On the Redundancy Detection in Keyframe-based SLAM

Author(s):
Schmuck, Patrik; Chli, Margarita [iD]

On the Redundancy Detection in Keyframe-based SLAM

Patrik Schmuck and Margarita Chli
Vision for Robotics Lab, ETH Zurich
8092 Zurich, Switzerland
`https://v4rl.ethz.ch/`

Abstract

Egomotion and scene estimation is a key component in automating robot navigation, as well as in virtual reality applications for mobile phones or head-mounted displays. It is well known, however, that with long exploratory trajectories and multi-session mapping for long-term autonomy or collaborative applications, the maintenance of the ever-increasing size of these maps quickly becomes a bottleneck. With the explosion of data resulting in increasing runtime of the optimization algorithms ensuring the accuracy of the Simultaneous Localization And Mapping (SLAM) estimates, the large quantity of collected experiences is imposing hard limits on the scalability of such techniques. Considering the keyframe-based paradigm of SLAM techniques, this paper investigates the redundancy inherent in SLAM maps, by quantifying the information of different experiences of the scene as encoded in keyframes. Here we propose and evaluate different information-theoretic and heuristic metrics to remove dispensable scene measurements with minimal impact on the accuracy of the SLAM estimates. Evaluating the proposed metrics in two state-of-the-art centralized collaborative SLAM systems, we provide our key insights into how to identify redundancy in keyframe-based SLAM.

1. Introduction

Building on top of state-of-the-art Simultaneous Localization And Mapping (SLAM) systems with considerable robustness and accuracy in the centimeter range [23] for single-robot scenarios, multi-agent system have been gaining growing popularity in various applications, ranging from inspection tasks to search-and-rescue missions. By sharing information amongst the participants or dividing up a task between multiple robots, robotic teams can increase the robustness, efficiency and accuracy of a robotic mission [16], and enable tasks impossible for a single robot. State-of-the-art single-agent [22, 23] and centralized multi-agent

[16, 25, 28] systems generate a global graph (the *SLAM graph* or *map*), estimating the trajectory of the participating robots as well as the 3D structure of the environment. With these SLAM systems being able to cover large areas during robotic missions, and the SLAM graph growing constantly during the mission, the graph quickly reaches a size, where operations on it, such as graph optimization, become computationally expensive, therefore affecting the performance of the SLAM system.

The front-end of keyframe-based SLAM runs *Visual Odometry* (VO) [22, 27, 28], using only camera feeds, or *Visual-Inertial Odometry* (VIO) [23, 16, 25], using camera images and measurements from an Inertial Measurement Unit (IMU), with the goal of estimating the local trajectory and 3D environment (the *local map*) of the robot. Keyframe-based systems choose a subset of the most representative frames from all incoming frames to be stored in the local map, the keyframes (KFs), and are now well-established in the SLAM literature [19, 22, 25], gaining substantial ground over filtering-based techniques [31]. For robust and accurate estimation of this local map, especially for fast motions and poorly textured areas, the SLAM front-end typically employs a generous KF creation policy, with the drawback of having potentially high redundancy encoded in consecutive KFs.

While the front-end considers only a local window incorporating the most recent KFs, the SLAM back-end stores and maintains all KFs created during the mission in a global SLAM graph (or global map). It introduces new links in the SLAM graph when returning to previously visited places of the environment (*loop closure*) and applies global optimization techniques, namely *Global Bundle Adjustment* (GBA), to the graph to increase the accuracy of the estimate. With cubic complexity of GBA in the size of the graph, this optimization quickly gets computationally expensive with growing graph size. Therefore, it is desirable to reduce the graph to a minimum size of most informative nodes that accurately represents the robots' trajectories and the structure of the surroundings.

While manifesting itself also in single-agent SLAM, this

---

problem of exploding SLAM maps is especially evident in collaborative systems, where multiple agents contribute data simultaneously to the SLAM estimate. As shown in [16] and [28], this issue is currently the main bottleneck for the scalability of centralized collaborative SLAM systems. Likewise, long-term autonomy [6] depends on efficient data management to ensure applicability when continuously feeding experiences of the environment to the system. To relieve the problem of exploding data and computation time, it is necessary to sparsify the SLAM graph by detecting redundancy in the graph and determining the informative value of every node with respect to all others in the graph, and subsequently remove the most redundant nodes.

