Does Sample Space Matter? Preliminary Results on Keyframe Sampling Optimization for LiDAR-based Place Recognition

Nikolaos Stathoulopoulos, Vidya Sumathy, Christoforos Kanellakis and George Nikolakopoulos

***Abstract*— Loop closures through place recognition are crucial for mitigating pose estimation drift in long term and large scale missions. Despite computational advancements, optimizing performance for real-time deployment, especially in resource-constrained mobile robots and multi-robot systems, remains a challenge. Conventional keyframe sampling practices, in the context of place recognition, often result in either retaining redundant information or overlooking relevant data. To address these concerns, we introduce a novel keyframe sampling approach for LiDAR-based place recognition based on the concepts of redundancy minimization and information preservation in the hyper-dimensional descriptor space, applicable to both learning-based and handcrafted descriptors.**

I. Introduction

In modern robotics, Simultaneous Localization and Mapping (SLAM) is essential for applications ranging from autonomous vehicles [1] to search and rescue missions [2]. For large-scale, long-term missions, accurate pose estimation and robust loop closures through place recognition are essential to mitigate the drift that accumulates over time. Interest in place recognition, especially for global localization, has grown significantly [3], [4], driven by advances in learning-based architectures. Despite the availability of better computational resources, which enable the use of complex algorithms, challenges persist in balancing performance, communication overhead [5], [6], and real-time mapping for resource-constrained platforms [7].

Current research evaluates place recognition using densely sampled datasets [8], [9], which, while improving performance, can overcomplicate long-term mapping by requiring frequent comparisons against an ever-expanding map database. Traditional approaches often rely on a fixed keyframe sampling interval, typically based on traveled distance [10], [11], risking redundancy or information loss, which in turn complicates the selection of an optimal interval for long-term missions. In this work, we propose a novel keyframe sampling approach designed to enhance the efficiency and scalability of global localization and long-term mapping tasks, such as place recognition for loop closure and multi-robot map merging [12]. Our method implements a continuous sliding window optimization strategy for real-time sampling, which adapts dynamically to changes in the environment. We begin by quantifying redundancy within a set of keyframes and assessing the correlation between descriptors and pose changes. By decomposing the correlation matrix, we transform the descriptors along their principal components, allowing us to measure information preservation within the keyframe set. Our framework jointly optimizes for redundancy minimization and information preservation within a unified objective function at each step, ensuring essential information is retained while discarding redundant keyframes. This process reduces memory usage and enables long-term, large-scale mapping without overwhelming system resources. An overview of the method and its impact is shown in Fig. 1.

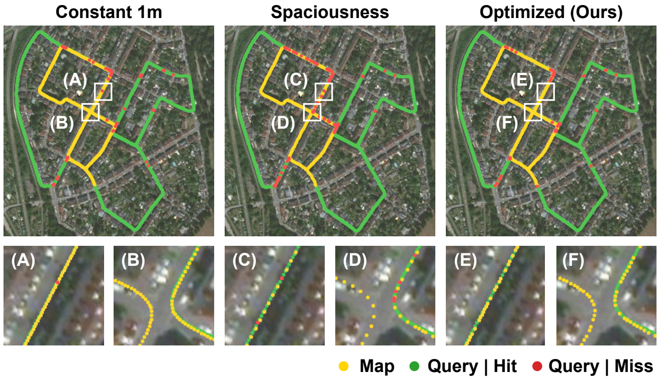

Fig. 1. Our approach outperforms other sampling methods like constant interval and LiDAR spaciousness-based adaptive techniques, as shown in a KITTI dataset example. Top figures show retrieval performance overview, while bottom figures highlight specific instances. Yellow: map samples, green: correctly classified query samples, red: false matches.

The contributions of this paper are as follows: (a) We identify limitations of fixed-interval sampling methods for long-term place recognition and assess their performance loss through an evaluation study. (b) We introduce the concepts of redundancy minimization and information preservation within the descriptor space, demonstrating their efficacy with both learning-based and handcrafted descriptors. (c) We propose an optimization framework for LiDAR-based place recognition that minimizes redundancy and preserves essential information, reducing memory and computational overhead while maintaining retrieval performance, which is critical for long-term mapping in resource-constrained systems. (d) Finally, our method is adaptive across various environments, eliminating the need for manual tuning required by traditional approaches, thus improving robustness.

II. Related Work

In LiDAR-(Inertial) Odometry, keyframe sampling often uses fixed intervals, based on either Euclidean distance [13] or time [14]. These keyframes support local map maintenance, loop closure, and global pose optimization [15]. Recent methods [16], [17] adapt the sampling interval based on environmental spaciousness, optimizing

The Authors are with the Robotics and AI Group, Department of Computer, Electrical and Space Engineering, Luleå University of Technology, 971 87 Luleå, Sweden. Corresponding Author's Email: `niksta@ltu.se`

scan-to-submap matching with k-nearest neighbor (kNN) search. However, they require manual tuning of spaciousness thresholds. Entropy-based methods, like the one by Zeng et al. [18], use information theory for keyframe selection in LiDAR odometry, focusing on local map updates and point cloud matching by calculating Fisher information and entropy. This approach can be less adaptable due to the need for adjusting the information change threshold for different environments. In [19], keyframes are selected based on displacement vector similarity to balance computational cost and map completeness. A local sliding window maintains a limited number of keyframes, generating new ones when the distance threshold is exceeded. Several LiDAR-based place recognition methods use fixed distance intervals for keyframe extraction [10], [11], primarily addressing front-end odometry without considering long-term keyframe accumulation. In contrast,[20] introduces a place recognition method for loop closure with a 3D bag of words[21], using LinK3D [22] features and avoiding redundancy by setting new keyframes based on matches between current and reference frames.

Drawing inspiration from video summarization techniques, we address the gap in optimized keyframe sampling for place recognition. Our approach streamlines keyframe extraction from LiDAR sequences to enable efficient global localization without performance compromise. Introducing two key optimization criteria, redundancy minimization and information preservation, we employ a sliding window framework to reduce redundant keyframes while preserving crucial information. Our aim is to eliminate the need for manual threshold tuning, enhancing adaptability to diverse environments.

