OpenLiDARMap: Zero-Drift Point Cloud Mapping Using Map Priors

Dominik Kulmer[†][a], Maximilian Leitenstern[b], Marcel Weinmann[c] and Markus Lienkamp[d]

Institute of Automotive Technology, Munich Institute of Robotics and Machine Intelligence,
Technical University of Munich, Garching, Germany
[†]

Keywords: Mapping, Localization, SLAM, Georeferencing.

Abstract: Accurate localization is a critical component of mobile autonomous systems, especially in Global Navigation Satellite Systems (GNSS)-denied environments where traditional methods fail. In such scenarios, environmental sensing is essential for reliable operation. However, approaches such as LiDAR odometry and Simultaneous Localization and Mapping (SLAM) suffer from drift over long distances, especially in the absence of loop closures. Map-based localization offers a robust alternative, but the challenge lies in creating and georeferencing maps without GNSS support. To address this issue, we propose a method for creating georeferenced maps without GNSS by using publicly available data, such as building footprints and surface models derived from sparse aerial scans. Our approach integrates these data with onboard LiDAR scans to produce dense, accurate, georeferenced 3D point cloud maps. By combining an Iterative Closest Point (ICP) scan-to-scan and scan-to-map matching strategy, we achieve high local consistency without suffering from long-term drift. Thus, we eliminate the reliance on GNSS for the creation of georeferenced maps. The results demonstrate that LiDAR-only mapping can produce accurate georeferenced point cloud maps when augmented with existing map priors.

1 INTRODUCTION

Localization is essential for mobile autonomous systems, enabling them to navigate and interact with their environment effectively. In recent years, significant advancements have been made in LiDAR odometry (Vizzo et al., 2023; Zheng and Zhu, 2024b), LiDAR-inertial odometry (Bai et al., 2022; Xu et al., 2022; Zheng and Zhu, 2024a), and Simultaneous Localization and Mapping (SLAM) (Dellenbach et al., 2022; Yifan et al., 2024; Koide et al., 2024b; Pan et al., 2024). Despite their advances, these methods remain prone to drift, especially during long-term operations. While loop closure can help reduce drift by aligning the current position with a previously visited location, it does not guarantee an accurate reconstruction of the intermediate path. Additionally, loops are not always present, especially in linear or open-ended trajectories.

Map-based localization offers an alternative by using pre-existing maps to constrain localization and
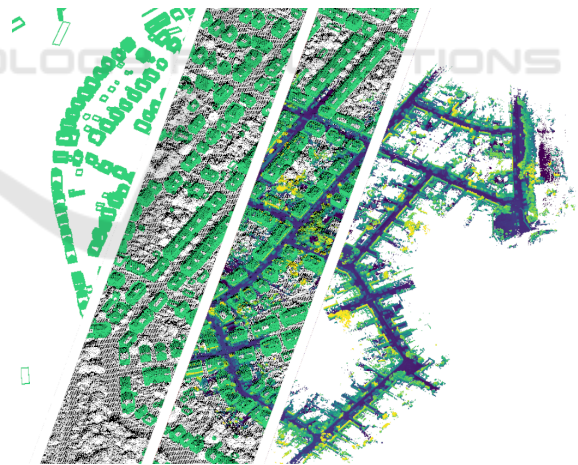

Figure 1: Representation of the different map priors and formats from the building approximations on the left to the final georeferenced point cloud map on the right for KITTI Seq. 00.

eliminate drift. However, these algorithms rely on the availability of maps, which limits their applicability in unknown environments where LiDAR odometry, LiDAR-inertial odometry, and SLAM excel. Open-

[a] https://orcid.org/0000-0001-7886-7550
[b] https://orcid.org/0009-0008-6436-7967
[c] https://orcid.org/0009-0008-7174-4732
[d] https://orcid.org/0000-0002-9263-5323

StreetMap[1] (OSM), a globally available web-based platform, provides maps that can be used for localization with minimal geographic constraints. However, these standard-definition (SD) maps cannot support high-precision localization due to their limited accuracy.

High-definition (HD) maps represent the environment at a resolution of 10 cm to 20 cm (Jeong et al., 2022) and are ideal for precise localization (Koide et al., 2019; Koide et al., 2024a). These maps typically consist of high accuracy point cloud maps. However, the map generation process suffers from the same drift and accuracy issues as the localization itself. Global Navigation Satellite Systems (GNSS) offer a global reference to mitigate drift and enable georeferenced mapping. However, their signals are highly susceptible to degradation in obstructed environments. This vulnerability significantly reduces their reliability for producing high-precision maps.

This work addresses these challenges by proposing a method for high-definition map generation that combines publicly available building maps and surface models with onboard LiDAR data. By leveraging these resources, the approach eliminates reliance on GNSS and enables the creation of accurate, georeferenced point cloud maps (Figure 1).

The main contribution of this paper is a pose-graph-based optimization algorithm that combines Iterative Closest Point (ICP) scan-to-scan with scan-to-map matching of publicly available building data and sparse surface models to generate accurate georeferenced point cloud maps without the need for GNSS data. While scan-to-map matching regulates long-term drift, scan-to-scan matching maintains local consistency between individual LiDAR scans. It also enables the approach to compensate for missing buildings and outdated map data or to bridge short areas not present on the sparse prior map.

Our approach uses neither learning-based methods, feature extraction techniques, nor loop closures. We show that the approach works with a single parameter set on different platforms, like vehicles and segways, with different LiDAR setups, ranging from a single 32-channel LiDAR to a modern multi-LiDAR setup, and in various environments, such as residential and rural regions.

In sum, we make four claims: Our approach is able to (I) map long sequences without accumulating drift over time; (II) automatically georeference the generated map without GNSS data; (III) keep a high local consistency of the generated map; (IV) yield promising results on multiple robotic platforms, LiDAR setups and environments without further tuning.

