t}_P,\,{}^{I}\boldsymbol{t}_C}{\arg\max} \cdot P\left(\boldsymbol{x}_I,\ {}^{W}\boldsymbol{t}_P,\ {}^{I}\boldsymbol{t}_C \mid z_I, z_C\right) \qquad (3)$$

Applying Bayes' theorem:

$$\boldsymbol{x}_I^*,\ {}^{W}\boldsymbol{t}_P^*,\ {}^{I}\boldsymbol{t}_C^* = \underset{\boldsymbol{x}_I,\,{}^{W}\boldsymbol{t}_P,\,{}^{I}\boldsymbol{t}_C}{\arg\max}\ P\left(z_I, z_C \mid \boldsymbol{x}_I,\ {}^{W}\boldsymbol{t}_P,\ {}^{I}\boldsymbol{t}_C\right) \cdot P\left(\boldsymbol{x}_I,\ {}^{W}\boldsymbol{t}_P,\ {}^{I}\boldsymbol{t}_C\right), \qquad (4)$$

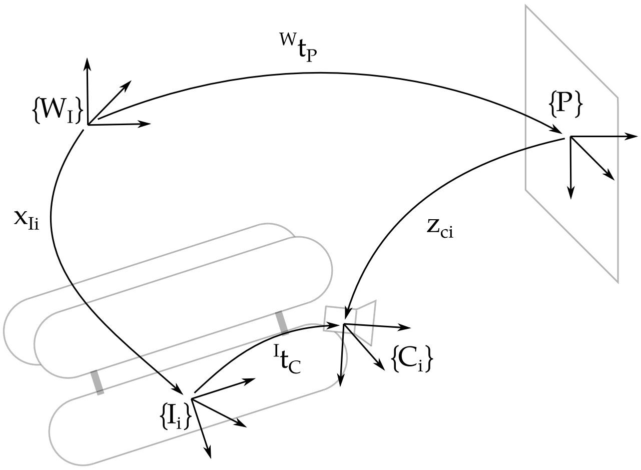


FIGURE 4. The observed camera poses z_{Ci} can be expressed as a function of the optimization variables x_{Ii}, ${}^{W}t_P$ and ${}^{I}t_C$.

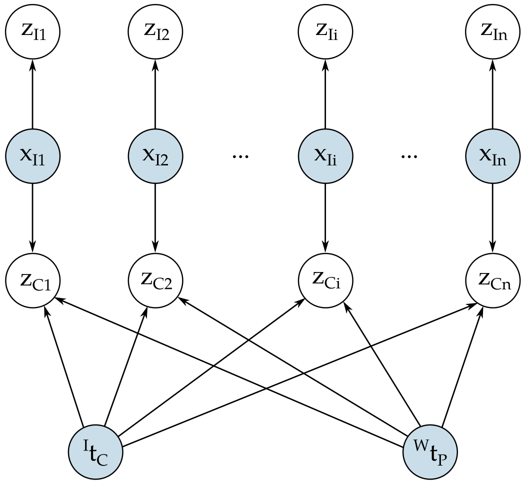


FIGURE 5. Pose graph representation of the minimization problem. Blue circles represent the minimization variables.

where $P\left(z_I, z_C \mid \boldsymbol{x}_I,\ {}^{W}\boldsymbol{t}_P,\ {}^{I}\boldsymbol{t}_C\right)$ are the observation models of the INS and the camera and $P\left(\boldsymbol{x}_I,\ {}^{W}\boldsymbol{t}_P,\ {}^{I}\boldsymbol{t}_C\right)$ are the prior beliefs of the optimization variables. Looking at figure 5 it can be seen that some variables are assumed independent: the trajectory of the robot $\boldsymbol{x}_I$ and the inertial measurements z_I are independent of ${}^{I}\boldsymbol{t}_C$ and ${}^{W}\boldsymbol{t}_P$, and camera measurements z_C are independent of z_I. Also, each measurement z_C and z_I is considered absolute and thus not dependant on previous measurements. Therefore:

$$\boldsymbol{x}_I^*,\ {}^{W}\boldsymbol{t}_P^*,\ {}^{I}\boldsymbol{t}_C^* = \underset{\boldsymbol{x}_I,\,{}^{W}\boldsymbol{t}_P,\,{}^{I}\boldsymbol{t}_C}{\arg\max}\ P\left(z_I \mid \boldsymbol{x}_I\right) P\left(z_C \mid \boldsymbol{x}_I,\ {}^{W}\boldsymbol{t}_P,\ {}^{I}\boldsymbol{t}_C\right) \cdot P\left(\boldsymbol{x}_I\right) P\left({}^{W}\boldsymbol{t}_P\right) P\left({}^{I}\boldsymbol{t}_C\right) \qquad (5)$$

We assume that there is no prior knowledge of the minimization variables available. Therefore, $P(\boldsymbol{x}_I)$, $P\left({}^{W}\boldsymbol{t}_P\right)$ and $P\left({}^{I}\boldsymbol{t}_C\right)$ are assumed uniform distributions and removed from the equation. If there were reasons to consider different distributions for these terms, they could be easily incorporated back in the equation. Expanding the equation to all the available observations:

$$\boldsymbol{x}_I^*,\ {}^W\boldsymbol{t}_P^*,\ {}^I\boldsymbol{t}_C^* = \underset{\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C}{\arg\max} \prod_{i=1}^{n} P\left(z_{Ii} \mid \boldsymbol{x}_{Ii}\right) \cdot P\left(z_{Ci} \mid \boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C\right) \quad (6)$$

These two remaining terms are the observation models of both sensors. They are both assumed to have Gaussian distributions. At the side of the INS, z_{Ii} is considered a random observation of a distribution centered in the expected value $\bar{\boldsymbol{x}}_{Ii}$ with a covariance matrix $\boldsymbol{\Sigma}_{Ii}$. Formally:

$$P\left(z_{Ii} \mid \boldsymbol{x}_{Ii}\right) \sim \mathcal{N}\left(\bar{\boldsymbol{x}}_{Ii}, \boldsymbol{\Sigma}_{Ii}\right) \propto exp\left(-\left[z_{Ii} - \bar{\boldsymbol{x}}_{Ii}\right]^T \boldsymbol{\Sigma}_{Ii}^{-1} \left[z_{Ii} - \bar{\boldsymbol{x}}_{Ii}\right]\right) \quad (7)$$

The camera pose can also be written using the optimization variables $\left(\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C\right)$ (see figure 4):

$$\hat{z}_{Ci}(\boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C) = \ominus\, {}^W\boldsymbol{t}_P \oplus \boldsymbol{x}_{Ii} \oplus {}^I\boldsymbol{t}_C \quad (8)$$

Therefore, the measured camera pose z_{Ci} is assumed a random observation of a normal distribution centered around the expected value $\bar{\hat{z}}_{Ci}(\boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C)$ with an associated covariance matrix $\boldsymbol{\Sigma}_{Ci}$:

$$\begin{aligned} &P\left(z_{Ci} \mid \boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C\right) \\ &\sim \mathcal{N}\left(\bar{\hat{z}}_{Ci}(\boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C), \boldsymbol{\Sigma}_{Ci}\right) \\ &\propto exp\left(-\left[z_{Ci} - \bar{\hat{z}}_{Ci}(\boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C)\right]^T \boldsymbol{\Sigma}_{Ci}^{-1} \right. \\ &\left. \cdot \left[z_{Ci} - \bar{\hat{z}}_{Ci}(\boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C)\right]\right) \quad (9) \end{aligned}$$

Combining equations (6), (7) and (9) and taking the negative logarithm:

$$\begin{aligned} \boldsymbol{x}_I^*,\ {}^W\boldsymbol{t}_P^*,\ {}^I\boldsymbol{t}_C^* = \underset{\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C}{\arg\min} \sum_{i=1}^{n} \boldsymbol{r}_{Ii}(\boldsymbol{x}_{Ii})^T\, \boldsymbol{\Omega}_{Ii}\, \boldsymbol{r}_{Ii}(\boldsymbol{x}_{Ii}) \\ + \boldsymbol{r}_{Ci}(\boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C)^T\, \boldsymbol{\Omega}_{Ci}\, \boldsymbol{r}_{Ci}(\boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C), \quad (10) \end{aligned}$$

where:

$$\begin{aligned} \boldsymbol{r}_{Ii}(\boldsymbol{x}_{Ii}) &= z_{Ii} - \boldsymbol{x}_{Ii} & (11) \\ \boldsymbol{r}_{Ci}(\boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C) &= z_{Ci} - \hat{z}_{Ci}(\boldsymbol{x}_{Ii},\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C) & (12) \\ \boldsymbol{\Omega}_{Ii} &= \boldsymbol{\Sigma}_{Ii}^{-1} & (13) \\ \boldsymbol{\Omega}_{Ci} &= \boldsymbol{\Sigma}_{Ci}^{-1} & (14) \end{aligned}$$

