RMS: Redundancy-Minimizing Point Cloud Sampling for Real-Time Pose Estimation

Pavel Petracek [ID], Kostas Alexis [ID], and Martin Saska [ID], *Member, IEEE*

Abstract—The typical point cloud sampling methods used in state estimation for mobile robots preserve a high level of point redundancy. This redundancy unnecessarily slows down the estimation pipeline and may cause drift under real-time constraints. Such undue latency becomes a bottleneck for resource-constrained robots (especially UAVs), requiring minimal delay for agile and accurate operation. We propose a novel, deterministic, uninformed, and single-parameter point cloud sampling method named RMS that minimizes redundancy within a 3D point cloud. In contrast to the state of the art, RMS balances the translation-space observability by leveraging the fact that linear and planar surfaces inherently exhibit high redundancy propagated into iterative estimation pipelines. We define the concept of *gradient flow*, quantifying the local surface underlying a point. We also show that maximizing the entropy of the *gradient flow* minimizes point redundancy for robot ego-motion estimation. We integrate RMS into the *point*-based KISS-ICP and *feature*-based LOAM odometry pipelines and evaluate experimentally on KITTI, Hilti-Oxford, and custom datasets from multirotor UAVs. The experiments demonstrate that RMS outperforms state-of-the-art methods in speed, compression, and accuracy in well-conditioned as well as in geometrically-degenerated settings.

Index Terms—Localization, range sensing, aerial systems: perception and autonomy.

I. INTRODUCTION

FOR the accurate and real-time ego-motion estimation of a resource-constrained robot, the amount of data provided in a 3D LiDAR point cloud is plentiful (see Fig. 1). To achieve convergence under real-time constraints (i.e., number of iterations, comp. time, convergence rate), the point clouds must be reduced. Apart from cardinality reduction, the objectives of such point cloud sampling are twofold — preserve the quality of the points and be computationally fast. While the latter is subject
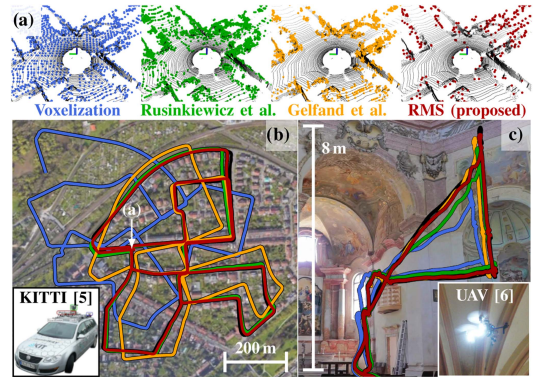

Fig. 1.   Fast and noise-filtering 3D point cloud sampling can speed up real-time estimation pipelines. (a) An example of a single-frame sampling at the crossroad highlighted in (b) by each of the given methods (input point cloud in black). (b)–(c) Although sampling in the input space is uninformed about point-map correspondences, such sampling can improve performance if the sampling is fast and preserves the quality of the points (e.g., removes non-informative points). (b) Trajectory estimated by KISS-ICP [7] odometry on KITTI seq. #00 when preceded by one of the sampling methods (ground truth in black). Similarly, (c) shows trajectory estimated on-board a UAV using LOAM [8] odometry in a vertically self-symmetrical church in Stará Voda [6].

to algorithm efficiency and available computational resources, the former must preserve the overall information available in the point cloud. In the task of point cloud matching, the contribution (i.e., information) of a point has been shown to be quantifiable via its point-map correspondence and the shape of the loss function [1], [2], [3], [4]. However, information-aware sampling of an input point cloud without the knowledge of these point-map correspondences (uninformed sampling) is non-causal and remains an ongoing challenge.

The typical uninformed point cloud sampling methods include feature extraction [8], [9], point-density normalization [7], [10], normal-space sampling [11], [12], and learning-based inference [13], [14], [15], [16], [17]. With the individual advantages and disadvantages of these widely used methodologies, the overall challenges remain in their effectiveness, latency minimization, and environment adaptability. The experimental part of this letter shows that finding optimal parameters of such methods is often a balance between speed and accuracy. Our analyses also show that the optimal parameters are rarely adaptable to different LiDAR sensors, estimation pipelines, and environment types; and need to be exhaustively tuned for every instance.

Lastly, fast and noise-removing uniformed sampling has been shown to improve the performance of real-time pipelines in well-conditioned settings [7], [8]. However, it has also been shown that uninformed methods may improve performance in environments with a low amount of salient geometrical structures if these salient structures are part of the sampled data.

Pavel Petracek and Martin Saska are with the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University, 121 00 Prague, Czech Republic (e-mail: pavel.petracek@fel.cvut.cz).

Kostas Alexis is with the Autonomous Robots Lab, Norwegian University of Science and Technology, 7034 Trondheim, Norway (e-mail: konstantinos.alexis@ntnu.no).

We denote these settings, where the point cloud contains only a handful of exploitable structures, *weakly* degenerate. These settings most notably emerge in geometrically symmetrical environments, such as subterranean tunnels [18] and caves [19], and vertically-symmetrical historical monuments [6].

II. RELATED WORK

A ubiquitous point cloud sampling method is uniform sampling (voxelization), which discretizes space into fixed-sized cubes, each containing N (typically 1) points at maximum. Typical voxel-filter implementations employ an octree structure [20] or use a simple numerical discretization, such as that implemented in PCL [10]. The feature extraction methods consist of learning-based solutions (such as PointNet++ [9]) and hand-crafted feature (most commonly plane and line features defined in LOAM [8]) extractors. Although these methods perform reasonably well in geometrically rich settings when tuned properly, they are sensitive to parametrization. Moreover, learning-based methods lack sampling guarantees and require each environment to be part of the training data.

A deterministic sampling method [11] selects points such that their normals uniformly fill the normal-vector space. The covariance-based sampling (CovS) [12] iteratively selects the points, which maximize the expected normal-based contribution to the DoF least constrained in the eigenspace of the sampled set. Both methods [11], [12] have shown that point normals can be a helpful mechanism in guiding the sampling under *weak* geometrical degeneracy. However, obtaining the point normals cheaply, correctly, and reliably is challenging, especially given the projection nature of modern 3D light detection and rangings (LiDARs) that generate data with uneven density and surface occlusions. PFilter [21] and ROI-cloud [22] are designed for use in a robot ego-motion estimation by employing previous LiDAR scans. PFilter [21] assigns each point a *persistency-index*, quantifying how persistent the point is over a short history of measurements. Static points, favorable in correspondence matching, tend to score higher in persistence. The ROI-cloud [22] divides space into cubes weighted by the amount of inlying geometrical features. [22] then propagates virtual particles representing past measurements and fuses them with the weighted cubes. Points are then sampled in areas where the weighted cubes align with the particles' distribution.

Among data-based sampling methods lies SampleNet [13], which learns task-specific sampling for object classification and geometry reconstruction. The method in [14] learns features and selects the points with the greatest contribution to the global max-pooling. DGCNN [15], FoldingNET [16], and KCNET [17] convert the point cloud into a graph and resample based on graph-based max-pooling, which takes the maximum features over the neighborhood of each vertex using a pre-built k-NN graph. The disadvantage of these methods is the absence of deterministic guarantees that the sampling will be invariant to the type of environment, and that it will maximize point relevancy in estimation.

Among the relevant redundancy-minimizing methods is [23]. Therein, the authors show that fewer correspondences are better in global registration, given that the correspondences are accurate. A map-compressing method [24] then applies concepts of feature similarity to select only one of the nearby features, marking the rest redundant and removing them. However, being formulated for expensive global registration and map compression, [23], [24] are inapplicable in front end of a real-time ego-motion estimation of a robot.
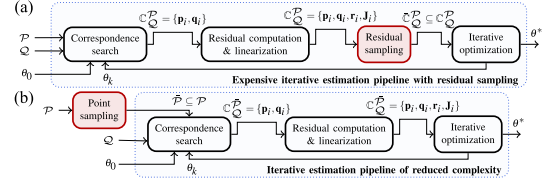

Fig. 2. Pipeline of an iterative pose estimation pipeline extended with (a) in-the-loop residual sampling and (b) a single-shot input data sampling. (a) The formulation (used in [2], [3], [4]) utilizes the full point cloud $\mathcal{P}$ and introduces a significant overhead in each iteration. (b) The proposed architecture includes a single-shot out-of-the-loop sampling, which lowers the overall complexity by reducing the input size.

