

Novel robotic mapping system for freshwater environments

Matias Mäki-Leppilampi, Heikki Hyyti, Harri Kaartinen

Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute FGI,
National Land Survey of Finland, Vuorimiehentie 5, 02150 Espoo, Finland
(matias.maki-leppilampi, heikki.hyyti, harri.kaartinen)@nls.fi

Keywords: Lidar, Positioning, Calibration, Sensor fusion, SLAM

Abstract

This paper presents an unmanned surface vehicle (USV) equipped with a mapping system designed to map boreal freshwater environments. The proposed system fuses satellite navigation, inertial measurements, and lidar data to provide accurate and precise three-dimensional (3D) point clouds from the environment around the USV's path. In order to achieve the required accuracy, we present several calibration methods used including a novel cost function for optimizing a rotation between lidar and inertial frames based on accelerometer measurements and point cloud registration. In the proposed positioning method, a post-processed high-end satellite navigation and inertial fusion trajectory is used as an initial guess of the USV's pose and for motion compensating lidar data. Pose graph based simultaneous localization and mapping (SLAM) algorithm is used to further refine the map and trajectory using normal distributions transform (NDT) distribution to distribution variant to compute lidar odometry and loop-closures offline after data collection. A method for rating loop-closures is adopted to select scan registration results to add into the pose graph. A factor graph is built using lidar odometry, detected loop-closures, and fused satellite navigation and inertial solution to optimize and solve the optimal trajectory. The conducted experiment demonstrates that the proposed graph-SLAM method significantly improves the overall consistency of the resulting 3D point cloud and the absolute trajectory error (ATE) of the optimized trajectory.

1. Introduction

Rivers play vital roles in their local ecology and economies (Tomsett and Leyland, 2019). Active monitoring and research on rivers is required adequately manage them. Lidars have enabled collecting high-density, high resolution (sub-centimetre in some cases) digital models, however, data collection with static human-operated systems is laborious and expensive (Harpold et al., 2015). Autonomous unmanned surface vehicles (USV) can be deployed at decreased labour cost, offering longer battery life, better visibility under bridges or tree canopies compared with unmanned aerial vehicle. Additionally, they allow deployment of sensors under water.

To successfully complete tasks autonomously USVs require robust means of positioning for mapping and to avoid collisions. Global navigation satellite systems (GNSS) are commonly used in open areas and in the air, but they might not work sufficiently well near waterfront forests, in canyons, or near embankments, even when fused with inertial navigation systems (INS) (see e.g., Kaartinen et al. 2015).

In survey applications GNSS/INS post-processed trajectories are a common solution used for mapping and geolocating point cloud data, which first solves the localization problem to map the environment (see e.g., Kukko et al. 2012; Di Stefano et al. 2021). However, the solutions often contain errors which are visible in the final point cloud model. Many Nordic natural freshwater environments are narrow rivers or small lakes with forest growing along the waterfront, or even leaning over water, which impedes GNSS signals. Consequently, the autonomous USV should also be able to localize and map the environment in these conditions.

Angular errors in the trajectory or calibration cause problematic blurring or duplication in the final point cloud model.

Moreover, longer trajectories may also contain drift in the position solution further compounding errors into the point cloud model.

In robotics simultaneous localization and mapping is a well established field of research, which allows to use the mapping data to solve the localization problem and vice versa. SLAM has been applied with great success in real-time systems in urban environments on ground based vehicles (Zhang and Singh, 2014; Shan et al., 2020). However, these methods still suffer from drift. Loop-closures can be added to remove drift, but incorrect or erroneous loop-closures introduce error or even cause a failure of the SLAM solution (Sunderhauf and Protzel, 2012).

Research on USV application and methods is gaining increased attraction. Application of SLAM in the context of Nordic boreal freshwater environments is a less explored area. Most research is focused on urban or semi-urban waterways or rivers (Chambers et al., 2011; Wang et al., 2019). Understanding the challenges and limitations of lidar-based mapping in these environments will open up new application areas, for example, for remote surveying and data collection. Additionally, robust mapping algorithms that do not rely solely on GNSS measurements ensures useful data can be collected even if the GNSS/INS solution is inadequate.

In this work, we have developed a novel robotic mapping system to be mounted on a USV, to enable the robot to position itself precisely and map the environment around its trajectory. In order to achieve the required accuracy, we present several calibration methods including a novel cost function for optimizing the rotation between lidar and inertial frames based on accelerometer measurements and point cloud registration. The positioning and mapping is done by fusing inertial measurements, satellite navigation, and point clouds produced by a rotating lidar sensor. Graph optimization is used to fine-tune the

map and trajectory. Planning and control of the USV is not tackled in this work.

The contribution of our work is the following: (1) the construction of a novel robotic mapping system for freshwater environments, (2) calibration of the setup based on Rényi quadratic entropy measure, and (3) proposing a novel graph-SLAM variant for freshwater environments.

In the following sections, we first present the methods and materials, including the used sensors on the USV, and the three different calibration methods needed in the work. Next, we propose our graph-SLAM variant for freshwater environments fusing post-processed GNSS/IMU and lidar measurements on the rotating tilted platform. Then, we explain the performed experiments done in Oulanka National park, Finland, using the USV and sensor payload additionally collecting reference position measurements. In the results section we present the extent to which the post-processed GNSS/IMU solution was improved upon, which is commonly used as a reference in previous work, using our calibration and graph-SLAM methods. Finally, we discuss the gained improvements and conclude the paper.

2. Methods and materials

2.1 Notation

A point cloud X is defined as a set of points $\{x_i\}_{i=1}^N$, where $x \in \mathbb{R}^3$ is the three-dimensional (3D) coordinates of a measured point in space, and $N \in \mathbb{R}$ is the number of points in X . Likewise, a Scan S_j is defined as a set of points $\{x_i\}_{i=1}^{N_j}$, where $N_j \in \mathbb{R}$ is the number of points in scan S_j . In this work scans refer to a set of points collected during one entire revolution of the rotating platform. Unlike with some lidar systems, the size of a scan varies depending on the number of returns.