III. Preliminaries

The *global localization* problem [3] in robotics is defined as the estimation of the state or pose of a robot, denoted as $\mathbf{x}_t \in \mathbf{X}$, within a known map $\mathbf{M}$, using a sensor observation $\mathbf{z}_t$. Mathematically, this can be formulated as a Maximum Likelihood Estimation (MLE) problem [23], given by:

$$\hat{\mathbf{x}}_t = \underset{\mathbb{X}}{\arg\max}\; p\left(\mathbf{z}_t \,|\, \mathbf{x}_t, \mathbf{M}\right). \tag{1}$$

Here, $\mathbb{X} \equiv \mathbf{X}$ represents the pose space. Unlike *local pose tracking*, *global localization* lacks prior pose information, expanding the search space significantly. To manage this complexity, a retrieval-based approach is adopted, commonly through *place recognition*. We define the keyframes $\mathbf{K} = \{\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_N\}$ as triplets, consisting of a pose coupled with a submap (typically represented by a single LiDAR scan) and a descriptor. The keyframe poses are denoted as $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ where $\mathbf{x} \in SE(3)$, while the corresponding keyframe submaps are represented by $\mathbf{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N\}$, where $\mathbf{z} \in \mathbb{R}^{L\times 3}$, L is the amount of points within a LiDAR scan and $N \in \mathbb{N}$ is the total number of keyframes describing the given map. As a common practice in the literature [3], the map representation is transformed into descriptive vectors $\mathbf{D} = \{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_N\}$ through a feature extraction process. Here, $\mathbf{d} \in \mathbb{R}^M$ and $M \in \mathbb{N}$ is the feature dimensionality of a given descriptor. This process, denoted as a function $F(\mathbf{z}_i, \mathbf{x}_i) = \mathbf{d}_i, \forall \mathbf{z}_i \in \mathbf{Z}, \forall \mathbf{x}_i \in \mathbf{X}$, is typically either learning-based [8] or handcrafted [9]. These descriptive vectors contribute to efficient matching and querying in subsequent stages of the *place recognition* pipeline. Therefore, the keyframes can be succinctly represented as: $\mathbf{K} = \big\{(\mathbf{x}_1, \mathbf{z}_1, \mathbf{d}_1), (\mathbf{x}_2, \mathbf{z}_2, \mathbf{d}_2), \ldots, (\mathbf{x}_N, \mathbf{z}_N, \mathbf{d}_N)\big\}$. The map can be represented by the descriptors, thus, Eq. (1) becomes:

$$\hat{\mathbf{x}}_k = \underset{\mathbf{X}}{\arg\max}\; p\left(\mathbf{z}_k \,|\, \mathbf{x}_k,\, \mathbf{D}\right) \tag{2}$$

where $\mathbf{X} \subset \mathbb{X}$ is the discrete subset of the pose space. The size of the keyframe list, N, determines the constrained search space $|\mathbf{X}| \ll |\mathbb{X}|$. The retrieval-based approach bounds the estimated pose within a range of the true pose, $\delta_{min} \leqslant \|\mathbf{x}_k - \hat{\mathbf{x}}_k\|_2 \leqslant \delta_{max}$, with an accuracy influenced by the descriptor extractor's quality.

The sampling interval between keyframes affects map representation and retrieval performance, as shown in Fig 1 and discussed in Section VI. Dense keyframe sampling improves retrieval accuracy by providing more data for pose estimation, but it can also introduce redundancy and noise from dynamic motions or disturbances. Additionally, a larger keyframe list increases computational load and memory usage, especially in large or long-term missions. Therefore, balancing keyframe density and list size is crucial for optimizing retrieval accuracy and computational efficiency in *global localization* tasks.

IV. Problem Formulation

Given a keyframe set $\mathbf{K}_{\mathrm{M}} = \{\mathbf{X}_{\mathrm{M}}^{\mathsf{T}}, \mathbf{Z}_{\mathrm{M}}^{\mathsf{T}}, \mathbf{D}_{\mathrm{M}}^{\mathsf{T}}\}$ that represents an environment, a query set $\mathbf{K}_{\mathrm{Q}} = \{\mathbf{X}_{\mathrm{Q}}^{\mathsf{T}}, \mathbf{Z}_{\mathrm{Q}}^{\mathsf{T}}, \mathbf{D}_{\mathrm{Q}}^{\mathsf{T}}\}$ and for a given descriptor extraction process $F(\cdot)$, the retrieval task for the whole query set can be defined as:

$$\begin{aligned} &\underset{\bar{\mathbf{K}}=\{\mathbf{k}_j:\mathbf{k}_j\in\mathbf{K}_{\mathrm{M}}\}}{\text{minimize}} \sum_{(\mathbf{k}_i,\mathbf{k}_j)\in\mathbf{K}_{\mathrm{M}}^{\mathrm{Q}}} f_\circ\left(\mathbf{k}_i,\, \mathbf{k}_j\right), \\ &\text{subject to}\;\; \forall\, \mathbf{k}_i \in \mathbf{K}_{\mathrm{Q}},\, \bar{\mathbf{K}} \subseteq \mathbf{K}_{\mathrm{M}},\, |\bar{\mathbf{K}}| = |\mathbf{K}_{\mathrm{Q}}| \neq 0 \end{aligned} \tag{3}$$

where $\mathbf{K}_{\mathrm{M}}^{\mathrm{Q}} = \mathbf{K}_{\mathrm{Q}} \times \mathbf{K}_{\mathrm{M}} = \{(\mathbf{k}_i, \mathbf{k}_j) \,|\, \mathbf{k}_i \in \mathbf{K}_{\mathrm{Q}},\, \mathbf{k}_j \in \mathbf{K}_{\mathrm{M}}\}$ is the space of all the possible combinations between the query and map set, and $f_\circ(\cdot)$ is the retrieval function given by the corresponding feature extraction framework. A common function used is the distance between two descriptors $f_\delta = \|\mathbf{d}_i - \mathbf{d}_j\|_M$, or a similarity measure $f_\sigma = 1/(1 + \|\mathbf{d}_i - \mathbf{d}_j\|_M)$. It is important to note that given the function $f_\circ(\cdot)$, it's essential to specify the sign of the function to align with the definition of minimization. This minimization problem aims to find the subset $\bar{\mathbf{K}} \subseteq \mathbf{K}_{\mathrm{M}}$ that contains the corresponding nearest candidate for every $\mathbf{k}_i$ in the query set $\mathbf{K}_{\mathrm{Q}}$. To reduce the search space of the map keyframes $\mathbf{K}_{\mathrm{M}}$, we propose the following formulation:

$$\underset{\substack{\mathbf{K}_{\mathrm{M}}^{*}\subset\mathbf{K}_{\mathrm{M}},\\ \bar{\mathbf{K}}\subseteq\mathbf{K}_{\mathrm{M}}^{*}}}{\arg\min}\; \mathrm{g}_\circ\left(\mathbf{K}_{\mathrm{Q}}, \mathbf{K}_{\mathrm{M}}\right) = \underset{\substack{\mathbf{K}_{\mathrm{M}}^{*}\subset\mathbf{K}_{\mathrm{M}},\\ \bar{\mathbf{K}}\subseteq\mathbf{K}_{\mathrm{M}}^{*}}}{\arg\min} \sum_{(\mathbf{k}_i,\mathbf{k}_j)\in\mathbf{K}_{\mathrm{M}}^{\mathrm{Q}}} f_\circ\left(\mathbf{k}_i,\, \mathbf{k}_j\right), \tag{4}$$

where $\mathbf{K}^*_{\mathrm{M}}$ is the minimum-cardinality subset of $\mathbf{K}_{\mathrm{M}}$ over which the minimization problem converges to the same optimum set $\bar{\mathbf{K}}$ as in the original formulation in Eq. (3):

$$\begin{aligned}\mathbf{K}^*_{\mathrm{M}} &= \underset{\mathbf{K}\subset\mathbf{K}_{\mathrm{M}}}{\arg\min}\ |\mathbf{K}| \\ \text{subject to } &\underset{\bar{\mathbf{K}}\subseteq\mathbf{K}^*_{\mathrm{M}}}{\arg\min}\ \mathrm{g}_\circ\left(\mathbf{K}_{\mathrm{Q}},\mathbf{K}^*_{\mathrm{M}}\right).\end{aligned} \tag{5}$$

To solve the aforementioned combinatorial optimization that refers to a minimum-cardinality problem, we must consider both sets $\mathbf{K}_{\mathrm{M}}$ and $\mathbf{K}_{\mathrm{Q}}$. However, this approach is impractical as our objective is to dynamically minimize the size of the map keyframe set $\mathbf{K}_{\mathrm{M}}$ in real-time, before its utilization with the query set $\mathbf{K}_{\mathrm{Q}}$. Determining the contribution of a keyframe to the optimization process is largely based upon knowledge of the target query set, thus, the system lacks causality as it necessitates future inputs. Furthermore, the problem falls within the NP-hard class due to its inherently combinatorial nature. More specifically, identifying the optimal subset $\mathbf{K}^*_{\mathrm{M}}$ necessitates evaluating all feasible combinations of keyframes, rendering it computationally infeasible for sets comprising more than $15-20$ keyframes.

V. Information Preservation And Redundancy Minimization

To address these inherent challenges, we propose an approximation method for solving the problem outlined in Eq. (3)-(5). Our approach involves defining, identifying, and eliminating redundancy within a keyframe set, all while retaining the crucial information encoded in the descriptors' space. Designed for real-time operations, our solution employs a sliding window combinatorial optimization technique with just two optional tunable parameters that control the balance between *information preservation* and *redundancy minimization*. The primary objective is twofold: (a) to eliminate redundant samples that might degrade performance in place recognition tasks while simultaneously, (b) we aim to retain the essential information encoded in the descriptors such that the retrieval performance either remains the same or is improved while the computational load is decreased.

A. Redundancy in keyframes

Depending on the scanning frequency of the sensor kit integrated into the robotic platform or autonomous vehicle, along with its moving speed and the characteristics of the environment, keyframe samples may capture redundant information if they are too closely spaced. First, we establish a definition for redundancy within a local set of keyframes and subsequently, we propose a metric to quantify the redundancy between poses based on their corresponding descriptor.

Definition 1: A keyframe $\mathbf{k}$ is deemed redundant within a keyframe set $\mathbf{K}$ if its removal does not affect the optimal solution set $\bar{\mathbf{K}}$ nor the minimum value of the query process defined in Eq. (3) and does not create discontinuities in the map representation. This can be expressed as:

$$\underset{\bar{\mathbf{K}}}{\arg\min}\ \mathrm{g}_\circ\left(\mathbf{K}_{\mathrm{Q}},\mathbf{K}_{\mathrm{M}}\right) = \underset{\bar{\mathbf{K}}'}{\arg\min}\ \mathrm{g}_\circ\left(\mathbf{K}_{\mathrm{Q}},\mathbf{K}_{\mathrm{M}}\backslash\{\mathbf{k}\}\right) \tag{6}$$

and

$$\mathrm{g}_\circ\left(\mathbf{K}_{\mathrm{Q}},\bar{\mathbf{K}}\right) = \mathrm{g}_\circ\left(\mathbf{K}_{\mathrm{Q}},\bar{\mathbf{K}}'\right) \tag{7}$$

$$\text{subject to}\quad \delta_l \leqslant \|\mathbf{x}_i - \mathbf{x}_{i+1}\|_2 \leqslant \delta_u,\ \forall\,\mathbf{x}\in\mathbf{X}. \tag{8}$$

Remark 1: The preceding equation describes redundancy within a keyframe set, however, we can also introduce the notion of *soft redundancy*. In this case, although the optimal set $\bar{\mathbf{K}}$ may vary following the removal of a keyframe, the minimum value should remain close to the original. The underlying concept is that even after removing a keyframe, there should exist another keyframe $\mathbf{k}'$ nearby that still offers a viable candidate for the place recognition task, ensuring that the query keyframe retains a potential match.

$$\begin{gathered}\left|\mathrm{g}_\circ\left(\mathbf{K}_{\mathrm{Q}},\bar{\mathbf{K}}\right) - \mathrm{g}_\circ\left(\mathbf{K}_{\mathrm{Q}},\bar{\mathbf{K}}'\right)\right| \leqslant \varepsilon, \\ \text{where}\quad f_\circ(\mathbf{k}_{\mathrm{q}},\mathbf{k}) \approx f_\circ(\mathbf{k}_{\mathrm{q}},\mathbf{k}')\ \text{ and }\ \|\mathbf{x}-\mathbf{x}'\|_2 \leqslant \delta,\end{gathered} \tag{9}$$

where $\delta,\varepsilon > 0$ are small positive real values, $\mathbf{x}$, $\mathbf{x}'$ are the corresponding poses to the keyframe tuples $\mathbf{k}$, $\mathbf{k}'$ and $\mathbf{k}_q \in \mathbf{K}_{\mathrm{Q}}$ is the candidate pair for the original keyframe $\mathbf{k}$ that, after removal, is matched with $\mathbf{k}'$.