It has also been proposed that sampling is to be performed at the optimization level once the point-to-map correspondences are found. The greedy-based method [2] selects the optimization residuals such that the log-determinant of the approximate Hessian of the optimization problem is maximized. KFS-LIO [3] does so similarly, but maximizes the inverse trace of the Hessian. X-ICP [4] filters out residuals with non-parallel plane normals per each DoF, similarly to the normal-space equalization proposed in [11]. Simplified version Xs-ICP [4] does similarly to X-ICP but reuses the residuals computed in the first iteration as a prior in subsequent iterations. The advantage of sampling at the optimization level (informed) is the possibility to relate to the information theory, allowing to formulate awareness to degeneracy in the optimization. In particular, [2], [3], [4] utilize the eigenspace of the information matrix to quantify the degeneracy in the optimization, as introduced in [1]. However, residual-space sampling is sensitive to noise in correspondences and variability in point density and comes at a cost of re-sampling in every iteration of an estimation pipeline (see Fig. 2). Uninformed input-space sampling is computed only once per point cloud, but cannot directly relate to the degeneracy without the correspondence pairings.

The contributions of this letter include a novel out-of-the-loop 3D point cloud sampling named Redundancy-Minimizing Sampling (RMS). The method minimizes point redundancy within a point cloud by maximizing the entropy of the *gradient flow* in the sampled set. It builds upon the fact that hyperplane surfaces (i.e., linear and planar surfaces) contain a high level of redundancy propagated into the iterative estimation pipeline. Instead of classifying points into surface types, we propose a *gradient flow heuristic* (GFH) quantifying the potential of a point to lie on a hyperplane surface based on its local point distribution. The method is fast, uninformed, and deterministic and does not need point-normals to be known, is independent on the environment, is effectively parametrizable by a single parameter only, and is integrable into most state-of-the-art LiDAR-based odometries and SLAMs, both *dense* (using entire point clouds) and *feature*-based.

III. PROBLEM DEFINITION

The underlying problem of a six DoF robot ego-motion estimation from LiDAR data is scan matching. Scan matching can be formulated as finding the parameters $\theta^* \in SE(3)$, minimizing the squared sum of the residual functions $\mathbf{r} \in \mathbb{R}^3$ over two point sets $\mathcal{P} = \{\mathbf{p} \in \mathbb{R}^3\}$ and $\mathcal{Q} = \{\mathbf{q} \in \mathbb{R}^3\}$

$$\arg\min_{\theta \in SE(3)} g_\theta(\mathcal{P}, \mathcal{Q}) = \arg\min_{\theta \in SE(3)} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_{\mathcal{Q}}^{\mathcal{P}}} \rho\left(||\mathbf{r}(\theta, \mathbf{p}, \mathbf{q})||_2^2\right), \quad (1)$$

where $\mathbb{C}_Q^{\mathcal{P}}$ represents the set of correspondence pairs from $\mathcal{P}$ to $\mathcal{Q}$ and ρ is a robust kernel with outlier rejection properties. Formulated as a pose estimation task, $\theta = \{\mathbf{t}, \mathbf{R}\}$ consists of a translation $\mathbf{t} \in \mathbb{R}^3$ and a rotation $\mathbf{R} \in SO(3)$ of the pose change from $\mathcal{P}$ to $\mathcal{Q}$. Note that, $\mathcal{P}$ and $\mathcal{Q}$ can be entire LiDAR scans in *dense* or extracted features in *feature*-based formulations, and that the most prevalent $\mathbf{r}$ functions in common iterative scan matchers are the *point-to-point*, *point-to-plane*, and *point-to-line* metrics, which are for a pair $(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\mathcal{P}}$ given as

$$\mathbf{r}^{\bullet} = \theta \mathbf{p} - \mathbf{q}, \quad \mathbf{r}^{\square} = (\mathbf{n}^{\mathsf{T}} \mathbf{r}^{\bullet} + d) \mathbf{n}, \quad \mathbf{r}^{|} = \mathbf{r}^{\bullet} - ((\mathbf{r}^{\bullet})^{\mathsf{T}} \mathbf{v}) \mathbf{v}, \tag{2}$$

where $(\mathbf{n}, d)$ is the parametrization of a plane that $\mathbf{q}$ lies on ($\mathbf{n}$ is a unit surface normal), $\mathbf{v}$ is a unit direction of a line that $\mathbf{q}$ lies on, and $\theta \mathbf{p} = \mathbf{R}\mathbf{p} + \mathbf{t}$.

In the related correspondence selection methods [2], [3], [4], the selection is formulated as finding a minimum-information correspondence subset that improves the performance of an iterative matching process in degenerate scenarios. Commonly, these works formulate the problem as a minimization task

$$\underset{\theta \in SE(3)}{\text{minimize}} \sum_{(\mathbf{p}, \mathbf{q}) \in \bar{\mathbb{C}}_Q^{\mathcal{P}}} \rho \left(||\mathbf{r}(\theta, \mathbf{p}, \mathbf{q})||_2^2 \right), \tag{3}$$

$$\text{subject to} \quad \bar{\mathbb{C}}_Q^{\mathcal{P}} \subseteq \mathbb{C}_Q^{\mathcal{P}}, \ \bar{\mathbb{C}}_Q^{\mathcal{P}} \neq \emptyset, \tag{4}$$

where $\bar{\mathbb{C}}_Q^{\mathcal{P}}$ is a fixed-cardinality subset of correspondences selected from $\mathbb{C}_Q^{\mathcal{P}}$ with respect to the log determinant [2] or inverse trace [3] of the information matrix, and as a sum of constraints per optimization direction in the objective function [4]. Finding point-map correspondences $\mathbb{C}_Q^{\mathcal{P}}$ and then identifying the optimal subset $\bar{\mathbb{C}}_Q^{\mathcal{P}}$ is expensive, especially when repeatedly computed within iterative algorithms.

Proposed formulation decreases the problem dimensionality by selecting points in the input scan $\mathcal{P}$ before the iterative process of correspondence search, linearization, residual sampling, and optimization. We formulate the pose estimation as

$$\arg\min_{\substack{\theta \in SE(3), \\ \bar{\mathcal{P}} \subseteq \mathcal{P}}} g_\theta(\bar{\mathcal{P}}, \mathcal{Q}) = \arg\min_{\substack{\theta \in SE(3), \\ \bar{\mathcal{P}} \subseteq \mathcal{P}}} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\bar{\mathcal{P}}}} \rho \left(||\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})||_2^2 \right), \tag{5}$$

where $\mathbb{C}_Q^{\bar{\mathcal{P}}}$ is a set of correspondence pairs from $\bar{\mathcal{P}}$ to $\mathcal{Q}$ and

$$\bar{\mathcal{P}} = \underset{\Omega \in \{\Theta \mid \Theta \subseteq \mathcal{P}, \Theta \neq \emptyset\}}{\arg\min} |\Omega|, \tag{6}$$

$$\text{subject to} \quad \underset{\theta \in SE(3)}{\arg\min} g_\theta(\mathcal{P}, \mathcal{Q}) = \underset{\theta \in SE(3), \bar{\mathcal{P}} \subseteq \mathcal{P}}{\arg\min} g_\theta(\bar{\mathcal{P}}, \mathcal{Q}). \tag{7}$$

In other words, we formulate the problem as finding a minimum-cardinality subset $\bar{\mathcal{P}} \subseteq \mathcal{P}$ over which the minimization problem converges to the same optimum as in the original formulation. Differences in iterative pipelines using formulations in (3) and (5) are shown in Fig. 2.

IV. INFORMATION REDUNDANCY MINIMIZATION

The problem formulated in (5)–(7) requires finding a minimal-cardinality non-empty subset of points $\bar{\mathcal{P}} \subseteq \mathcal{P}$ over which the estimation converges to the optimum without prior knowledge about the correspondences among point sets $\mathcal{P}$ and $\mathcal{Q}$. This makes the formulation NP-hard and non-causal as the information about a point contribution to the optimization is unknown without its target correspondence. When a correspondence $(\mathbf{p}_i, \mathbf{q}_i)$ is known, the related works [2], [3], [4] define

its contribution in relation to the eigenspace of the information matrix ${}^i\mathbf{J}_\theta^{\mathsf{T}}{}^i\mathbf{J}_\theta$, where

$$^i\mathbf{J}_\theta = \left[\frac{\partial \mathbf{r}_\theta(\mathbf{p}_i, \mathbf{q}_i)}{\partial \mathbf{t}}, \frac{\partial \mathbf{r}_\theta(\mathbf{p}_i, \mathbf{q}_i)}{\partial \mathbf{R}} \right] \tag{8}$$

is the Jacobian of the residual function $\mathbf{r}_\theta$ (e.g., $\mathbf{r}_\theta^{\square}$ from (2) in [2]), or with relation to the approximate Hessian of the opt. problem [1] given as $({}^{\mathcal{P}}\mathbf{J}_\theta)^{\mathsf{T}}{}^{\mathcal{P}}\mathbf{J}_\theta$, where ${}^{\mathcal{P}}\mathbf{J}_\theta = \sum_{i=1}^{|\mathbb{C}_Q^{\mathcal{P}}|} {}^i\mathbf{J}_\theta$. Although this makes the problem causal, finding the optimal minimum-cardinality subset is still NP-hard; and remains an open challenge.