In this spirit, this article investigates the question of how the redundancy of KFs in a SLAM graph can be assessed, to sparsify the graph with minimum decrease in accuracy of the estimate, with the goal of boosting the efficiency and scalability of SLAM maps. To this end, we investigate information-theoretic approaches to classify the redundancy of KFs in the SLAM graph, and evaluate their performance under different scenarios. While mutual information is shown to suggest effective graph compression with minimal loss in accuracy, it is computationally too expensive in realistic applications. Therefore, we propose an efficient heuristic that exploits the structure of the SLAM graph to find redundant KFs, being able to run online in parallel to the SLAM estimation process. We evaluate the approaches in two collaborative SLAM systems of different modalities, namely the visual-inertial CVI-SLAM [16] and the monocular CCM-SLAM [28], attesting to its practical applicability while compressing the SLAM graph up to more than 50% with only small reductions in accuracy.

2. Related Work

Information-based measures have been widely used for various tasks throughout the computer vision and robotics communities, all the way from showing how mutual information can guide the feature matching process in visual SLAM in [4], to using positional covariance to construct skeletal graphs for large Structure-from-Motion problems, in order to speed up the expensive computation process, in [30]. Schneider *et al.* [29] evaluate trajectory segments regarding their entropy with respect to the calibration parameters of the sensor suite in order to estimate this calibration, while Mu *et al.* [21] use entropy to select the most important landmarks for collision avoidance. In [11], information is quantified in order to to calculate the optimal path of a UAV for 3D reconstruction of a scene of interest.

While all aforementioned systems employ vision-based sensors, information-based measures are also used for LIDAR[1]-based systems in the literature. Kretzschmar and

---
[1]*LI*ght *D*etection *A*nd *R*anging

Stachniss [18] formulate measures for information gain and mutual information in a LIDAR SLAM system in connection with occupancy grid cells. Using these measures, the authors identify measurements providing a small amount of information and discard them. In [5], the authors aim at finding a set of reduced landmarks and poses by minimizing an objective function that takes into account memory requirements and estimation accuracy. Carlone *et al.* [3] reduce the pose graph by looking for the maximal subset of measurements that are internally "coherent" and observable. While these approaches show promising results in terms of sparsification of pose graphs, the approaches cannot directly be applied to visual SLAM, since LIDAR and visual SLAM are conceptually different, most notably regarding the fact that scene depth cannot be measured directly in visual SLAM, but has to be recovered and estimated during the SLAM process.

Many recent works proposed efficient strategies to deal with the densification of the Hessian matrix of the pose graph, resulting for example from node marginalization. Huang *et al.* [13] aim to consistently approximate the original dense graph by a sparse one using l_1-regularization. In [2], the authors introduce *generic linear constraint* factors, which replace the dense information matrix of the markov blanket of the marginalized node by a sparse approximation, using Chow-Liu trees. Mazuran *et al.* [20] go one step further, proposing *nonlinear factor recovery* that allows to replace dense factors by arbitrarily defined "virtual" nonlinear measurements. Vallvé *et al.* [33] propose a factor descent algorithm for belief sparsification. The recent work of [12] does not aim for a sparsification of the global pose graph, but for an efficient sparse approximation of the dense prior resulting from marginalization of the past measurement in sliding-window SLAM approaches. While all these works provide efficient strategies for sparsification of pose graphs under specific condition, they focus on how to remove nodes from the graph, while this work aims to answer the question of which data in the SLAM graph is most redundant and therefore most dispensable for the sake of bounded computation.