Remark 2: The constraint specified in Eq. (8) and the concept of map discontinuity are essential to ensuring a comprehensive coverage of the map rather than achieving a continuous surface representation, given that we are dealing with 3D point clouds. More specifically, this constraint ensures the presence of a candidate keyframe across the entirety of the map. The lower and upper distance between poses, δ_l and δ_u respectively, typically ranges between $1-5$ meters.

We proceed to quantify the *redundancy* term within a keyframe set. This metric effectively captures the similarity or correlation between descriptors across pairs of keyframes. Utilizing the previously mentioned (IV) similarity function f_σ, we measure the redundancy between consecutive keyframe pairs using the following expression:

$$\rho_\tau(\mathbf{K}) = \frac{1}{N-1}\sum_{i=1}^{N-1} f_\sigma(\mathbf{k}_i,\mathbf{k}_{i+1}), \tag{10}$$

where $0 < \rho_\tau(\mathbf{K}) \leqslant 1$ and N denotes the cardinality of the keyframe set $\mathbf{K}$. Higher redundancy scores, indicate high similarity between consecutive descriptors, while lower values indicate lower correlation between the keyframes.

B. Information Preservation in keyframes

In Section III, we defined the function $F:\mathbb{R}^3 \rightarrow \mathbb{R}^M$ to obtain the descriptive vectors, effectively mapping each keyframe from 3D space to an M-dimensional representation. By computing the Jacobian $\mathbf{J}$ of F with respect to the poses $\mathbf{x}$, we gain insight into the sensitivity of the descriptors to pose changes. Considering the poses in 3D space, we simplify dimensionality by regarding the distance between poses as the Euclidean norm $\|\cdot\|_2$. If we consider the descriptors $\mathbf{d}\in\mathbb{R}^M$ as random variables and the poses $\mathbf{x}\in SE(3)$ as samples, then the product of the Jacobian $\mathbf{J}$ and its transpose $\mathbf{J}^\intercal$ gives us an estimate of the covariance matrix between the descriptors. Essentially, each element of

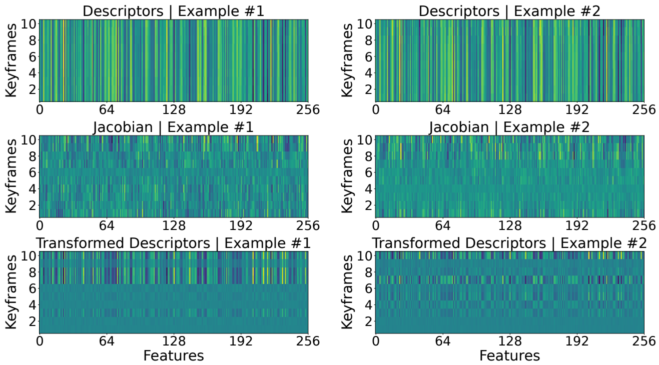

Fig. 2. An example showing the descriptors, the Jacobian, and the transformed descriptors for two window keyframe sets. The Jacobian indicates how the features of the descriptors change between keyframes, while the transformed descriptors demonstrate how these same features appear after transformation to the principal components of that set.

this matrix indicates how much two descriptors covary across poses and can be defined as:

$$\mathbf{J}_{\mathbf{F}}^{\mathsf{T}}\mathbf{J}_{\mathbf{F}} = \left(\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right)^{\mathsf{T}}\left(\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right) = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{-1}, \tag{11}$$

where $\boldsymbol{\Lambda}$ is the $N \times N$ diagonal matrix with the eigenvalues, $\mathrm{diag}(\boldsymbol{\Lambda}) = \{\lambda_1, \lambda_2, \dots, \lambda_N\}$ and $\mathbf{V}$ is the $N \times N$ matrix whose columns are the eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ resulting from the decomposition process. These eigenvectors represent the principal directions of variation in the descriptor space, while the eigenvalues represent their magnitudes. As an additional preprocessing step before computing the information preservation term, we transform the descriptors using the eigenvectors obtained from the eigenvalue decomposition, $\mathbf{D}' = \sqrt{\boldsymbol{\Lambda}} \cdot \mathbf{V} \cdot \mathbf{D}$. This aligns the descriptors with the principal directions of maximum variability captured by the eigenvectors. Simultaneously, the descriptive vectors are scaled by the square root of eigenvalues. This scaling ensures that each dimension in the transformed data space captures variance proportional to the original data's variability. An example of the descriptors across keyframes, the Jacobian and the transformed descriptors is illustrated in Fig. 2. Next, we define the information preservation term, which measures how well the poses in the dataset preserve the variability captured by the descriptive vectors.

Definition 2: The *information preservation* term π_τ for a keyframe set $\mathbf{K}$ is expressed through the transformation of the descriptors $\mathbf{D}$ to $\mathbf{D}'$ using the principal components $\mathbf{V}$ and the eigenvalues $\boldsymbol{\Lambda}$ from the decomposition of the covariance matrix $\mathbf{J}_{\mathbf{F}}^{\mathsf{T}}\mathbf{J}_{\mathbf{F}}\big|_{\mathbf{K}}$ and is computed as:

$$\pi_\tau(\mathbf{K}) = -\frac{1}{N-1}\sum_{i=1}^{N-1} f_\delta(\mathbf{d}'_i, \mathbf{d}'_{i+1}), \tag{12}$$

where $-1 \leqslant \pi_\tau(\mathbf{K}) < 0$, f_δ is the distance function between two descriptors and $\mathbf{d}' \in \mathbf{D}'$ are the transformed descriptors. Higher values of the information preservation term indicate that the relationships and patterns captured by the descriptive vectors are well-preserved as poses change.