If these variables are stacked into vectors:

$$\begin{aligned} \boldsymbol{r}(\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C) &= \left[\boldsymbol{r}_{I1} \ldots \boldsymbol{r}_{In}\, \boldsymbol{r}_{C1} \ldots \boldsymbol{r}_{Cn}\right]^T & (15) \\ \boldsymbol{\Omega} &= \text{diag}\left[\boldsymbol{\Omega}_{I1} \ldots \boldsymbol{\Omega}_{In}\, \boldsymbol{\Omega}_{C1} \ldots \boldsymbol{\Omega}_{Cn}\right], & (16) \end{aligned}$$

the final expression is:

$$\boldsymbol{x}_I^*,\ {}^W\boldsymbol{t}_P^*,\ {}^I\boldsymbol{t}_C^* = \underset{\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C}{\arg\min}\ \boldsymbol{r}(\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C)^T \cdot \boldsymbol{\Omega}\, \boldsymbol{r}(\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C), \quad (17)$$

which is the standard equation of non-linear weighted least squares. Since the weight vector is built using the inverse of the covariance matrix of each residual, the optimized variables are found by minimizing the sum of the squared Mahalanobis distance of the residuals. It should be highlighted that all the variables that refer to poses, observations, displacements or transformations are actually 6 × 1 vectors because they belong to $SE(3)$. Covariance and information matrices are therefore 6 × 6 and the size of $\boldsymbol{r}(\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C)$ is $12\,n \times 1$.

In our current implementation of equation 17 on Ceres [55], the minimizer is fed with a weighted residual array rather than its squared sum, since this approach was found to convergence faster in practice. In order to do so, each term in $\boldsymbol{r}(\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C)$ is weighted with the upper triangular factorization of the corresponding element of $\boldsymbol{\Omega}$:

$$\boldsymbol{r}^T \boldsymbol{\Omega}\, \boldsymbol{r} = \boldsymbol{r}^T \boldsymbol{L} \boldsymbol{L}^T \boldsymbol{r} = (\boldsymbol{L}^T \boldsymbol{r})^T\ (\boldsymbol{L}^T \boldsymbol{r}), \quad (18)$$

where

$$\boldsymbol{\Omega} = \boldsymbol{L}\boldsymbol{L}^T \quad (19)$$

is the Cholesky factorization.

Finally:

$$\boldsymbol{x}_I^*,\ {}^W\boldsymbol{t}_P^*,\ {}^I\boldsymbol{t}_C^* = \underset{\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C}{\arg\min}\ \boldsymbol{L}^T\, \boldsymbol{r}(\boldsymbol{x}_I,\ {}^W\boldsymbol{t}_P,\ {}^I\boldsymbol{t}_C) \quad (20)$$

The uncertainty of each observation of the INS $\boldsymbol{\Sigma}_{Ii}$ is directly given by the sensor, whereas the uncertainty of each camera observation $\boldsymbol{\Sigma}_{Ci}$ is computed with the Monte Carlo algorithm by applying random noise to the image frame.

V. SIMULATION RESULTS

The calibration method explained in the previous section was validated on synthetic data. First, the extrinsic calibration was simulated by generating the front-end data which was compared later on to the ground truth (see section V-A). This proved useful to help relate the sensors' noises with the final uncertainty of the calibration result. Then, the motion distortion correction was evaluated to have a better grasp of how errors in the calibration propagated to the final 3D reconstruction (see section V-B).

A. CALIBRATION ON SYNTHETIC DATA

The goal of the simulation presented in this section is to validate whether a given trajectory has enough information to calibrate all DoF and quantify its robustness to noise. In our case, the trajectory is constrained in two different forms. First of all, the Girona1000 is designed to be controlled in surge, sway, heave, and yaw, being passively stable in roll and pitch. However, it is possible to control pitch in a short-range. Also, it is important to test if the calibration can be performed in a constrained space, such as a water tank, noticeably reducing resource and logistic costs.

A ground truth of the INS and camera trajectories $(\boldsymbol{x}_I, \boldsymbol{x}_c)$ has been generated given a known ${}^I\boldsymbol{t}_C$ and ${}^W\boldsymbol{t}_p$. The INS trajectory is generated considering a $6\,m \times 5\,m \times 5\,m$ region where the robot can safely move, and keeping a static pattern inside the FoV of the camera. Then, (z_I, z_c) are generated taking random samples of the measurement models, defined as follows:

$$\begin{aligned} z_{Ii} &= z_{Ii-1} \oplus \boldsymbol{\Delta x}_{Ii} \oplus \boldsymbol{w}_{\Delta I} \ ; \ \boldsymbol{w}_{\Delta I} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\Delta I}) \\ \boldsymbol{\Sigma}_{\Delta I} &= diag([\sigma_x^2 \ \sigma_y^2 \ \sigma_z^2 \ \sigma_\psi^2 \ \sigma_\theta^2 \ \sigma_\phi^2]) \end{aligned} \tag{21}$$

$$\begin{aligned} z_{ci} &= \boldsymbol{x}_{ci} \oplus \boldsymbol{w}_c \ ; \ \boldsymbol{w}_c \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_c) \\ \boldsymbol{\Sigma}_c &= diag([\sigma_x^2 \ \sigma_y^2 \ \sigma_z^2 \ \sigma_\psi^2 \ \sigma_\theta^2 \ \sigma_\phi^2]). \end{aligned} \tag{22}$$

The uncertainties in (21) are composed as:

$$\boldsymbol{\Sigma}_{Ii} = \boldsymbol{J}_{i-i\oplus}^T \left(\boldsymbol{J}_{i-i\ominus}^T \boldsymbol{\Sigma}_{Ii-1} \boldsymbol{J}_{i-i\ominus} \right) \boldsymbol{J}_{i-i\oplus} + \boldsymbol{J}_{i\oplus}^T \boldsymbol{\Sigma}_{Ii} \boldsymbol{J}_{i\oplus}. \tag{23}$$

where $\boldsymbol{J}_\oplus$ and $\boldsymbol{J}_\ominus$ are the Jacobians of the composition and of the inverse composition, respectively.

The INS position measurements (21) are modeled to drift, including a noise added to the displacement. Here, $\Sigma_{\Delta I}$ is assumed to be constant, and has been set to:

$$\begin{aligned} \sigma_x &= 1 \ mm \quad \sigma_\psi = 6 \times 10^{-7} \ ^\circ \\ \sigma_y &= 1 \ mm \quad \sigma_\theta = 6 \times 10^{-7} \ ^\circ \\ \sigma_z &= 0.1 \ mm \quad \sigma_\phi = 6 \times 10^{-7} \ ^\circ. \end{aligned} \tag{24}$$

Note that, although the INS filter receives updates (e.g. pressure sensor), this is not modeled for simplicity.

The camera position measurements (22) are modeled as absolute measurements, with a constant uncertainty Σ_{ci}, set to:

$$\begin{aligned} \sigma_x &= 2 \ mm \quad \sigma_\psi = 1 \ ^\circ \\ \sigma_y &= 2 \ mm \quad \sigma_\theta = 1 \ ^\circ \\ \sigma_z &= 2 \ mm \quad \sigma_\phi = 1 \ ^\circ. \end{aligned} \tag{25}$$

In general, the measurement models are pessimistic, either by over-conservative uncertainties or neglecting navigation updates. The simulated trajectory is shown in figure 6. It accounts for a ±20° pitch range of motion. The accumulated uncertainty at the last INS pose of the generated trajectory is around:

$$\begin{aligned} \sigma_x &= 316 \ mm \quad \sigma_\psi = 2.3 \ ^\circ \\ \sigma_y &= 315 \ mm \quad \sigma_\theta = 1.8 \ ^\circ \\ \sigma_z &= 100 \ mm \quad \sigma_\phi = 3.1 \ ^\circ. \end{aligned} \tag{26}$$

The calibration procedure has been run 15000 times re-generating the measurement samples. The introduced error in the ${}^I\boldsymbol{t}_C$ initialization was within ±100 mm for translation and ±20° for rotation, which is considerably worse than the accuracy that can be achieved measuring by hand. Results are shown in figure 7. Most of the ${}^I\boldsymbol{t}_C$ components are recovered with less than 1 cm and 0.5 degree error. However, it can be seen that the z and pitch components show the highest error bias, mainly due to the poor observability of the calibration parameters. As studied in [39], [40], if the rotations of the robot always happen around the same local axis, some components of the transformation between sensors reference frames cannot be observed. In order to better resemble the actual dynamics of Girona1000, the simulated dataset of robot poses included very limited rotations around its local x and y axes. Consequently, it is natural that the z and pitch components of the calibrated transformation show the highest errors. In order to assess the influence of these errors in the final point cloud reconstruction, another set of simulations is reported in section V-B.