To tackle this problem, we propose to approximate the solution to the problem formulated in (5)–(7) by defining, finding, and removing redundancy within a point set without knowledge about the correspondences. When applied to a typical iterative process of a robot's ego-motion estimation, the proposed solution inherently removes noise and lowers the computational latency. When under real-time termination criteria (e.g., number of iterations, rate of change), the lowered cost improves the rate and accuracy of convergence.

A. Redundancy in a Point Set

Every perceived environment can be decomposed into a set of S atomic surfaces $\mathbb{S} = \bigcup_{s \in \langle 1, S \rangle} \mathbb{S}_s$ of arbitrary complexities, ranging from linear and planar to quadratic and other nonlinear areas. In this work, the environment is assumed to be decomposable into linear and planar (hyperplane) surfaces. An input point set $\mathcal{P}$ can then be understood as a discretization of the observed hyperplane surfaces $\mathcal{P} = \bigcup_{s \in \langle 1, S \rangle} \mathcal{P}_s$, where $\mathcal{P}_s$ represents a set of points observed on the surface $\mathbb{S}_s$.

Definition 1: A single-surface point set $\mathcal{P}_s$ contains information redundancy if removing one or multiple points from the set does not change its rate of information (average entropy) regarding the optimization problem.

Remark 1: In the optimization task defined in (1)–(5), the redundancy represents points that generate identical (parallel and of the same magnitude) residuals, whose removal does not alter the loss function, nor does it change the global optimum of the objective function.

Without the loss of generality, the robust kernel in (1)–(5) can be omitted for now, and a set-residual function (the objective function) can be defined as the sum of the point residuals

$$\mathbf{r}_\theta(\mathcal{P}) = \sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\mathcal{P}}} ||\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})||_2^2 \tag{9}$$

to be minimized. Given the set of atomic surfaces $\mathbb{S}$ and their corresponding point sets $\mathcal{P}_s$, (9) can be equivalently defined as a sum of surface-subset residuals

$$\mathbf{r}_\theta(\mathbb{S}) = \sum_{\mathcal{P}_s \in \mathbb{S}} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\mathcal{P}_s}} ||\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})||_2^2. \tag{10}$$

Definition 2: Without altering the translational optimum, the objective function can be defined as a sum of set-residual rates

$$\bar{\mathbf{r}}_\theta(\mathbb{S}) = \sum_{\mathcal{P}_s \in \mathbb{S}} \bar{\mathbf{r}}_\theta(\mathcal{P}_s) = \sum_{\mathcal{P}_s \in \mathbb{S}} \frac{1}{\left| \mathbb{C}_Q^{\mathcal{P}_s} \right|} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\mathcal{P}_s}} ||\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})||_2^2. \tag{11}$$

Proof: The Jacobian of the obj. function defined in (9) is

$$^{\mathcal{P}}\mathbf{J}_\theta = \frac{\partial \left(\sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\mathcal{P}}} ||\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})||_2^2 \right)}{\partial \theta}$$

$$= \sum_{(\mathbf{p},\mathbf{q})\in\mathbb{C}_Q^{\mathcal{P}}} \frac{\partial ||\mathbf{r}_\theta(\mathbf{p},\mathbf{q})||_2^2}{\partial\theta} = 2\sum_{(\mathbf{p},\mathbf{q})\in\mathbb{C}_Q^{\mathcal{P}}} \mathbf{r}_\theta(\mathbf{p},\mathbf{q}) \quad (12)$$

and the Jacobian of (10) is given as

$$^{\mathbb{S}}\mathbf{J}_\theta = \frac{\partial\left(\sum_{\mathcal{P}_s\in\mathbb{S}}\sum_{(\mathbf{p},\mathbf{q})\in\mathbb{C}_Q^{\mathcal{P}_s}} ||\mathbf{r}_\theta(\mathbf{p},\mathbf{q})||_2^2\right)}{\partial\theta}$$

$$= \sum_{\mathcal{P}_s\in\mathbb{S}} |\mathcal{P}_s| \frac{\partial ||^s\mathbf{r}_\theta(\mathbf{p},\mathbf{q})||_2^2}{\partial\theta} = 2\sum_{\mathcal{P}_s\in\mathbb{S}} |\mathcal{P}_s|\, ^s\mathbf{r}_\theta(\mathbf{p},\mathbf{q}), \quad (13)$$

where $^s\mathbf{r}_\theta$ is a common residual for the redundant surface s. As each hyperplane surface s contains $|\mathcal{P}_s|$ identical residuals (see Definition 3), the simplification $\sum_{(\mathbf{p},\mathbf{q})\in\mathbb{C}_Q^{\mathcal{P}_s}} ||\mathbf{r}_\theta(\mathbf{p},\mathbf{q})||_2^2 = |\mathcal{P}_s|\cdot||^s\mathbf{r}_\theta||_2^2$ makes the Jacobians in (12) and (13) identical, assuming perfect point-to-surface associations. The Jacobian of (11) is derived similarly as in (12) and (13) as

$$^{\mathbb{S}}\bar{\mathbf{J}}_\theta = \frac{\partial\left(\sum_{\mathcal{P}_s\in\mathbb{S}} \bar{\mathbf{r}}_\theta(\mathcal{P}_s)\right)}{\partial\theta}$$

$$= \sum_{\mathcal{P}_s\in\mathbb{S}} \frac{1}{|\mathcal{P}_s|} \frac{\partial\left(\sum_{(\mathbf{p},\mathbf{q})\in\mathbb{C}_Q^{\mathcal{P}_s}} ||\mathbf{r}_\theta(\mathbf{p},\mathbf{q})||_2^2\right)}{\partial\theta}$$

$$= \sum_{\mathcal{P}_s\in\mathbb{S}} \frac{1}{|\mathcal{P}_s|} \frac{\partial\left(|\mathcal{P}_s|\cdot||^s\mathbf{r}_\theta||_2^2\right)}{\partial\theta} = 2\sum_{\mathcal{P}_s\in\mathbb{S}} \, ^s\mathbf{r}_\theta. \quad (14)$$

The Hessian matrices of all three formulations are given as

$$^{\mathcal{P}}\mathbf{H}_\theta = 2\sum_{i=1}^{|\mathcal{P}|} \, ^i\mathbf{J}_\theta, \qquad ^{\mathbb{S}}\mathbf{H}_\theta = 2\sum_{s=1}^{|\mathcal{P}_s|} |\mathcal{P}_s|\, ^s\mathbf{J}_\theta,$$

$$^{\mathbb{S}}\bar{\mathbf{H}}_\theta = 2\sum_{s=1}^{|\mathcal{P}_s|} \, ^s\mathbf{J}_\theta, \quad (15)$$

where $^s\mathbf{J}_\theta$ is the Jacobian of $^s\mathbf{r}_\theta$, as per (16)–(18) below.

The Jacobians of the residual functions (defined in (2)) are given analytically according to (8) as

$$^i\mathbf{J}_\theta^\bullet = [\mathbf{I}, \; -\mathbf{R}[\mathbf{p}_i]_\times], \quad (16)$$

$$^i\mathbf{J}_\theta^\square = [\mathbf{n}_i^\mathsf{T}\mathbf{n}_i, \; -\mathbf{n}_i^\mathsf{T}\mathbf{n}_i\mathbf{R}[\mathbf{p}_i]_\times], \quad (17)$$

$$^i\mathbf{J}_\theta^\mid = [\mathbf{I} - \mathbf{v}_i\mathbf{v}_i^\mathsf{T}, \; -(\mathbf{I} - \mathbf{v}_i\mathbf{v}_i^\mathsf{T})\mathbf{R}[\mathbf{p}_i]_\times], \quad (18)$$

given that $\mathbf{I}\in\mathbb{R}^{3\times3}$ and $\frac{\partial}{\partial\mathbf{R}}(\mathbf{R}\mathbf{p}_i) = -\mathbf{R}[\mathbf{p}_i]_\times$, where $[\mathbf{p}_i]_\times\in\mathbb{R}^{3\times3}$ is the skew-symmetric matrix of $\mathbf{p}_i$. It is clear that in the translational space (the $\partial/\partial\mathbf{t}$ part of the Jacobians), the change in residuals is, for the most common metrics, either constant or a function of the surface parameters. Since the translational change depends only on the surface s, selecting a single residual $\mathbf{r}_s$ per surface preserves the basis of the translational eigenspace of both the Jacobian and Hessian matrices. Thus, the global optimum in the translational space of the objective function remains unchanged. ∎

Remark 2: Although this reformulation does not alter the translational optimum, it reshapes the respective part of the objective func. g_θ to $\bar{g}_\theta$ without changing its monotonic intervals

$$\forall\mathbf{x},\mathbf{y}\in\mathbb{R}^D, \quad g_\theta(\mathbf{x})\odot g_\theta(\mathbf{y}) \Rightarrow \bar{g}_\theta(\mathbf{x})\odot\bar{g}_\theta(\mathbf{y}), \quad (19)$$

where $\odot$ is any linear inequality operator and D is the problem dimensionality.