Remark 3: Each eigenvector $\mathbf{v}_k$ denotes the directions of maximal variability in the descriptor space, occurring from changes in pose-to-pose distances. Within these eigenvectors, the components $\mathbf{v}_{kj}$ signify the relative contributions of each descriptor feature to the variability induced by changes in pose-to-pose distances.

Remark 4: The k-th eigenvalue λ_k of the covariance matrix quantifies the amount of variance described by each eigenvector $\mathbf{v}_k$. Larger eigenvalues indicate that the corresponding eigenvectors capture more significant patterns of variability in the data, highlighting the importance of each principal component in explaining the overall variability in the descriptor space with respect to the pose-to-pose distance.

In summary, while both ρ_τ and π_τ utilize a similarity or distance function, they differ in their objectives. The redundancy term focuses on capturing local redundancy or similarity within the keyframe set, whereas the information preservation term evaluates the conservation of information structure within the same local context, considering the variability induced by changes in pose-to-pose distances. Despite sharing similarities in computation, their distinct goals result in differing interpretations and implications for assessing the relationships between descriptive vectors within the keyframe set.

C. Proposed Sliding Window Optimization

After defining these two terms, we propose a keyframe sampling strategy tailored for LiDAR-based place recognition. Our objective is to approximate the optimal map keyframe set $\mathbf{K}_{\mathrm{M}}^{*}$ as previously defined in Section IV and Eq. (4)-(5). To address the challenges posed by the computational complexity of optimizing such a large keyframe set and the non-causality of proactively selecting the best keyframes for future query sets, we introduce a sliding window optimization sampling method. By continuously optimizing window keyframe sets $\mathbf{K}_t$ over the mission duration T, we accumulate them to approximate the optimal keyframe set $\mathbf{K}_{\mathrm{M}}^{*}$ as $\mathbf{K}_{\mathrm{M}}^{*} \cong \bigcup_{t \in T} \mathbf{K}_t^{*}$, where $\mathbf{K}_t^{*}$ represents the minimum cardinality subset of $\mathbf{K}_t$ that retains the maximum information. To compute the desired subset, we leverage the *redundancy* and *information preservation* terms defined earlier in Sections V-A and V-B respectively. The process begins with the initialization of a window keyframe set $\mathbf{K}_t$, containing N keyframes. The time step t progresses as soon as there are N new keyframes available, and the optimization has converged to the optimum window keyframe set $\mathbf{K}_t^{*}$. Notably, for every window, the last chosen keyframe becomes the first keyframe in the next time step, rendering the window adaptive and flexible regarding the number of chosen keyframes. The process of finding the optimum window keyframe set is as follows: Given a window keyframe set $\mathbf{K}_t$, we begin by generating all possible keyframe subsets, denoted as the power set $\mathbb{P}(\mathbf{K}_t)$. The cardinality of this power set is $|\mathbb{P}(\mathbf{K}_t)| = 2^N$, providing insights into the computational complexity relative to the window size chosen. To align with the definition of the redundancy term, we impose constraints on the power set, retaining only the subsets that satisfy the constraints posed in Eq. (8), namely the minimum and maximum distance between consecutive poses. These constraints substantially reduce the size of the power set, typically by $5 - 10$ times. We refer to this constrained power set as $\bar{\mathbb{P}}(\mathbf{K}_t) = \{\mathbf{K}_t^{\mathbb{S}} \in \mathbb{P}(\mathbf{K}_t) : \delta_l \leqslant \|\mathbf{x}_i - \mathbf{x}_{i+1}\|_2 \leqslant \delta_u, \forall \mathbf{x} \in \mathbf{K}_t^{\mathbb{S}}\}$, where $\mathbf{K}_t^{\mathbb{S}}$ represents each subset in the power set, and δ_l and δ_u denote the lower and upper distance limits for the pose-to-pose distance, typically ranging within $1-5$ meters. We then formulate the following optimization problem, leveraging the two terms defined earlier, to search for the optimal set $\mathbf{K}_t^*$ within the constrained power set $\bar{\mathbb{P}}(\mathbf{K}_t)$:

$$\mathbf{K}_t^* = \underset{\mathbf{K}_t^{\mathbb{S}}}{\arg\min} \;\; \left(\rho_\tau\left(\mathbf{K}_t^{\mathbb{S}}\right) + \alpha\right) \big/ \left(\pi_\tau(\mathbf{K}_t^{\mathbb{S}}) - \beta\right) \qquad (13)$$
$$\text{where} \quad \alpha, \beta > 0 \;\text{ and }\; \mathbf{K}_t^{\mathbb{S}} \in \bar{\mathbb{P}}(\mathbf{K}_t).$$

The rationale behind this formulation is to minimize redundancy and maximize information preservation within the keyframe set. To achieve this, the minimization problem outlined above is solved through an exhaustive search. For every subset, we compute the information matrix $\mathbf{J}_\mathbf{F}^\top \mathbf{J}_\mathbf{F}$ to capture the relationship between poses and descriptors. Subsequently, we quantify this relationship using the information preservation term from Eq. (12). Similarly, we compute the redundancy term for every subset and proceed to identify the subset with the best combined score, as per Eq. (13).

VI. Experimental Evaluation and Results

For the experimental evaluation, we utilize two publicly available datasets: KITTI [24] and Ford Campus [25], both featuring LiDAR scans from an HDL-64E Velodyne sensor. We employ two different descriptor extraction frameworks: OverlapTransformer [8] for learning-based descriptors with a 1×256 feature vector and Scan Context [9] for handcrafted descriptors with a $1 \times 20 \times 60$ feature vector. The comparative analysis includes various methods, with three constant sampling intervals (at 1, 3, and 5 meters) and adaptive intervals based on spaciousness [16], [17] and LiDAR scan entropy [18]. All methods are compared against the baseline, which utilizes all samples from the datasets. Our objective is to assess the adaptability of our proposed keyframe sampling approach against common methods. For the following evaluations, we set the parameters as, $\alpha = 1$, $\beta = 1$, and $N = 10$, while in subsection VI-C, we explore the method's response to different parameter settings through an ablation study.

