Research Article

A Minimal Subset Approach for Efficient and Scalable Loop Closure

Nikolaos Stathoulopoulos[1], Christoforos Kanellakis[1], George Nikolakopoulos[1]

1. Robotics and AI Group, Department of Computer, Electrical and Space Engineering, Lulea University of Technology, Sweden

Loop closure detection in large-scale and long-term missions can be computationally demanding due to the need to identify, verify, and process numerous candidate pairs to establish edge connections for the pose graph optimization. Keyframe sampling mitigates this by reducing the number of frames stored and processed in the back-end system. In this article, we address the gap in optimized keyframe sampling for the combined problem of pose graph optimization and loop closure detection. Our Minimal Subset Approach (MSA) employs an optimization strategy with two key factors, redundancy minimization and information preservation, within a sliding window framework to efficiently reduce redundant keyframes, while preserving essential information. This method delivers comparable performance to baseline approaches, while enhancing scalability and reducing computational overhead. Finally, we evaluate MSA on relevant publicly available datasets, showcasing that it consistently performs across a wide range of environments, without requiring any manual parameter tuning.

Corresponding author: Nikolaos Stathoulopoulos, niksta@ltu.se

1. Introduction

Simultaneous Localization and Mapping (SLAM) in large-scale and long-term missions presents significant challenges, particularly in maintaining accuracy and efficiency as the environment grows over time. A key component of SLAM is Pose Graph Optimization (PGO), which ensures consistent mapping by optimizing the global trajectory. However, this process relies heavily on loop closure detection, to reduce accumulated drift, which is computationally expensive, especially in large-scale environments, due to the vast number of candidate pairs that need to be identified, verified, and processed to establish node connections in the pose graph. As the mission progresses, the number of loop closure candidates grows rapidly, leading to significant delays in processing times, with some datasets generating over 150,000 candidates, requiring four and a half hours to process, exceeding that of the mission time[1]. This complexity is further compounded by the computational intensity of data association and sequence matching during map optimization, where the processing time can range from seconds to several minutes, depending on the size of the data[2][3].

Due to these computational challenges, many SLAM systems in large-scale environments have opted to either avoid or minimize loop closure detection to reduce the burden on limited computational resources. Some approaches, particularly in expansive environments, forego loop closures entirely, as the growing number of candidates becomes increasingly difficult to handle in real-time applications, as highlighted recently in[4]. While loop closure detection is critical for correcting drift

and ensuring map consistency, the computational complexity can overwhelm systems if not managed effectively, often leading to false positives that degrade the overall quality of the map[5].

A. Related Work

Efficient implementations of pose graph optimization, such as GTSAM[6], g2o[7] and iSAM2[8] have significantly enhanced its scalability by leveraging factor graphs and incremental optimization techniques. Additionally, sparsification methods[9][10][11] and hierarchical approaches[12][13] have further improved computational efficiency by reducing the number of active nodes in the optimization process. However, despite these advances, loop closure detection remains a major bottleneck due to the large number of candidate pairs that must be checked and processed to add meaningful constraints to the pose graph.

Keyframe sampling helps reduce the computational demands of loop closure detection by limiting the number of frames stored and processed in SLAM systems. In LiDAR-(Inertial) Odometry, systems such as LIO-SAM[14][15][16] or back-end global optimization systems like LAMP[17][18], use fixed Euclidean distance intervals (1–2 meters) for keyframe generation to maintain local maps and optimize global poses. Although effective, these methods lack adaptability in dynamic environments with varying densities of loop closure candidates. Recent advancements have introduced adaptive keyframe sampling techniques, adjusting intervals based on environmental spaciousness[19][20], although manual threshold adjustment is still required. Entropy-based methods, like the one proposed by[21], use information theory to select keyframes, but face similar adaptability issues. Other approaches, such as[22], focus on displacement vector similarity to balance computational cost and map completeness while managing keyframes in a sliding window.

In this article, we address the gap in optimized keyframe sampling for pose graph optimization and loop closure detection by streamlining keyframe extraction. Our Minimal Subset Approach uses two key criteria, *redundancy minimization* and *information preservation*, within a sliding window framework to efficiently reduce redundant keyframes while maintaining essential information. This method enhances adaptability across various environments and eliminates the need for manual threshold tuning, with an example of its performance illustrated on Fig. 1.

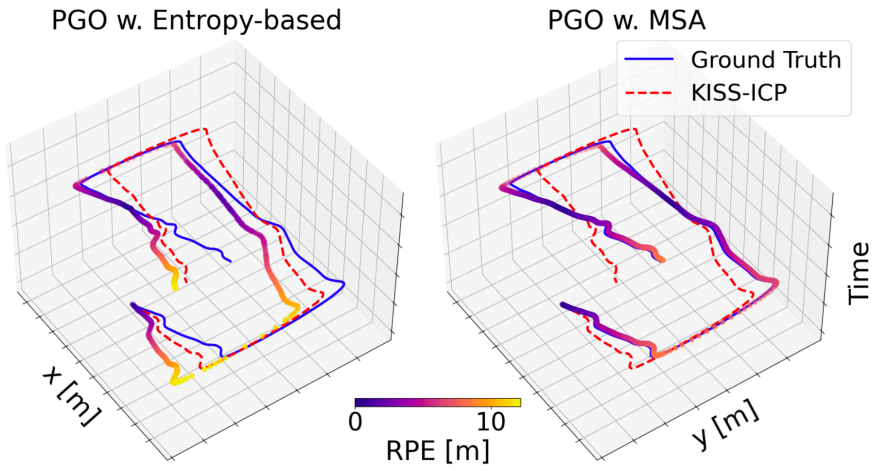

Figure 1. An example on KITTI 06. Comparison of two sampled trajectories: one using an entropy-based approach and the other using the proposed Minimal Subset Approach (MSA). The MSA-sampled trajectory achieves a lower Relative Pose Error (RPE) after the pose graph optimization.

B. Contributions

Based on the aforementioned, the contributions of this article can be summarized as follows: (a) We propose a Minimal Subset Approach (MSA) that efficiently addresses the problem of loop closure detection and pose graph optimization by leveraging a subset of keyframes. This method achieves comparable results to baseline approaches but with improved scalability and efficiency, reducing computational demands. (b) The proposed approach is environment agnostic, avoiding manual parameter tuning and offering consistent performance across diverse environments. This is achieved through a sliding window optimization in the hyper-dimensional descriptor space, eliminating the need for threshold adjustments. (c) The MSA addresses the combined problem of pose graph optimization and loop closure detection. Unlike many existing methods that focus solely on extracting keyframes for scan matching and pose graph optimization, MSA takes into account how sampling intervals impact loop closure detection performance through place recognition, providing a more integrated solution to these interrelated challenges.