---

[1]www.openstreetmap.org

To build on our work, we provide an open-source implementation at: https://github.com/TUMFTM/OpenLiDARMap

2 RELATED WORK

When available, high-precision RTK-GNSS signals are often used for georeferenced mapping, requiring precise time synchronization between sensors and optimal signal reception along the mapped route. However, GNSS signal interruptions, such as in urban canyons or underpasses, pose a challenge to continuous mapping. Hybrid pipelines have been proposed to deal with signal interruption. These typically involve creating an initial map using LiDAR odometry or SLAM, followed by post-processing to georeference the map using GNSS data (Leitenstern et al., 2024). While this approach can bridge short gaps without GNSS, it remains dependent on intermittent signal availability. SLAM techniques fused with GNSS signals can also produce georeferenced maps (Koide et al., 2019; Cramariuc et al., 2023; Dellaert, 2022), but require careful tuning of sensor weights and suffer from the same dependency on GNSS signal quality.

OSM represents an alternative to GNSS for global localization and mapping, relying on widely available crowdsourced map data. (Floros et al., 2013) improved global localization accuracy by combining visual odometry with Monte Carlo localization, using chamfer matching to align trajectories with OSM maps. (Suger and Burgard, 2017) proposed a probabilistic navigation method that aligns 3D-LiDAR sensor data with OSM tracks using semantic terrain information and a Markov-Chain Monte Carlo framework. (Yang et al., 2017) introduced a Gaussian-Gaussian cloud model for visual odometry, where OSM road constraints help mitigate drift and resolve scale ambiguities. (Yan et al., 2019) used OSM to create orthophoto-style images of roads and building footprints and generated semantic descriptors to match LiDAR data. (Ballardini et al., 2021) introduced a localization method that detects building facades using stereo image-based point clouds and matches them with 3D building models from OSM. Similarly, (Cho et al., 2022) computed angular distances to buildings within OSM and produced descriptors that match LiDAR data to achieve localization. (Elhousni et al., 2022) used a particle filter to integrate LiDAR point clouds with OSM constraints such as road boundaries, improving accuracy by exploiting map geometry. (Frosi et al., 2023) extended this concept by combining SLAM with OSM priors, integrating 2D

building geometry into trajectory estimation through LiDAR scan matching.

These methods highlight the diverse applications of OSM in reducing drift for mapping and localization in the absence of GNSS signals. However, challenges such as limited accuracy and dependence on map accuracy remain.

Satellite or aerial imagery is another alternative for georeferenced mapping by aligning ground-level observations with overhead map features. (Miller et al., 2021) presented a cross-view localization framework that leverages semantic LiDAR point clouds alongside top-down RGB satellite imagery for georeferenced mapping. Similarly, (Xia et al., 2024) refined cross-view localization techniques to operate effectively in areas lacking fine-grained ground truth, relying on coarse satellite or map data to improve accuracy. However, the inherent misalignment between ground observations and aerial imagery remains a persistent challenge, often resulting in significant localization errors.

Pure LiDAR odometry and SLAM methods lack georeferencing, while GNSS-based approaches remain vulnerable to environmental interference and signal loss. OSM-based methods, while independent of GNSS, rely on the accuracy of crowd-sourced maps that may be outdated or inaccurate. Satellite data provides broad coverage but often suffers from alignment problems between ground and aerial imagery.

These limitations highlight the need for novel approaches that can achieve georeferenced mapping without relying on GNSS.

3 GEOREFENCED POINT CLOUD MAPPING

In this section, we explain our LiDAR-based point cloud mapping approach. The main idea is to combine scan-to-scan matching with scan-to-map matching of reference maps of publicly available building information and sparse surface models in a pose-graph optimization framework. The advantage of our method is that we maintain high local consistency while eliminating long-term drift without the need for additional sensors. Our mapping procedure can be summarized in five steps (Figure 3); while the first one is done beforehand, the others are performed for each input frame:

1. Generating a sparse reference point cloud map from openly available building data and surface models.

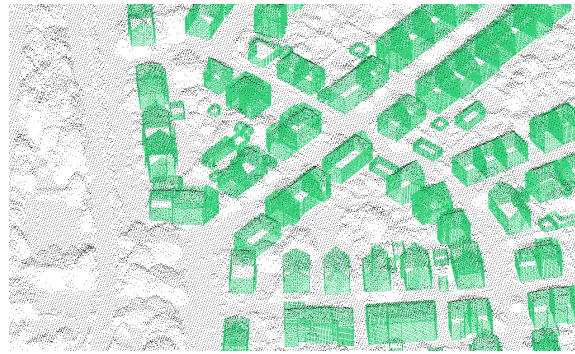2. Scan matching of the current LiDAR scan and the

Figure 2: Exemplary illustration of a combined sparse point cloud map of (green) approximated building data and (black) the surface model with a ground sampling distance of 1 m around the starting position of KITTI Seq. 00.

sparse reference map.

3. Scan matching of the current LiDAR scan and a local submap of previous LiDAR scans.

4. Performing a graph optimization of the resulting poses.

5. Estimating the initial guess for the next LiDAR scan from a constant distance and rotation assumption.

3.1 Generating Sparse Reference Maps

OSM provides, besides road infrastructure data, building outlines for much of the world. In addition to geometric outlines, OSM can store semantic attributes such as building height or number of stories. Beyond OSM, other publicly available local data sources, such as Germany's open data portal[2], provide detailed building information. From this data, sparse georeferenced three-dimensional point clouds can be generated. Depending on data availability, a simplified building representation is derived from OSM data or more sophisticated building models, such as those from the German open data portal. While building models can directly be approximated as a point cloud due to their inherent spatial representation, OSM data requires an estimation for building heights. We assume a height of 4 m per floor, with a default height of 8 m for untagged buildings. The resulting point cloud approximates the building shape using a tessellation with an edge length of 0.5 m.