A transformation $T \in SE(3)$ is given by

$$T(x, y, z, \gamma, \beta, \alpha) = \begin{bmatrix} R(\gamma, \beta, \alpha) & t(x, y, z) \\ \mathbf{0}_{3 \times 1} & 1 \end{bmatrix}, \quad (1)$$

$$d(T) = \|t(x, y, z)\|_2, \quad (2)$$

$$\angle(T) = \arccos\left(\frac{\text{Tr}(R) - 1}{2}\right), \quad (3)$$

where $t = [x, y, z]^T$, $R(\gamma, \beta, \alpha)$ is a rotation matrix $R \in SO(3)$, and γ, β, α are the yaw, pitch, and roll angles according to the intrinsic Tait-Bryan $R_{zy'x''}$ convention, d is the norm of the translation component of T , and $\angle$ the magnitude of R .

2.2 Sensors and hardware

An unmanned surface vehicle (USV), The Otter by Maritime Robotics is used as a platform in this study (see Figure 1). The following sensors are mounted on top of it to map the surrounding environment.

A two-dimensional (2D) laser scanner, Riegl miniVUX-1UAV is mounted on a rotating platform to collect three-dimensional (3D) scans of the environment (see details in Mäki-Leppilampi 2024). The laser scanner measures at a rate of up to 100 scan lines per second with up to 100 000 points per second with a beam divergence of 1.6×0.5 mrad. The rotating platform is capable of up to 90 revolutions per minute. The angle of the rotating platform is measured using an encoder with a resolution



Figure 1. Picture of the USV equipped with the rotating platform and Riegl's miniVUX-1UAV. Photo by Ville Kankare.

of 0.0035° . The lidar is angled at 50° from the normal resulting in a scanning plane of 40° from the horizontal, which results in a cross-hatch scan pattern. The combined system has a field of view of $360^\circ \times 80^\circ$.

A Novatel CPT7 (rev. 2) combined GNSS receiver and inertial measurement unit (IMU), with a measurement rate of 400 Hz, is used as the primary navigation device. Virtual GNSS base-station data is downloaded along with satellite ephemeris data from a service provider after performing measurements to compute a differential tightly-coupled GNSS/INS trajectory. The post-processing is done using NovAtel Inertial Explorer software (NovAtel, 2023).

2.3 Synchronization and calibration

A pulse per second (PPS) signal and a serial NMEA data stream is used for synchronizing the time between the GNSS receiver, micro-controller, and lidar. The micro-controller receives the pulse and serial data from GNSS, and sends a separate PPS pulse and NMEA serial message to the lidar irrespective of the GNSS receiver. This ensures that the two can be time synchronized without a GNSS fix.

2.3.1 Extrinsic calibration of the 3D lidar The custom setup necessitates calibration to determine the extrinsic transformation between the laser scanner and the rotating platform under which the IMU is mounted. The parametrized transformation $T(x, y, z, \gamma, \beta, \alpha)$ given by Eq. (1) is used with the translation in z axis set to 0, because the platform rotates around z axis, thus not containing information for its calibration.

To optimize the transformation, we use the Rényi quadratic entropy measure

$$E(x) = -\log_2 \int p(x)^2 dx, \quad (4)$$

where $p(x)$ is a probability density function (PDF) for continuous variables (Rényi, 1961). For discrete variables, such as point clouds, the Rényi's quadratic entropy (RQE) estimator is solved using a Gaussian kernel to estimate the PDF around each point (Principe and Xu, 1999). The RQE of a point cloud X is thus given by

$$E_G(X) = -\log_2 \frac{1}{N^2 C} \sum_i \sum_j e^{\left(\frac{1}{2} (x_i - x_j)^T \Sigma_\sigma (x_i - x_j)\right)}, \quad (5)$$

where x_i and x_j are points in X , N is the number of points in X , $C = \sqrt{(2\pi)^D |\Sigma_\sigma|}$ is a normalizing constant, D is the

dimension of $\mathbf{x}$ (here $D = 3$), $\Sigma_\sigma = \frac{2}{\sigma^2} I$, and σ is a freely chosen parameter. The entropy is a measure of point cloud coherence, and has been used as an objective function in literature, for minimizing a set of parameters without manually specified calibration targets (see e.g., Putkiranta 2020).

A downside of RQE is its expensive computational cost. However, since the effect of points several σ away are negligible on Eq. (5), it is sufficient to use the immediate k nearest neighbors (kNN), which can be efficiently found using kNN-search (Maddern et al., 2012; Sheehan et al., 2012). Further, with a rotating scanner, two redundant points sets S_L and S_R are measured, consisting of points collected in the left and right halves of an entire revolution, respectively. Exploiting the redundant point sets, and using voxel downsampling the calibration is sped up (Oberländer et al., 2015). Eq. (5) is modified to drop unnecessary constants to calculate the combined entropy of S_L and S_R

$$Q(S_L, S_R) = - \sum_i \sum_j e^{((\mathbf{x}_i^L - \mathbf{x}_j^R)^T \Sigma_\sigma (\mathbf{x}_i^L - \mathbf{x}_j^R))}, \quad (6)$$

where $\mathbf{x}_i^L$ and $\mathbf{x}_j^R$ are points belonging to S_L and S_R , respectively.

Data for the extrinsic calibration was collected outdoors with several surfaces in different directions around the scanner. The rotating platform was mounted horizontally on a tripod and the motor was rotated at a slow and constant angular velocity. The calibration parameters were optimized with MATLAB `fminsearch`-function, which implements the Nelder-Mead simplex search method (Lagarias et al., 1998). Table 1 presents the initial parameters based on the CAD-model, and the final calibrated parameters. Total time taken was 511 seconds with pre-processing taking 263 seconds, and optimization 248 seconds.

	x (mm)	y (mm)	α (°)	β (°)	γ (°)	Q
init.	31.60	20.00	40.00	0.00	0.00	-32034
final	37.65	14.29	40.97	0.13	-0.18	-52284

Table 1. Initial and final parameters with their non-normalized entropy values Q in Eq. (6) with $\sigma = 0.05$ and using 4 nearest neighbors in the kNN search.