II. Problem Formulation

A. Pose Graph Optimization

The objective of PGO is to find the most likely configuration of a robot's trajectory by minimizing the error in a graph-based representation of the poses. The pose graph is represented as a set of nodes and edges, where each node represents a robot pose and each edge encodes a spatial constraint between two poses[23]. Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ be the set of N robot poses, where, $\mathbf{x} \in \mathrm{SE}(3)$, $\mathbf{z}_{ij} = h(\mathbf{x}_i, \mathbf{x}_j)$ be the observed spatial constraint between a pair of poses $\langle i, j \rangle$ and $\hat{\mathbf{z}}_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ be the predicted value of that constraint. The error $\mathbf{e}_{ij} \in \mathbb{R}^6$ quantifies the difference between the observed relative transformation and the predicted transformation between poses $\mathbf{x}_i$ and $\mathbf{x}_j$ and is denoted as:

doi.org/10.32388/6QNWB4

$$\mathbf{e}_{ij} = \mathbf{e}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{z}_{ij} - \hat{\mathbf{z}}_{ij}(\mathbf{x}_i, \mathbf{x}_j) \tag{1}$$

We can then define the set of edges $\mathcal{E} = \{\langle i, j\rangle \mid \exists \mathbf{z}_{ij}\}$, that contains all the pairs in which a relative constraint exists. The goal of pose graph optimization is to find the optimal configuration of poses $\mathbf{X}^*$ that minimizes the overall error function $E(\mathbf{X})$, denoted as:

$$\mathbf{X}^* = \arg\min_{\mathbf{X}} \sum_{\langle i,j\rangle \in \mathcal{E}} \mathbf{e}(\mathbf{x}_i, \mathbf{x}_j)^\top \mathbf{\Omega}_{ij} \mathbf{e}(\mathbf{x}_i, \mathbf{x}_j), \tag{2}$$

where $\mathbf{\Omega}_{ij} \in \mathbb{R}^{6\times 6}$ is the information matrix that reflects the certainty of the relative pose measurement for each pair.

B. Odometry and Loop Closure Edges

Pose graph optimization is widely used as a global optimization method to mitigate accumulated pose drift during large-scale and long-term missions. This would not be possible without loop closures. Relying solely on pose-to-pose odometry constraints is insufficient to maintain global consistency, hence, loop closures are essential. Given this, it is reasonable to assume that the set of edges $\mathcal{E}$ in the pose graph can be partitioned into two distinct subsets. The first subset, $\mathcal{E}_o \subseteq \mathcal{E}$, consists of *odometry edges*, which represent consecutive movements of the robot based on local motion estimates, while the second subset, $\mathcal{E}_l \subseteq \mathcal{E}$, consists of *loop closure edges*, which correspond to observations of previously visited locations. Thus, the overall error function can be decomposed into two components:

$$E(\mathbf{X}) = \sum_{\langle i,j\rangle \in \mathcal{E}_o} \mathbf{e}_{ij}^\top \mathbf{\Omega}_{ij} \mathbf{e}_{ij} + \sum_{\langle i,j\rangle \in \mathcal{E}_l} \mathbf{e}_{ij}^\top \mathbf{\Omega}_{ij} \mathbf{e}_{ij}, \tag{3}$$

where the edge sets satisfy the following two relationships: $\mathcal{E} = \mathcal{E}_o \cup \mathcal{E}_l$ and $\mathcal{E}_o \cap \mathcal{E}_l = \varnothing$.

C. Loop Closure Detection

Detecting loop closures is challenging, particularly in large-scale missions, where accumulated drift can cause traditional radius-based searches to fail by missing nearby candidates. A common solution is *place recognition*, which compares current sensory data (e.g., images or LiDAR scans) against a database of past observations to identify matches. To formalize this, we define a keyframe set $\mathbf{K} = \{(\mathbf{x}_i, \mathbf{d}_i, \mathbf{z}_i) \mid i = 1, \ldots, N\}$, where $\mathbf{x}_i \in SE(3)$ is the pose of keyframe i, $\mathbf{d}_i \in \mathbb{R}^M$ represents its descriptor which encodes distinct features[24][25], and $\mathbf{z}_i$ is the set of associated measurements. The goal is to find loop closure candidates $\mathcal{L}$ by comparing descriptors:

$$\mathcal{L} = \{\langle i, j\rangle \mid f(\mathbf{d}_i, \mathbf{d}_j) > \tau, \forall \langle \mathbf{k}_i, \mathbf{k}_j\rangle \in \mathbf{K} \times \mathbf{K}, \mathbf{k}_i \neq \mathbf{k}_j\}, \tag{4}$$

where $f(\mathbf{d}_i, \mathbf{d}_j)$ is a similarity function (e.g., cosine similarity, Euclidean distance) and τ is a threshold for considering two keyframes representing the same place. This approach generalizes to various scenarios, such as comparing a current keyframe with the k nearest past keyframes in single-robot detection or comparing keyframes across multiple robots.

D. Reducing Search Space in Place Recognition

The detection and verification of loop closure edges $\mathcal{E}_l \triangleq \mathcal{L}$ is computationally demanding due to the large number of potential keyframe pairs, which scales quadratically as $O(n^2)$, where $n = |\mathbf{K}|$ is the number of keyframes. While using descriptors and methods like k-nearest neighbor (k-NN) search[26] can accelerate the process, additional steps are still required to remove outliers, mitigate false positives, or prioritize certain nodes[1], while extracting the relative pose transformation between loop closure nodes also involves computationally intensive techniques such as General Iterative

Closest Point (GICP)[27] or Sample Consensus Initial Alignment (SAC-IA)[28]. This pipeline does not scale well for extensive missions, often resulting in processing times of 3–4 hours or more[4][17][18], and the challenge becomes even greater in multi-robot systems[2][3].

To address this computational complexity, we propose a *sampling strategy* to reduce the number of keyframes considered for matching, while maintaining minimal impact on the loop closure set $\mathcal{L}$. Let $\mathbf{K}_\mathrm{S} \subset \mathbf{K}$ represent a subset of keyframes selected through a sampling process $S : \mathbf{K} \to \mathbf{K}_\mathrm{S}$ such that $|\mathbf{K}_\mathrm{S}| \ll |\mathbf{K}|$. Our objective is to reduce the search space by confining the loop closure detection process to only the selected keyframes in $\mathbf{K}_\mathrm{S}$. The goal is to identify a reduced set $\mathbf{K}_\mathrm{S}^*$ that minimizes redundant keyframes while preserving the set of loop closures $\mathcal{L}$, a process denoted as:

$$\mathbf{K}_\mathrm{S}^* = \underset{\mathbf{K}_\mathrm{S} \subseteq \mathbf{K}}{\operatorname{argmin}} \left(|\mathbf{K}_\mathrm{S}| \right), \quad \text{subject to } \mathcal{L}^* = \mathcal{L}, \text{ where} \tag{5}$$

$$\mathcal{L}^* = \left\{ \langle i, j \rangle \mid f(\mathbf{d}_i, \mathbf{d}_j) > \tau, \forall \langle \mathbf{k}_i, \mathbf{k}_j \rangle \in \mathbf{K}_\mathrm{S}^* \times \mathbf{K}_\mathrm{S}^* \right\}, \tag{6}$$

In this approach, the overall pose graph optimization is reformulated as follows:

$$\mathbf{X}_\mathrm{S}^* = \underset{\mathbf{X}_\mathrm{S}}{\operatorname{argmin}} \sum_{\langle i,j \rangle \in \mathcal{E}_o^*} \mathbf{e}_{ij}^\top \mathbf{\Omega}_{ij} \mathbf{e}_{ij} + \sum_{\langle i,j \rangle \in \mathcal{E}_l^*} \mathbf{e}_{ij}^\top \mathbf{\Omega}_{ij} \mathbf{e}_{ij}, \tag{7}$$

where $\mathbf{X}_\mathrm{S}$ represents the poses corresponding to the optimally sampled keyframe set $\mathbf{K}_\mathrm{S}^*$, and $\mathcal{E}_o^*, \mathcal{E}_l^* \triangleq \mathcal{L}^*$ denote the odometry and loop closure edge pairs between the sampled keyframe poses $\mathbf{X}_\mathrm{S}$, respectively.