In addition to building data, three-dimensional surface measurements are available for many regions of the world[2345]. These surface measurements are

---

[2]www.govdata.de

[3]www.data.europa.eu
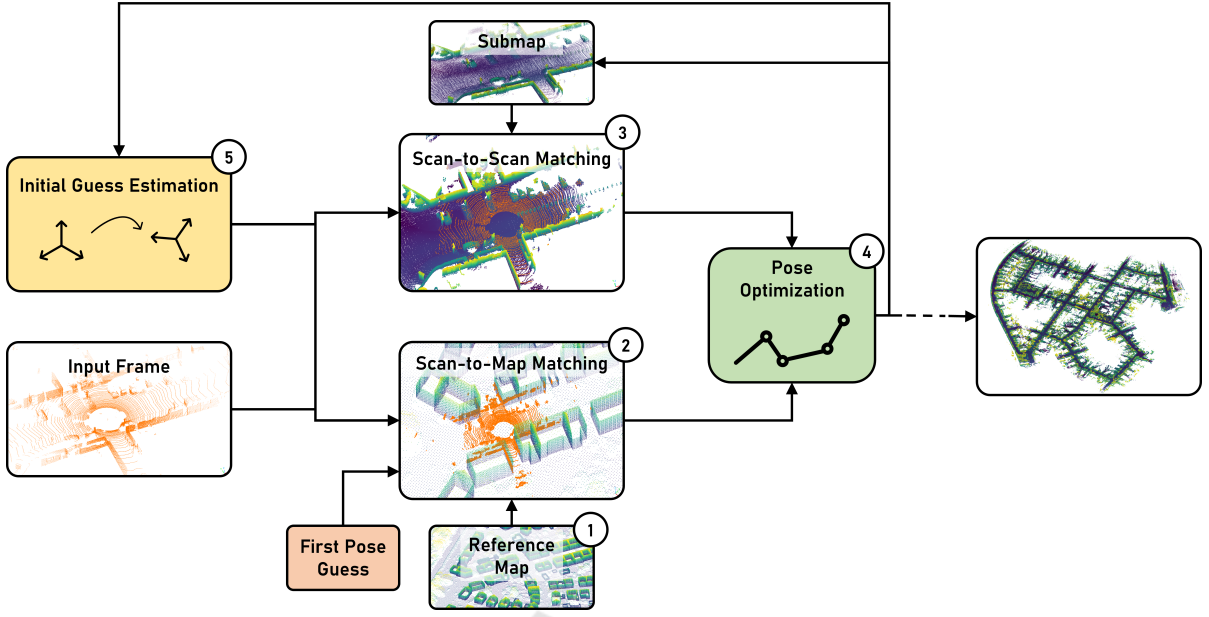
[4]www.data.gov.uk

[5]www.usgs.gov

Figure 3: Pipeline overview that shows our main steps to generate georeferenced point cloud maps. Starting with (1) the sparse reference map, which is used for (2) the scan-to-map matching. (3) shows the scan-to-scan matching. The two results are then optimized with (4) a pose-graph optimization, resulting in the final pose, which is then used for (5) the initial guess estimation for the next LiDAR frame.

typically provided as either surface models or elevation models with varying ground sampling distances, often in the region of 1 m to 5 m. Surface models capture raw LiDAR scans or a downsampled subset, while elevation models represent the terrain alone, excluding vegetation and structures. This data is just another representation of point clouds. Therefore, it can be combined with the building point cloud data to create a sparse georeferenced point cloud map (Figure 2). We use this map as a reference in the following scan-to-map matching.

3.2 Scan-to-Map Matching

Our approach is based on *small_gicp* (Koide, 2024), a lightweight, header-only C++ library designed for point cloud preprocessing and scan matching. We adopt a voxel hash map as our data structure to efficiently handle large point clouds, allowing fast nearest-neighbor searches. The data within each voxel is stored according to the linear iVox principle (Bai et al., 2022).

The current input frame is pre-processed with a voxel-based downsampling before the scan matching. The parameters for our entire processing chain can be taken from subsection 3.6.

A manual initial pose estimate on the map is provided for the first input frame and subsequently refined through scan-to-map matching. For subsequent frames, the initial estimate is derived from the re-

sults of the pose-graph optimization, as described in subsection 3.5. The scan matching process uses the ICP algorithm for its simplicity and robustness. Outliers are managed using the Geman-McClure robust kernel, and the optimization is carried out using the Gauss-Newton method. The scan matching process returns a global pose within our reference map.

3.3 Scan-to-Scan Matching

The same procedure used for scan-to-map matching is applied to scan-to-scan matching, using the same pre-processing, data structure, and optimization.

For scan-to-scan matching, a local submap of the environment is incrementally constructed. The first input frame serves as the registration target for the subsequent scan frame. If the relative transformation between consecutive frames is below 0.1 m, the system assumes a static state. Therefore, no pose is provided for the optimization, and the optimization process for both scan-to-scan and scan-to-map matching is skipped. In dynamic scenarios, as defined by this threshold, the scan-to-scan matching outputs the relative transformation.

After the pose-graph optimization, the current frame is integrated into the local submap using the determined transformation. To maintain a local representation of the environment, any voxels in the submap that are more than 100 m from the current position are removed. From the second frame onward,

scan matching is performed against this evolving local submap rather than solely against the previous frame. This approach is inspired by KISS-ICP (Vizzo et al., 2023) and leverages the initial pose estimation described in subsection 3.5.

3.4 Pose-Graph Optimization

Our approach integrates scan-to-scan and scan-to-map matching into an optimization framework using the Ceres solver (Agarwal and Mierle, 2023). The optimization problem is formulated as a non-linear least squares problem (Equation 1), where residuals are minimized to align the current pose estimate with the scan matching results.

$$\min_x \frac{1}{2} \sum_i \rho_i \left(||f_i(x_{i_1}, \cdots, x_{i_k})||^2 \right) \qquad (1)$$

ρ_i represents a loss function to reduce the influence of outliers on the solution. $f_i(\ )$ is the cost function, which depends on the parameter block $(x_{i_1}, \cdots, x_{i_k})$.