Remark 3: When constrained to an ego-motion estimation task, we can assume the rotation changes to be small. Under this assumption, the first-order linearization of $\mathbf{R}$ is given as $\mathbf{R}\approx$
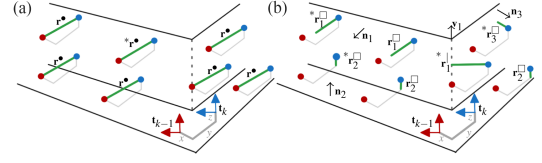

Fig. 3. Simplistic case of point redundancy (Definitions 2 and 3) for a robot translating from the position $\mathbf{t}_{k-1}$ (● points) to $\mathbf{t}_k$ (● points). (a) The *point-to-point* metric generates identical residuals, which makes the residual rate constant for any positive number of residuals used, e.g., a single residual generates $\bar{\mathbf{r}}(\mathcal{P}) = {}^*\mathbf{r}^\bullet$. (b) The point-to-hyperplane metrics generate identical residuals per surface (in this example, the surfaces comprise three planes and a single line). The minimized objective function remains constant if any positive number of points is sampled per each surface. In this case, using the minimum amount of samples yields the residual rate $\bar{\mathbf{r}}(\mathcal{P}) = {}^*\mathbf{r}_1^\mid + \sum_{i=1}^3 {}^*\mathbf{r}_i^\square$. This example assumes perfect correspondences, which is unrealistic under noise and rotation. The point sampling method proposed in Section IV-B is designed to be robust to cases where this assumption is not met.

$\mathbf{I} + [\mathbf{b}]_\times$, where $\mathbf{b} = [\alpha, \beta, \gamma]^\mathsf{T}$ is vector of the three rotational degrees of freedoms (DoFs). Then, the rotational space in (16)–(18) reduces to a function of surface parameters and $[\mathbf{p}_i]_\times$, which denotes that sensitivity to rotations increases with point distance. This means that two points are also redundant in the rotational space if they belong to the same surface and have equal $[\mathbf{p}_i]_\times$.

Definition 3: Assuming zero noise, every hyperplane surface $\mathbb{S}_s$ generates $|\mathcal{P}_s| - 1$ redundant residuals.

Proof: Given the hyperplane surfaces and their point-set observations $\mathcal{P}_s$, the set-residual rate $\bar{\mathbf{r}}_\theta(\mathcal{P}_s) = \bar{\mathbf{r}}_\theta(\Pi)$ applies for all $\Pi\in\{\pi\,|\,\pi\subset\mathcal{P}_s, \pi\neq\emptyset\}$. (14) then shows that reducing the cardinality of $\mathcal{P}_s$ from $|\mathcal{P}_s|$ to 1 preserves the translational optimum, which implies that $|\mathcal{P}_s| - 1$ residuals are redundant. ∎

Fig. 3 shows an idealized case demonstrating redundancy in surface-point sets, as defined in Definitions 2 and 3.

Definition 4: Assuming the presence of noise, Definitions 2 and 3 can be generalized to find the min-cardinality non-empty subset $\hat{\mathcal{P}}_s\subseteq\mathcal{P}_s$ whose set-residual rate matches the one of its superset

$$\hat{\mathcal{P}}_s = \underset{\Omega\in\{\Theta\,|\,\Theta\subseteq\mathcal{P}_s,\Theta\neq\emptyset\}}{\arg\min} |\Omega| \quad (20)$$

$$\text{subject to} \quad \bar{\mathbf{r}}_\theta(\Omega) = \bar{\mathbf{r}}_\theta(\mathcal{P}_s), \quad (21)$$

for each set of surface points $\mathcal{P}_s$. Given this formulation, each surface contains $|\mathcal{P}_s\setminus\hat{\mathcal{P}}_s|$ redundant residuals. Substituting (20) into (11) yields the obj. function in the form of

$$\mathbf{r}_\theta(\mathbb{S}) = \sum_{\mathcal{P}_s\in\mathbb{S}} \bar{\mathbf{r}}_\theta(\hat{\mathcal{P}}_s). \quad (22)$$

Remark 4: The reformulation is feasible since the Definition 2 maintains the convergence properties exploitable by the nonlinear solvers. The optimum consistency further satisfies (7).

In practice, the data are usually unstructured and are subjected to noise, making it expensive to segment the input point set $\mathcal{P}$ into a set of surfaces, even trivially. Instead of finding and segmenting the underlying surfaces (as defined in Definitions 3 and 4), we propose in Section IV-B a heuristic for the direct quantification of the redundancy without point-surface associations. In Section IV-C, we then propose a redundancy-minimizing algorithm robust towards noise, independent of correspondence matching, and invariant to small rotations.
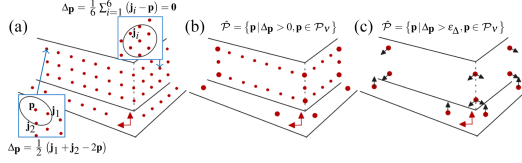
Fig. 4. Proposed *gradient flow heuristic* for quantifying redundancy in a point set. (a) GFH is computed for each point in a voxelized point set $\mathcal{P}_\nu$. (b) Points on the perceived borders have generally non-zero $\Delta_\mathbf{p}$, whereas (c) corner points yield the maximum $\Delta_\mathbf{p}$ (herein thresholded by an abstract value ϵ_Δ). Keeping the max-$\Delta_\mathbf{p}$ subset (c) ensures that all directions remain constrained, as shown by the black axes representing which translational directions the points constraint.

B. Quantifying the Redundancy

As discussed at the beginning of Section IV, our objective is to find redundancy within a point set $\mathcal{P}$ without knowing the correspondences $\mathbb{C}^\mathcal{P}_\mathcal{Q}$ beforehand. We tackle this by introducing a *gradient flow heuristic* quantifying the uniqueness of a point by local flow of a geometric gradient. The GFH maximizes the potential of points in bringing unique information to the optimization once their correspondences are found. Instead of expensive segmentation of the set $\mathcal{P}$ into surface observations $\mathcal{P}_s$ (as formulated in Definitions 3 and 4), the GFH quantifies whether a point is locally a part of any hyperplane. Since generating multiple residuals on a single hyperplane is a source of the redundancy (as defined in Definitions 1–4), this opens a way to the redundancy minimization discussed in Section IV-C.

The GFH emerges from Definitions 2 and 3, which define that identical (in orientation and magnitude) residuals are redundant in structuring the objective function and that on a single hyperplane, the residuals are identical inherently. To quantify the uniqueness of points (and thus, the potential of future residuals), the GFH finds the neighbors of each point $\mathbf{p} \in \mathcal{P}$ within a spherical neighborhood with radius $\lambda_\mathbf{p}$ (m)

$$\mathcal{N}_\mathbf{p} = \{\mathbf{j} \mid ||\mathbf{j} - \mathbf{p}||_2 < \lambda_\mathbf{p}, \mathbf{j} \neq \mathbf{p}, \mathbf{j} \in \mathcal{P}\}, \quad (23)$$

and defines the *gradient flow* (in meters) as

$$\Delta_\mathbf{p} = \frac{1}{|\mathcal{N}_\mathbf{p}|} \sum_{\mathbf{j} \in \mathcal{N}_\mathbf{p}} \mathbf{j} - \mathbf{p}. \quad (24)$$

As demonstrated in Fig. 4, the GFH scores high for points lying on the borders of a surface and low for points lying inside the borders (inliers). Maximizing GFH thus leads to prioritizing the borders of surfaces rather than the surface inliers, which is important for two reasons. First, the borders in $\mathcal{P}$ have the largest potential for correct correspondence matching with the borders of the corresponding physical surface. Second, the inliers can generate erroneous local minima and resist sliding along the directions of a hyperplane when using the *point-to-point* metric. As discussed in [25], the point-to-hyperplane metrics do not suffer from this deficiency, but it is still valuable to remove the redundancy to increase efficiency.