E. Underlying Challenges

Addressing the combined problem of pose graph optimization and loop closure detection through place recognition presents several underlying challenges that affect both the efficiency and accuracy of the solution.

1. Impact of Sampling

While the sampling strategy S aims to preserve key loop closure candidates, in practice it must retain a subset $\overline{\mathcal{L}} \subseteq \mathcal{L}$ that still ensures an effective solution for pose graph optimization. Redundant loop closure edges can lead to issues, such as overfitting, increased computational burden, and numerical instability in the optimization process. Moreover, because the sampling process reduces the set of poses, the odometry edge set $\mathcal{E}_o^* \subset \mathcal{E}_o$ is also reduced, which can affect the overall accuracy of the optimization[29], with an example illustrated on Fig. 2.

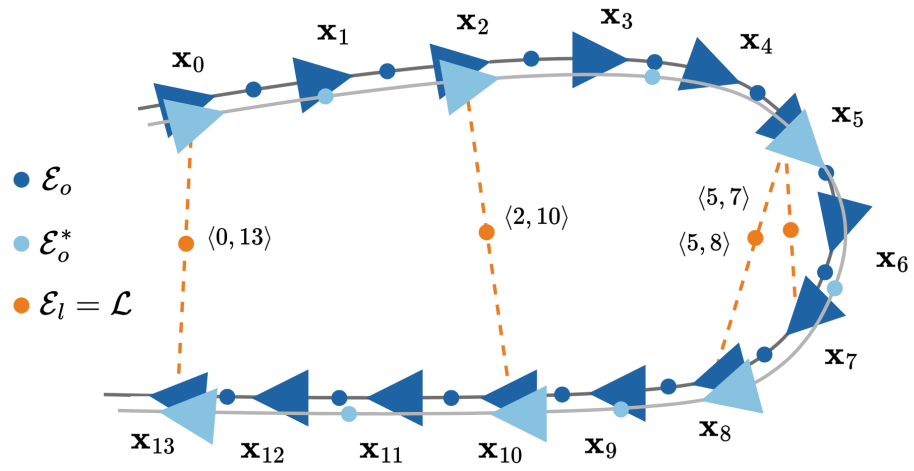

Figure 2. Redundant Keyframes Effect. Example of removing redundant keyframes (light blue). The updated odometry edge set, $\mathcal{E}_o^*$, influences the pose graph optimization, while the sparsified loop closure edges, $\overline{\mathcal{L}} = \mathcal{E}_l \backslash \langle 5, 7 \rangle$, can further reduce the computational complexity.

2. Combined Problem

Combining loop closure detection through place recognition with pose graph optimization presents challenges due to their differing characteristics. Loop closure detection relies on matching high-dimensional feature descriptors, which is computationally expensive and sensitive to environmental factors. Conversely, pose graph optimization relies on geometric consistency and is affected by noise and outliers in loop closures. Balancing these require careful consideration of both feature reliability and constraint robustness to prevent false positives and reduced efficiency from degrading optimization performance.

3. Dynamic Keyframe Selection

To minimize the keyframe set size in real-time, we must consider the entire set $\mathbf{K}$; however, this is impractical for maintaining computational feasibility in loop closure detection. The challenge is to determine the contribution of each keyframe, which requires anticipating future keyframes and introduces a lack of casualty. Moreover, finding the optimal subset $\mathbf{K}_S^*$ is an NP-hard problem, involving an exhaustive search of all keyframe combinations, which becomes computationally infeasible for sets larger than 15–20 keyframes.

The goal of this research is to show that reducing the number of keyframes through an informed sampling strategy can significantly decrease the search space for loop closure detection, improving computational efficiency by reducing memory usage and query time without compromising SLAM accuracy. The proposed approach seeks to balance computational complexity and optimization precision, ensuring scalability and robustness.

III. Minimal Subset Approach

To address these challenges, we propose a Minimal Subset Approach (MSA) to approximate a solution to the problems in Eq. (5)-(6). The proposed MSA identifies and eliminates redundancy within a keyframe set while preserving crucial information. Designed for real-time implementation, it employs a sliding window combinatorial optimization with two distinct objectives, balancing *information preservation* and *redundancy minimization*.

A. Redundancy in Keyframes

The frequency of a sensor, the platform's speed, and the environmental characteristics can cause keyframe samples to capture redundant information if they are too closely spaced, with more details available on [30]. To address this, we define redundancy within a keyframe set and propose a metric to quantify it based on the descriptor space. A keyframe $\mathbf{k}$ is redundant within a set $\mathbf{K}$ if its removal does not affect the loop closure edge set $\mathcal{L}$ and does not create discontinuities in the map representation:

$$\begin{aligned} \overline{\overline{\mathcal{L}}} \equiv \mathcal{L} = \left\{ \langle i, j \rangle \mid f(\mathbf{d}_i, \mathbf{d}_j) > \tau, \forall \langle \mathbf{k}_i, \mathbf{k}_j \rangle \in \overline{\overline{\mathbf{K}}} \times \overline{\overline{\mathbf{K}}} \right\}, \\ \text{subject to } \delta_l \leq \|\mathbf{x}_i - \mathbf{x}_{i+1}\|_2 \leq \delta_u, \forall \mathbf{x} \in \mathbf{X}, \end{aligned} \tag{8}$$

where $\overline{\overline{\mathbf{K}}} = \mathbf{K} \backslash \{\mathbf{k}\}$. The spatial constraint in Eq. (8) ensures comprehensive map coverage and that consecutive odometry edges can be computed safely. The lower and upper pose distances, δ_l and δ_u, usually range from 1 to 5 meters[24]. We quantify redundancy in a keyframe set using a metric that captures the similarity between consecutive keyframes, utilizing any similarity function f_σ, provided by the corresponding descriptor extraction framework:

$$\rho_\tau(\mathbf{K}) = \frac{1}{N-1} \sum_{i=1}^{N-1} f_\sigma(\mathbf{k}_i, \mathbf{k}_{i+1}), \tag{9}$$

where $0 < \rho_\tau(\mathbf{K}) \leq 1$ and N is the number of keyframes in $\mathbf{K}$. Higher values indicate greater redundancy.