Unlike batch optimization methods, which process all frames simultaneously, the proposed approach employs frame-by-frame optimization. For each frame, the residuals are computed as the deviation between the initial pose estimate and the results obtained from scan-to-scan and scan-to-map matching. Scan-to-map constraints represent absolute pose estimates derived from aligning the current frame to the sparse map. Scan-to-scan constraints represent relative transformations between consecutive frames, linking absolute poses through relative measurements. These assemble a classic pose-graph optimization problem.

To mitigate the influence of outliers, residuals are weighted using a robust loss function to reduce the impact of large deviations on the final solution. We use a Tukey loss to aggressively suppress large deviations for the scan-to-map constraints. For the scan-to-scan matching, we use the softer Cauchy loss. These weights reduce the influence of outliers, which are particularly likely in scan-to-map matching due to deviations between the sparse reference map and the current LiDAR frame. To maintain the reliability of scan-to-map constraints, they are only incorporated into the optimization process if the number of point inliers exceeds 50 % of the correspondences between the current frame and the sparse map. This criterion ensures that only sufficiently aligned frames contribute to the optimization, mitigating the effect of poor matches.

The optimization framework balances relative constraints from scan-to-scan matching with absolute constraints from scan-to-map matching, providing accurate and drift-corrected pose estimates. The result of the optimization is a georeferenced pose computed from the results of the scan-to-scan and scan-to-map matching.

3.5 Estimating the Initial Guess

We use a constant distance and rotation model for the initial pose estimation. The delta translation between the previous and current position is applied to the current position to estimate the next initial position.

$$\mathbf{t}_{x+1} = \mathbf{t}_x + (\mathbf{t}_x - \mathbf{t}_{x-1}) \qquad (2)$$

For the rotation we use a spherical quaternion extrapolation through slerp (Equation 3) with a scalar of $s = 2$.

$$slerp(\mathbf{q}_{x-1}, \mathbf{q}_x, s) = \mathbf{q}_{x-1}(\mathbf{q}_{x-1}^{-1}\mathbf{q}_x)^s \qquad (3)$$

This straightforward method eliminates the need for time stamping across point clouds or individual points, simplifying the implementation. In addition, the lack of a complex kinematic model enhances the versatility of the approach, allowing seamless application across different robotic platforms without the need to modify the methodology.

3.6 Parameters

We use a single set of parameters for both scan-to-scan and scan-to-map matching, as well as for all subsequent evaluations. This design choice is aimed at simplicity, minimizing the number of parameters and allowing for a more straightforward system configuration.

Each input frame is downsampled with a 1.5 m voxel filter to reduce the computational load. The voxel maps are constructed with a resolution of 1 m and can store up to 10 points per voxel, maintaining a minimum distance of 0.1 m between individual points.

For the nearest neighbor search, we consider 27 neighboring voxels within a 3x3x3 cube around the query point. A correspondence threshold of 6 m is applied to filter out invalid associations during scan matching. For all robust kernels used in the optimization process, we use a static kernel width of 1.0 to ensure consistent outlier handling throughout the pipeline.

4 EXPERIMENTAL EVALUATION

We present our experiments to show the capabilities of our method. The results of our experiments sup-

port our key claims, namely that we can (I) map long sequences without accumulating drift over time; (II) automatically georeference the generated map without GNSS data; (III) keep a high local consistency of the generated map; (IV) yield promising results on multiple robotic platforms, LiDAR setups and environments without further tuning.

4.1 Experimental Setup

We evaluate our method on the three different long-distance datasets KITTI Seq. 00 (Geiger et al., 2012; Geiger et al., 2013), NCLT 2013-01-10 (Carlevaris-Bianco et al., 2016), and EDGAR Campus, each using different robot platforms and sensor setups. The characteristics can be found in Table 1. The KITTI dataset is a widely accepted benchmark for evaluating localization and mapping algorithms. Following the approach demonstrated by IMLS-SLAM (Deschaud, 2018) and later adopted by others, such as KISS-ICP (Vizzo et al., 2023) and CT-ICP (Dellenbach et al., 2022), we apply a correction to the intrinsic calibrations around the vertical axis. Specifically, the point clouds are adjusted by an angle of $0.375°$ to ensure accurate alignment.

The NCLT dataset employs a Segway platform, introducing significant challenges due to the vehicle's high dynamics and diverse environmental conditions. The dataset encompasses outdoor scenes ranging from wide-open parking areas to narrow building passages.

The EDGAR dataset is the longest of the three tested datasets and covers the Technical University of Munich campus in Garching, Germany. It includes a variety of environments, from building complexes and extensive parking areas to country roads, with vehicle speeds ranging from low speeds up to $70\,\mathrm{km\,h^{-1}}$. The route traverses both dense and sparsely built-up areas. Unlike KITTI and NCLT, EDGAR features a multi-LiDAR setup, consisting of two spinning and two solid-state LiDARs, which enhances data richness and coverage. While the first two datasets are freely available, the EDGAR Campus dataset is proprietary. Information about the sensor setup can be found in (Karle et al., 2023; Kulmer et al., 2024).

All vehicles are equipped with RTK-GNSS systems to provide the ground truth data.

The quality of the maps created cannot be answered with a single metric. A distinction is made between global deviations, such as drift, and local inconsistencies.

Table 1: Caracteristic of the datasets used for the evaluation.

Dataset	Length [m]	Frames [#]	Sensor Setup	Scenario
KITTI Seq. 00	3724	4541	Velodyne HDL-64	outdoor, residential, car
NCLT 2013-01-1	1311	5120	Velodyne HDL-32E	outdoor/indoor, residential, segway
EDGAR Campus	5289	9104	2x Ouster OS1-128 2x Seyond Falcon	outdoor, rual/residential, car

4.2 Global Displacement Evaluation

We evaluate our approach using the Absolute Trajectory Error (ATE) and the KITTI metric for the Relative Trajectory Error (RTE). The KITTI metric samples the trajectory into segments of varying lengths, ranging from $100\,\mathrm{m}$ to $800\,\mathrm{m}$, and calculates the average relative translational and rotational errors. While the ATE measures the global deviation of the estimated trajectory from the ground truth, the KITTI metric provides insights into the accuracy over medium-length segment pieces.