Every point-set matching algorithm more or less voxelizes the input set by a constant voxel size factor ν in order to reduce the cardinality of the input point set to $\mathcal{P}_\nu \subset \mathcal{P}$. We exploit this by setting $\lambda_\mathbf{p} = 2\nu$ for unstructured point sets. In structured point sets coming from a rotating 3D LiDAR (e.g., Ouster), the neighborhood radius instead respects the projective properties of these sensors as

$$\lambda_\mathbf{p} = 2 \max \left[\nu, ||\mathbf{p}||_2 \max \left(\sin \frac{2\pi}{C}, \sin \frac{\theta_v}{R-1} \right) \right], \quad (25)$$

where θ_v is vertical and 2π is a horizontal field of view of the sensor, which data are generated in a matrix form with R rows and C columns.

The neighborhood search of (23) is the only expensive part of the proposed methodology. We construct a KD-tree from the voxelized point set $\mathcal{P}_\nu$ to lower the cost. Using $\mathcal{P}_\nu$ lowers the construction cost of the KD-tree and reduces the number of KD-tree queries to $|\mathcal{P}_\nu|$. With construction complexity $\mathcal{O}(n \log n)$ and worst-case complexity of n-query radius search being $\mathcal{O}(n^2 \log n)$ (where n is $|\mathcal{P}|$ in the full and $|\mathcal{P}_\nu|$ in the voxelized case), the overall complexity is reduced since $|\mathcal{P}_\nu| < |\mathcal{P}|$. We show in Section V that the overhead for computing the GFH for all the points lowers the complexity of the ego-motion estimation and accelerates the full pipeline.

C. Removing the Redundancy

Although a redundancy might be beneficial for reducing the effects of noise and outliers, it makes the iterative process of correspondence finding, residual generation, linearization, and optimization more expensive. Under the presence of termination criteria, the process may be undesirably hindered by accurate in-time convergence.

To find a solution to the NP-hard problem formulated in (20)–(22), we could propose to solve an optimization task

$$\hat{\mathcal{P}} = \underset{\Omega \in \{\Theta \mid \Theta \subseteq \mathcal{P}_\nu, \Theta \neq \emptyset\}}{\arg\min} \Gamma_\Delta(\Omega), \quad \text{subject to} \quad |\Omega| = N_\Omega, \quad (26)$$

minimizing redundancy Γ_Δ in the *gradient flow* of the subset Ω under a constraint on fixed cardinality of the output set $N_\Omega \in (1, |\mathcal{P}_\nu|\rangle$. Although the concept of a fixed-cardinality constraint is common within the related works [2], [3], the notion of redundancy allows for a more rigorous formulation. Since minimizing redundancy in data can be understood as maximizing the data entropy, we instead define a dual task

$$\hat{\mathcal{P}} = \underset{\Omega \in \{\Theta \mid \Theta \subseteq \mathcal{P}_\nu, \Theta \neq \emptyset\}}{\arg\max} H_\Delta(\Omega),$$

$$\text{subject to} \quad \bar{H}_\Delta(\Omega) \leq \lambda_{\bar{H}}, \quad (27)$$

maximizing the entropy of information H_Δ in the *gradient flow* of the subset Ω under the termination criteria on the relative information rate $\bar{H}_\Delta$ (defined in (30)), given a maximum relative entropy rate $\lambda_{\bar{H}}$ (%). The termination criteria in (27) replaces the constraint on a fixed cardinality, which allows the selection to emergently adapt to the distribution of the points, making this method invariant to the type of environment. By thresholding the relative information rate via $\lambda_{\bar{H}}$, a certain level of redundancy is introduced into the system, possibly increasing robustness towards noise and outliers.

Let an entropy rate be an average entropy per point in set Ω

$$\bar{\mu}_\Delta(\Omega) = \frac{1}{|\Omega|} H_\Delta(\Omega), \quad (28)$$

where the entropy of the set is given as

$$H_\Delta(\Omega) = -\sum_{\mathbf{p} \in \Omega} p(\Delta_\mathbf{p}) \log p(\Delta_\mathbf{p}), \quad (29)$$

and p represents the probability of observing the GFH value $\Delta_\mathbf{p}$. The relative entropy rate (conditioned in (27)) is defined as the normalized entropy rate

$$\bar{H}_\Delta(\Omega) = \frac{1}{\bar{\mu}^*_\Delta(\Omega)} \bar{\mu}_\Delta(\Omega), \quad (30)$$

where

$$\bar{\mu}_\Delta^*(\Omega) = \max_{\Psi \in \{\Theta \,|\, \Theta \subseteq \Omega, \Theta \neq \emptyset\}} \bar{\mu}_\Delta(\Psi) \qquad (31)$$

represents the maximum entropy rate of all non-empty subsets $\Psi \subseteq \Omega$. Note that the entropy rate is an inverse function to redundancy within the data, which allows us to formulate the dual task in (27).

The probability function p is a function of the data $\mathcal{P}_\nu$, which are a function of the environment. To maintain invariance to the environment, p can not be modeled with a probability density function. Instead, we propose to use a frequentist's approach to approximate the probability function p. First, the GFH of each point in $\mathcal{P}_\nu$ is converted to its normalized norm

$$\bar{\Delta}_{\mathcal{P}_\nu} = \left\{ \frac{||\Delta_{\mathbf{p}}||_2}{\max ||\Delta_{\mathcal{P}_\nu}||_2} \,\middle|\, \mathbf{p} \in \mathcal{P}_\nu \right\}, \qquad (32)$$

where $\max ||\mathcal{P}_\nu||_2$ represents the maximum $||\Delta_{\mathbf{p}}||_2$ of any point $\mathbf{p} \in \mathcal{P}_\nu$. Second, a histogram $\mathcal{H}_\Delta$ with K bins bounded in interval $\langle 0, 1 \rangle$ is created out of the normalized GFH norms $\bar{\Delta}_{\mathcal{P}_\nu}$, where each bin $k \in \langle 1, K \rangle$ holds a point set $^k\mathcal{H}_\Delta$. The probability of a bin k is then approximated by $p_k = \frac{|^k\mathcal{H}_\Delta|}{|\mathcal{P}_\nu|}$.

In Algorithm 1, we propose a point sampling routine following formulation in (27). Given the fact that the uniform distribution function yields a maximum entropy, Algorithm 1 maximizes uniformity in the GFH by normalizing GFH values in histogram $\mathcal{H}_\Delta$. The routine constructs an empty histogram $\hat{\mathcal{H}}_\Delta$ and iteratively moves points from $\mathcal{H}_\Delta$ to $\hat{\mathcal{H}}_\Delta$. This is done by per-row sampling from $\mathcal{H}_\Delta$ via cyclic iterative selection, going from greater to lower bins and moving a single point in each of the bins k (if there is any) to the corresponding bin k in $\hat{\mathcal{H}}_\Delta$. The primary and secondary keys of sampling from a bin k are

$$\mathbf{p}_k = \arg\max_{\mathbf{p} \in {}^k\mathcal{H}_\Delta} \Delta_{\mathbf{p}}, \quad \mathbf{p}_k = \arg\max_{\mathbf{p} \in {}^k\mathcal{H}_\Delta} ||\mathbf{p}||_2. \qquad (33)$$

RMS does not balance rotation-space observability but exploits the fact that the rotational rate of residuals is a function of $||\mathbf{p}||_2$, as defined in Remark 3. This is done via the secondary key in (33), which values points by their potential for being part of a large-magnitude residual in the later correspondence-matching part of the estimation. Note that when the assumption of small rotations in Remark 3 is not met, the invariance to rotations no more applies, leading to suboptimal sampling.

The iterative sampling process is terminated once the termination criteria in (27) is satisfied and $i \geq K$. Since the entropy reaches its maximum at K steps, the maximum entropy rate $\bar{\mu}_\Delta^*$ is guaranteed to be found at K steps at maximum. The $i \geq K$ condition thus allows redefining (31) as

$$\bar{\mu}_\Delta^*(\mathcal{P}_\nu) = \max_{i=\{1,\ldots,K\}} \left(\bar{\mu}_\Delta \left({}^i\hat{\mathcal{P}} \right) \right), \qquad (34)$$

where $^i\hat{\mathcal{P}}$ is the sampled point-set at iteration i. After terminating the routine at iteration $i \geq K$, the sampled points $\hat{\mathcal{P}} = {}^i\hat{\mathcal{P}}$ equal to all the points sampled up to iteration i.