2.3.2 Rotation calibration between the lidar and IMU

Let B denote the Body (IMU) frame and L denote the 3D Lidar frame (see Figure 2). We measure a point cloud at an initial orientation of the Body frame B_0 with a rigidly connected Lidar frame L_0 , and a gravity vector $\mathbf{g}_0$. Further measurements $\mathbf{g}_i$ are taken by tilting the rigidly connected lidar-IMU setup at multiple different orientations B_i with an unknown rotation R_{L_i, L_0} , which is a rotation from the rotated Lidar frame L_i back into L_0 . The measured gravity vectors in the rotated Body frame B_i can be rotated into B_0 via

$$\hat{\mathbf{g}}_{i, B_0} = R_{L, B} R_{L_i, L_0} R_{L, B}^T \mathbf{g}_{B_i}, \quad (7)$$

where $R_{L, B}$ is the rotation from Lidar to Body frame. Given the rotations R_{L_i, L_0} and $R_{L, B}$ are correct, then $\mathbf{g}_0 = \hat{\mathbf{g}}_{i, B_0}$.

Combining N measurements at different orientations, the gravity residual E_C and solution for $R_{L, B}^*$ is given by

$$E_C = \sum_{i=1}^N \|\mathbf{g}_0 - \hat{\mathbf{g}}_{i, B_0}\|^2 \quad (8)$$

$$R_{L, B}^* = \arg \min_{R_{L, B}} E_C,$$

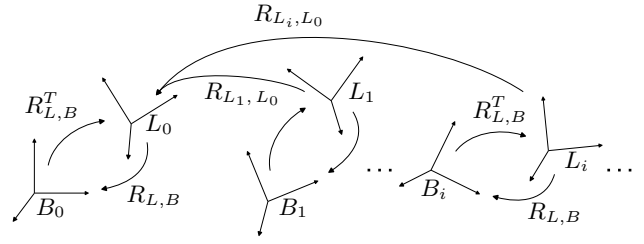


Figure 2. Link rotations of the Body and Lidar frames at different orientation indices i used in the calibration.

where $R_{L, B}^*$ is the optimized Lidar to Body frame rotation.

The rotation R_{L_i, L_0} can be found via scan registration of point clouds P_{L_0} and P_{L_i} that are measured in the Lidar frames L_0 and L_i , respectively. The minimization problem can then be solved as a manifold optimization problem in the group $SO(3)$ using Manopt (Absil et al., 2007; Boumal et al., 2014).

The method was tested by measuring at four different orientations (B_0 and $\{B_i\}_{i=1}^{N=3}$) using a tripod. Scans were registered using Distribution-to-Distribution (D2D) Normal Distributions Transform (NDT) proposed by Stoyanov (2012), and further developed by Kivioja (2022). The source and target cell sizes were set to 1 m, and initial transformations were defined manually. Parameterized results for $R_{L, B}^*$ are given in Table 2 along with the initial ($k = 0$) and final ($k = 70$) cost values, in which k is the number of iterations in the optimization. In total the time taken was 421 seconds from which pre-processing took 416 seconds, registration 4 seconds, and optimization 1 second.

k	α (°)	β (°)	γ (°)	E_C
0	0.00	0.00	0.00	122.762
70	-0.01	0.22	0.33	0.005

Table 2. Initial and final rotation parameters between the Lidar to Body (IMU) frame using parametrization defined in Eq. (1). The minimized E_C in Eq. (8) is shown along the parameters.

2.3.3 Heading correction for the post-processed GNSS/IMU

A bias in the rotations of the collected post-processed trajectories was observed, especially w.r.t the heading (yaw-axis) (Mäki-Leppilampi, 2024). We reason that there is either systematic error in collecting, calculating the post-processed trajectory, or in the data due to the environmental conditions (latitude, multi-path, or sensor biases).

To correct for the biases, a small subset of N consecutive scans starting from an index j , $S^* = \{S_i\}_{i=j}^{j+N}$ is selected, and used to solve for an arbitrary rotation R by minimizing Eq. (5) of the combined point cloud $\hat{X} = \{S_i \cup S_{i+1} \dots \cup S_{i+N}\}$. Individual scans S_i are transformed into their local coordinate frame, the optimal rotation is applied, and the scans are transformed back into the global frame

$$S_i = T_i R_{\text{opt}} T_i^{-1} S_i, \quad (9)$$

where T_i^{-1} is the transformation into the local frame i , T_i is the transformation back into the global frame, and R_{opt} is the optimal rotation matrix. Table 3 gives the initial ($k = 0$) and final ($k = 85$) results of the parametrized trajectory rectifying rotation. MATLAB `fminsearch` function was used as the optimizer. Preprocessing took 3 seconds, and optimization 204 seconds.

k	α (°)	β (°)	γ (°)	E_G
0	0.00	0.00	0.00	0.888
85	-0.01	0.27	2.64	0.801

Table 3. Calibration results to correct rotational bias in the GNSS/INS trajectory using parametrization defined in Eq. (1). The minimized E_G in Eq. (5) is shown along with the parameters.

2.4 Localization and Mapping

The post-processed GNSS/INS trajectory is used as the starting point for refining the map and trajectory. A two-step process is used to remove the motion from the collected lidar measurements. First, the lidar data is projected from the rotating scanner frame to the Body (IMU) frame (see Section 2.3). Second, the post-processed differential GNSS/INS solution is used to transform the measurements into the national ETRS-TM35FIN reference frame into a point cloud.