We compare our approach with three OSM-based global localization and mapping methods and a satellite imagery-based approach for the ATE evaluation. Additionally, we include KISS-ICP as a LiDAR odometry baseline for comparison. However, open-source code is unavailable for all three of the OSM-based methods, and the repository for the imagery-based method is outdated. As a result, we rely on the published data from their respective published works. To enable a comparison with the ground truth, which is positioned in a local coordinate system originating in the first frame, we transform our georeferenced results into the first frame to obtain the same local representation. Then, we calculate the mean and maximum ATE using the ground truth data with *evo*[6].

The results listed in Table 2 demonstrate that our approach reduces the mean ATE by more than half compared to the next best method. Furthermore, our method achieves a mean ATE that is a magnitude lower than the LiDAR odometry approach. Figure 4 shows that our method accumulates no drift over the traversed distance. Notably, we observe an unusually high deviation in a specific section compared to the rest of the trajectory. Figure 4 (I) and (II) show the point cloud generated from the ground truth data and our approach for this section. We identify significant deviations in the provided "ground truth" data for the two passes of the section, particularly with vertical offsets exceeding $1\,\mathrm{m}$ in some regions. Hence, the ground truth provided by KITTI cannot be regarded as accurate values, affecting the overall evaluation and the metrics presented.

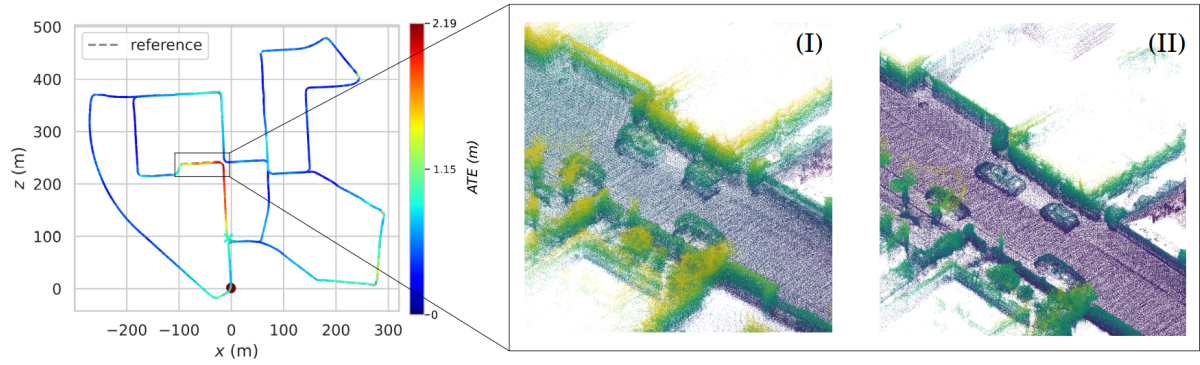[6]https://github.com/MichaelGrupp/evo

Figure 4: Absolute Trajectory Error (ATE) of our approach for KITTI Seq. 00. (I) Point cloud created from the "ground truth" data. (II) Point cloud created with our approach.
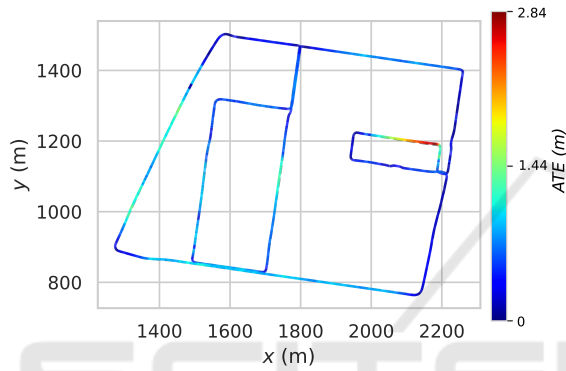

Figure 5: Absolute Trajectory Error (ATE) of our approach for the EDGAR Campus dataset.
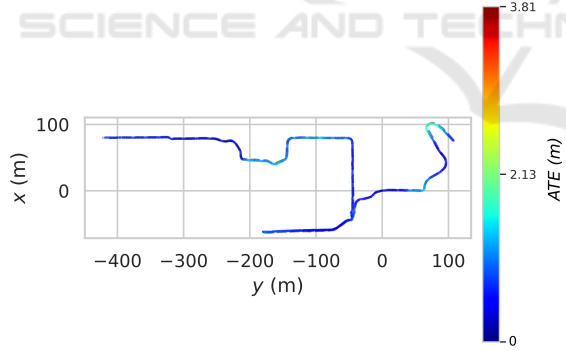

Figure 6: Absolute Trajectory Error (ATE) of our approach for the NCLT 2013-01-1 dataset.

Beyond the KITTI dataset, we demonstrate similar performance for the NCLT and EDGAR datasets, where our method exhibits no drift over time. On the EDGAR dataset, again the largest ATEs occur in the section with bad GNSS signals, most likely due to the signal blockage caused by underpasses and tall building walls (Figure 5). For NCLT, although deviations are the largest, our pose estimation remains robust despite the challenging dynamics and even across an unmapped indoor section (Figure 6).

Figure 7: KITTI Seq. 00 point cloud map, created with our approach, plotted to the orthophoto of Karlsruhe.

We obtain a global pose directly from our approach and only transform it into a local coordinate system for evaluation. On Figure 7, we show the overload of the directly generated map by our approach on georeferenced orthophotos of Karlsruhe[2].