This entropy-maximizing approach normalizes the spectrum of Δ, and thus introduces a certain level of redundancy defined in Section IV-A (by including points with low Δ). A certain level of redundancy helps in maintaining the original spatial distribution of the points (similar to voxelization), which is beneficial in reducing the effects of noise and outliers. It has been verified experimentally that the entropy maximization of GFH is more resilient than maximizing the cumulative sum of GFH, which tends to under-constrain the problem and is sensitive to noise and outliers.

Algorithm 1: Information-Maximizing Point Selection.

1: **Input:**
2: $\quad \mathcal{P} = \{\mathbf{p}\}, \mathbf{p} \in \mathbb{R}^3$ $\qquad\qquad\qquad$ ▷ input point set
3: $\quad \nu \in \mathbb{R}^+$ $\qquad\qquad\qquad\qquad$ ▷ voxel size in meters
4: $\quad K \in \mathbb{Z}^+$ $\qquad\qquad\qquad$ ▷ number of histogram bins
5: $\quad \lambda_{\bar{H}} \in \langle 0, 1 \rangle$ $\qquad$ ▷ entropy-rate termination criteria (Eq. (27))
6: $\quad C, R \in \mathbb{Z}^+$ ▷ number of columns and rows (if $\mathcal{P}$ in matrix form)
7: $\quad \theta_v \in \mathbb{R}^+$ ▷ vertical field of view of the sensor (if $\mathcal{P}$ in matrix form)
8: **Output:**
9: $\quad \hat{\mathcal{P}} \subseteq \mathcal{P}$ $\qquad$ ▷ point subset maximizing GFH entropy, Eq. (27)
10: **Begin:**
11: $\quad \mathcal{P}_\nu = \text{voxelize}(\mathcal{P}, \nu)$
12: $\quad \mathbb{K}_\nu = \text{KDTree}(\mathcal{P}_\nu)$ $\quad$ ▷ construct KD-tree for efficient NN search
13: $\quad \Delta_{\mathcal{P}_\nu} = \text{GFH}(\mathcal{P}_\nu, \mathbb{K}_\nu, C, R, \theta_v)$ $\qquad$ ▷ Eq. (23)–(25)
14: $\quad \bar{\Delta}_{\mathcal{P}_\nu} = \text{normalizeGFH}(\Delta_{\mathcal{P}_\nu})$ $\qquad\qquad$ ▷ Eq. (32)
15: $\quad$ ▶ Construct a histogram of GFH values
16: $\quad \mathcal{H}_\Delta = \text{histogram}(\bar{\Delta}_{\mathcal{P}_\nu}, K)$ $\quad$ ▷ discretize $\bar{\Delta}_{\mathcal{P}_\nu}$ into K fixed-sized bins
17: $\quad \hat{\mathcal{H}}_\Delta = \text{histogram}(\emptyset, K)$ $\quad$ ▷ empty histogram of K fixed-sized bins
18: $\quad$ ▶ Compute maximum entropy rate $\bar{\mu}_\Delta^*$
19: $\quad \bar{\mu}_\Delta^* = 0$
20: $\quad$ **for** each $k \in \langle 1, K \rangle$ **do** $\qquad$ ▷ iterate each bin exactly once
21: $\quad\quad ^k\mathcal{H}_\Delta = \text{sort}(^k\mathcal{H}_\Delta)$ $\qquad$ ▷ sort bin k in desc. order by Eq. (33)
22: $\quad\quad ^k\hat{\mathcal{H}}_\Delta = {}^k\hat{\mathcal{H}}_\Delta \cup \{^k\mathcal{H}_\Delta.\text{pop}()\}$ $\quad$ ▷ move highest-value point between bins k
23: $\quad\quad \bar{\mu}_\Delta^* = \max\{\bar{\mu}_\Delta^*, \bar{\mu}_\Delta(\hat{\mathcal{H}}_\Delta)\}$ $\qquad\qquad$ ▷ Eq. (34)
24: $\quad$ ▶ Entropy-maximizing selection
25: $\quad k = K$ $\qquad\qquad\qquad$ ▷ current bin-lookup index
26: $\quad$ **while** $|\mathcal{H}_\Delta| > 0$ **and** $\bar{\mu}_\Delta(\hat{H}_\Delta)/\bar{\mu}_\Delta^* > \lambda_{\bar{H}}$ **do** $\quad$ ▷ terminating via Eq. (27) and (30)
27: $\quad\quad ^k\hat{\mathcal{H}}_\Delta = {}^k\hat{\mathcal{H}}_\Delta \cup \{^k\mathcal{H}_\Delta.\text{pop}()\}$ $\quad$ ▷ move first point in bin k
28: $\quad\quad k = k - 1$ **if** $k > 1$ **else** K $\quad$ ▷ cyclic right-left iteration
29: $\quad \hat{\mathcal{P}} = \bigcup_{k \in \langle 1, K \rangle} {}^k\hat{\mathcal{H}}_\Delta$ $\qquad$ ▷ extract all selected points

---

V. EXPERIMENTAL ANALYSES

Let us compare the proposed approach with three state-of-the-art point cloud sampling methods:

- **V•**: uniform sampling[1] with voxel size $\nu = \bullet$ cm,
- **NS•**: normal-space voxelization [11] with angular resolution $\pi = \bullet \,^\circ$ in both azimuth and elevation, and
- **CovS•**: covariance-based sampling[2] [12] with sampled-to-all point ratio of $\rho = \bullet$ %.

All the state-of-the-art methods and the proposed approach were integrated into two state-of-the-art odometry (no loop closures) pipelines: KISS-ICP [7] and LOAM [8]. KISS-ICP is a state-of-the-art implementation of the ICP algorithm, a typical case of a *dense* method utilizing the *point-to-point* metric. LOAM is a *feature*-based odometry method extracting plane and line features. LOAM represents a basis for the majority of the *feature*-based state-of-the-art methods. Since the proposed sampling method is algorithm-independent, it has the potential for positively improving all the other LiDAR-based odometry and SLAM methods building upon ICP and LOAM algorithms. To remain close to the core principles and to reduce the effects of any additional concepts, these two representative odometry pipelines have been chosen for their minimalism on purpose.

To ensure a fair comparison, the best parametrizations balancing convergence and real-time processing were fine-tuned manually for all methods, both odometry pipelines, and all datasets. These parametrizations are given in Table IV. All the

---

[1] Open-source implementation taken from KISS-ICP [7].
[2] Open-source implementation taken from PointMatcher [26].

TABLE I
TABLE OF USED DATASETS

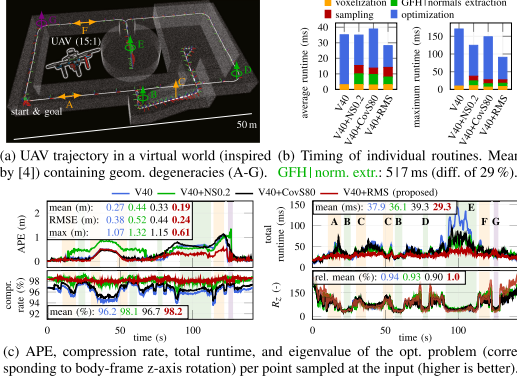| ID	Dataset	Work	Platform	Real world	Point count
D1	X-ICP	[4]	Drone	✗	64×1024 @ 10 Hz
D2	Dronument	[6][3]	Drone	✓	16×1024 @ 10 Hz
D3	DARPA SubT	[27][3]	Drone	✓	64×1024 @ 10 Hz
D4	KITTI	[5]	Car	✓	16×1024 @ 10 Hz
D5	Hilti-Oxford	[28]	Handheld	✓	32×2000 @ 10 Hz

Fig. 5. Output of KISS-ICP [7] 6-DoF odometry when preceded by different point cloud sampling methods. Parametrization: best performing for each method, robot: multirotor UAV, sensor range: 30 m, sensor noise: none. (a) Shows areas of translational (A, C, F) and body-frame z-axis rotational (B, D, E, G) degeneracy. At D and G, the degeneracy arises (see low values of the R_z eigenvalue) from large UAV tilt, which orients the LiDAR such that its data are degenerate around the z-axis. At G, a "loop closing" emerges naturally (see APE). (b) Due to the high compression rate and by balancing the translational space, RMS samples points such that they yield the fastest optimization convergence. (c) RMS yields the lowest drift, removes the largest amount of points, produces stable and lowest runtime, and preserves the highest information rate for optimization (only R_z shown).
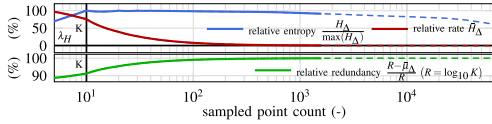

Fig. 6. Rel. entropy, rel. entropy rate, and rel. redundancy of GFH at time 60 s of experiment in Fig. 5. **V40+RMS** sampled about 2% of points out of 65 k total. Dashed lines represent the non-sampled points.