Next, we split the georeferenced point cloud into consecutive scans or frames such that one full revolution of the platform defines a scan. Scan odometry, i.e., relative transformations between consecutive frames $T_{j,j+1}^S$ are measured with scan registration using the GNSS/INS trajectory as an initial guess by calculating relative transformations

$$T_{i,j} = T_j^T T_i, \quad (10)$$

where transformations T_i and T_j are from frame i and j to the global frame, respectively.

A sliding window submap $M_k = \{S_i \cup T_{i-1,i} S_{j-1} \cdots \cup T_{i-m,i} S_{i-m}\}$ of the previous m frames is used as the target point set when computing scan odometry. Additionally, every scan S_i , where $i \bmod m = 0$ and $i > 0$, is chosen as a key-frame to form a corresponding k th submap M_k .

A relative transformations $T_{i,j}$ between between poses i and j is obtained by solving the point clouds registration problem of registering S_j (source) to S_i (target) scans associated with their respective poses. Different registration algorithms variants exist, such as generalized iterative closest point (GICP) (Segal et al., 2009), the NDT point to distribution (P2D by Magnusson 2009), and distribution to distribution (D2D by Stoyanov 2012) variants.

Potential loop-closures $T_{j,i}$ are computed between submaps that are within a range of $d \in R$ away from each other, as measured using the GNSS/INS trajectory. Additionally, the GNSS/INS trajectory is used as an initial guess for the scan registration algorithm using Eq. (10). Thus we have a large set of transformations between keyframes, $T_M = \{T_{i,j}\}$, where $i = 1 \dots N$ and $j = 1 \dots N$. Furthermore, we limit to submap indices $i < j$ so that the same transformation is computed only once.

Since scan registration is prone to getting stuck in local minima, thus returning an erroneous transformation estimate, a strategy of finding successful estimates, according to some measure, is employed. Different potential measures exist to measure point clouds alignment, such as the NDT score-function or root-mean-square between the source and target clouds (Almqvist et al., 2018). Alternatively, the entropy of point clouds before and after registration can be compared (Adolfsson et al., 2021). In the aforementioned methods a threshold level needs to be

set, for example, by training a classifier via machine learning to classify registration results as failed or successful. However, this necessitates having labeled results to train the model parameters. Additionally, different environments and varying degrees of overlap between source and target clouds limits the transferability of the models to different environments or sensor configurations.

Instead of the aforementioned, we adopted the method by Shen et al. (2025). The authors use loops of closed transformations from i through indices j and k back to i to rate each graph edge for consistency

$$\hat{T}_{i,i} := T_{i,j} T_{j,k} T_{k,i}. \quad (11)$$

All loops are searched from the set of registrations between submaps. A loop is considered a success if the translation component $d(\hat{T}_{i,i})$ Eq. (2) and angular deviation $\angle(\hat{T}_{i,i})$ Eq. (3) between the start and end location are within a distance threshold $d_{thr} \in \mathbb{R}$ and $\angle_{thr} \in \mathbb{R}$, respectively. The successes and failures for each edge are counted and individual edges are accepted if their ratio of successes to occurrences exceeds a given ratio $r_{thr} \in \mathbb{R}$.

To further refine the map and trajectory estimates, the problem is solved as a pose graph-SLAM problem of finding an optimal graph, consisting of nodes describing the sensor pose $\mathcal{P}_i \in \text{SE}(3)$ associated with scans S_i , and edges corresponding to constraints on the nodes derived from measurements, i.e., the GNSS/INS solution and scan registration results T_M . To form loops from T_M , the inverse transformations T^{-1} are added to the set as well.

The accepted scan matching results, scan odometry, and GNSS measurements are combined in a factor graph depicted in Figure 3 with pre-defined covariances (Dellaert and Kaess, 2017; Dellaert, 2021). The first pose in the graph is constrained with T_1 given by the GNSS/INS solution. The prior position-only GNSS factors $G_i = t_i$ are added to every n keyframes to constrain the elevation of the solution.

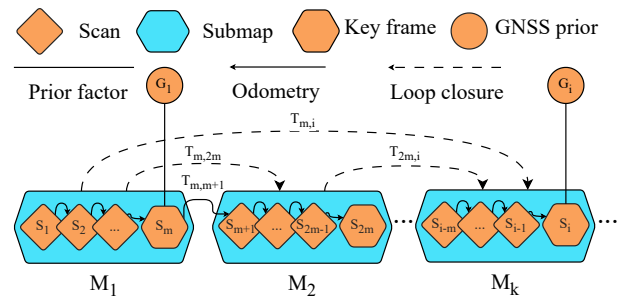


Figure 3. Structure of the factor graph used to solve for the trajectory given GNSS priors, and scan registration estimates.

2.5 Performed experiments

Measurements were collected next to the Oulanka National Park in eastern Finland in September 2024. The USV used in the experiment is a Maritime Robotics Otter provided by the University of Turku, which was controlled to cover the river bend. The measurement lasted 43 minutes during which the USV travelled a distance of 2.2 km moving at a mean speed of 0.84 ms^{-1} . The river was non-turbulent with a flow speed of approximately 0.7 ms^{-1} . A reference trajectory of the USV was collected with a total station and reflector prism installed

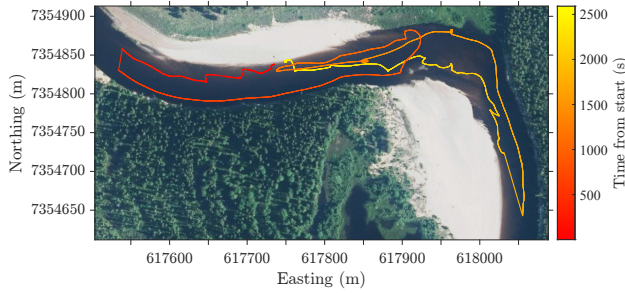


Figure 4. Orthophoto of the survey area with the reference trajectory projected on top. The trajectory is colored based on time. Orthophoto by National Land Survey of Finland.

underneath the GNSS antenna. An orthophoto with the reference trajectory projected on top is shown in Figure 4.