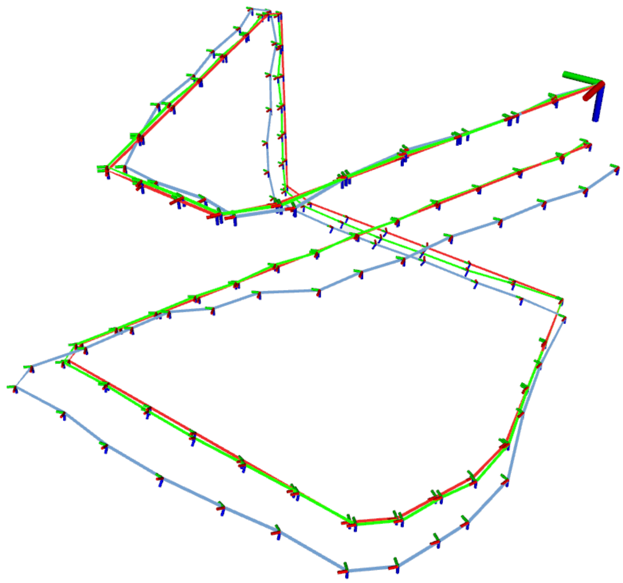


FIGURE 6. Synthetic INS trajectory generated to test the calibration procedure on simulation, considering a 6*m* × 5*m* × 5 *m* region. The red trajectory corresponds to the ground truth. The blue trajectory corresponds to a noisy measured trajectory, following the model in (21). The green trajectory corresponds to the optimized trajectory after the calibration procedure is completed.

B. MOTION DISTORTION ON SYNTHETIC DATA

The goal of the simulations presented in this section is to show the magnitude of the reconstruction error depending on the accuracy of the calibration between the INS and the camera. The simulated experimental set-up is schematically shown in figure 8. The simulated experiment is the inspection of an object (a sphere S of 100 mm diameter placed at a known position) using a laser scanner. The robot describes a smooth, realistic trajectory around the sphere at a distance of between 1 m and 3 m while sweeping a laser plane to scan the scene. Along its trajectory, the robot moves in all 3 axes but it rotates only around 2, resembling the movement of Girona1000. The robot moves at a linear speed of 0.5 m/s and a rotational speed of 7°/s. This set-up was chosen because it was possible to recreate experimentally in the water tank (see section VI-A).

At a given instant in time t, ${}^W\boldsymbol{x}_I$ represents the current 6-DoF pose $\{I\}$ of the INS with respect to the world reference frame $\{W\}$. The current ground truth pose of the camera $\{C\}$ is computed by composing ${}^W\boldsymbol{x}_I$ and ${}^I\boldsymbol{t}_C$, where ${}^I\boldsymbol{t}_C$ is the ground truth transformation between the INS and the camera. Similarly, the current ground truth pose of the laser $\{L\}$ is computed by further composing $\{C\}$ and ${}^C\boldsymbol{t}_L$. The laser plane projected at time t is π_L. The curve described by the intersection between π_L and S is discretized in a finite number

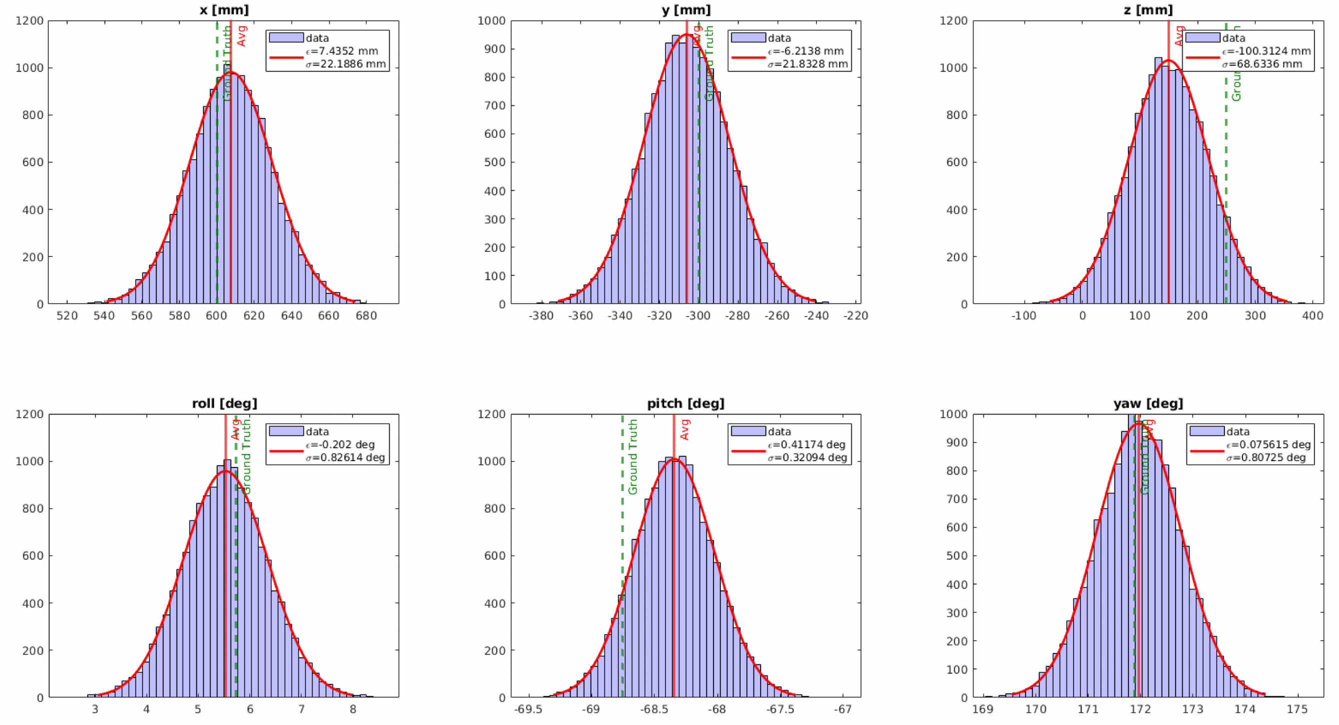


FIGURE 7. Histograms (blue) of ${}^{I}t_{C}^{*}$ from 15000 simulations. The vertical red line is the average solution, and the vertical green line is the ground truth. The solutions are fit into a Gaussian distribution, marked in red. The legend describes the standard deviation and the average error.

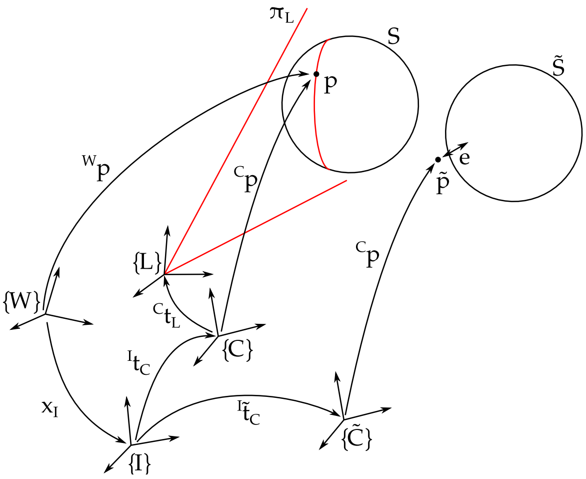


FIGURE 8. Scheme of the simulated experiment.

of points. In figure 8 only one of these points $\boldsymbol{p}$ is shown for the sake of clarity.

In the ideal case where ${}^{I}\boldsymbol{t}_{C}$ is perfectly known, the set of all points $\boldsymbol{p}$ lie exactly on the surface of S. However, the non-ideal calibration process may introduce errors, so the estimated pose between INS and camera ${}^{I}\tilde{\boldsymbol{t}}_{C}$ would differ from ${}^{I}\boldsymbol{t}_{C}$. Therefore, the non-ideal reconstructed point $\tilde{\boldsymbol{p}}$ would not coincide with $\boldsymbol{p}$, introducing a reconstruction error. It is expected that the larger the calibration error is, also the larger the reconstruction error will be.