TABLE II
INFLUENCE OF THE MAXIMUM RELATIVE ENTROPY RATE $\lambda_{\bar{H}}$ ON
PERFORMANCE OF THE PROPOSED METHOD IN EXPERIMENT
PRESENTED IN FIG. 5

$\lambda_{\bar{H}}$ (%)	0.1	0.2	0.3	0.4	0.5	0.7	1.0
RMSE (m)	0.31	**0.22**	0.25	0.24	0.27	0.29	0.43
avg. time (ms)	42	37	32	29	27	24	23
compr. rate (%)	95.3	97.0	97.8	98.2	98.5	98.9	99.2

experiments were performed on AMD Ryzen 7 PRO 4750 U (comparable performance verified on Intel Core i7-10710 U).

A. Datasets

The datasets used in evaluation are summarized in Table I. Their selection includes custom data from UAVs (D1-D3) covering full six-DoF movements in different degraded contexts, and KITTI (D4) and Hilti-Oxford (D5) as two of the most prevalent datasets used in evaluating LiDAR-based methods in the related literature. Only 3D LiDAR data are used.

TABLE III
QUANTITATIVE PERFORMANCE OF KISS-ICP AND LOAM POSE ESTIMATION
PIPELINES WHEN PRECEDED BY FOUR DIFFERENT 3D LiDAR POINT CLOUD
SAMPLING TECHNIQUES: UNIFORM SAMPLING **V•**, NORMAL-SPACE SAMPLING
V•+NS• [11], COVARIANCE SAMPLING **V•+CovS•** [12], AND
REDUNDANCY-MINIMIZING SAMPLING **V•+RMS** (PROPOSED)

Dataset	Metric	KISS-ICP				LOAM			
		V•	**V40+ NS•**	**V40+ CovS•**	**V40+ RMS**	**V•**	**V•+ NS•**	**V•+ CovS•**	**V•+ RMS**
D2: Stará Voda #00	$\delta|_{rmse}$					0.85	0.31	0.19	**0.12**
	$\delta|_{mean}$					0.39	0.18	0.15	**0.09**
	$\delta|_{max}$					2.56	0.91	0.45	**0.39**
	$^\Delta\delta_{rmse}$		failed in estimating			0.06	0.02	**0.01**	0.02
	$^\Delta\delta_{mean}$		vertical motion			**0.01**	**0.01**	**0.01**	**0.01**
	$^\Delta\delta_{max}$					1.57	0.39	**0.12**	0.18
	τ_{mean}					69.0	61.4	37.9	**31.0**
	τ_{max}					109.6	138.8	74.3	**69.7**
	χ_{mean}					74.8	82.0	90.9	**95.7**
D3: urban corridor	$\delta|_{rmse}$	0.59	0.63	0.67	**0.42**	1.20	2.01	**0.75**	1.20
	$\delta|_{mean}$	0.50	0.52	0.63	**0.38**	1.14	1.78	**0.69**	1.16
	$\delta|_{max}$	1.24	1.34	1.11	**0.86**	1.84	4.00	**1.41**	1.78
	$^\Delta\delta_{rmse}$	0.04	0.04	0.04	0.04	0.03	0.04	0.03	0.03
	$^\Delta\delta_{mean}$	0.03	0.03	0.04	0.03	0.02	0.03	0.03	0.02
	$^\Delta\delta_{max}$	0.23	0.29	0.24	**0.23**	0.12	0.18	0.17	**0.12**
	τ_{mean}	**13.5**	18.1	24.1	16.7	13.0	15.1	**11.0**	13.0
	τ_{max}	61.3	56.8	77.9	**55.7**	36.3	32.7	24.9	28.6
	χ_{mean}	95.8	**96.2**	93.4	95.9	94.8	99.0	**99.1**	98.5
D4: KITTI #00	$\delta|_{rmse}$	248.60	14.31	56.01	**8.35**	20.72	25.11	**12.28**	12.76
	$\delta|_{mean}$	205.97	11.27	45.29	**7.48**	16.34	21.05	**9.50**	9.78
	$\delta|_{max}$	458.96	32.59	116.34	**16.20**	49.87	51.82	29.96	**29.45**
	$^\Delta\delta_{rmse}$	1.41	**1.27**	**1.27**	**1.27**	**1.27**	**1.27**	**1.27**	**1.27**
	$^\Delta\delta_{mean}$	1.26	**1.17**	**1.17**	**1.17**	**1.17**	**1.17**	**1.17**	**1.17**
	$^\Delta\delta_{max}$	14.86	14.71	**14.64**	14.67	14.79	14.72	14.70	**14.69**
	τ_{mean}	43.3	57.1	**34.2**	35.7	83.0	76.3	74.9	**66.9**
	τ_{max}	769.1	270.8	88.4	**87.7**	180.8	196.9	168.1	**159.6**
	χ_{mean}	96.6	97.2	99.2	**99.3**	89.4	99.2	99.1	**99.4**
D4: KITTI #09	$\delta|_{rmse}$	481.02	17.48	25.66	**15.76**	16.08	21.15	12.64	**10.77**
	$\delta|_{mean}$	418.98	13.89	19.31	**13.12**	11.94	15.72	9.18	**8.07**
	$\delta|_{max}$	772.04	46.24	68.67	**32.61**	38.86	50.01	31.49	**25.59**
	$^\Delta\delta_{rmse}$	2.19	1.65	1.75	**1.58**	1.58	1.58	1.58	**1.58**
	$^\Delta\delta_{mean}$	1.71	1.57	1.62	**1.52**	1.52	1.52	1.52	**1.52**
	$^\Delta\delta_{max}$	25.22	4.28	7.72	**3.58**	3.60	3.60	3.62	3.65
	τ_{mean}	71.3	67.0	59.8	**42.8**	70.2	59.1	59.33	**50.8**
	τ_{max}	1748.3	811.3	700.5	**118.0**	143.4	119.2	127.3	**92.7**
	χ_{mean}	95.9	96.4	97.1	**98.8**	87.1	99.0	98.5	**99.4**
D5: Hilti-Oxford #04	$\delta|_{rmse}$					0.34	0.63	0.29	**0.23**
	$\delta|_{mean}$					0.30	0.59	0.26	**0.21**
	$\delta|_{max}$					0.74	1.08	0.67	**0.58**
	$^\Delta\delta_{rmse}$		failed in estimating			**0.14**	**0.14**	**0.14**	**0.14**
	$^\Delta\delta_{mean}$		quick rotational motions			**0.10**	**0.10**	**0.10**	**0.10**
	$^\Delta\delta_{max}$					0.56	0.61	0.54	**0.51**
	τ_{mean}					67.9	55.7	54.1	**42.8**
	τ_{max}					138.5	133.9	120.0	**77.1**
	χ_{mean}					90.5	95.4	95.3	**97.7**
D5: Hilti-Oxford #14	$\delta|_{rmse}$					2.22	2.85	0.84	**0.76**
	$\delta|_{mean}$					1.86	2.35	**0.62**	0.62
	$\delta|_{max}$					3.69	4.82	2.06	**1.64**
	$^\Delta\delta_{rmse}$		failed in estimating			**0.16**	**0.16**	**0.16**	**0.16**
	$^\Delta\delta_{mean}$		quick rotational motions			**0.09**	**0.09**	**0.09**	**0.09**
	$^\Delta\delta_{max}$					1.91	1.91	1.91	1.91
	τ_{mean}					20.7	18.2	17.3	**17.2**
	τ_{max}					45.1	47.6	40.4	**30.3**
	χ_{mean}					97.9	98.7	98.7	**98.8**

Metrics: APE $|\delta|$ (m), RPE $^\Delta\delta$(m), total runtime τ(ms), and compression rate χ (%). The best results are in bold. Trajectories of experiments D2 and D4: KITTI #00 are shown in Fig. 1.