The laser scanner was set to measure at its maximum ≈ 100 lines per second, and the rotating motor was set to 30 RPM. Points x_k were partitioned into scans S_i every full rotation of the scanner. A sliding window and submap length of $m = 5$ was used. GPS measurements were added every $n = 10$ key-frames.

Scan registrations were computed with the NDT D2D variant. Cell sizes of 1.5 m and 1 m, were used for scan odometry and computing the exhaustive matches between submaps, respectively. The point density was not sufficient for smaller cell sizes, especially when registering individual scans due to the sparseness of the data. Consequently, the minimum amount of points to create a NDT cell was set to 4, which is the theoretical minimum. Iteration of the algorithm was terminated after reaching either the maximum number of iterations, 20 for scan odometry and 10 for matching submaps, or a tolerance of 0.001 in the change in the transformation parameters.

Georgia Institute of Technology Smoothing and Mapping library (GTSAM) was used to optimize the graph-SLAM problem with a Levenberg-Marquardt optimizer with default settings (Dellaert and Contributors, 2022). Loops were considered to be consistent with parameters $d_{thr} = 0.03$ m and $\angle_{thr} = 0.5^\circ$. Consecutive edges and edges with a ratio $r \geq 0.7$ were added into the factor graph.

Table 4 gives the standard deviations of the diagonal covariance matrices of the factors added to the graph. The scan-to-scan odometry covariance is denoted by $\Sigma_{T_{i,i+1}}$. The covariances of the submap loop closure factors Σ_{T_M} are set to $\angle_{thr}$, and d_{thr} accordingly. The total processing time was 562 seconds from which pre-processing took 45 seconds, scan registration 506 seconds, loop-searching 10 seconds, and graph optimization 1 second.

Reference position of the USV was measured using Leica Nova TS60 total station. It automatically follows and measures the distance and angles to a prism mounted under the GNSS antenna in the USV. The Cartesian coordinates of the target are

	α ($^\circ$)	β ($^\circ$)	γ ($^\circ$)	x (m)	y (m)	z (m)
Σ_{T_1}	0.57	0.57	0.57	0.01	0.01	0.01
$\Sigma_{T_{i,i+1}}$	5.00	5.00	5.00	0.30	0.30	0.30
Σ_{T_M}	0.50	0.50	0.50	0.03	0.03	0.03
Σ_G	-	-	-	5.00	5.00	0.05

Table 4. Diagonal elements of the covariance matrices of the factors added to the factor graph.

obtained by the total station from its own known location and orientation, and its measurements are transformed into ETRS-TM35FIN and UTC time. The system can only track one prism and thus we only have reference position measurements and no knowledge about the reference orientation.

2.6 Evaluation metrics

The performance of the methods was evaluated numerically with ground truth values from a total station. The position accuracy of the aligned GNSS and SLAM solutions are compared against each other to evaluate overall consistency of the solution with absolute trajectory error (ATE)

$$ATE = \sqrt{\frac{1}{N} \sum_{i=1}^N \|p_i - \hat{p}_i\|^2}, \quad (12)$$

where N is the length of p_i and $\hat{p}_i$, $\{\hat{p}_i\}_1^M$ is the set of positions in the trajectory being evaluated, and $\{p_i\}_1^M$ is the set of reference positions given by the total station. Positions p_i are 3D vectors of Easting, Northing, and Elevation in a global coordinate system in which Easting and Northing are according to ETRS-TM35FIN (EPSG:3067) and Elevation is the ellipsoidal elevation (EPSG:224). The equation is a position only version ATE of the one presented in (Zhang and Scaramuzza, 2018) as the rotations of the reference positions are unknown. A delay of 0.155 s was estimated and corrected for in the reference trajectory.

As a second evaluation metric, the consistency of the generated point cloud models are evaluated numerically with Rényi's quadratic entropy measure in Eq. (5) using 10 nearest neighbors in the k NN search, and $\sigma = 0.05$. Finally, the generated point cloud models are evaluated qualitatively.

3. Results

We compared the post-processed GNSS/INS (shortened as GNSS) and graph-SLAM (shortened SLAM) solutions using the experiment conducted at Oulanka river. For both, we used all other calibrations, but compared the results with and without the heading correction. Table 5 gives the quantitative results, in which experiments with postfixes A correspond with data without heading correction, and B with the heading correction. As shown in the table, the position only ATE measure used in this work is unaffected by the rotation correction when comparing the two GNSS trajectories. However, GNSS B trajectory has smaller entropy values due to the optimized R_{opt} rotation which suggests the bias remains constant throughout the trajectory.

SLAM A increased the ATE by 21.5% relative to the GNSS trajectory. Conversely, SLAM B decreased the ATE by 2.8%. The only difference is the initial guess provided by the GNSS trajectory, which has been corrected for bias in rotations in case of SLAM B. As a result, the scan registration benefits from better

	GNSS A	GNSS B	SLAM A	SLAM B
ATE (m)	0.191	0.191	0.232	0.186
E_G	1.861	1.271	1.220	1.169

Table 5. Quantitative results in which B uses optimized R_{opt} in Eq (9), and A does not. E_G is the entropy in Eq. (5).