When it comes to redundancy detection for SLAM, a straight-forward heuristic to limit redundancy is to enforce a spatial distribution of the graph nodes, as done in [17] and [15]. This decouples the growth of the SLAM graph and exploration time, however, Euclidean distance heuristics do not account for the fact that the structure of the explored environment might not be homogeneous and therefore, some parts of it may need more measurements for a robust estimate than others. In [9], the authors propose a heuristic that removes nodes, for which the number of edges exceeds a pre-defined threshold. While this ensures sparsity of the pose graph, it does also not account for the structure of the environment. Ila *et al.* [14] move the dis-

tance criterion to the information space and apply it to filter-based SLAM, adding only measurements to the estimate that are distant to already existing ones in terms of mutual information. Also aiming for filter-based systems instead of graph-based SLAM, Vial *et al.* [34] formulate the sparsification as an optimization problem, minimizing the *Kullback-Leibler-Divergence* between the sparse and true estimate. Paull *et al.* [24] treat node removal and sparsifiyng the marginalized graph as an entirety in consideration of resource constraints, such as memory requirement and communication bandwidth. While these constraints are more limiting in a distributed multi-robot setup, in a centralized system that accumulates all experiences of all participating agents at one central instance, memory usage is not a limiting factor, and communication bandwidth can be held constant for each agent, independent of the number of participant [28]. As discussed in [28], a major bottleneck of non-distributed (collaborative) SLAM systems based on GBA is its increasing computational time in the number of nodes in the graph. However, most recent systems in the literature, such as [10, 26, 8] and also [25] which allows estimating joint trajectories using a multi-session functionality, do not include redundancy detection. ORB-SLAM [22] proposes a simple algorithm for redundancy detection, based on common landmark observations. This scheme is adopted by recent visual [28] and visual-inertial [16] collaborate SLAM systems. As shown in [22], this heuristic bounds the growth of the SLAM with respect to the size of exploration space. However, as this is a heuristic, as demonstrated in our analysis, it does not reach the accuracy of an information-theoretic approach.

This work proposes a new information-theoretic measure for redundancy based on mutual information, formalizing the amount of information that a node in the graph adds to the SLAM graph. The benefit of this new evaluation metric compared to "direct" mutual information is quantified in the experimental evaluation. Furthermore, a real-time capable, efficient, landmark-based heuristic for redundancy detection is proposed and experimentally evaluated. The metrics for redundancy detection are implemented and tested in a state of the art visual-inertial [28] as well as a monocular [16] centralized collaborative SLAM system, confirming their applicability to state-of-the-art SLAM.

3. Methodology

3.1. Keyframe-based SLAM

The algorithms presented in this article are designed for application in keyframe-based SLAM systems. From all frames captured by the camera, these systems only keep a subset of the most representative frames, the *keyframes* (KFs). Other frames are only used to estimate the current pose of the camera, but are dropped subsequently. From the camera images, 2D feature keypoints, encoding salient parts
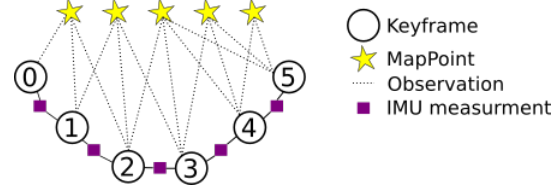

Figure 1: Example SLAM graph derived with keyframes (KFs), Map Points (MPs) and edges induced by observations.

of the image, are extracted and used to triangulate scene landmarks. These scene landmarks are stored as 3D map points (MPs) in a *map*, together with the KFs. If a feature keypoint of KF_i is associated to MP_j, KF_i *observes* MP_j. The set of all KFs $\mathcal{V}$ and MPs $\mathcal{L}$ form together the *SLAM graph G*, where $\mathcal{V}$ and $\mathcal{L}$ are the set of nodes, and any edges are induced by the observations, as illustrated in Fig. 1. In a visual-inertial system, such as [16] and [25], the inertial measurements induce additional edges between the KFs.

Shared observations of MPs (i.e. MP_a is observed by KF_i and KF_j) induce so-called covisibility-edges between the KFs. If KF_i and KF_j have at least one shared observation, they are connected by a *covisibility edge c_{ij}*. The weight $\omega(c_{ij})$ of the edge is the number of shared observations between KF_i and KF_j. This notion of covisibility induces a *covisibility graph $C = \{\mathcal{V}, \mathcal{C}\}$*, where $\mathcal{C}$ is the set of weighted covisibility edges, as illustrated in Fig. 2. We refer to the KFs connected to KF_i via a direct edge in the covisibility graph as the *neighbors* of KF_i. In practice, we include only edges with $\omega > 10$ in the graph, to limit the density of the graph and ensure a minimum overlap of the scene observed by two KFs. The *Markov Blanket* of KF_i, denoted as $MB(KF_i)$, is the set of all nodes that are connected to KF_i by a covisibility edge (i.e. all neighbors of KF_i).
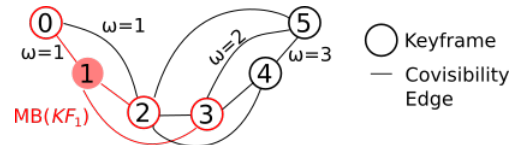