In addition to the ATE, we also evaluated the RTE using the KITTI methodology (Table 3). For the KITTI dataset, our approach is outperformed by the LiDAR odometry algorithm KISS-ICP. This result can be attributed to the high consistency and smooth trajectory generated by KISS-ICP, which minimizes drift over the short averaged segments used in the KITTI metric. However, on the NCLT dataset, which features high dynamics and a challenging environment, our method demonstrates a better performance.

Table 2: Mean and max Absolute Trajectory Error (ATE) in [m] for the evaluated datasets. Bold represents the best results and underscores the second-best results.

Method	Reference	KITTI Seq. 00		NCLT 2013-01-1		EDGAR Campus	
		mean ↓	max ↓	mean ↓	max ↓	mean ↓	max ↓
RC-MVO	(Yang et al., 2017)	3.76	14.01	-	-	-	-
AWYLaI	(Miller et al., 2021)	2.0	12.0	-	-	-	-
LiDAR-OSM	(Elhousni et al., 2022)	1.37	3.34	-	-	-	-
OSM-SLAM	(Frosi et al., 2023)	3.15	11.06	-	-	-	-
KISS-ICP	(Vizzo et al., 2023)	6.71	15.20	3.26	15.99	66.40	259.13
OpenLiDARMap	(ours)	**0.66**	**2.19**	**1.02**	**3.81**	**0.52**	**2.84**

Similarly, the advantages of our approach are evident on the EDGAR dataset, where we maintain consistent trajectories over long distances without drift, leading to lower relative errors.

To assess how well our approach performs compared to pure scan-to-map algorithms, we tried to evaluate the current state of the art on our sparse reference map. However, none of LiLoc (Fang et al., 2024), BM-Loc (Feng et al., 2024), HDL_localization (Koide et al., 2019), DLL (Caballero and Merino, 2021), FAST_LIO_LOCALIZATION[7], and lidar_localization_ros2[8] was able to finish KITTI Seq. 00, leaving us without any results to compare.

4.3 Local Consistency Evaluation

For quantitative evaluation of map consistency, we use the Mean Map Entropy (MME) metric (Razlaw et al., 2015). Specifically, we compare the MME between the point cloud map generated by OpenLiDARMap and the map generated from the ground truth transformations of the individual scans.

To facilitate this comparison, we preprocess the point clouds by cropping them to a 100×100 m region and downsampling individual clouds to a voxel size of 0.25 m. Notably, the composite map remains unfiltered to avoid introducing bias into the evaluation. This preprocessing step is essential to manage the computational complexity caused by the size of the maps.

In the case of the KITTI dataset, we can also use the MME to show that the ground truth is not exact. Our map shows a better (lower) score compared to the ground truth (Table 4). The same can also be shown for the NCLT results, where our approach reaches a lower score than the map generated from the ground truth data. For the EDGAR dataset, the MME generated with the ground truth is lower than the results of our approach.

4.4 Out-of-Date Reference Maps

Finally, we demonstrate that our method does not rely on up-to-date reference maps. Across all datasets, we show that neither current map data nor temporal alignment between the maps and the LiDAR frames is necessary. For instance, the surface data used in the KITTI dataset originates from 2000 to 2023, while the building data stems from 2023, and the evaluated dataset was collected in 2011. Similarly, the NCLT dataset's data sources span from 2013 to 2024, whereas the EDGAR dataset spans from 2023 to 2024.

The accompanying Figure 8 and Figure 9 highlight discrepancies between the sparse maps and the datasets, such as missing buildings or structural differences. These variations emphasize the minimal requirements for input data in our approach. For instance, building outlines for the NCLT dataset were extracted from OSM data, whereas those for KITTI and EDGAR were derived from spatial building models. Moreover, the resolution and type of input data vary significantly: KITTI and NCLT datasets rely on 1 m-resolution GeoTIFFs, while the EDGAR dataset benefits from raw LiDAR point clouds obtained through aerial scans. This diversity highlights the adaptability of our method to sparse maps with significantly different compositions.

As a final note, our approach focuses on mapping, making runtime not a concern for us. Despite that, we are able to run the entire pipeline in about 30 ms per KITTI frame on a modern PC with an AMD Ryzen 7700.

---

[7]https://github.com/HViktorTsoi/fast_lio_localization

[8]https://github.com/rsasaki0109/lidar_localization_ros2

Table 3: Relative Trajectory Error (RTE) using the KITTI methodology for the evaluated datasets.

Method	Reference	KITTI Seq. 00		NCLT 2013-01-1		EDGAR Campus	
		trans. ↓ [%]	rot. ↓ [deg/m]	trans. ↓ [%]	rot. ↓ [deg/m]	trans. ↓ [%]	rot. ↓ [deg/m]
KISS-ICP	(Vizzo et al., 2023)	**0.51**	**0.0017**	2.31	0.0161	2.77	0.0095
OpenLiDARMap	(ours)	0.53	0.0025	**1.93**	**0.0116**	**1.44**	**0.0046**

Table 4: Mean Map Entropy for the frames transformed with the ground truth data and OpenLiDARMap results. Lower ↓ is better.