B. Parametrization of RMS

It has been validated empirically that out of the three sensor-agnostic parameters in Algorithm 1, K and ν have limited effect on the performance. Thus, $K = 10$ remains fixed in all the presented experiments and ν is selected such that uniform sampling **V•** is the most accurate and computes in real time in the given dataset. Table II presents an ablation study on the maximum relative entropy rate $\lambda_{\bar{H}}$. The table demonstrates that the algorithm is stable once $\lambda_{\bar{H}}$ lies in a reasonable interval, here 0.2–0.7%. Based on Table II, we use $\lambda_{\bar{H}} = 0.4\%$ in all our KISS-ICP [7] experiments as a balance between runtime, accuracy, and stability. Since the stable interval is pipeline-dependent, similar grid-search has been done to find optimal $\lambda_{\bar{H}}$ for the *feature*-based LOAM [8] estimation pipeline used in Section V-D. In LOAM, one instance of Algorithm 1 runs independently for each of the feature types, with $\lambda_{\bar{H}}$ being fixed to $\lambda_{\bar{H}} = 0.8\%$ for plane and $\lambda_{\bar{H}} = 15\%$ for line features.

TABLE IV
TABLE OF PARAMETERS USED IN THE EXPERIMENTS IN TABLE III

Pipeline Sampling	KISS-ICP 3D points				LOAM plane features				line features			
Parameter Units	v (m)	π (°)	ρ (%)	λ_{fl} (%)	v (m)	π (°)	ρ (%)	λ_{fl} (%)	v (m)	π (°)	ρ (%)	λ_{fl} (%)
D2	✗	✗	✗	✗	0.4	0.2	40		0.2	0.2	20	
D3	0.4	0.2	50		0.8	0.2	60		0.4	0.2	30	
D4: #00	1.0*	1.0	10	0.4	0.4	0.1	80	0.8	0.2	0.1	20	15
D4: #09	1.0*	1.0	30		1.0	0.1	50		0.5	0.1	2.5	
D5: #04	✗	✗	✗	✗	0.4	1.0	50		0.2	1.0	50	
D5: #14	✗	✗	✗	✗	0.4	1.0	80		0.2	1.0	2.5	

The values marked with* are for the V• method only (the V40 parametrizations use $v = 40$cm). Among the compared methods, only the proposed method's parametrization is fixed (λ_{fl}), showing the method's unique adaptability to different sensors and environments.

C. Convergence Analysis

Fig. 5 demonstrates an experiment designed to compare performance of the four sampling techniques (all fine-tuned to the environment). In the experiment, a UAV performs a loop inside a challenging simulation world (D1) designed to contain various geometrical degeneracies (translational along narrow corridors and rotational within a circular room). The experiment shows that the proposed method outperforms the baseline methods in terms of speed, accuracy, and robustness (even to geometrical degeneracies), all while sampling the least amount of points. The data show superior timing and compression consistency of the proposed method, with both reaching almost constant values with a limited number of outliers. This is a particularly important attribute for deployment of small and agile resource-constrained robots with real-time constraints, such as UAVs. Fig. 5(c) shows that information rate (measured as eigenvalue per point sampled on input) extracted in the optimization from the problem Hessian is highest in RMS (only z-axis rotational eigenvalue R_z is shown). Additionally, Fig. 6 showcases a single-frame sampling of Algorithm 1.

D. Quantitative Analysis

Table III presents a quantitative analysis comparing the effects of the sampling methods on the two odometry pipelines. Together with Table IV, the two tables show that the fixed parametrization adapts well to various different sensors, environments, and conditions. This is a significant practical advantage, which improves the method's applicability by reducing the need for tuning the proposed method to every domain. The data show superior performance of the proposed method in
- improving performance in well-conditioned settings,
- reducing odometry drift in degenerated conditions,
- sampling the least amount of points in general, and
- computing the fastest while being the most accurate.

The reported timings are for the entire pipeline, including point or feature sampling, and the optimization. The accuracy gains are associated with lower comp. time enabling use of all data in real-time as well as with high noise and outlier removal.

REFERENCES

[1] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2016, pp. 809–816.

[2] J. Jiao et al., "Greedy-based feature selection for efficient LiDAR SLAM," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 5222–5228.

[3] W. Li, Y. Hu, Y. Han, and X. Li, "KFS-LIO: Key-feature selection for lightweight LiDAR inertial odometry," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 5042–5048.

[4] T. Tuna, J. Nubert, Y. Nava, S. Khattak, and M. Hutter, "X-ICP: Localizability-aware LiDAR registration for robust localization in extreme environments," *IEEE Trans. Robot.*, vol. 40, pp. 452–471, 2024.

[5] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.

[6] P. Petracek et al., "New era in cultural heritage preservation: Cooperative aerial autonomy for fast digitalization of difficult-to-access interiors of historical monuments," *IEEE Robot. Automat. Mag.*, early access, Feb. 2023, doi: 10.1109/MRA.2023.3244423.

[7] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, "KISS-ICP: In defense of point-to-point ICP – simple, accurate, and robust registration if done the right way," *IEEE Robot. Automat. Lett.*, vol. 8, no. 2, pp. 1029–1036, Feb. 2023.

[8] J. Zhang et al., "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot.: Sci. Syst.*, 2014, pp. 1–9.

[9] C. R. Qi et al., "PointNet: Deep hierarchical feature learning on point sets in a metric space," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5105–5114.

[10] R. B. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2011, pp. 1–4.

[11] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. 3rd Int. Conf. 3-D Digit. Imag. Model.*, 2001, pp. 145–152.

[12] N. Gelfand, L. Ikemoto, S. Rusinkiewicz, and M. Levoy, "Geometrically stable sampling for the ICP algorithm," in *Proc. 4th Int. Conf. 3-D Digit. Imag. Model.*, 2003, pp. 260–267.

[13] I. Lang, A. Manor, and S. Avidan, "SampleNet: Differentiable point cloud sampling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 7575–7585.

[14] E. Nezhadarya, E. Taghavi, R. Razani, B. Liu, and J. Luo, "Adaptive hierarchical down-sampling for point cloud classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12953–12961.

[15] Y. Wang et al., "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 146:1–146:12, 2019.

[16] Y. Yang, C. Feng, Y. Shen, and D. Tian, "FoldingNet: Point cloud auto-encoder via deep grid deformation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 206–215.

[17] Y. Shen, C. Feng, Y. Yang, and D. Tian, "Mining point cloud local structures by kernel correlation and graph pooling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4548–4557.

[18] T. Ozaslan et al., "Autonomous navigation and mapping for inspection of penstocks and tunnels with MAVs," *IEEE Robot. Automat. Lett.*, vol. 2, no. 3, pp. 1740–1747, Jul. 2017.

[19] M. Tranzatto et al., "CERBERUS in the DARPA subterranean challenge," *Sci. Robot.*, vol. 7, no. 66, 2022, Art. no. eabp9742.

[20] R. Schnabel et al., "Octree-based point-cloud compression," in *Proc. IEEE VGTC Conf. Point-Based Graph.*, 2006, pp. 111–121.

[21] Y. Duan, J. Peng, Y. Zhang, J. Ji, and Y. Zhang, "PFilter: Building persistent maps through feature filtering for fast and accurate LiDAR-based SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 11087–11093.

[22] Z. Zhou, M. Yang, C. Wang, and B. Wang, "ROI-cloud: A key region extraction method for LiDAR odometry and localization," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 3312–3318.

[23] H. Lim et al., "A single correspondence is enough: Robust global registration to avoid degeneracy in urban environments," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2022, pp. 8010–8017.

[24] F. Li et al., "Reducing redundancy in maps without lowering accuracy: A geometric feature fusion approach for simultaneous localization and mapping," *Int. J. Geo-Inf.*, vol. 12, no. 6, 2023, Art. no. 235.

[25] H. Pottmann et al., "Geometry of the squared distance function to curves and surfaces," in *Proc. Visualization Math. III*, 2003, pp. 221–242.

[26] F. c. Pomerleau et al., "Comparing ICP variants on real-world data sets," *Auton. Robots*, vol. 34, no. 3, pp. 133–148, 2013.

[27] M. Petrlik et al., "UAVs beneath the surface: Cooperative autonomy for subterranean search and rescue in DARPA SubT," *Field Robot.*, vol. 3, no. 1, pp. 1–68, 2023.

[28] L. Zhang et al., "Hilti-Oxford dataset: A millimeter-accurate benchmark for simultaneous localization and mapping," *IEEE Robot. Automat. Lett.*, vol. 8, no. 1, pp. 408–415, Jan. 2023.