A. Retrieval Performance and Memory Allocation

We start off with a quantitative analysis, presenting results for both datasets and descriptor extraction frameworks across all sampling methods in Table I. The metrics provided include the Area Under the Precision-Recall Curve (AUC), the F1-max score, and the memory allocated for storing the keyframes. Memory allocation is normalized against the baseline, where it always appears as 1.00. Additionally, precision-recall curves for all methods and datasets are depicted for the OverlapTransformer in Fig. 3, illustrating that our proposed approach manages to maintain the original performance in most of the scenarios, while improving the performance for others. From both the table and the figure, it is evident that the fixed interval sampling struggles, as different sequences favor different intervals. Similarly, the spaciousness and entropy-based methods have a varying performance depending on the dataset and sequence.

TABLE I
Results for the performance of all evaluated methods using the OverlapTransformer (OT) and Scan Context (SC) descriptors. Metrics include the Area Under the Curve (AUC), F1-max score, and memory allocated (MEM).

Dataset	Metric	All Samples		Constant at 1m		Constant at 3m		Constant at 5m		Spacious -ness		Entropy -based		Optimized (Ours)	
		OT	SC	OT	SC	OT	SC	OT	SC	OT	SC	OT	SC	OT	SC
KITTI 00	F1-MAX	0.94	0.92	**0.94**	**0.91**	0.92	0.89	0.85	0.82	0.90	0.83	0.92	0.90	**0.93**	**0.91**
	AUC	0.99	0.93	**0.99**	**0.92**	0.98	0.88	0.92	0.85	0.97	0.89	0.98	0.87	**0.98**	**0.92**
	MEM.	1.00		0.51		0.22		0.14		0.24		0.40		0.38	0.45
KITTI 02	F1-MAX	0.78	0.95	**0.79**	**0.95**	0.75	0.93	0.68	0.84	**0.82**	**0.96**	0.66	0.89	0.78	0.95
	AUC	0.82	0.98	**0.83**	**0.98**	0.72	0.92	0.65	0.87	**0.85**	**0.97**	0.64	0.86	0.84	0.96
	MEM.	1.00		0.82		0.30		0.19		0.40		0.33		0.68	0.66
KITTI 05	F1-MAX	0.91	0.95	0.91	0.94	**0.91**	**0.95**	0.86	0.90	**0.91**	0.93	0.86	0.88	0.90	**0.96**
	AUC	0.98	0.97	0.97	0.95	**0.97**	**0.96**	0.93	0.92	**0.97**	0.95	0.93	0.91	0.97	**0.98**
	MEM.	1.00		0.65		0.24		0.15		0.28		0.34		0.51	0.60
KITTI 06	F1-MAX	0.96	0.94	**0.96**	**0.95**	0.94	0.93	0.89	0.90	0.94	0.93	0.92	0.92	**0.95**	**0.96**
	AUC	0.99	0.97	**0.99**	**0.99**	0.97	0.96	0.94	0.94	0.97	0.98	0.96	0.97	**0.98**	**0.99**
	MEM.	1.00		0.85		0.30		0.21		0.30		0.37		0.60	0.63
KITTI 07	F1-MAX	0.67	0.91	0.64	0.83	0.69	**0.87**	**0.76**	0.82	0.65	0.77	0.67	0.87	**0.70**	**0.90**
	AUC	0.77	0.90	0.73	0.87	0.75	**0.90**	**0.82**	0.86	0.72	0.75	0.76	0.89	**0.80**	**0.93**
	MEM.	1.00		0.51		0.19		0.12		0.24		0.30		0.38	0.42
KITTI 08	F1-MAX	0.29	0.69	**0.30**	**0.63**	0.28	0.52	0.24	0.50	0.28	**0.60**	0.28	0.55	**0.33**	0.59
	AUC	0.22	0.68	**0.18**	**0.65**	0.16	0.55	0.14	0.52	0.16	**0.61**	0.18	0.52	**0.20**	0.60
	MEM.	1.00		0.62		0.25		0.16		0.28		0.31		0.48	0.46
FORD 01	F1-MAX	0.82	0.83	**0.87**	0.76	0.85	**0.84**	0.85	0.83	0.85	0.83	0.88	0.86	**0.88**	**0.89**
	AUC	0.58	0.83	**0.65**	0.79	0.63	**0.80**	0.65	0.80	0.65	0.81	0.64	0.79	**0.69**	**0.84**
	MEM.	1.00		0.51		0.20		0.13		0.13		0.68		0.23	0.24
FORD 02	F1-MAX	0.55	0.64	0.53	**0.67**	**0.59**	0.58	0.57	0.59	0.57	0.63	0.57	0.66	**0.61**	**0.68**
	AUC	0.27	0.60	0.23	**0.61**	0.34	0.55	**0.37**	0.56	**0.37**	0.56	0.29	0.62	0.32	**0.62**
	MEM.	1.00		0.62		0.24		0.15		0.15		0.67		0.27	0.28

B. Qualitative Analysis

For the qualitative analysis, we conducted a statistical examination of the results presented in Table I. Figure 4 illustrates the deviations for both the AUC and F1-max metrics, as well as memory allocation, computed against the baseline and expressed as a percentage of overall performance gain or loss. The top figure displays the results for the OverlapTransformer descriptors, while the bottom shows those for the Scan Context descriptors. The median, minimum, and maximum values demonstrate the effectiveness of our proposed approach in maintaining superior performance compared to both adaptive and fixed interval methods. Deviations in AUC and F1-max performance indicate consistent retention of performance, with a tendency towards improved performance and lower minimum values compared to other methods using both descriptor extraction frameworks. In addition, the large deviation in memory allocation underscores the adaptability and functionality of our proposed method to adjust and retain as many samples as necessary to maintain performance. In contrast, both entropy-based and spaciousness methods fail to adapt in all scenarios. These results highlight the contribution of our proposed approach, which does not require tuning and can adapt to the environment dynamically. In contrast, fixed intervals offer varying benefits in different environments, and even entropy-based and spaciousness methods require tuning