Figure 2: Example covisibility graph with some edge weights displayed, derived from the SLAM graph in Fig. 1. The Markov Blanket of KF_1 ($MB(KF_1)$) is indicated in red.

To increase the accuracy of the map, estimated by the SLAM graph, GBA is performed on it, minimizing the global reprojection error of the MP observations (and the IMU factors, in a visual-inertial system). In addition to minimizing the error residuals of the SLAM graph, GBA estimates the covariance matrix of the SLAM graph as the inverse of the Hessian matrix of the non-linear least-squares optimization problem.

We evaluate our metrics described in Sec. 3.2 and Sec. 3.3 in a centralized collaborative SLAM system, where the

problem of decreasing performance with increasing data in the system is most evident, because the number of KFs grows faster in those systems with multiple agents contributing KFs simultaneously. However, the metrics presented in this article are not limited to these systems, but are generally applicable to any keyframe-based SLAM system. For more details on the SLAM- and covisibility-graph, we kindly refer the reader to [22]. The collaborative visual-inertial SLAM system used for our tests, including the GBA scheme, is described in detail in [16].

3.2. Redundancy Detection using Mutual Information

From GBA, we retrieve the covariance matrix Σ for all KFs $i \in \mathcal{V}$, consisting the marginal covariance matrices Σ_{ii} for each KF_i as well as the correlation matrices for Σ_{ij} for KFs $i, j \in \mathcal{V}$, as shown in Eq. (1).

$$
\Sigma = \begin{bmatrix} \Sigma_{00} & \Sigma_{01} & \dots & \Sigma_{0n} \\ \Sigma_{10} & \Sigma_{11} & \dots & \Sigma_{1n} \\ \vdots & \ddots & \ddots & \vdots \\ \Sigma_{n0} & \Sigma_{n1} & \dots & \Sigma_{nn} \end{bmatrix} \quad (1)
$$

For 6 Degree-of-Freedom (DoF) poses, the dimension of the covariance matrix for a KF is 6×6. Using Σ, we can calculate the *Mutual Information* (MI) between two KFs [7] as:

$$
\text{MI}(KF_i, KF_j) = \frac{1}{2} \log_2 \left(\frac{\det(\Sigma_{ii})}{\det(\Sigma_{ii} - \Sigma_{ij}\Sigma_{jj}^{-1}\Sigma_{ji})} \right), \quad (2)
$$

where $\det(\cdot)$ denotes the matrix determinant. Intuitively, the MI between two KFs is a measure of how much information we get about the pose of one KF by knowing the pose of the other KF. With Eq. (2) encoding how much information one KF encodes about another, we derive a measure $\psi(\cdot)$ of how much information KF_i adds to the estimated SLAM map by summing up the MI between i and its neighbors in the covisibility graph (i.e. the Markov Blanket $\text{MB}(KF_i)$):

$$
\psi(KF_i) = \frac{1}{|\text{MB}(KF_i)|} \sum_{j \in \text{MB}(KF_i)} \text{MI}(KF_i, KF_j) \quad (3)
$$

where $|\text{MB}(KF_i)|$ is the cardinality of the Markov Blanket. Applying $\psi(\cdot)$ to all KFs $i \in \mathcal{V}$ assigns a value to each KF, quantifying its "informativeness". A high value for $\psi(KF_i)$ means that KF_i adds distinctive information to its neighbors, and therefore, measures the redundancy of KF_i. However, directly measuring pairwise MI between individual KFs does not take into account that information might be also be encoded by other KFs as illustrated in Fig. 3; although the shared information between KF_i and its neighbors (KF_{n_1} and KF_{n_2}) is high, the information is also contained in the other KFs (KF_{k_1} and KF_{k_2}) in the graph.
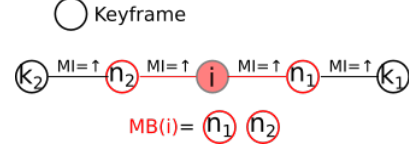