B. Information Preservation in Keyframes

Let $F : \mathbb{R}^3 \to \mathbb{R}^M$, represent the descriptor extraction function, either learning-based[24,31] or handcrafted[25,32], which maps each observation from 3D space to an M-dimensional representation. This function depends on the input observation (e.g. LiDAR scan), but it is understood that $\mathbf{z}_t \propto f(\mathbf{x}_t)$ implies that each observation depends on the viewpoint and pose. To understand how the descriptors are sensitive to pose changes, we compute the Jacobian $\mathbf{J}$ of F with respect to poses $\mathbf{x}$. Explicitly deriving the Jacobian can be impractical because of the intricate nature of functions like deep neural networks[30]. Consequently, we employ numerical approximations to estimate the rate of change, as shown in Fig. 3. Moreover, since poses are encompassed within $SE(3)$, we reduce the dimensional complexity by using the Euclidean norm for distance measurement instead of individual axis derivatives. It is essential to use yaw-invariant descriptors, as suggested in[24,25,33], to ensure the Jacobian calculations are meaningful by preventing orientation changes. Considering the descriptors $\mathbf{d} \in \mathbb{R}^M$ as random variables and poses $\mathbf{x} \in SE(3)$ as samples, the product $\mathbf{J}_\mathbf{F}^\top \mathbf{J}_\mathbf{F}$ estimates the covariance matrix:

$$\mathbf{J}_\mathbf{F}^\top \mathbf{J}_\mathbf{F} = \left(\frac{\partial \mathbf{F}}{\partial \mathbf{x}} \right)^\top \left(\frac{\partial \mathbf{F}}{\partial \mathbf{x}} \right) = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^{-1}, \tag{10}$$

where $\mathbf{\Lambda}$ is the diagonal matrix of eigenvalues and $\mathbf{V}$ is the matrix of eigenvectors from the decomposition. The eigenvectors represent the principal directions of variation, and the eigenvalues represent their magnitudes. The descriptors are transformed using the eigenvectors and scaled by the square root of the eigenvalues, $\mathbf{D}' = \sqrt{\mathbf{\Lambda}} \cdot \mathbf{V} \cdot \mathbf{D}$, aligning them with the main directions of variability. Therefore, we can define the *information preservation* term for a keyframe set as:

$$\pi_\tau(\mathbf{K}) = -\frac{1}{N-1} \sum_{i=1}^{N-1} f_\delta(\mathbf{d}'_i, \mathbf{d}'_{i+1}), \tag{11}$$

where $-1 \leq \pi_\tau(\mathbf{K}) < 0$, f_δ is the distance function between descriptors, and $\mathbf{d}' \in \mathbf{D}'$ are the transformed descriptors. Higher values of π_τ indicate that the poses better preserve the variability in the descriptor space. Eigenvectors $\mathbf{v}_k$ denote directions of maximal variability, and the k-th eigenvalue λ_k quantifies the variance described by each eigenvector. Larger eigenvalues indicate more significant patterns of variability.

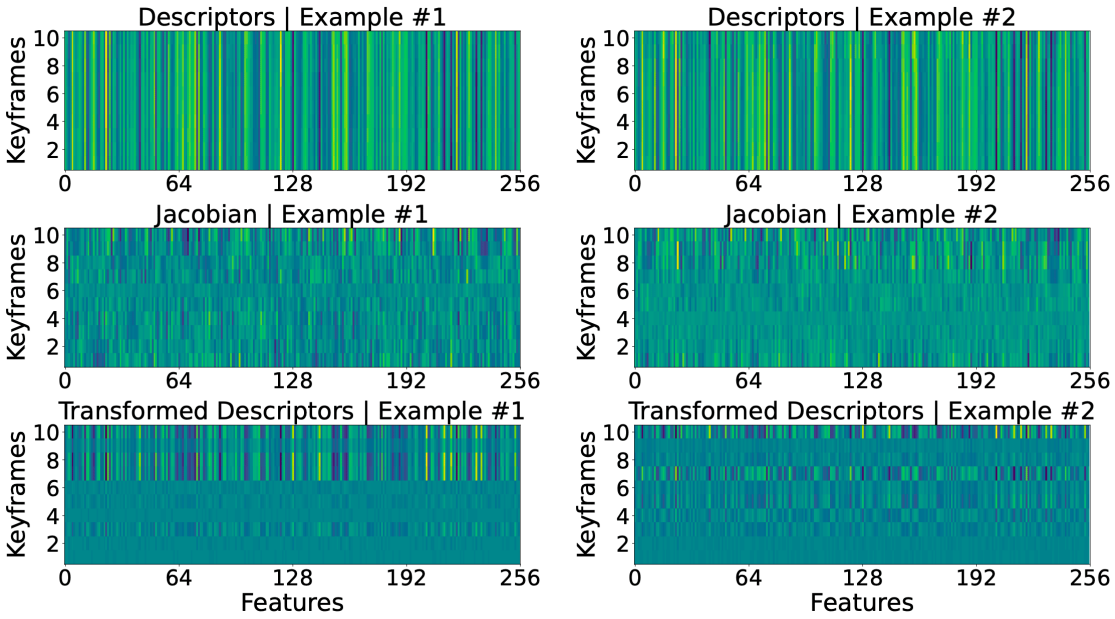

Figure 3. Descriptors across poses. An example showing the descriptors, the Jacobian, and the transformed descriptors for two window keyframe sets. The Jacobian indicates how the features of the descriptors change between keyframes, while the transformed descriptors demonstrate how these same features appear after transformation to the principal components of that set.

In summary, while both ρ_τ and π_τ utilize a similarity or distance function, they differ in objectives: the redundancy term focuses on local redundancy within the keyframe set, while the information preservation term evaluates the preservation of information structure with respect to pose changes. The distinct goals lead to different interpretations of relationships between descriptive vectors within the keyframe set.

doi.org/10.32388/6QNWB4

C. Sliding Window Optimization

To manage the computational complexity of optimizing a large keyframe set and proactively selecting keyframes for future queries, we introduce a sliding window optimization method. This approach continuously optimizes keyframe sets $\mathbf{K}_t$ over the mission duration T, approximating the optimal keyframe set as:

$$\mathbf{K}_S^* \cong \bigcup_{t\in[0,T]} \mathbf{K}_t^*, \tag{12}$$

where $\mathbf{K}_t^*$ is the minimal subset of $\mathbf{K}_t$ that retains maximum information. To compute this subset, we use the redundancy and information preservation terms defined earlier.

The process begins by initializing a window keyframe set $\mathbf{K}_t$ containing N keyframes. The time step t advances once N new keyframes are available, and the optimization converges to the optimal window keyframe set $\mathbf{K}_t^*$. To find this optimum, we generate all possible subsets of $\mathbf{K}_t$, forming the power set $\mathbb{P}(\mathbf{K}_t)$ with cardinality $|\mathbb{P}(\mathbf{K}_t)| = 2^N$. To reduce computational complexity, we apply constraints on the power set, retaining only subsets that satisfy minimum and maximum distances between consecutive poses, as per Eq. (8), reducing the power set by 5–10 times. The constrained power set can be denoted as:

$$\bar{\mathbb{P}}(\mathbf{K}_t) = \left\{ \mathbf{K}_t^{\mathbb{S}} \in \mathbb{P}(\mathbf{K}_t) \mid \delta_l \leq \|\mathbf{x}_i - \mathbf{x}_{i+1}\|_2 \leq \delta_u, \forall \mathbf{k} \in \mathbf{K}_t^{\mathbb{S}} \right\}, \tag{13}$$

where δ_l and δ_u are the lower and upper limits for the distance between poses. The window optimization problem can be formulated as:

$$\mathbf{K}_t^* = \arg\min_{\mathbf{K}_t^{\mathbb{S}}} \left(\rho_\tau(\mathbf{K}_t^{\mathbb{S}}) + \alpha\right) / \left(\pi_\tau(\mathbf{K}_t^{\mathbb{S}}) - \beta\right) \tag{14}$$

$$\text{where } \alpha, \beta > 0 \text{ and } \mathbf{K}_t^{\mathbb{S}} \in \bar{\mathbb{P}}(\mathbf{K}_t). \tag{15}$$

The minimization problem is solved through an exhaustive search, evaluating each subset's information matrix $\mathbf{J}_\mathbf{F}^\mathsf{T}\mathbf{J}_\mathbf{F}$ and quantifying its effectiveness using the information preservation and redundancy terms, selecting the subset with the best combined score.