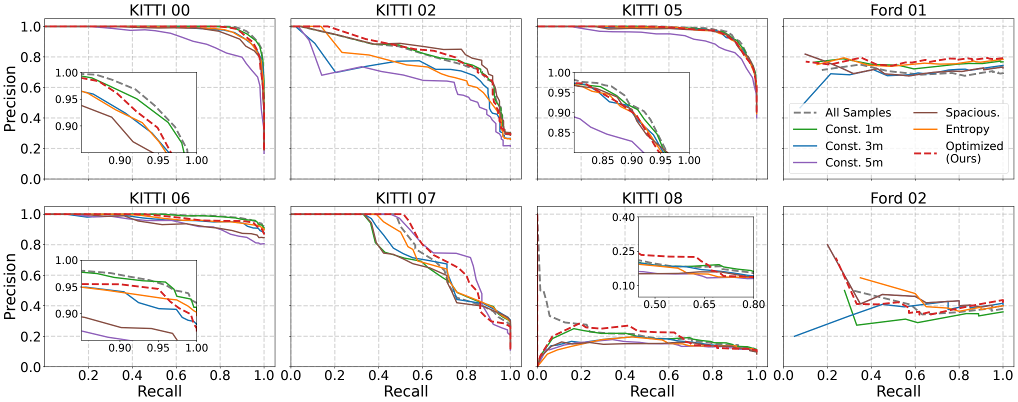

Fig. 3. Precision-recall curves for the KITTI and Ford Campus datasets using the OverlapTransformer descriptors.

to adapt, making them difficult and less efficient to use.

C. Ablation Study and Computational Complexity

Additionally, we conducted an ablation study to investigate the individual contributions of the two optimization terms, $\rho_{\mathcal{T}}$ and $\pi_{\mathcal{T}}$, as well as the effects of changing the optimization parameters α and β, and the window size N on performance and computational complexity. The results are presented in Table II and refer to the mean and standard deviation of the difference from the baseline, across all datasets. First, focusing on optimizing each term independently, we observe distinct behaviors. Minimizing redundancy ($\rho_{\mathcal{T}}$) significantly reduces memory allocation but leads to larger deviations in performance, particularly in terms of the AUC. Conversely, optimizing solely for information preservation ($\pi_{\mathcal{T}}$) results in higher memory allocation with lower deviations in AUC and F1-max, indicating a more robust performance across datasets. Next, we explore the impact of tuning the α and β parameters. Standard tuning ($\alpha = \beta = 1$) strikes a balance between performance and memory allocation, while increasing both parameters to 4, one at a time, demonstrates varying performance and memory consumption, providing flexibility for users based on their memory constraints and performance requirements. Considering the window size, it affects computational speed as the number of keyframes increases. However, due to constraints on pose distance, not all subsets are evaluated, resulting in a computational complexity of $\mathcal{O}(n \log n)$ instead of $\mathcal{O}(2^n)$. Despite significant changes in computational time with different window sizes (5, 10, and 15 keyframes), performance remains relatively stable. A window size of 10 keyframes balances performance and computational time, averaging at 44.6ms, making it suitable for real-time operations. All the results were realized on an 11th Gen Intel® Core™ i7-1165G7 @ 2.80GHz × 8 with 32GB of RAM.

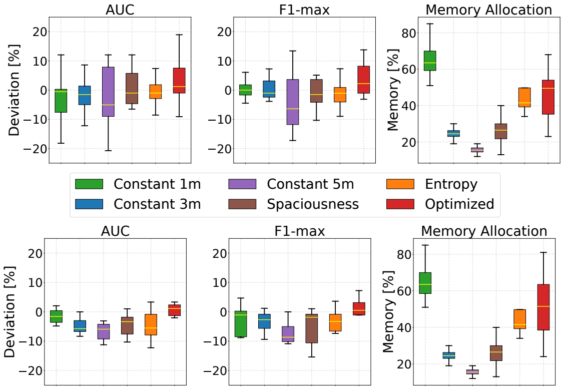

Fig. 4. Median, min, and max deviations from the baseline for all methods evaluated on both KITTI and Ford Campus datasets. The top figure corresponds to results using the OverlapTransformer descriptors, while the bottom figure refers to the results using the Scan Context descriptors.

VII. Conclusions

In conclusion, we present a novel framework for improving global localization in robotics, focusing on LiDAR-based keyframe selection for place recognition. Our method minimizes redundancy while preserving information, reducing memory usage without requiring precise tuning, unlike other fixed or adaptive methods. Evaluations on public datasets, using both learning-based and handcrafted descriptors, demonstrate the robustness and effectiveness of our approach across diverse scenarios. Our findings highlight the potential for more efficient robotic applications, paving the way for future research into optimizing keyframe selection and further exploring sampling strategies in place recognition.

TABLE II

Results of the ablation study are presented as the mean and standard deviation for the AUC, F1-max score, and memory, as well as the min./avg./max. values respectively for the window optimization times in milliseconds.

Window Size	**N = 5**				**N = 10**				**N = 15**			
Metrics	AUC [%]	F1-MAX [%]	MEM. [%]	TIME	AUC [%]	F1-MAX [%]	MEM. [%]	TIME	AUC [%]	F1-MAX [%]	MEM. [%]	TIME
only $\rho_{\mathcal{T}}$	02.58 ± 22.18	00.17 ± 07.36	29.63 ± 04.41	0.16/1.23/8.94	15.06 ± 41.68	4.00 ± 14.76	24.00 ± 04.45	1.77/44.6/168.1	12.55 ± 32.80	01.42 ± 10.45	23.12 ± 13.70	59/1270/3517
only $\pi_{\mathcal{T}}$	04.66 ± 09.32	02.20 ± 05.51	59.13 ± 08.84		05.18 ± 16.89	01.68 ± 03.91	58.00 ± 12.21		04.76 ± 13.70	02.31 ± 04.54	57.75 ± 13.58	
$\alpha = 1,\ \beta = 1$	04.93 ± 15.03	02.54 ± 04.80	48.62 ± 11.61		04.08 ± 09.18	03.90 ± 05.84	45.88 ± 14.57		03.66 ± 10.85	03.53 ± 05.47	45.50 ± 15.00	
$\alpha = 4,\ \beta = 1$	07.45 ± 18.36	05.13 ± 09.63	36.37 ± 05.43		00.88 ± 09.92	00.12 ± 05.48	37.60 ± 10.73		02.93 ± 11.03	02.13 ± 06.97	39.62 ± 12.55	
$\alpha = 1,\ \beta = 4$	05.06 ± 14.20	03.07 ± 04.20	55.87 ± 14.77		01.22 ± 09.34	00.75 ± 03.11	51.00 ± 16.08		00.58 ± 07.81	00.80 ± 03.12	50.00 ± 16.53	

References

[1] G. Mehr, P. Ghorai, C. Zhang, A. Nayak, D. Patel, S. Sivashangaran, and A. Eskandarian, “X-CAR: An Experimental Vehicle Platform for Connected Autonomy Research,” *IEEE Intelligent Transportation Systems Magazine*, vol. 15, pp. 41–57, 2022.