Figure 3: Example covisibility graph, where KF_i shares information with its neighbors KF_{n_1} and KF_{n_2}, however, the same information is also provided by other KFs in the graph (KF_{k_1} and KF_{k_2}).
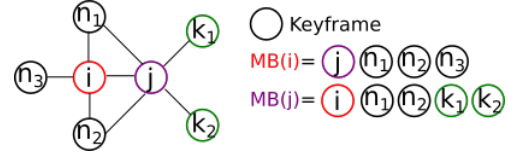

Figure 4: Example covisibility graph to illustrate the proposed method to calculate the unique amount of information that KF_i adds to KF_j.

Therefore, in the presence of KF_{k_1} and KF_{k_2}, KF_i does not add much additional information to the estimated SLAM map and should be considered redundant. To this end, we modify $\psi(\cdot)$ to an extended measure $\widehat{\psi}(\cdot)$, taking into account the information of neighbors using conditional covariance matrices $\hat{\Sigma}$ for the KFs to calculate MI, as detailed below. We illustrate the approach using the graph given in Fig. 4, where we want to quantify the information that KF_i adds to KF_j. To obtain conditional covariances, we first construct a local covariance matrix $\Sigma_{\text{MB}(j)}$ from the full covariance matrix Σ from Eq. (1) for the Markov Blanket of KF_j:

$$
\Sigma_{\text{MB}(j)}
$$
$$
= \begin{bmatrix}
\Sigma_{ii} & \Sigma_{ij} & \Sigma_{in_1} & \Sigma_{in_2} & \Sigma_{ik_1} & \Sigma_{ik2} \\
\Sigma_{ji} & \Sigma_{jj} & \Sigma_{jn_1} & \Sigma_{jn_2} & \Sigma_{jk_1} & \Sigma_{jk2} \\
\Sigma_{n_1i} & \Sigma_{n_1j} & \Sigma_{n_1n_1} & \Sigma_{n_1n_2} & \Sigma_{n_1k_1} & \Sigma_{n_1k2} \\
\Sigma_{n_2i} & \Sigma_{n_2j} & \Sigma_{n_2n_1} & \Sigma_{n_2n_2} & \Sigma_{n_2k_1} & \Sigma_{n_2k2} \\
\Sigma_{k_1i} & \Sigma_{k_1j} & \Sigma_{k_1n_1} & \Sigma_{k_1n_2} & \Sigma_{k_1k_1} & \Sigma_{k_1k2} \\
\Sigma_{k_2i} & \Sigma_{k_2j} & \Sigma_{k_2n_1} & \Sigma_{k_2n_2} & \Sigma_{k_2k_1} & \Sigma_{k_2k2}
\end{bmatrix}
$$
$$
\quad (4)
$$

and split $\Sigma_{\text{MB}(j)}$ as follows:

$$
\Sigma_{(i,j)} = \begin{bmatrix} \Sigma_{ii} & \Sigma_{ij} \\ \Sigma_{ji} & \Sigma_{jj} \end{bmatrix} \quad (5)
$$

$$
\Sigma_{(MB)} = \begin{bmatrix}
\Sigma_{n_1n_1} & \Sigma_{n_1n_2} & \Sigma_{n_1k_1} & \Sigma_{n_1k2} \\
\Sigma_{n_2n_1} & \Sigma_{n_2n_2} & \Sigma_{n_2k_1} & \Sigma_{n_2k2} \\
\Sigma_{k_1n_1} & \Sigma_{k_1n_2} & \Sigma_{k_1k_1} & \Sigma_{k_1k2} \\
\Sigma_{k_2n_1} & \Sigma_{k_2n_2} & \Sigma_{k_2k_1} & \Sigma_{k_2k2}
\end{bmatrix} \quad (6)
$$

$$
\Sigma_{(i,MB)} = \begin{bmatrix}
\Sigma_{in_1} & \Sigma_{in_2} & \Sigma_{ik_1} & \Sigma_{ik2} \\
\Sigma_{jn_1} & \Sigma_{jn_2} & \Sigma_{jk_1} & \Sigma_{jk2}
\end{bmatrix} \quad (7)
$$