In order to quantify the reconstruction error due to a mis-calibration of ${}^{I}\boldsymbol{t}_{C}$, the following process is simulated. At a given time t, the current laser plane π_L is first referenced with respect to $\{W\}$:

$$ {}^{W}\pi_{L} = {}^{W}\boldsymbol{x}_{I} \oplus {}^{I}\boldsymbol{t}_{C} \oplus {}^{C}\boldsymbol{t}_{L} \oplus {}^{L}\pi_{L} \tag{27} $$

The left superscript indicates the reference frame. Then, a set of points on the intersection curve between the laser plane and the sphere are computed. Each of these points complies with:

$$ \boldsymbol{p} \in \pi_{L} \cap S. \tag{28} $$

The erroneous reconstruction of each point $\boldsymbol{p}$ is then computed as:

$$ {}^{W}\tilde{\boldsymbol{p}} = {}^{W}\boldsymbol{x}_{I} \oplus {}^{I}\tilde{\boldsymbol{t}}_{C} \oplus {}^{C}\boldsymbol{p}, \tag{29} $$

where:

$$ {}^{C}\boldsymbol{p} = \ominus {}^{I}\boldsymbol{t}_{C} \ominus {}^{W}\boldsymbol{x}_{I} \oplus {}^{W}\boldsymbol{p}. \tag{30} $$

Note that if ${}^{I}\tilde{\boldsymbol{t}}_{C} = {}^{I}\boldsymbol{t}_{C}$, then ${}^{W}\tilde{\boldsymbol{p}} = {}^{W}\boldsymbol{p}$, as it should be.

This process is repeated for each pose in the simulated trajectory. Finally, a sphere $\tilde{S}$ is fitted to the set of all the points $\tilde{\boldsymbol{p}}$ in the scan using least squares. Then, the error e is computed as the distance between each point $\tilde{\boldsymbol{p}}$ and the surface of $\tilde{S}$.

$$ e = \mathrm{dist}(\tilde{\boldsymbol{p}}, \tilde{S}) \tag{31} $$

The root mean square sum of all these errors and the error of the fitted sphere radius are calculated for each scan. Then, these two metrics are averaged for all the scans in the simulation.

TABLE 1. Color code of ${}^{I}\tilde{t}_C$ for figure 9. The erroneous transformation represented by each color has error both in translation and rotation.

Color	Orange	Yellow	Purple
Error translation norm ${}^{I}\tilde{t}_C$ [mm]	86.6	17.3	1.7
Error rotation norm ${}^{I}\tilde{t}_C$ [°]	11.2	1.8	0.2

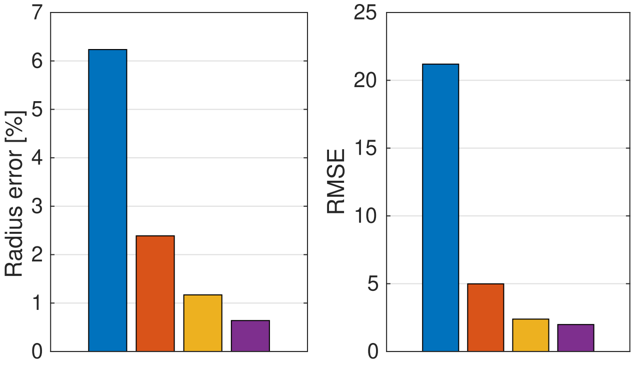


FIGURE 9. Error in the simulated scan of a sphere. Blue bars correspond to rigid scanning (no motion distortion compensation is applied). Orange, yellow and purple bars correspond to different levels of error in the calibration (see table 1).

Non-ideal cumulative noise was also added to the robot positions. At the end of the simulation, the robot had accumulated a drift of 32 mm and 1.5° over a total displacement of 7 m. Under these conditions, the 3D reconstruction error of the sphere is compared for different errors of ${}^{I}\tilde{t}_C$ in figure 9. It can be seen that not accounting for the motion distortion and treating each scan as rigid (blue) results in large errors. The other three bars show that the better the calibration, the lower the reconstruction error (as expected). However, it should be highlighted that noise in INS readings naturally plays a role as well in degrading the quality of the reconstruction: even for very accurate calibrations the error in the point cloud is still noticeable.

VI. EXPERIMENTAL RESULTS

The proposed motion distortion algorithm was also evaluated experimentally. The experiments were carried out using the hardware explained in section III in the water tank of the CIRS lab. In the water tank there was a ChArUco board, which was used to retrieve camera poses for the calibration algorithm, and some objects to be scanned (see figure 10).

The initial guess for the 6-DoF transformation between the INS and the camera measured by hand was:

$$ {}^{I}\tilde{t}_C = [0.60,\ -0.25,\ 0.20,\ 0.1,\ -1.57,\ 3.14] \tag{32}$$

The first three values are the displacements in x, y and z, respectively, all in meters, whereas the last three are the Euler angles roll, pitch and yaw around the x, y and z axes, respectively, applied in ZYX order, in radians.

A dataset containing 1100 images of the ChArUco and 32000 INS poses was gathered, since the output rate of the INS was much higher than the frame rate of the camera. Data coming from both sensors was associated according to their time stamps. Assuming that the robot moved smoothly and knowing that the rate of the INS was much higher than the camera, the corresponding INS poses were found using linear interpolation for the translation and spherical linear interpolation (SLERP) for the rotations. These pairs of poses were then fed to the back-end minimizer explained in section IV. The result of the calibration was:

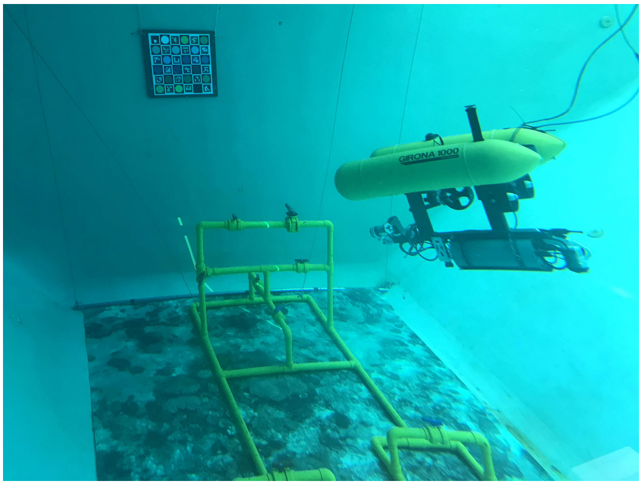


FIGURE 10. Experimental underwater setup consisting of a ChArUco board and a mock-up structure. The ChArUco board was only used as a reference to gather camera poses for the calibration algorithm.