[2] A. Agha, K. Otsu, B. Morrell, D. Fan, R. Thakker, A. Santamaria-Navarro, S.-K. Kim, A. Bouman, *et al.*, “NeBula: TEAM CoSTAR's Robotic Autonomy Solution that Won Phase II of DARPA Subterranean Challenge,” *Field Robotics*, vol. 2, no. 1, pp. 1432–1506, 2022.

[3] H. Yin, X. Xu, S. Lu, X. Chen, R. Xiong, S. Shen, C. Stachniss, and Y. Wang, “A Survey on Global LiDAR Localization: Challenges, Advances and Open Problems,” *International Journal of Computer Vision*, Mar 2024.

[4] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, “Visual Place Recognition: A Survey,” *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 1–19, 2016.

[5] N. Stathoulopoulos, M. A. Saucedo, A. Koval, and G. Nikolakopoulos, “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, pp. 4883–4889.

[6] G. Damigos, N. Stathoulopoulos, A. Koval, T. Lindgren, and G. Nikolakopoulos, “Communication-Aware Control of Large Data Transmissions via Centralized Cognition and 5G Networks for Multi-Robot Map merging,” *Journal of Intelligent & Robotic Systems*, vol. 110, no. 1, p. 22, Jan 2024.

[7] Y. Tian, Y. Chang, F. Herrera Arias, C. Nieto-Granda, J. P. How, and L. Carlone, “Kimera-Multi: Robust, Distributed, Dense Metric-Semantic SLAM for Multi-Robot Systems,” *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2022–2038, 2022.

[8] J. Ma, J. Zhang, J. Xu, R. Ai, W. Gu, and X. Chen, “OverlapTransformer: An Efficient and Yaw-Angle-Invariant Transformer Network for LiDAR-Based Place Recognition,” *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6958–6965, 2022.

[9] G. Kim and A. Kim, “Scan Context: Egocentric Spatial Descriptor for Place Recognition Within 3D Point Cloud Map,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4802–4809.

[10] P. Liu, S. Bao, L. Du, J. Yuan, H. Zhang, and R. Luo, “ECH: An Enhanced Cart Histogram for place recognition,” *2023 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1–6, 2023.

[11] H. Gao, Q. Qiu, S. Zhang, W. Hua, Z. Su, and X. Zhang, “MPC-MF: Multi-Point Cloud Map Fusion Based on Offline Global Optimization for Mobile Robots,” *Chinese Control Conference, CCC*, vol. 2023-July, pp. 4237–4242, 2023.

[12] N. Stathoulopoulos, B. Lindqvist, A. Koval, A.-A. Agha-Mohammadi, and G. Nikolakopoulos, “FRAME: A Modular Framework for Autonomous Map Merging: Advancements in the Field,” *IEEE Transactions on Field Robotics*, vol. 1, pp. 1–26, 2024.

[13] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and R. Daniela, “LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5135–5142.

[14] Y. Huang, T. Shan, F. Chen, and B. Englot, “DiSCo-SLAM: Distributed Scan Context-Enabled Multi-Robot LiDAR SLAM with Two-Stage Global-Local Graph Optimization,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1150–1157, 2022.

[15] K. Ebadi, Y. Chang, M. Palieri, A. Stephens, A. Hatteland, E. Heiden, A. Thakur, N. Funabiki, B. Morrell, S. Wood, L. Carlone, and A. A. Agha-Mohammadi, “LAMP: Large-Scale Autonomous Mapping and Positioning for Exploration of Perceptually-Degraded Subterranean Environments,” pp. 80–86, 2020.

[16] B. Kim, C. Jung, D. H. Shim, and A. Agha–mohammadi, “Adaptive Keyframe Generation based LiDAR Inertial Odometry for Complex Underground Environments,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 3332–3338.

[17] K. Chen, B. T. Lopez, A.-a. Agha-mohammadi, and A. Mehta, “Direct LiDAR Odometry: Fast Localization With Dense Point Clouds,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2000–2007, 2022.

[18] Q. Zeng, D. Liu, Y. Zhou, and Y. Peng, “Entropy-based Keyframe Established and Accelerated Fast LiDAR Odometry and Mapping,” *ITOEC 2023 - IEEE 7th Information Technology and Mechatronics Engineering Conference*, vol. 7, pp. 347–354, 2023.

[19] F. Ou, Y. Li, Z. Miao, and J. Zhou, “Lidar Odometry Key Frame Selection Based on Displacement Vector Similarity,” *Chinese Control Conference, CCC*, vol. 2021-July, pp. 3588–3593, 2021.

[20] Y. Cui, Q. Wu, Y. Hao, Y. Kong, Z. Lin, and F. Zhu, “Fast Relocalization and Loop Closing in Keyframe-Based 3D LiDAR SLAM,” *2022 IEEE International Conference on Robotics and Biomimetics, ROBIO 2022*, pp. 590–595, 2022.

[21] Y. Cui, X. Chen, Y. Zhang, J. Dong, Q. Wu, and F. Zhu, “BoW3D: Bag of Words for Real-Time Loop Closing in 3D LiDAR SLAM,” *IEEE Robotics and Automation Letters*, vol. 8, no. 5, pp. 2828–2835, 2023.

[22] Y. Cui, Y. Zhang, J. Dong, H. Sun, X. Chen, and F. Zhu, “LinK3D: Linear Keypoints Representation for 3D LiDAR Point Cloud,” *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2128–2135, 2024.

[23] M. Bosse, G. Agamennoni, and I. Gilitschenski, “Robust Estimation and Applications in Robotics,” *Foundations and Trends® in Robotics*, vol. 4, no. 4, pp. 225–269, 2016.

[24] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[25] G. Pandey, J. R. McBride, and R. M. Eustice, “Ford Campus vision and lidar data set,” *The International Journal of Robotics Research*, vol. 30, no. 13, pp. 1543–1552, 2011.