Using Schur's complement and Eq. (5) - Eq. (7), we can calculate the conditional covariance matrix $\hat{\Sigma}_{i,j|\text{MB}(j)\setminus i}$, containing the covariance matrices of KF_i and KF_j conditioned

on the Markov Blanket of KF_j (i.e. $\mathrm{MB}(j)\backslash i$).

$$
\begin{aligned}
\hat{\boldsymbol{\Sigma}}_{i,j|\mathrm{MB}(j)\backslash i} &= \begin{bmatrix} \hat{\boldsymbol{\Sigma}}_{ii} & \hat{\boldsymbol{\Sigma}}_{ij} \\ \hat{\boldsymbol{\Sigma}}_{ji} & \hat{\boldsymbol{\Sigma}}_{jj} \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{\Sigma}_{ii|\mathrm{MB}(j)\backslash i} & \boldsymbol{\Sigma}_{ij|\mathrm{MB}(j)\backslash i} \\ \boldsymbol{\Sigma}_{ji|\mathrm{MB}(j)\backslash i} & \boldsymbol{\Sigma}_{jj|\mathrm{MB}(j)\backslash i} \end{bmatrix} \\
&= \boldsymbol{\Sigma}_{(i,j)} - \boldsymbol{\Sigma}_{(i,MB)}\boldsymbol{\Sigma}_{(MB)}^{-1}\boldsymbol{\Sigma}_{(i,MB)}^T \quad (8)
\end{aligned}
$$

Since the covariance matrices of KF_i and KF_j are now conditioned on $\mathrm{MB}(j)\backslash i$, using these to calculate the MI quantifies solely the amount of information uniquely shared between KF_i and KF_j:

$$
\widehat{\mathrm{MI}}(KF_i, KF_j) = \frac{1}{2}\log_2\left(\frac{\det(\hat{\boldsymbol{\Sigma}}_{ii})}{\det(\hat{\boldsymbol{\Sigma}}_{ii} - \hat{\boldsymbol{\Sigma}}_{ij}\hat{\boldsymbol{\Sigma}}_{jj}^{-1}\hat{\boldsymbol{\Sigma}}_{ji})}\right) \quad (9)
$$

Calculating the $\widehat{\mathrm{MI}}$ score for all neighbors j of a KF i, we can quantify how much information KF_i adds to the estimate of the SLAM map, as:

$$
\widehat{\psi}(KF_i) = \frac{1}{|\mathrm{MB}(KF_i)|}\sum_{j\in\mathrm{MB}(KF_i)}\widehat{\mathrm{MI}}(KF_i, KF_j) \quad (10)
$$

A high value for $\widehat{\psi}(KF_i)$ means that KF_i adds significant information to its environment, while a small value can be interpreted in a way that all information that KF_i contributes is also contributed by other KFs in the graph, hence, can be considered redundant. Since the calculation of the Schur complement in Eq. (8) requires a costly matrix inversion, we limit this operation to the 20 neighbors with the highest edge weights in the covisibility graph.

3.3. Structure-based heuristics

The measures proposed in Eq. (3) and Eq. (10) quantify redundancy from an information-theoretic viewpoint, and are, therefore, expected to perform better in terms of accuracy, compared to hand-crafted heuristics. However, GBA and retrieval of the covariance matrix of KFs in the SLAM graph are time-consuming steps, especially on large graphs. Therefore, the information-theoretic graph compression is well suited to sparsify the SLAM graph between robotic missions when the robots are not exploring their environment, but not online during the mission. For this reason, we propose an efficient, structure-based heuristic for redundancy detection, that is capable to run online, in parallel to the SLAM estimation process.

A simple redundancy detection scheme using MPs is used in ORB-SLAM [22]. For each KF_i, the number η of associated MPs with more than 3 observations is calculated. If η exceeds a pre-defined threshold (90% of all the MPs observed by KF_i in [22]), it is considered redundant and removed from the map. While this is an efficient strategy, it only classifies KFs as redundant or not, and does not assign a value for redundancy to each KF.