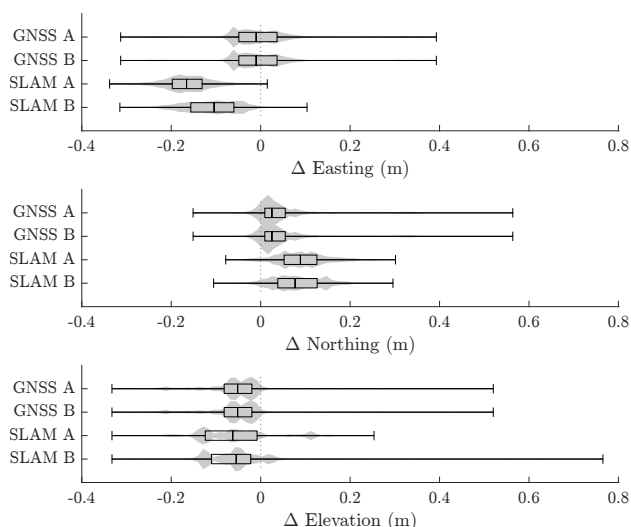


Figure 5. Box and whisker plots of the errors in the GNSS and SLAM trajectories with respect to error in East, North, and Elevation coordinates. The distribution of the data is drawn behind the box and whisker plots.

initialization, which is reflected in the final ATE and entropy E_G values.

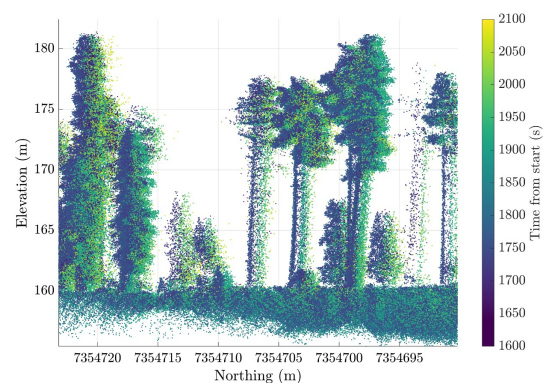
Figure 5 depicts box and whisker plots of the element-wise errors of the GNSS and SLAM trajectories. The SLAM solutions' Easting error means have shifted away from the mean, but the maximum errors have decreased. Likewise the Northing mean errors are slightly shifted, and the maximum errors have decreased. SLAM B Elevation errors are less spread compared with SLAM A, but there is a singular outlier data point increasing the maximum error significantly.

Figure 6 depicts a cross-section of the final point cloud which compares the GNSS/INS, and SLAM results. Versions of the GNSS/INS trajectory with and without heading corrections applied are given in Figure 6a and 6b, respectively. Figure 6a contains clear duplicate targets due to the angular biases. In comparison the trees in Figure 6b are more distinct and better resolved. Both SLAM methods outperform the GNSS/INS solutions as show in Figure 6. Finally, with the heading corrections a slight increase in the model sharpness is observed when comparing SLAM B in Figure 6d to SLAM A in Figure 6c. These qualitative observations are consistent with the entropy values (smaller is better) computed for the chosen sections which also indicate improved consistency.

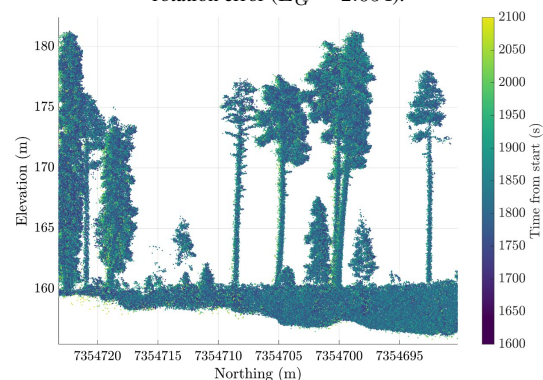
4. Discussion

Several calibration methods were used in the work to reduce errors caused by the extrinsic lever arm offset and rotation from the Lidar frame to the Body frame. The proposed method to solve the orientation between IMU and lidar allowed us to do accurate positioning and mapping using the setup. However, the evaluation of the method is limited due to the additional rotation errors in the post-processed GNSS/INS trajectory, which was tackled separately relying on the Rényi's quadratic entropy. In future work rotation biases should be modeled and estimated within the factor graph.

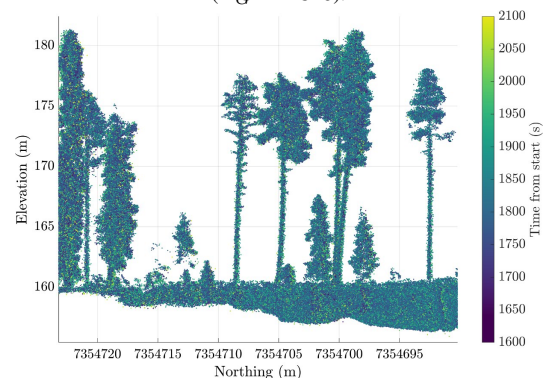
The 19.8% smaller ATE of SLAM B w.r.t SLAM A, seen in Table 5, suggests that errors in the initial guess of the orientation between two frames impacts the scan registration results,



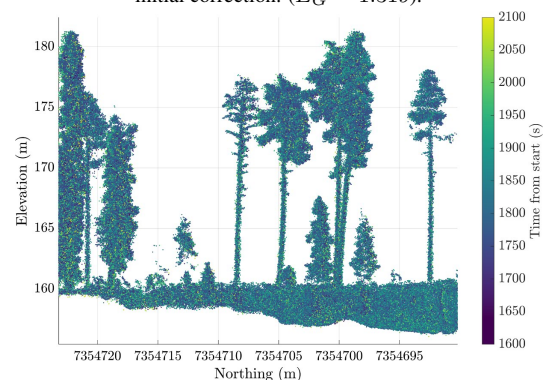
(a) GNSS A: point cloud without correcting trajectory rotation error ($E_G = 2.054$).



(b) GNSS B: point cloud example with corrected rotation ($E_G = 1.340$).



(c) SLAM A: SLAM solution point cloud example without initial correction. ($E_G = 1.319$).



(d) SLAM B: SLAM solution point cloud example with corrected rotation ($E_G = 1.277$).

Figure 6. Qualitative comparison of the GNSS only, heading corrected, and SLAM point cloud models as well as the Quadratic Entropies Eq. (5) of the selected area.

which get stuck in local minima. The performance of different registration algorithms was not explored in this work.