Method	KITTI Seq. 00	NCLT 2013-01-1	EDGAR Campus
Ground Truth	-6.293	-6.212	**-6.317**
OpenLiDARMap (ours)	**-6.448**	**-6.339**	-6.231

Figure 9: Overload of the sparse reference map and the final point cloud map created with our approach for the EDGAR Campus dataset. Highlighted is a part of the map that still shows (I) the start of construction with multiple cranes in the reference map, while (II) the generated map from the LiDAR frames shows the completed building facades.
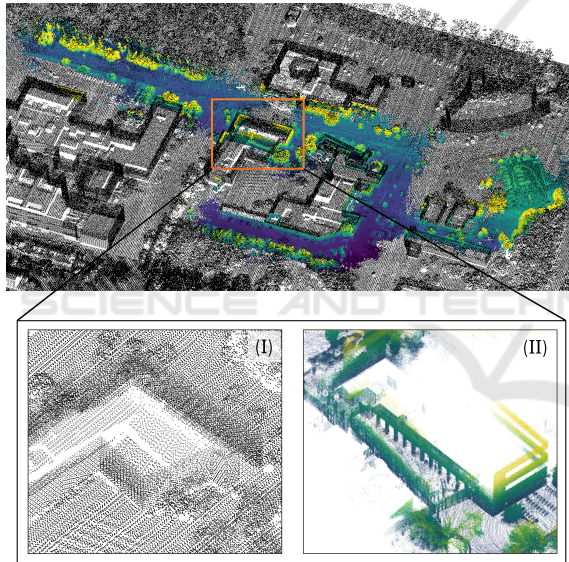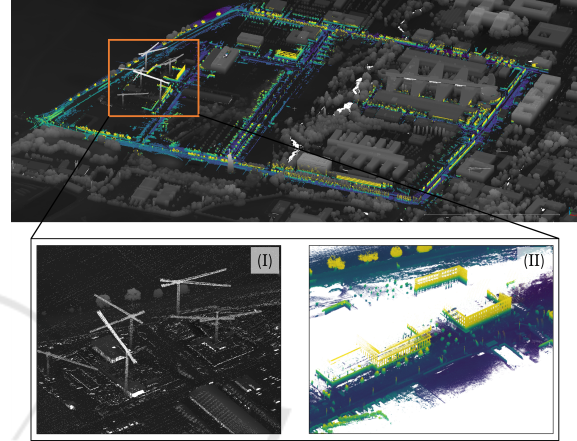

Figure 8: Overload of the sparse reference map and the final point cloud map created with our approach for the NCLT 2013-01-1 dataset. (I) shows part of the reference map, and (II) shows the point cloud map generated with our approach for an indoor section of the dataset that we were able to map despite the missing correspondences between the onboard LiDAR frames and the reference map.

5 LIMITATIONS AND FUTURE WORK

Unlike the ATE analysis, where our approach showed clear improvements, the RTE analysis yielded less conclusive results. This result can be primarily at-

tributed to rotational errors, which have a particularly strong impact on the KITTI metric. Future work could address this limitation by improving the initial estimation, such as using homogeneous transforms or incorporating the time differences between frames, as well as using Lie Algebra in the optimization to better account for small displacements.

A key requirement for the functionality of our approach is the availability of a digital surface model. While such data is available for large parts of the world, it is not universally accessible. A notable example is South Korea, where the lack of digital surface models prevents us from evaluating our approach on widely used datasets such as MulRan (Kim et al., 2020) or HeLiPR (Jung et al., 2024).

Due to the simple ICP-based scan-to-scan matching strategy, our method is limited in handling unmapped regions, such as long tunnels or extensive indoor areas. However, the modular design of our approach allows for future integration of more advanced LiDAR Odometry algorithms to address these challenges. Additionally, significant environmental

changes can lead to erroneous results, emphasizing the need for further investigations across diverse scenarios to evaluate robustness under varying conditions.

6 CONCLUSIONS

In this paper, we presented an approach for georeferenced point cloud mapping that operates without the need for GNSS. By combining a scan-to-map and scan-to-scan point cloud registration method within an optimization framework, we achieved accurate mapping across diverse environments, vehicles, and LiDAR setups. Our experiments suggest that the proposed method effectively eliminates drift over long distances and is robust to variations in environmental conditions and sensor configurations. Our experiments suggest that the proposed method enables accurate georeferenced point cloud mapping without relying on GNSS sensors.

ACKNOWLEDGEMENTS

REFERENCES

Agarwal, S. and Mierle, K. (2023). Ceres solver. https://github.com/ceres-solver/ceres-solver.

Bai, C., Xiao, T., Chen, Y., Wang, H., Zhang, F., and Gao, X. (2022). Faster-lio: Lightweight tightly coupled lidar-inertial odometry using parallel sparse incremental voxels. *IEEE Robotics and Automation Letters*, 7:4861–4868.

Ballardini, A. L., Fontana, S., Cattaneo, D., Matteucci, M., and Sorrenti, D. G. (2021). Vehicle localization using 3d building models and point cloud matching. *Sensors*, 21:5356.

Caballero, F. and Merino, L. (2021). Dll: Direct lidar localization. a map-based localization approach for aerial robots. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5491–5498. IEEE.

Carlevaris-Bianco, N., Ushani, A. K., and Eustice, R. M. (2016). University of michigan north campus long-term vision and lidar dataset. *The International Journal of Robotics Research*, 35:1023–1035.

Cho, Y., Kim, G., Lee, S., and Ryu, J.-H. (2022). Openstreetmap-based lidar global localization in urban environment without a prior lidar map. *IEEE Robotics and Automation Letters*, 7:4999–5006.

Cramariuc, A., Bernreiter, L., Tschopp, F., Fehr, M., Reijgwart, V., Nieto, J., Siegwart, R., and Cadena, C. (2023). maplab 2.0 – a modular and multi-modal mapping framework. *IEEE Robotics and Automation Letters*, 8:520–527.

Dellaert, F. (2022). borglab/gtsam. https://github.com/borglab/gtsam, 10.5281/zenodo.5794541.

Dellenbach, P., Deschaud, J.-E., Jacquet, B., and Goulette, F. (2022). Ct-icp: Real-time elastic lidar odometry with loop closure. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 5580–5586. IEEE.

Deschaud, J.-E. (2018). Imls-slam: Scan-to-model matching based on 3d data. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2480–2485. IEEE.

Elhousni, M., Zhang, Z., and Huang, X. (2022). Lidar-osm-based vehicle localization in gps-denied environments by using constrained particle filter. *Sensors*, 22:5206.

Fang, Y., Li, Y., Qian, K., Tombari, F., Wang, Y., and Lee, G. H. (2024). Liloc: Lifelong localization using adaptive submap joining and egocentric factor graph.