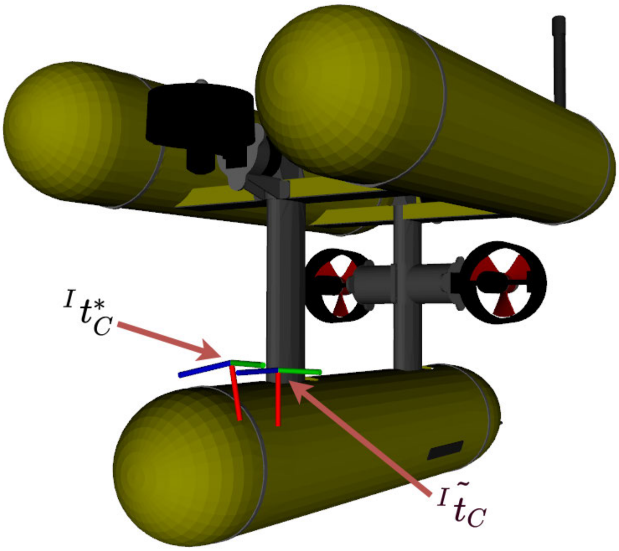
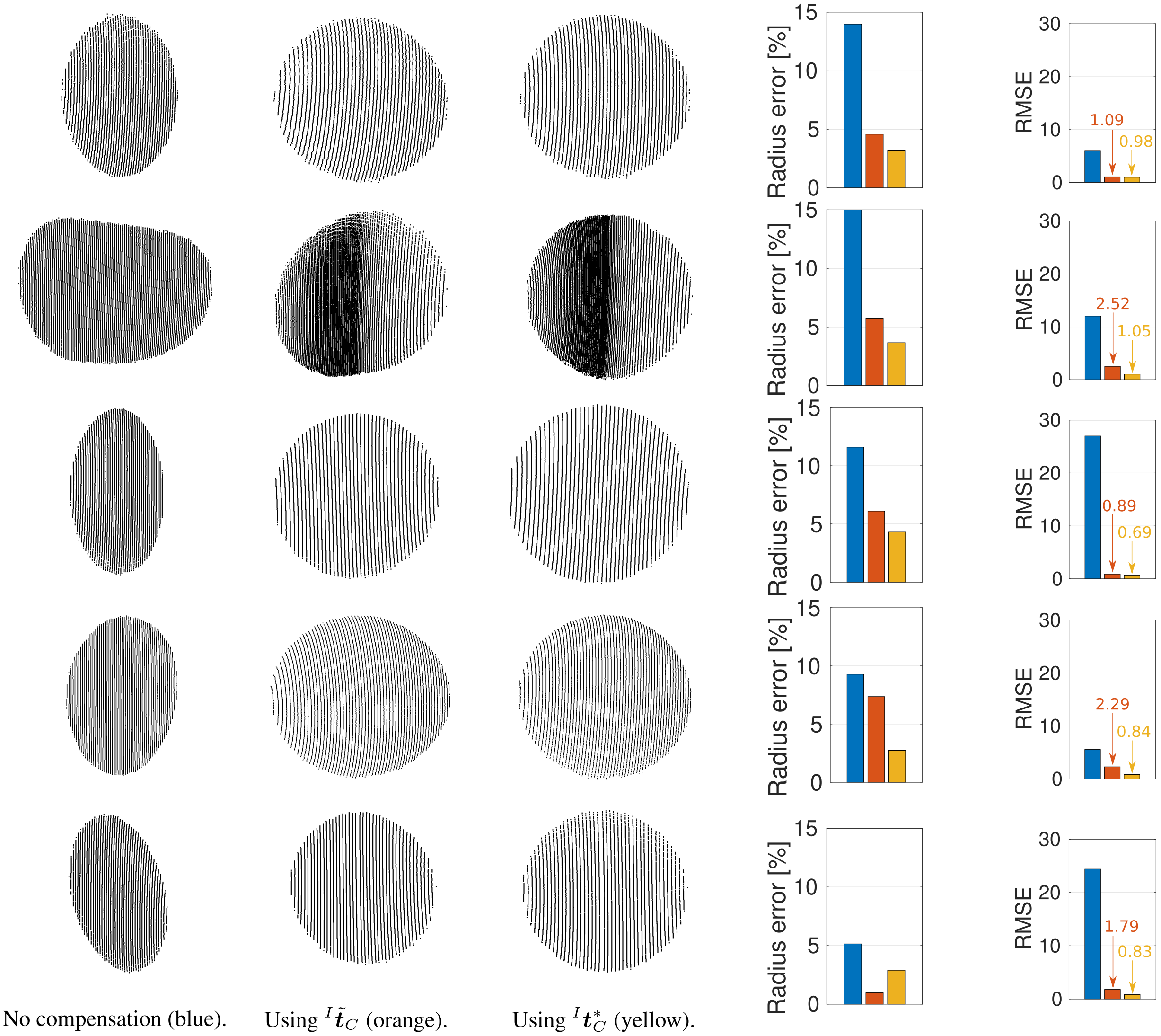


FIGURE 11. Position of the camera with respect to the robot using of ${}^{I}\tilde{t}_C$ measured by hand and the optimal ${}^{I}t_C^*$.

$$ {}^{I}t_C^* = [0.713,\ -0.237,\ 0.182,\ 0.0130,\ -1.394,\ 3.453] \tag{33}$$

The difference between the initial ${}^{I}\tilde{t}_C$ and the optimized ${}^{I}t_C^*$ was therefore a translation of 115.2 mm and a rotation of 16.4°. These are reasonably low values considering the difficulties of measuring by hand the translation and rotation



No compensation (blue). Using ${}^{I}\tilde{\boldsymbol{t}}_C$ (orange). Using ${}^{I}\boldsymbol{t}^*_C$ (yellow).

FIGURE 12. Evaluation of the 3D reconstruction of a sphere scanned dynamically using three different approaches. Each of the five rows represents one scan of a calibration sphere of known radius $r = 100$ mm. The first column was reconstructed treating the scan as rigid (no motion distortion compensation was applied). The spheres in the second column were reconstructed compensating for motion distortion using the value of ${}^{I}\tilde{t}_C$ measured by hand, whereas the optimized value ${}^{I}t^*_C$ was used for the third column. The two bar plots of each row measure the sphere fitting error. The plot in the left shows the percentage error in the fitted radius. The plot in the right shows the root-mean-squared distances of the point cloud to the fitted sphere. Blue, orange and yellow represent first, second and third approach, respectively.

between the center of the robot and the focal point of the camera at a non-standard position. Both transforms can be visualized in figure 11.

In order to assess the benefits of the calibration, two sets of experiments were designed. In both of them, the goal was evaluating the performance of the final 3D reconstruction following three approaches: (i) no motion distortion compensation (that is, treating the scans as rigid), (ii) compensation using the value of ${}^{I}\tilde{\boldsymbol{t}}_C$ measured by hand, and (iii) compensation using the optimized value ${}^{I}\boldsymbol{t}^*_C$. The first experiment consisted in scanning a calibration sphere (see section VI-A). In the second one, the target was a model of an underwater industrial structure (see section VI-B).

A. CALIBRATION SPHERE

The results of the experiment scanning a calibration sphere are shown in figure 12. Fives examples using the three aforementioned approaches are compared in the figure using two metrics: the radius error gives a measure of how well the size of the real sphere is reconstructed, whereas the fitting error RMSE is lower the closer the reconstructed point cloud resembles a sphere. It can be seen in the bar plots of the figure that trying to reconstruct the 3D shape of an object scanned dynamically without taking the robot displacement into account yields unusable results (first column of spheres). The reconstruction improves largely when motion distortion compensation is applied. However, using the optimized ${}^{I}\boldsymbol{t}^*_C$