IV. Experimental Setup and Results

We evaluate our approach using three datasets: `KITTI Odometry`[34], `MulRan`[35], and `Apollo-SouthBay`[36], which cover diverse environments such as urban areas, rural landscapes, and complex structures like bridges and tunnels. Each dataset provides 3D LiDAR scans and GNSS-based ground truth poses, with `MulRan` being particularly challenging due to obstructed LiDAR views. For descriptor extraction, we use `OverlapTransformer (OT)`[24] with a 1×256 feature vector. In[30], we demonstrate the generalizability of our method to other descriptors, like `Scan Context`[25]. Comparisons are made with constant intervals of 1, 2, and 3 meters[14][17][18][15][16], as well as with adaptive sampling based on LiDAR spaciousness[19] and entropy[21]. All experiments were conducted on a 14th Gen Intel Core i9-14900K with 128GB DDR5 RAM. The parameters of MSA are set to the defaults of $\alpha = \beta = 1$ and $N = 10$. The window optimization is solved, on average, in 14.4ms, making it suitable for real-time sampling.

A. Quantitative and Qualitative Analysis

For the batch experiments in Table I, the following pipeline is used: keyframes are sampled for all methods, and a pose graph is constructed using `GTSAM`[6] with odometry provided by `KISS-ICP`[37]. Potential loop closure matches are identified using OT descriptors, with a similarity threshold of 0.8. Ground truth data are used to classify candidates as true or false positives, based on a 1-meter radius. True positives are passed to `small_gicp`[38] for relative transformation estimation, with matches being verified if the registration residual is below 0.3 meters. Verified loop closures are then added as edges to the pose graph, which is optimized using the Levenberg-Marquardt algorithm.

Table I summarizes the results for all datasets and methods, where bold values represent the best performing adaptive method and underlined values indicate the best performing constant interval. The metrics include the Absolute Trajectory Error (ATE) for both translation and rotation, presented as the percent improvement in the `KISS-ICP` trajectory after pose graph optimization with each method. Additionally, the False Positive Ratio (FPR) for descriptor matching is provided, as sampling affects place recognition performance[30]. The results show that the proposed MSA consistently achieves the best performance, while constant intervals show varying results across different algorithms. In Fig. 4, the data from Table I are compiled into box plots, showing the deviation of each method's ATE performance compared to using all samples, along with variations in memory usage and total execution time of the whole pipeline. The proposed approach maintains overall improvements with minimal performance loss in both translation and rotation, while significantly reducing memory usage and providing reasonable execution times. Although other methods reduce memory overhead and processing time, their performance varies considerably, lacking robustness. Finally, Figs. 1 and 5 visually compares ground truth poses, raw KISS-ICP poses, and the corrected trajectories for MSA and the entropy-based method after optimization. For all illustrated trajectories, the proposed approach consistently achieves lower translation and rotation RPE than the other methods, while the segments containing loop closures remain very close to the ground truth, demonstrating the approach's ability to retain crucial keyframes.

Datasets	**KITTI Odometry (Seq. 00 / 06)**			**MulRan (DCC / KAIST)**			**Apollo-SouthBay (SanJose / Columbia)**		
Metrics	**t. ATE [%]**	**rot. ATE [%]**	**FPR [%]**	**t. ATE [%]**	**rot. ATE [%]**	**FPR [%]**	**t. ATE [%]**	**rot. ATE [%]**	**FPR [%]**
All Samples	06.9 / 78.6	-06.1 / 51.1	01.4 / 05.6	29.8 / 67.4	24.9 / 59.5	08.0 / 21.2	96.2 / 98.1	95.3 / 97.5	30.7 / 07.5
Const. 1m	12.2 / 78.7	12.3 / 55.3	02.8 / 07.0	59.3 / 87.1	56.4 / 71.0	20.2 / 23.8	85.8 / 97.1	84.1 / 97.2	25.1 / 04.1
Const. 2m	10.5 / 44.9	06.5 / 23.7	01.3 / 03.1	70.0 / 80.9	68.4 / 73.7	07.6 / 11.6	06.8 / 75.9	06.8 / 77.2	25.3 / 02.3
Const. 3m	05.1 / 22.6	-03.9 / 13.8	03.0 / 09.5	63.6 / 87.7	64.6 / 79.1	04.4 / 07.7	00.0 / 83.6	00.0 / 84.4	16.6 / 02.4
Entropy	09.6 / 24.1	-03.8 / 09.1	01.5 / 05.4	69.2 / 68.7	69.5 / 63.1	16.4 / 14.6	77.6 / 69.4	76.2 / 72.6	**21.0** / 03.1
Spacious.	04.4 / 35.5	02.0 / 39.1	02.3 / 06.9	68.4 / 68.6	68.1 / 63.1	08.5 / 16.1	03.5 / 67.9	03.5 / 71.7	24.1 / 04.9
MSA (Ours)	**11.1 / 78.8**	**09.7 / 55.5**	**01.1 / 04.9**	**70.1 / 91.4**	**69.7 / 77.8**	**08.2 / 11.4**	**88.7 / 84.0**	**88.3 / 85.1**	21.5 / **01.6**

Table I. Translational (t) and Rotational (rot) Absolute Trajectory Error Improvement after PGO and the False Positive Rate (FPR)

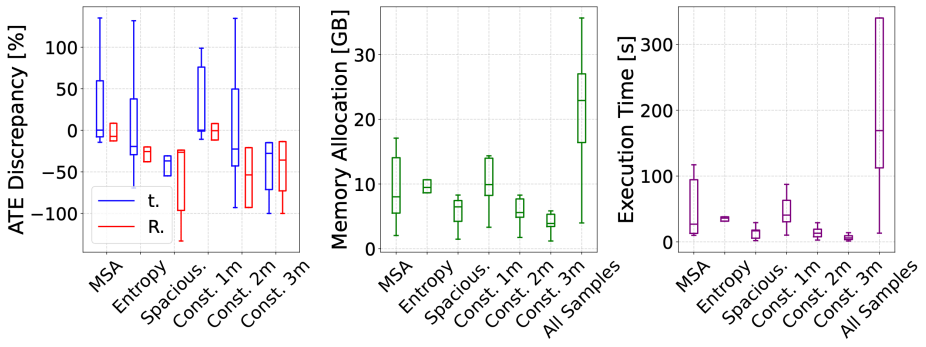

Figure 4. Box plot comparisons. Translational (t) and Rotational (R) Absolute Trajectory Error (ATE) difference of every method compared to the baseline of All Samples, as well as the Memory Allocation in Gigabytes (GB) and the total Execution Time in seconds (s).