At present there remains challenges in robust scan matching in freshwater environments due to the lack of features and geometry in close proximity to the USV, especially planar features parallel to the xy-plane. As a result, it was necessary to add apriori GNSS measurements to the graph. Despite this, the SLAM Elevation histograms shown in Figure 5 are more spread, albeit reducing maximum errors. Future work should address this via estimating the water level from, for example, the shoreline in order to constrain the elevation when computing scan registration results. The information could also be used to aid vehicle navigation or other sensor systems.

Figure 7 depicts the output of the edge selection algorithm as a heatmap of the ratios of the different edges of computed transformations. Darker areas of the graph correspond with higher loop success to occurrence ratio. Note that the brightness along the diagonal varies which highlights differences in the quantity and quality of features of scans in that particular area. The triangular sections in the middle and end of the graph correspond with periods in time where the USV was holding its position.

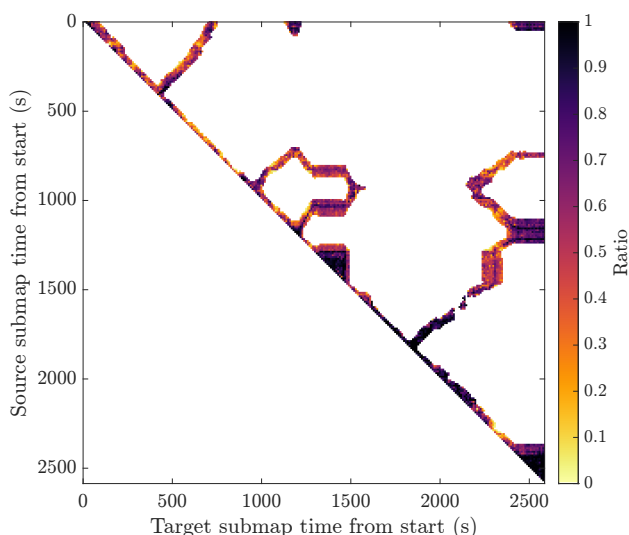


Figure 7. Computed ratios of likes to occurrences for each scan registration between submaps from found loops of length 3.

In Figure 7, it is seen that loop closures are quite rare (a lot of white in the figure) and there exists long chains of consecutive submaps registered together. Even a small bias in the registration will accumulate into large error in the map in this environment with features only available on the surrounding banks.

However, the work showed that the proposed graph-SLAM method improves the consistency of generated point cloud models shown quantitatively in the decrease in Rényi's quadratic entropy, and qualitatively in the visual conformity of features such as trees and ground shown in Figure 6.

Our work also addressed the issue of choosing successful registration results without relying on trained machine learning models. Additionally, the method parameters can be used in the covariance estimates of the factor graph factors, since current scan registration estimates remain overly optimistic. The method leverages the parameters' relation to bound the desired uncertainty in the estimated transformations.

Currently all research was done using a post-processing solutions, which is a limitation if real-time solution is needed. Real-time operation would allow using the sensor for control and navigation of the USV as well as for mapping. The implementation is technically feasible and in the plans as future work.

5. Conclusion

In this paper we have addressed the issue of 3D mapping in boreal freshwater environments using an autonomous USV in conjunction with a unique scanner setup. We used established entropy-based calibration method for calibrating the rotating platform extrinsic parameters and finding a bias in a post-processed GNSS/INS trajectory. Additionally, we introduce a cost function for solving the extrinsic Lidar to Body frame rotation.

The mapping solution is solved with a pose-graph SLAM problem using a post-processed GNSS/INS trajectory as the initial guess, which was refined further with lidar odometry and computing a large set of scan registration results forming loop-closures. Loop-closures were selected into the graph by rating transformations based on their measured consistency by forming larger loops.

The method improved the overall consistency of the point cloud as measured by Rényi's quadratic entropy, and visualizing the generated models. Additionally, given a good initialization of the scan registration, the absolute trajectory error was also observed to decrease.

A limitation of the current system is the lack of active control and obstacle avoidance capabilities, when the USV is equipped with our custom sensor payload. Additionally, the system is limited to slow moving and non-turbid waters in good weather conditions. For future work we propose improving the robustness of scan registration in freshwater environments by devising a constraint on the elevation of the registration solution. Additionally, a real-time implementation of the system is planned to enable obstacle avoidance and further autonomy of the USV.

6. Acknowledgements

Funded by European Union - The Next Generation EU recovery instrument (RFF) through Research Council of Finland projects Hydro RI Platform (Decision no 346162) and Green-Digi-Basin (Decision no 347702), Research Council of Finland Flagship project Digital Waters (Decision no 359249), and European Regional Development Fund through project UOMARI (grant number A80681).

References

- Absil, P.-A., Baker, C. G., Gallivan, K. A., 2007. Trust-region methods on Riemannian manifolds. *Foundations of Computational Mathematics*, 7(3), 303–330.
- Adolfsson, D., Magnusson, M., Liao, Q., Lilienthal, A. J., Andreasson, H., 2021. CorAl – Are the point clouds Correctly Aligned? *2021 European Conference on Mobile Robots (ECMR)*, 1–7.
- Almqvist, H., Magnusson, M., Kucner, T. P., Lilienthal, A. J., 2018. Learning to detect misaligned point clouds. *Journal of Field Robotics*, 35(5), 662–677. <https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21768>.