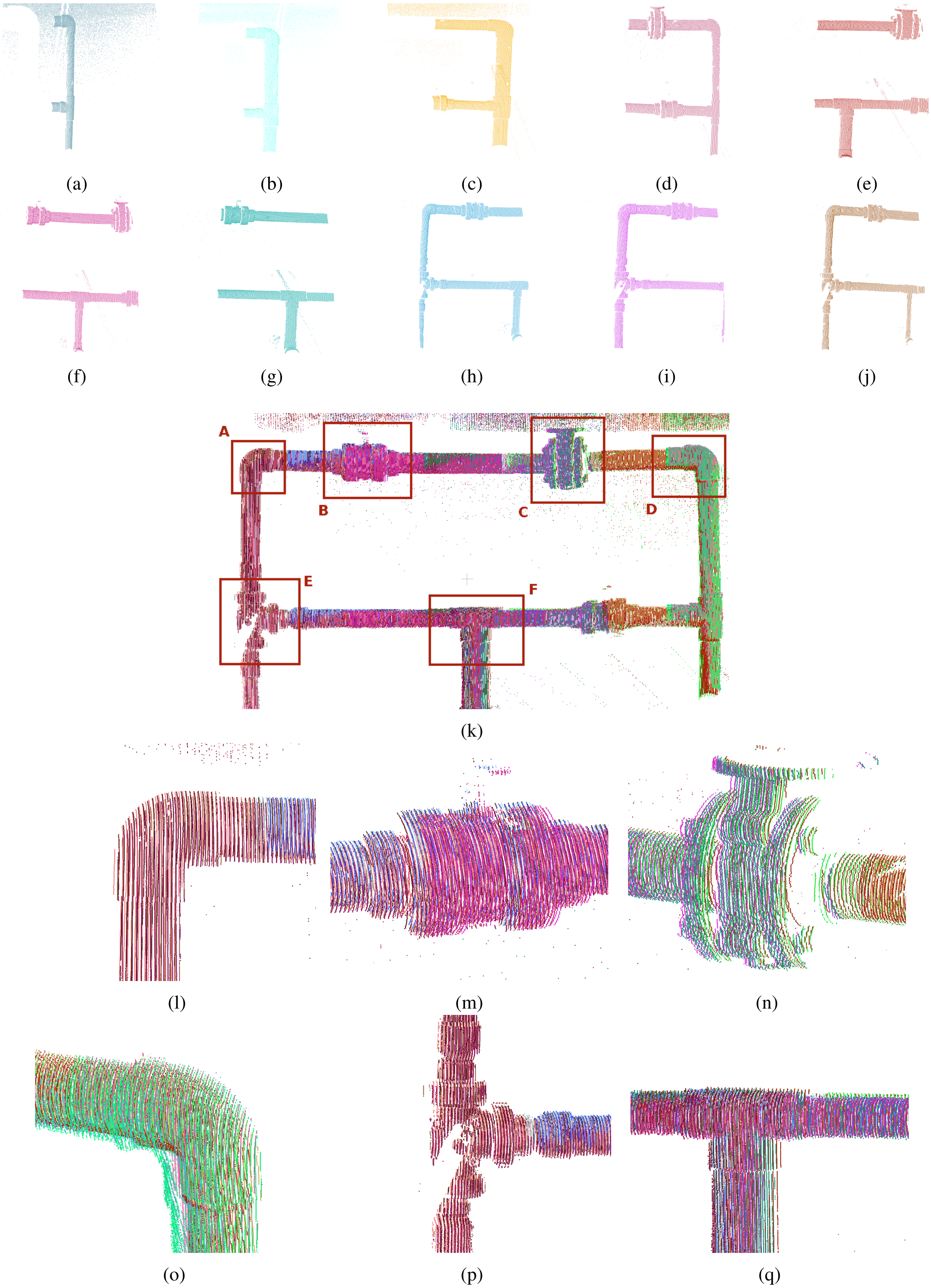
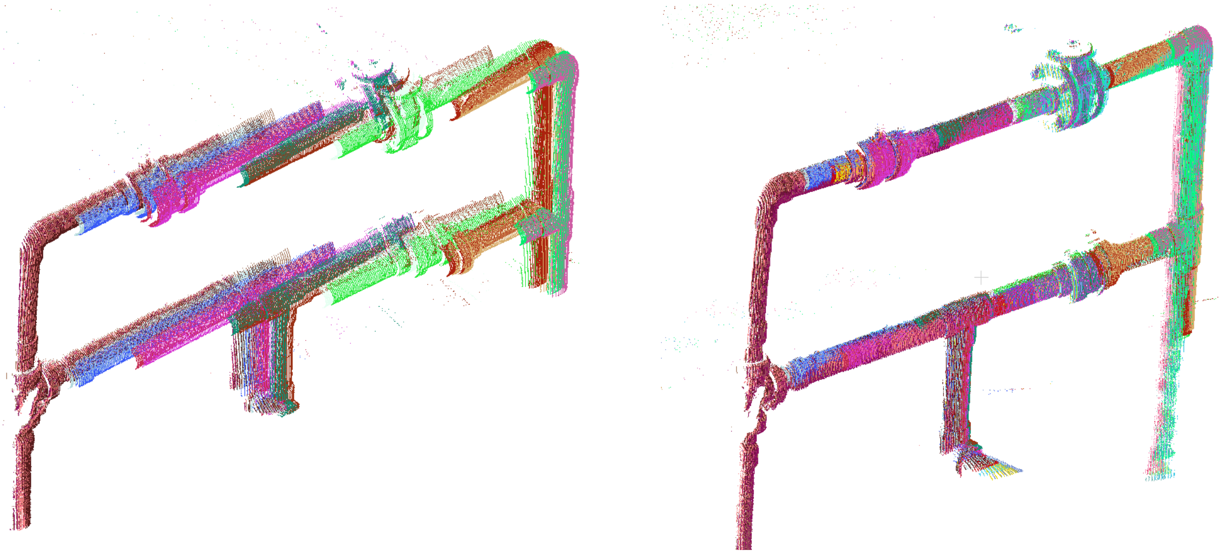


FIGURE 13. Combining partial dynamic scans into a single point cloud of the object of interest. Point clouds (a) to (j) show some examples of these partial dynamic scans. Note that they are affected by motion distortion. Point cloud (k) is the combination of 25 scans using the optimal ${}^{I}t_{C}^{*}$. Point clouds (l) to (q) show the details A to F.



(a) Using ${}^{I}\tilde{t}_C$ measured by hand.

(b) Using optimal ${}^{I}t^*_C$.

FIGURE 14. Comparison of the structure reconstruction using (a) ${}^{I}\tilde{t}_C$ measured by hand and (b) the optimal ${}^{I}t^*_C$. Using ${}^{I}\tilde{t}_C$ only achieves a rough alignment, where as a much better registration results using ${}^{I}t^*_C$.

generally results in better reconstructions. An unexpected value can be seen in the radius error plot of the last row. There, the orange bar (measured by hand) achieves the lowest error. This is most likely due to the fact that the motion distortion accidentally compensates for a distortion introduced by the internal calibration of the scanner. It is nonetheless a very particular case. The RMSE is however better when using ${}^{I}t^*_C$ in all cases. A qualitative visual inspection agrees that the third approach is best.

B. STRUCTURE INSPECTION

A more complete experiment was designed to inspect dynamically the model structure shown in figure 10. During the experiment, the robot navigated near the structure and the reconstructed point clouds were aggregated using only odometry readings from the INS. No alignment or registration between point clouds was performed. A demonstrative video was recorded and uploaded to: https://youtu.be/OytUI9 × 3cWw. The results are shown in figure 13. The two first rows of the figures show examples of single, partial scans. They are shown as originally retrieved from the scanner, so they suffer from motion distortion. In order to generate a single point cloud, 25 of these scans are combined using navigation data and the optimized ${}^{I}t^*_C$. Significant parts of the structure such as joints and valves are detailed in the last two rows. It can be seen that despite not using any registration algorithm, the navigation data and the calibrated ${}^{I}t^*_C$ are accurate enough to provide a generally consistent global reconstruction of the structure. Some errors can be seen in details D and F, for example. They are likely caused by an error accumulation coming from different sources, namely the navigation data, the estimation of ${}^{I}t^*_C$ and the internal calibration of the laser scanner.

A comparison of this odometry-based mapping using the ${}^{I}\tilde{t}_C$ measured by hand and the optimal ${}^{I}t^*_C$ is shown in figure 14. It can be clearly seen that the reconstruction using the optimal ${}^{I}t^*_C$ is much better aligned.

VII. CONCLUSION AND FUTURE WORK

This paper has described how the distortion that affects dynamic scans can be corrected by making use of the navigation data of the robotic platform on which the scanner is mounted. It has been shown using both synthetic and real data that, in order to achieve a satisfactory motion distortion compensation, the pose of the camera with respect to the inertial frame of the robot needs to be accurately known. A probabilistic approach has been followed to calibrate this parameter. In fact, this calibration can be applied to retrieve the 6-DoF pose between any pair of moving sensors by using observations of their respective trajectories.

A complete literature review on underwater 3D scanning and on visual-inertial extrinsic calibration has also been presented. Moreover, both the calibration and the undistortion processes have been simulated in order to numerically assess how the different error sources affect the final 3D reconstruction.

The two-fold goal of this paper was achieved: the 6-DoF roto-translation between a camera and an INS was estimated and this result was used experimentally to compensate for the motion distortion present in dynamic scans. The main limitation that has been found is that the effect of non-ideal sensors constrains the number of scans that can be referred to a common reference frame. These sources of error were mainly the cumulative noise of the inertial sensor and the internal calibration of the laser scanner.

Finally, prospective works in this line of research may include studies on how to take advantage of the undistorted scans for dynamic mapping and navigation and to apply them in realistic undersea scenarios.

APPENDIX

TRANSFORMATIONS IN *SE*(3)

The 6-DoF pose of reference frame $\{B\}$ with respect to $\{A\}$ can be written as ${}^{A}\boldsymbol{t}_B \in SE(3)$. It is made up of a rotation matrix ${}^{A}\boldsymbol{R}_B \in SO(3)$ and a translation vector ${}^{A}\boldsymbol{v}_B \in \mathbb{R}^3$. The rotation part can also be parameterized using the Euler angles $(\phi\ \theta\ \psi)$ in yaw-pitch-roll (ZYX) order. The resulting rotation matrix is therefore

$$ {}^{A}\boldsymbol{R}_B = \boldsymbol{Rot}_z(\psi)\,\boldsymbol{Rot}_{y'}(\theta)\,\boldsymbol{Rot}_{x''}(\phi). \tag{34} $$

Two successive transformations can be concatenated with the *composition* operator to compute the total transformation:

$$ {}^{A}\boldsymbol{t}_C = {}^{A}\boldsymbol{t}_B \oplus {}^{B}\boldsymbol{t}_C \tag{35} $$

This operator actually encodes two operations:

$$ {}^{A}\boldsymbol{v}_C = {}^{A}\boldsymbol{v}_B + {}^{A}\boldsymbol{R}_B\,{}^{B}\boldsymbol{v}_C \tag{36} $$

$$ {}^{A}\boldsymbol{R}_C = {}^{A}\boldsymbol{R}_B\,{}^{B}\boldsymbol{R}_C \tag{37} $$

Similarly, the composition operator can be used to express a point in another reference frame:

$$ {}^{A}\boldsymbol{p} = {}^{A}\boldsymbol{t}_B \oplus {}^{B}\boldsymbol{p}, \tag{38} $$

which is actually implemented as:

$$ {}^{A}\boldsymbol{p} = {}^{A}\boldsymbol{v}_B + {}^{A}\boldsymbol{R}_B\,{}^{B}\boldsymbol{p} \tag{39} $$

Finally, it is sometimes necessary to compute the inverse of a transform. This is done using the *inverse composition* operator

$$ {}^{B}\boldsymbol{t}_A = \ominus\,{}^{A}\boldsymbol{t}_B, \tag{40} $$

which actually encodes two operations:

$$ {}^{B}\boldsymbol{R}_A = \left({}^{A}\boldsymbol{R}_B\right)^T \tag{41} $$

$$ {}^{B}\boldsymbol{v}_A = -\,{}^{B}\boldsymbol{R}_A\,{}^{A}\boldsymbol{v}_B \tag{42} $$

The interested reader can find a detailed tutorial on *SE*(3) transformation parameterizations in [56].