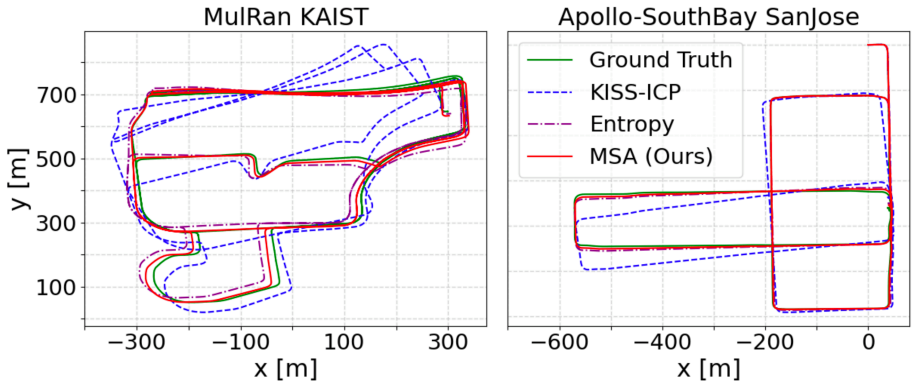

Figure 5. Trajectory comparisons. Comparison of the ground truth and KISS-ICP initial poses, and the sampled poses after the pose graph optimization for the proposed Minimal Subset Approach (MSA) and the entropy-based approach on the KAIST sequence of the MulRan dataset and the SanJoseDowntown sequence of the Apollo-SouthBay dataset.

B. Large-Scale Evaluation

For the second part, we conduct a more realistic online experiment using the `SunnyvaleBigloop` sequence from `Apollo-SouthBay`, covering over 100 kilometers. Poses are processed sequentially, simulating real-time operation. Keyframes are sampled online using the previously mentioned methods, appended to the pose graph, and optimized at each step with `iSAM2`[8]. Simultaneously, descriptors are queried against past keyframes to detect potential loop closures, which are verified with `small_gicp` and integrated into the graph. This online approach provides insights into how well each sampling method handles large-scale missions. Table II presents the translation and rotation ATE, FPR, total allocated memory, and processing time. Notably, when using all samples, the system runs out of memory approximately 70% into the sequence, so no metrics are provided. The remaining sampling methods reduce memory usage while offering excellent performance. This is likely because the mission revisits the same locations multiple times, allowing frequent loop closure edges that

doi.org/10.32388/6QNWB4

prevent significant accumulated drift. In Fig. 6, the optimized trajectory of the entropy-based method is compared to MSA, with the color gradient indicating the translational RPE, showcasing the better performance while maintaining lower memory allocation and false positive ratio. As a final step, Figs. 7 and 8 provide further insights into execution time and memory allocation. Fig. 7 shows memory usage over time, illustrating how the sampling methods scale more efficiently compared to retaining all samples or using a constant 1-meter interval. Fig. 8 presents the moving average of loop closure detection time (left) and incremental pose graph optimization time (right). The top figures show constant interval methods, while the bottom shows adaptive methods. The proposed approach scales similarly to other adaptive methods while offering more robust performance, as demonstrated earlier. The shaded areas represent the minimum and maximum deviations, with the moving average trend indicating the challenge of increasing processing time in large-scale missions. While query time exceeds 1 second per new sample, and the incremental pose graph optimization reaches up to 0.15s for the constant 1-meter sampling method, MSA retains query times of less than 0.5s and incremental optimization times of less than 25ms, maintaining the real-time performance of the system throughout the large-scale mission.

Metrics	t. ATE	rot. ATE	FPR	MEM.	TIME
All Samples	nan	nan	nan	> 128GB	nan
Const. 1m	98.5%	98.4%	12%	90.9GB	7340s
Const. 2m	98.1%	97.3%	17%	47.2GB	2264s
Const. 3m	97.9%	97.0%	28%	35.4GB	1515s
Entropy	95.5%	93.4%	15%	51.0GB	4945s
Spacious.	94.9%	97.5%	24%	38.3GB	1752s
MSA (Ours)	**98.3%**	**94.1%**	**12%**	41.2GB	1951s

Table II. Results for the Large-Scale Evaluation

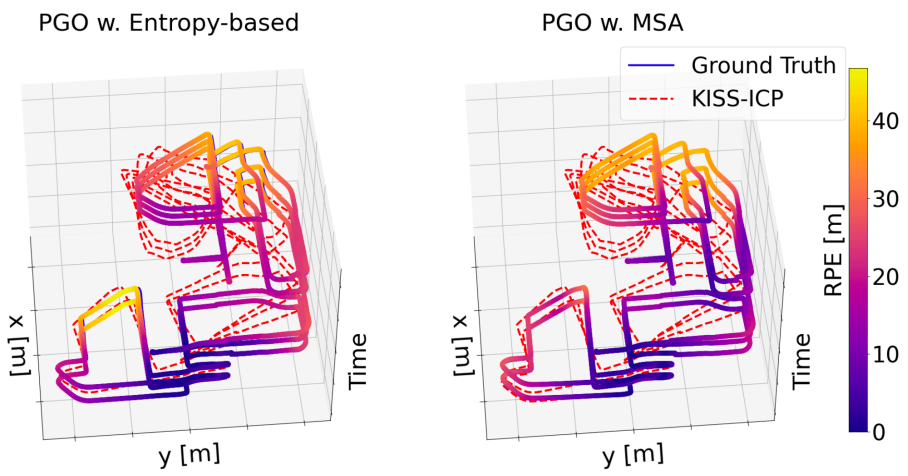

Figure 6. Trajectory comparisons. Comparison of the ground truth and KISS-ICP initial poses, and the sampled poses after the pose graph optimization for the proposed Minimal Subset Approach (MSA) and the entropy-based approach on the SunnyvaleBigloop sequence of the Apollo-SouthBay dataset where the color gradient denotes the RPE. A grid block corresponds to 0.5x0.5 km^2

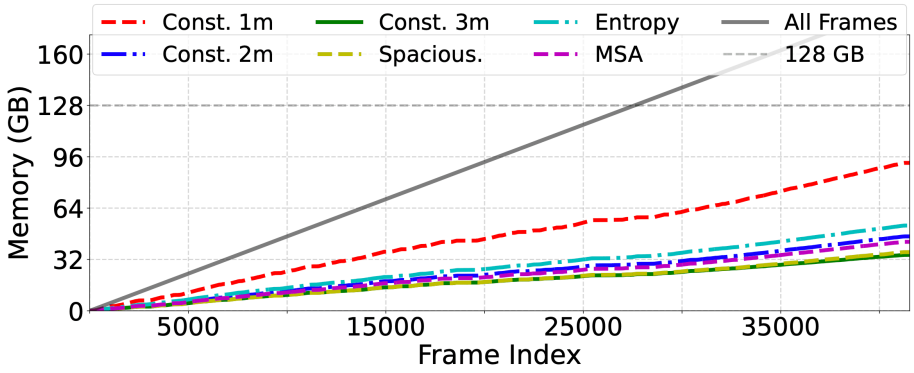

Figure 7. Memory allocation. Memory against mission time for the different sampling methods. Retaining all samples exceeds system memory of 128GB, demonstrating the need for efficient and scalable algorithms.