- Boumal, N., Mishra, B., Absil, P.-A., Sepulchre, R., 2014. Manopt, a Matlab Toolbox for Optimization on Manifolds. *Journal of Machine Learning Research*, 15(42), 1455–1459. <https://www.manopt.org>.
- Chambers, A., Achar, S., Nuske, S., Rehder, J., Kitt, B., Chamberlain, L., Haines, J., Scherer, S., Singh, S., 2011. Perception for a river mapping robot. *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 227–234.
- Dellaert, F., 2021. Factor Graphs: Exploiting Structure in Robotics. *Annual Review of Control, Robotics, and Autonomous Systems*, 4(1), 141–166. <https://www.annualreviews.org/doi/10.1146/annurev-control-061520-010504>.
- Dellaert, F., Contributors, G., 2022. borglab/gtsam.
- Dellaert, F., Kaess, M., 2017. Factor Graphs for Robot Perception. *Foundations and Trends® in Robotics*, 6(1-2), 1–139. <https://www.nowpublishers.com/article/Details/ROB-043>.
- Di Stefano, F., Chiappini, S., Gorreja, A., Balestra, M., Pierdicca, R., 2021. Mobile 3D scan LiDAR: A literature review. *Geomatics, natural hazards and risk*, 12(1), 2387–2429.
- Harpold, A. A., Marshall, J. A., Lyon, S. W., Barnhart, T. B., Fisher, B. A., Donovan, M., Brubaker, K. M., Crosby, C. J., Glenn, N. F., Glennie, C. L., Kirchner, P. B., Lam, N., Mankoff, K. D., McCreight, J. L., Molotch, N. P., Musselman, K. N., Pelletier, J., Russo, T., Sangireddy, H., Sjöberg, Y., Swetnam, T., West, N., 2015. Laser vision: lidar as a transformative tool to advance critical zone science. *Hydrology and Earth System Sciences*, 19(6), 2881–2897. <https://hess.copernicus.org/articles/19/2881/2015/>.
- Kaartinen, H., Hyypä, J., Vastaranta, M., Kukko, A., Jaakkola, A., Yu, X., Pyörälä, J., Liang, X., Liu, J., Wang, Y. et al., 2015. Accuracy of kinematic positioning using global satellite navigation systems under forest canopies. *Forests*, 6(9), 3218–3236.
- Kivioja, T., 2022. Estimating the covariance of point set registration based on the distribution-to-distribution normal distributions transform. Master's thesis, Aalto University, Espoo, Finland.
- Kukko, A., Kaartinen, H., Hyypä, J., Chen, Y., 2012. Multiplatform mobile laser scanning: Usability and performance. *Sensors*, 12(9), 11712–11733.
- Lagarias, J. C., Reeds, J. A., Wright, M. H., Wright, P. E., 1998. Convergence Properties of the Nelder–Mead Simplex Method in Low Dimensions. *SIAM Journal on Optimization*, 9(1), 112–147. <https://epubs.siam.org/doi/abs/10.1137/S1052623496303470>.
- Maddern, W., Harrison, A., Newman, P., 2012. Lost in translation (and rotation): Rapid extrinsic calibration for 2D and 3D LIDARs. *2012 IEEE International Conference on Robotics and Automation*, 3096–3102. ISSN: 1050-4729.
- Magnusson, M., 2009. *The Three-Dimensional Normal-Distributions Transform an Efficient Representation for Registration, Surface Analysis, and Loop Detection*. Örebro University, Örebro. OCLC: 938894806.
- Mäki-Leppilampi, M., 2024. Robotic mapping system for freshwater and forest environments. Master's thesis, Aalto University.
- NovAtel, 2023. Waypoint software 9.00. User Manual v11, Hexagon—NovAtel.
- Oberländer, J., Pfitzer, L., Roennau, A., Dillmann, R., 2015. Fast calibration of rotating and swivelling 3-D laser scanners exploiting measurement redundancies. *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3038–3044.
- Principe, J. C., Xu, D., 1999. Information-theoretic learning using Renyi's quadratic entropy. *Proceedings of the first international workshop on independent component analysis and signal separation*, 1, Citeseer.
- Putkiranta, P., 2020. Geometric calibration of rotating multi-beam lidar systems. Master's thesis, Aalto University.
- Rényi, A., 1961. On measures of entropy and information. *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, 4, University of California Press, 547–562.
- Segal, A., Haehnel, D., Thrun, S., 2009. Generalized-icp. *Robotics: science and systems*, 2, Seattle, WA, 435. Issue: 4.
- Shan, T., Englot, B., Meyers, D., Wang, W., Ratti, C., Rus, D., 2020. LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Las Vegas, NV, USA, 5135–5142.
- Sheehan, M., Harrison, A., Newman, P., 2012. Self-calibration for a 3D laser. *The International Journal of Robotics Research*, 31(5), 675–687. <http://journals.sagepub.com/doi/10.1177/0278364911429475>.
- Shen, J., Yu, H., Wu, J., Yang, W., Xia, G.-S., 2025. LiDAR-enhanced 3D Gaussian Splatting Mapping.
- Stoyanov, T. D., 2012. Reliable autonomous navigation in semi-structured environments using the three-dimensional normal distributions transform (3D-NDT). PhD thesis, Örebro University, Örebro, Sweden.
- Sunderhauf, N., Protzel, P., 2012. Towards a robust back-end for pose graph SLAM. *2012 IEEE International Conference on Robotics and Automation*, IEEE, St Paul, MN, USA, 1254–1261.
- Tomsett, C., Leyland, J., 2019. Remote sensing of river corridors: A review of current trends and future directions. *River Research and Applications*, 35(7), 779–803. <https://onlinelibrary.wiley.com/doi/abs/10.1002/rra.3479>.
- Wang, W., Gheneti, B., Mateos, L. A., Duarte, F., Ratti, C., Rus, D., 2019. Roboat: An Autonomous Surface Vehicle for Urban Waterways. *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 6340–6347. ISSN: 2153-0866.
- Zhang, J., Singh, S., 2014. LOAM: Lidar Odometry and Mapping in Real-time. *Robotics: Science and Systems X*, Robotics: Science and Systems Foundation.
- Zhang, Z., Scaramuzza, D., 2018. A Tutorial on Quantitative Trajectory Evaluation for Visual-(Inertial) Odometry. *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Madrid, 7244–7251.