ABBREVIATIONS

AUV	autonomous underwater vehicle
CAD	computer-aided design
DoF	degree of freedom
DVL	doppler velocity log
FoG	fiber optic gyroscope
FoV	field of view
GPS	global positioning system
I-AUV	intervention AUV
ICP	iterative closest point
IMU	inertial measurement unit
INS	inertial navigation system
IR	infrared
LiDAR	light detection and ranging
LLS	laser line scanner
NED	north-east-down
RMSE	root-mean-square error
ROV	remotely operated vehicle
SfM	structure from motion
SLERP	spherical linear interpolation
SONAR	sound navigation ranging
ToF	time of flight
USBL	ultra-short baseline
UUV	unmanned underwater vehicle

ACKNOWLEDGMENT

(Miguel Castillón and Roger Pi contributed equally to this work.)

REFERENCES

[1] A. Palomer, P. Ridao, and D. Ribas, "Inspection of an underwater structure using point-cloud SLAM with an AUV and a laser scanner," *J. Field Robot.*, vol. 36, no. 8, pp. 1333–1344, Dec. 2019.

[2] K. Himri, P. Ridao, and N. Gracias, "Underwater object recognition using point-features, Bayesian estimation and semantic information," *Sensors*, vol. 21, no. 5, pp. 1–27, Mar. 2021.

[3] R. Pi, P. Cieslak, P. Ridao, and P. J. Sanz, "TWINBOT: Autonomous underwater cooperative transportation," *IEEE Access*, vol. 9, pp. 37668–37684, 2021.

[4] F. R. Dalgleish, S. Tetlow, and R. L. Allwood, "Experiments in laser-assisted visual sensing for AUV navigation," *Control Eng. Pract.*, vol. 12, no. 12, pp. 1561–1573, Dec. 2004.

[5] M. Massot-Campos and G. Oliver-Codina, "Optical sensors and methods for underwater 3D reconstruction," *Sensors*, vol. 15, no. 12, pp. 31525–31557, Dec. 2015.

[6] P. Risholm, J. Thorstensen, J. T. Thielemann, K. Kaspersen, J. Tschudi, C. Yates, C. Softley, I. Abrosimov, J. Alexander, and K. H. Haugholt, "Real-time super-resolved 3D in turbid water using a fast range-gated CMOS camera," *Appl. Opt.*, vol. 57, no. 14, pp. 3927–3937, 2018.

[7] M. Massot-Campos and G. Oliver-Codina, "Underwater laser-based structured light system for one-shot 3D reconstruction," in *Proc. IEEE SENSORS*, Nov. 2014, pp. 1138–1141.

[8] M. Bleier and A. Nüchter, "Low-cost 3D laser scanning in air or water using self-calibrating structured light," in *Proc. Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci. ISPRS Arch.*, 2017, vol. 42, no. 2W3, pp. 105–112.

[9] P. Risholm, T. Kirkhus, and J. T. Thielemann, "High-resolution structured light d sensor for autonomous underwater inspection," in *Proc. OCEANS MTS/IEEE Charleston*, Oct. 2018, pp. 1–5.

[10] D. M. Kocak, F. M. Caimi, P. S. Das, and J. A. Karson, "A 3-D laser line scanner for outcrop scale studies of seafloor features," in *Proc. MTS/IEEE. Riding Crest 21st Century. Conf. Exhib. Conf.*, vol. 3, 1999, pp. 1105–1114.

[11] F. R. Dalgleish, F. M. Caimi, W. B. Britton, and C. F. Andren, "Improved LLS imaging performance in scattering-dominant waters," *Proc. SPIE, Ocean Sens. Monit.*, vol. 7317, Apr. 2009, Art. no. 73170E, doi: 10.1117/12.820836.

[12] F. Lopes, H. Silva, J. M. Almeida, A. Martins, and E. Silva, "Structured light system for underwater inspection operations," in *Proc. OCEANS Genova*, May 2015, pp. 1–6.

[13] S. Chi, Z. Xie, and W. Chen, "A laser line auto-scanning system for underwater 3D reconstruction," *Sensors*, vol. 16, no. 9, p. 1534, Sep. 2016.

[14] A. Palomer, P. Ridao, J. Forest, and D. Ribas, ''Underwater laser scanner: Ray-based model and calibration,'' *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 5, pp. 1986–1997, Oct. 2019.

[15] M. Castillón, A. Palomer, J. Forest, and P. Ridao, ''Underwater 3D scanner model using a biaxial MEMS mirror,'' *IEEE Access*, vol. 9, pp. 50231–50243, 2021.

[16] M. Castillón, A. Palomer, J. Forest, and P. Ridao, ''State of the art of underwater active optical 3D scanners,'' *Sensors*, vol. 19, no. 23, p. 5161, Nov. 2019.

[17] S. T. Digumarti, G. Chaurasia, A. Taneja, R. Siegwart, A. Thomas, and P. Beardsley, ''Underwater 3D capture using a low-cost commercial depth camera,'' in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–9.

[18] A. Anwer, S. S. A. Ali, A. Khan, and F. Meriaudeau, ''Underwater 3-D scene reconstruction using Kinect v2 based on physical models for refraction and time of flight correction,'' *IEEE Access*, vol. 5, pp. 15960–15970, 2017.

[19] S. Chourasiya, P. K. Mohapatra, and S. Tripathi, ''Non-intrusive underwater measurement of mobile bottom surface,'' *Adv. Water Resour.*, vol. 104, pp. 76–88, Jun. 2017.

[20] C.-K. Liang, L.-W. Chang, and H. H. Chen, ''Analysis and compensation of rolling shutter effect,'' *IEEE Trans. Image Process.*, vol. 17, no. 8, pp. 1323–1330, Aug. 2008.

[21] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and R. Medina-Carnicer, ''Generation of fiducial marker dictionaries using mixed integer linear programming,'' *Pattern Recognit.*, vol. 51, pp. 481–491, Mar. 2016.

[22] F. J. Romero-Ramirez, R. Muñoz-Salinas, and R. Medina-Carnicer, ''Speeded up detection of squared fiducial markers,'' *Image Vis. Comput.*, vol. 76, pp. 38–47, Aug. 2018.

[23] 2G Robotics. *Subsea 7 Spool Metrology*. Accessed: Jan. 26, 2021. [Online]. Available: http://www.2grobotics.com/subsea-7-spool-metrology/

[24] Leica Geosystems AG. *Leica Cyclone 3D Point Cloud Processing Software*. Accessed: Jan. 26, 2021. [Online]. Available: https://leica-geosystems.com/en-us/products/laser-scanners/software/leica-cyclone/

[25] H. Kondo and T. Ura, ''Navigation of an AUV for investigation of underwater structures,'' *Control Eng. Pract.*, vol. 12, no. 12, pp. 1551–1559, Dec. 2004.

[26] G. C. Karras and K. J. Kyriakopoulos, ''Localization of an underwater vehicle using an IMU and a laser-based vision system,'' in *Proc. Medit. Conf. Control Autom.*, Jun. 2007, pp. 1–6.

[27] A. Palomer, P. Ridao, D. Youakim, D. Ribas, J. Forest, and Y. Petillot, ''3D laser scanner for underwater manipulation,'' *Sensors*, vol. 18, no. 4, pp. 1–14, 2018.

[28] M. Bleier, J. Van Der Lucht, and A. Nüchter, ''SCOUT3D—An underwater laser scanning system for mobile mapping,'' in *Proc. Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci. ISPRS Arch.*, 2019, vol. 42, no. 2/W18, pp. 13–18.