doi.org/10.32388/6QNWB4

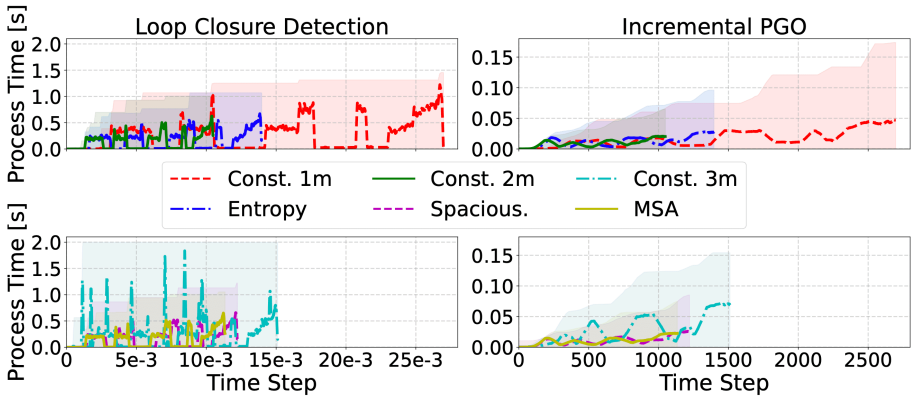

Figure 8. Processing times. The top row shows constant interval methods, while the bottom row shows adaptive methods. The left column represents loop closure detection time, and the right column represents incremental pose graph optimization convergence time. All times in seconds (s).

V. Conclusions

In conclusion, as the robotics community advances towards larger-scale and long-term missions, addressing system limitations with efficient and scalable solutions becomes crucial. This article introduced the Minimal Subset Approach (MSA) to address the combined challenges of loop closure detection and pose graph optimization by reducing redundant samples while maintaining robust performance. Experimental evaluations across multiple datasets show that MSA outperforms in terms of FPR, as well as achieving superior ATE and RPE after pose graph optimization. Unlike other adaptive sampling methods, MSA requires minimal to no parameter tuning, while using a fraction of the system's memory and maintaining comparable computational times.

Data Availability

This paper is supported by code and demonstration files, available at https://github.com/LTU-RAI/opt-key.git.

References

1. [a], [b] *Denniston C, Chang Y, Reinke A, Ebadi K, Sukhatme GS, Carlone L, Morrell B, Agha-mohammadi A (2022). "Loop Closure Prioritization for Efficient and Scalable Multi-Robot SLAM". IEEE Robotics and Automation Letters. 7: 9651–9658.*
2. [a], [b] *Yin P, Zhao S, Lai H, Ge R, Zhang J, Choset H, Scherer S (2023). "AutoMerge: A Framework for Map Assembling and Smoothing in City-Scale Environments". IEEE Transactions on Robotics. 39 (5): 3686–3704. doi:10.1109/TRO.2023.3290448.*
3. [a], [b] *Stathoulopoulos N, Lindqvist B, Koval A, Agha-Mohammadi AA, Nikolakopoulos G (2024). "FRAME: A Modular Framework for Autonomous Map Merging: Advancements in the Field". IEEE Transactions on Field Robotics. 1: 1–26. doi:10.1109/TFR.2024.3419439.*
4. [a], [b] *Ebadi K, Bernreiter L, Biggie H, Catt G, Chang Y, Chatterjee A, Denniston CE, Deschênes S-P, Harlow K, Khattak S, Nogueira L, Palieri M, Petráček P, Petrlík M, Reinke A, Krátký V, Zhao S, Agha-mohammadi A-a, Alexis K, Heckman C, Khosoussi K, Kott*

ege N, Morrell B, Hutter M, Pauling F, Pomerleau F, Saska M, Scherer S, Siegwart R, Williams JL, Carlone L. "Present and Future of SLAM in Extreme Environments: The DARPA SubT Challenge." IEEE Transactions on Robotics. 40: 936-959, 2024. doi:10.1109/TRO.2023.3323938.

5. ^*Cadena C, Carlone L, Carrillo H, Latif Y, Scaramuzza D, Neira J, Reid I, Leonard JJ (2016). "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age". IEEE Transactions on Robotics. 32 (6): 1309–1332. doi:10.1109/TRO.2016.2624754.*
6. a, b*Dellaert F, GTSAM Contributors. borglab/gtsam [software]. Version 4.2a8. Georgia Tech Borg Lab; 2022 May. Available from: https://github.com/borglab/gtsam). doi:10.5281/zenodo.5794541.*
7. ^*Kummerle R, Grisetti G, Strasdat H, Konolige K, Burgard W. "g2o: A General Framework for Graph Optimization." In: 2011 IEEE International Conference on Robotics and Automation. IEEE; 2011. p. 3607-3613. ISBN: 9781612843858.*
8. a, b*Kaess M, Johannsson H, Roberts R, Ila V, Leonard J, Dellaert F (2011). "iSAM2: Incremental Smoothing and Mapping with Fluid Relinearization and Incremental Variable Reordering." In: 2011 IEEE International Conference on Robotics and Automation. 2011. p. 3281–3288.*
9. ^*Huang G, Kaess M, Leonard JJ. "Consistent Sparsification for Graph Optimization." In: 2013 European Conference on Mobile Robots; 2013. p. 150–157.*
10. ^*Bai F, Vidal-Calleja T, Grisetti G (2021). "Sparse Pose Graph Optimization in Cycle Space". IEEE Transactions on Robotics. 37 (5): 1381–1400. doi:10.1109/TRO.2021.3050328.*
11. ^*Nam J, Hyeon S, Joo Y, Noh D, Shim H (2024). "Spectral Trade-Off for Measurement Sparsification of Pose-Graph SLAM". IEEE Robotics and Automation Letters. 9 (1): 723–730. doi:10.1109/LRA.2023.3337590.*
12. ^*Guadagnino T, Di Giammarino LD, Grisetti G (2022). "HiPE: Hierarchical Initialization for Pose Graphs". IEEE Robotics and Automation Letters. 7 (1): 287–294. doi:10.1109/LRA.2021.3125046.*
13. ^*Tazaki Y. "A Spanning Tree-Based Multi-Resolution Approach for Pose-Graph Optimization." IEEE Robotics and Automation Letters. 7 (4): 10033–10040, 2022. doi:10.1109/LRA.2022.3185779.*
14. a, b*Shan T, Englot B, Meyers D, Wang W, Ratti C, Rus D (2020). "LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping." In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5135-5142.*
15. a, b*Huang Y, Shan T, Chen F, Englot B (2022). "DiSCo-SLAM: Distributed Scan Context-Enabled Multi-Robot LiDAR SLAM with Two-Stage Global-Local Graph Optimization". IEEE Robotics and Automation Letters. 7 (2): 1150–1157. doi:10.1109/LRA.2021.3138156.*
16. a, b*Zhao S, Zhang H, Wang P, Nogueira L, Scherer S (2021). "Super Odometry: IMU-centric LiDAR-Visual-Inertial Estimator for Challenging Environments." In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 8729-8736. doi:10.1109/IROS51168.2021.9635862.*
17. a, b, c*Ebadi K, Chang Y, Palieri M, Stephens A, Hatteland A, Heiden E, Thakur A, Funabiki N, Morrell B, Wood S, Carlone L, Agha-Mohammadi AA. "LAMP: Large-Scale Autonomous Mapping and Positioning for Exploration of Perceptually-Degraded Subterranean Environments." In: 2020 IEEE International Conference on Robotics and Automation (ICRA). 2020. p. 80-86. doi:10.1109/ICRA40945.2020.9197082. ePrint 2003.01744.*
18. a, b, c*Chang Y, Ebadi K, Denniston CE, Ginting MF, Rosinol A, Reinke A, Palieri M, Shi J, Chatterjee A, Morrell B, Agha-mohammadi A, Carlone L (2022). "LAMP 2.0: A Robust Multi-Robot SLAM System for Operation in Challenging Large-Scale Undergro*