Feng, Y., Jiang, Z., Shi, Y., Feng, Y., Chen, X., Zhao, H., and Zhou, G. (2024). Block-map-based localization in large-scale environment. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1709–1715. IEEE.

Floros, G., van der Zander, B., and Leibe, B. (2013). Openstreetslam: Global vehicle localization using openstreetmaps. In *2013 IEEE International Conference on Robotics and Automation*, pages 1054–1059. IEEE.

Frosi, M., Gobbi, V., and Matteucci, M. (2023). Osm-slam: Aiding slam with openstreetmaps priors. *Frontiers in Robotics and AI*, 10.

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32:1231–1237.

Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE.

Jeong, J., Yoon, J. Y., Lee, H., Darweesh, H., and Sung, W. (2022). Tutorial on high-definition map generation for automated driving in urban environments. *Sensors*, 22:7056.

Jung, M., Yang, W., Lee, D., Gil, H., Kim, G., and Kim, A. (2024). Helipr: Heterogeneous lidar dataset for inter-lidar place recognition under spatiotemporal variations. *The International Journal of Robotics Research*, 43:1867–1883.

Karle, P., Betz, T., Bosk, M., Fent, F., Gehrke, N., Geisslinger, M., Gressenbuch, L., Hafemann, P., Huber, S., Hübner, M., Huch, S., Kaljavesi, G., Kerbl, T., Kulmer, D., Maierhofer, S., Mascetta, T., Pfab, F., Rezabek, F., Rivera, E., Sagmeister, S., Seidlitz, L., Sauerbeck, F., Tahiraj, I., Trauth, R., Uhlemann, N., Würsching, G., Zarrouki, B., Althoff, M., Betz, J., Bengler, K., Carle, G., Diermeyer, F., Ott, J., and Lienkamp, M. (2023). Edgar: An autonomous driving research platform – from feature development to real-world application.

Kim, G., Park, Y. S., Cho, Y., Jeong, J., and Kim, A. (2020). Mulran: Multimodal range dataset for urban place recognition. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6246–6253. IEEE.

Koide, K. (2024). small_gicp: Efficient and parallel algorithms for point cloud registration. *Journal of Open Source Software*, 9:6948.

Koide, K., Miura, J., and Menegatti, E. (2019). A portable three-dimensional lidar-based system for long-term and wide-area people behavior measurement. *International Journal of Advanced Robotic Systems*, 16.

Koide, K., Oishi, S., Yokozuka, M., and Banno, A. (2024a). Tightly coupled range inertial localization on a 3d prior map based on sliding window factor graph optimization. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1745–1751. IEEE.

Koide, K., Yokozuka, M., Oishi, S., and Banno, A. (2024b). Glim: 3d range-inertial localization and mapping with gpu-accelerated scan matching factors. *Robotics and Autonomous Systems*, 179:104750.

Kulmer, D., Tahiraj, I., Chumak, A., and Lienkamp, M. (2024). Multi-lica: A motion- and targetless multi - lidar-to-lidar calibration framework. In *2024 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 1–7. IEEE.

Leitenstern, M., Sauerbeck, F., Kulmer, D., and Betz, J. (2024). Flexmap fusion: Georeferencing and automated conflation of hd maps with openstreetmap.

Miller, I. D., Cowley, A., Konkimalla, R., Shivakumar, S. S., Nguyen, T., Smith, T., Taylor, C. J., and Kumar, V. (2021). Any way you look at it: Semantic crossview localization and mapping with lidar. *IEEE Robotics and Automation Letters*, 6:2397–2404.

Pan, Y., Zhong, X., Wiesmann, L., Posewsky, T., Behley, J., and Stachniss, C. (2024). Pin-slam: Lidar slam using a point-based implicit neural representation for achieving global map consistency. *IEEE Transactions on Robotics*, 40:4045–4064.

Razlaw, J., Droeschel, D., Holz, D., and Behnke, S. (2015). Evaluation of registration methods for sparse 3d laser scans. In *2015 European Conference on Mobile Robots (ECMR)*, pages 1–7. IEEE.

Suger, B. and Burgard, W. (2017). Global outer-urban navigation with openstreetmap. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1417–1422. IEEE.

Vizzo, I., Guadagnino, T., Mersch, B., Wiesmann, L., Behley, J., and Stachniss, C. (2023). Kiss-icp: In defense of point-to-point icp – simple, accurate, and robust registration if done the right way. *IEEE Robotics and Automation Letters*, 8:1029–1036.

Xia, Z., Shi, Y., Li, H., and Kooij, J. F. P. (2024). *Adapting Fine-Grained Cross-View Localization to Areas Without Fine Ground Truth*, pages 397–415.

Xu, W., Cai, Y., He, D., Lin, J., and Zhang, F. (2022). Fastlio2: Fast direct lidar-inertial odometry. *IEEE Transactions on Robotics*, 38:2053–2073.

Yan, F., Vysotska, O., and Stachniss, C. (2019). Global localization on openstreetmap using 4-bit semantic descriptors. In *2019 European Conference on Mobile Robots (ECMR)*, pages 1–7. IEEE.

Yang, S., Jiang, R., Wang, H., and Ge, S. S. (2017). Road constrained monocular visual localization using gaussian-gaussian cloud model. *IEEE Transactions on Intelligent Transportation Systems*, 18:3449–3456.

Yifan, D., Zhang, X., Li, Y., You, G., Chu, X., Ji, J., and Zhang, Y. (2024). Cellmap: Enhancing lidar slam through elastic and lightweight spherical map representation.

Zheng, X. and Zhu, J. (2024a). Traj-lio: A resilient multilidar multi-imu state estimator through sparse gaussian process.

Zheng, X. and Zhu, J. (2024b). Traj-lo: In defense of lidar-only odometry using an effective continuous-time trajectory. *IEEE Robotics and Automation Letters*, 9:1961–1968.