[29] Kraken Robotics. *Seavision*. Accessed: Mar. 4, 2021. [Online]. Available: https://krakenrobotics.com/products/seavision/

[30] 3D at Depth. *Dynamic Lidar Moving Platform*. Accessed: Jan. 26, 2021. [Online]. Available: https://www.3datdepth.com/product/dynamic-lidar-moving-platform

[31] D. McLeod, J. Jacobson, M. Hardy, and C. Embry, ''Autonomous inspection using an underwater 3D LiDAR,'' in *Proc. OCEANS*, San Diego, CA, USA, 2013, pp. 1–8.

[32] 2G Robotics. *NOAA U-Boat Scanning*. Accessed: Jan. 26, 2021. [Online]. Available: http://www.2grobotics.com/noaa-u-boat/

[33] iXblue. *Phins Subsea: High-Performance Subsea ins for Deep Waters*. Accessed: Jan. 26, 2021. [Online]. Available: https://www.ixblue.com/products/phins-subsea

[34] D. Ribas, N. Palomeras, P. Ridao, M. Carreras, and A. Mallios, ''Girona 500 AUV: From survey to intervention,'' *IEEE/ASME Trans. Mechatronics*, vol. 17, no. 1, pp. 46–53, Feb. 2012.

[35] C. Roman, G. Inglis, and J. Rutter, ''Application of structured light imaging for high resolution mapping of underwater archaeological sites,'' in *Proc. OCEANS IEEE SYDNEY*, May 2010, pp. 1–9.

[36] K. Istenič, N. Gracias, A. Arnaubec, J. Escartín, and R. Garcia, ''Automatic scale estimation of structure from motion based 3D models using laser scalers in underwater scenarios,'' *ISPRS J. Photogramm. Remote Sens.*, vol. 159, pp. 13–25, Jan. 2020.

[37] J. Zhang and S. Singh, ''LOAM: Lidar odometry and mapping in real-time,'' in *Proc. Conf., Robot., Sci. Syst. Conf. (RSS)*, vol. 2, no. 9, 2014.

[38] C. Le Gentil, T. Vidal-Calleja, and S. Huang, ''3D lidar-IMU calibration based on upsampled preintegrated measurements for motion distortion correction,'' in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. Piscataway, NJ, USA: Institute of Electrical and Electronics Engineers, May 2018, pp. 2149–2155.

[39] J. Brookshire and S. Teller, ''Automatic calibration of multiple coplanar sensors,'' in *Proc. 7th Annu. Robot., Sci. Syst. Conf.*, vol. 7. Los Angeles, CA, USA: Univ. Southern California, 2011, pp. 33–40. [Online]. Available: https://ieeexplore.ieee.org/servlet/opac?bknumber=6276859

[40] J. Brookshire and S. Teller, ''Extrinsic calibration from per-sensor egomotion,'' in *Proc. 8th Annu. Robot., Sci. Syst. (RSS) Conf.*, vol. 8. Sydney, NSW, Australia: Univ. Sydney, Jul. 2012, pp. 25–32. [Online]. Available: https://ieeexplore.ieee.org/servlet/opac?bknumber=6574627

[41] Z. Taylor and J. Nieto, ''Motion-based calibration of multimodal sensor arrays,'' in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. Piscataway, NJ, USA: Institute of Electrical and Electronics Engineers, May 2015, pp. 4843–4850.

[42] S. Schneider, T. Luettel, and H.-J. Wuensche, ''Odometry-based online extrinsic sensor calibration,'' in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 1287–1292.

[43] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, ''Robust visual inertial odometry using a direct EKF-based approach,'' in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2015, pp. 298–304.

[44] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, ''Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback,'' *Int. J. Robot. Res.*, vol. 36, no. 10, pp. 1053–1072, 2017.

[45] P. Furgale, J. Rehder, and R. Siegwart, ''Unified temporal and spatial calibration for multi-sensor systems,'' in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 1280–1286.

[46] M. Fleps, E. Mair, O. Ruepp, M. Suppa, and D. Burschka, ''Optimization based IMU camera calibration,'' in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 3297–3304.

[47] T. Schneider, M. Li, C. Cadena, J. Nieto, and R. Siegwart, ''Observability-aware self-calibration of visual and inertial sensors for ego-motion estimation,'' *IEEE Sensors Journal*, vol. 19, no. 10, pp. 3846–3860, Jan. 2019, doi: 10.1109/JSEN.2019.2893809.

[48] F. M. Mirzaei and S. I. Roumeliotis, ''A Kalman filter-based algorithm for IMU-camera calibration: Observability analysis and performance evaluation,'' *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1143–1156, Oct. 2008.

[49] T. Lupton and S. Sukkarieh, ''Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions,'' *IEEE Trans. Robot.*, vol. 28, no. 1, pp. 61–76, Feb. 2012.

[50] J. Rehder and R. Siegwart, ''Camera/IMU calibration revisited,'' *IEEE Sensors J.*, vol. 17, no. 11, pp. 3257–3268, Jun. 2017.

[51] J. Heikkila and O. Silven, ''A four-step camera calibration procedure with implicit image correction,'' in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 1997, pp. 1106–1112.

[52] A. Sedlazeck and R. Koch, *Perspective and Non-Perspective Camera Models in Underwater Imaging—Overview and Error Analysis* (Lecture Notes in Computer Science: Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 7474. Berlin, Germany: Springer, 2012, pp. 212–242. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-34091-8_10

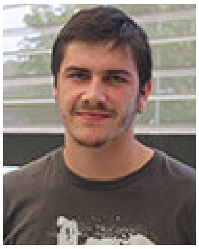
[53] T. Treibitz, Y. Schechner, C. Kunz, and H. Singh, ''Flat refractive geometry,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 1, pp. 51–65, Jan. 2012.

[54] G. Grisetti, R. Kummerle, C. Stachniss, and W. Burgard, ''A tutorial on graph-based SLAM,'' *IEEE Intell. Transp. Syst. Mag.*, vol. 2, no. 4, pp. 31–43, Winter 2010.

[55] S. Agarwal and K. Mierle. *Ceres Solver*. Accessed: Mar. 28, 2020. [Online]. Available: http://ceres-solver.org

[56] J. Blanco, ''A tutorial on SE(3) transformation parameterizations and on-manifold optimization,'' Univ. Malaga, Málaga, Spain, Tech. Rep. 3, 2013.

MIGUEL CASTILLÓN received the B.Sc. degree in industrial engineering from the University of Zaragoza, Spain, in 2015, and the M.Sc. degree in mechanical engineering from KU Leuven, Belgium, in 2018. He is currently pursuing the Ph.D. degree in robotics with the University of Girona, Spain. His research interests include computer vision applied to robotics and autonomous navigation.



ROGER PI received the B.S. degree in computer engineering from the Universitat de Girona, in 2017, and the joint M.Sc. degree in computer vision and robotics from the University of Burgundy, Universitat de Girona, and Heriot-Watt University, in 2019, where he also received the Best Master Student Award. He is currently pursuing the Ph.D. degree with the Underwater Robotics Research Center (CIRS) under the supervision of Dr. Pere Ridao. His research interest includes autonomous motion planning for underwater vehicle manipulator systems (UVMSs).

NARCÍS PALOMERAS received the M.Sc. degree, in 2004, and the Ph.D. degree, in 2011. He is currently the Coordinator of a Joint Degree Erasmus Mundus about Intelligent Field Robotic Systems, a Postdoctoral Researcher with the University of Girona (UdG), and a member of the Computer Vision and Robotics Group (VICOROB). He has participated in several research projects, all related to underwater robotics, both national and European (TRIDENT, PANDORA, MORPH, MERBOTS, LOONDOCK, TWINBOT, 3DAUV, and ATLANTIS) as well as in different European competitions for AUVs such as SAUC-E and ERL. His research interests include underwater robotics in topics like planning, exploration, intelligent control architectures, mission control, and localization.

PERE RIDAO (Member, IEEE) received the Ph.D. degree in industrial engineering from the University of Girona, Spain, in 2001. He is currently the Director of the Computer Vision and Robotics Research Institute (VICOROB), the Head of the Underwater Robotics Research Center (CIRS), and an Associate Professor with the Department of Computer Engineering, University of Girona. Since 1997, he has participated in 24 research projects (15 European and nine national). He is the author of more than 100 publications. He has directed nine Ph.D. theses (four more under direction) and 14 M.S. theses. He is also the coauthor of four licenses and one Spanish/European patent, being the Co-Founder of Iqua Robotics S.L. Spin-Off Company. His research interests include designing and developing autonomous underwater vehicles for 3-D mapping and intervention. He served as the Chair for the IFAC's Technical Committee on Marine Systems.