und Environments." IEEE Robotics and Automation Letters. 7 (4): 9175-9182. doi:10.1109/LRA.2022.3191204.

19. ^a^ ^b^*Chen K, Lopez BT, Agha-mohammadi A, Mehta A (2022). "Direct LiDAR Odometry: Fast Localization With Dense Point Clouds." IEEE Robotics and Automation Letters. 7 (2): 2000–2007. doi:10.1109/LRA.2022.3142739.*
20. ^*Kim B, Jung C, Shim DH, Agha-mohammadi A. "Adaptive Keyframe Generation based LiDAR Inertial Odometry for Complex Underground Environments." In: 2023 IEEE International Conference on Robotics and Automation (ICRA); 2023. p. 3332-3338. doi:10.1109/ICRA48891.2023.10161207.*
21. ^a^ ^b^*Zeng Q, Liu D, Zhou Y, Peng Y (2023). "Entropy-based Keyframe Established and Accelerated Fast LiDAR Odometry and Mapping". ITOEC 2023 - IEEE 7th Information Technology and Mechatronics Engineering Conference. 7: 347–354. doi:10.1109/ITOEC57671.2023.10291983.*
22. ^*Ou F, Li Y, Miao Z, Zhou J (2021). "Lidar Odometry Key Frame Selection Based on Displacement Vector Similarity". Chinese Control Conference, CCC. **2021-July**: 3588–3593. doi:10.23919/CCC52363.2021.9549411.*
23. ^*Grisetti G, Kümmerle R, Stachniss C, Burgard W (2010). "A Tutorial on Graph-Based SLAM". IEEE Intelligent Transportation Systems Magazine. 2 (4): 31–43. doi:10.1109/MITS.2010.939925.*
24. ^a^ ^b^ ^c^ ^d^ ^e^*Ma J, Zhang J, Xu J, Ai R, Gu W, Chen X (2022). "OverlapTransformer: An Efficient and Yaw-Angle-Invariant Transformer Network for LiDAR-Based Place Recognition". IEEE Robotics and Automation Letters. 7 (3): 6958–6965. doi:10.1109/LRA.2022.3178797.*
25. ^a^ ^b^ ^c^ ^d^*Kim G, Choi S, Kim A (2022). "Scan Context++: Structural Place Recognition Robust to Rotation and Lateral Variations in Urban Environments". IEEE Transactions on Robotics. 38 (3): 1856–1874. doi:10.1109/TRO.2021.3116424.*
26. ^*Yin H, Xu X, Lu S, Chen X, Xiong R, Shen S, Stachniss C, Wang Y. "A Survey on Global LiDAR Localization: Challenges, Advances and Open Problems." International Journal of Computer Vision. 132(8):3139-3171. doi:10.1007/s11263-024-02019-5. Aug 2024.*
27. ^*Segal A, Haehnel D, Thrun S (2009). "Generalized-ICP." In: Robotics: science and systems. 2(4): 435. Seattle, WA.*
28. ^*Rusu RB, Blodow N, Beetz M (2009). "Fast Point Feature Histograms (FPFH) for 3D registration." In: 2009 IEEE International Conference on Robotics and Automation. pp. 3212-3217. doi:10.1109/ROBOT.2009.5152473.*
29. ^*Placed JA, Strader J, Carrillo H, Atanasov N, Indelman V, Carlone L, Castellanos JA (2023). "A Survey on Active Simultaneous Localization and Mapping: State of the Art and New Frontiers." IEEE Transactions on Robotics. 39(3): 1686–1705. doi:10.1109/TRO.2023.3248510.*
30. ^a^ ^b^ ^c^ ^d^*Stathoulopoulos N, Sumathy V, Kanellakis C, Nikolakopoulos G (2024). "Why Sample Space Matters: Keyframe Sampling Optimization for LiDAR-based Place Recognition". arXiv. Available from: https://arxiv.org/abs/2410.02643.*
31. ^*Stathoulopoulos N, Saucedo MA, Koval A, Nikolakopoulos G. "RecNet: An Invertible Point Cloud Encoding through Range Image Embeddings for Multi-Robot Map Sharing and Reconstruction." In: 2024 IEEE International Conference on Robotics and Automation (ICRA); 2024. p. 4883-4889. doi:10.1109/ICRA57147.2024.10611602.*
32. ^*Salti S, Tombari F, Di Stefano L (2014). "SHOT: Unique signatures of histograms for surface and texture description". Computer Vision and Image Understanding. 125: 251–264. doi:10.1016/j.cviu.2014.04.011.*
33. ^*Komorowski J, Wysoczanska M, Trzcinski T (2022). "EgoNN: Egocentric Neural Network for Point Cloud Based 6DoF Relocalization at the City Scale". IEEE Robotics and Automation Letters. 7 (2): 722-729. doi:10.1109/LRA.2021.3133593.*

34. ^Geiger A, Lenz P, Urtasun R. "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite." In: 2012 IEEE conference on computer vision and pattern recognition (CVPR). 2012. p. 3354–3361.

35. ^Kim G, Park YS, Cho Y, Jeong J, Kim A (2020). "MulRan: Multimodal Range Dataset for Urban Place Recognition." In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 6246-6253. doi:10.1109/ICRA40945.2020.9197298.

36. ^Lu W, Zhou Y, Wan G, Hou S, Song S (2019). "L3-Net: Towards Learning Based LiDAR Localization for Autonomous Driving." In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6382-6391. doi:10.1109/CVPR.2019.00655.

37. ^Vizzo I, Guadagnino T, Mersch B, Wiesmann L, Behley J, 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 (RA-L). 8 (2): 1029--1036. doi:10.1109/LRA.2023.3236571. Code available here.

38. ^Koide K. "small_gicp: Efficient and parallel algorithms for point cloud registration." Journal of Open Source Software. 9 (100): 6948, Aug. 2024. doi:10.21105/joss.06948.

Declarations

Funding: No specific funding was received for this work.

Potential competing interests: No potential competing interests to declare.