Here, we modify this approach to quantify how redundant the observations of a MP are, with the goal of assigning a value $\phi \in [0, 1]$ to each MP classifying the redundancy of the observations, with higher value indicating higher redundancy. If a MP is observed by many KFs, its 3D position can still be precisely estimated when dropping one observation, while this would affect the accuracy more if there exist only few observations. Let $obs(MP_i)$ denote the number of observations of MP_i. Then, with a minimum of two observations necessary to triangulate the 3D position of one MP, we define the redundancy $\tau(obs(MP_i)) = 0$ for $obs(MP_i) \leq 2$. As five observations can be considered sufficient to robustly estimate the 3D position of an MP, we set $\tau(obs(MP_i)) = 1$ for $obs(MP_i) > 5$. Finally, accounting for the fact that with increasing number of observations the redundancy of one observation increases, we choose $\tau(x)$ as follows:

$$
\tau(x) = \begin{cases} 0, & \text{if } x \leq 2 \\ 0.4, & \text{if } x = 3 \\ 0.7, & \text{if } x = 4 \\ 0.9, & \text{if } x = 5 \\ 1, & \text{if } x > 5 \end{cases} \quad (11)
$$

In continuous space, $\tau(x)$ would be chosen as a sigmoid, exhibiting the desired behavior described above. However, since $obs(MP_i)$ can only take positive integer values, we assign a discrete function for efficiency, since a SLAM graph quickly includes > 1000 MPs for which $\tau(x)$ has to be computed. With $\mathfrak{L}_i \subseteq \mathcal{L}$ denoting the set of MPs observed by KF_i, we can again calculate a redundancy value $\phi(\cdot)$ for each KF using $\tau(\cdot)$:

$$
\phi(KF_i) = \frac{1}{|\mathfrak{L}_i|}\sum_{j\in\mathfrak{L}_i}\tau(obs(MP_j)) \quad (12)
$$

3.4. Data Removal

The underlying idea of KF-based SLAM is that a subset of carefully selected frames is sufficient to for a robust estimate of the camera trajectory and 3D structure of the environment. However, to ensure robust pose estimation in the SLAM front-end, state-of-the-art systems, such as [22, 16, 25, 28], employ generous strategies for KF creation, generating many more KFs than actually necessary for a robust estimation of the SLAM graph. Therefore, we completely remove redundant KFs from the graph, without performing any marginalization. The results presented in Sec. 4 support this assumption. Furthermore, directly dropping KFs automatically keeps the Hessian matrix of the underlying optimization problem sparse, and does not require potentially time-consuming sparsification procedures such as [2] or [20] to remove the dense connections introduced by

marginalization. MPs are removed from the SLAM graph in case they are observed by less than two KFs and their 3D position is therefore not fully constraint anymore, or in case they are classified as outliers by GBA due to a high reprojection error.

4. Experimental Results

4.1. Setup

We firstly implement all presented metrics to remove redundant KFs into CVI-SLAM [16], a centralized visual-inertial collaborative SLAM system. CVI-SLAM takes inputs from multiple robotic agents, equipped with a visual-inertial sensor suite, and estimates a SLAM graph (a common *global map*) from all input data on a central instance, the "Server". GBA is applied to this global SLAM graph using the Ceres[2] solver, which allows to estimate the covariance matrix of the system from the optimization problem. In order to eliminate the time-constraint induced between KFs in visual-inertial SLAM, results are also shown using CCM-SLAM, a collaborative SLAM system with a similar architecture to [16], however operating on monocular sensors feeds. While there are no additional constraints for KF removal in CCM-SLAM, CVI-SLAM imposes the constraint of a maximum time of 2s between two KFs, to bound inaccuracies from IMU integration, as described in [16]. For the evaluation, we use an Intel NUC 7i7BNH (3.5 GHz 4, 16 GB RAM) to run our algorithms.

4.2. Datasets

We use the publicly available EuRoC benchmark dataset [1] for our evaluation, which provides accurate ground-truth position data from a Leica Total Station. From this dataset, we choose three pairs of sequences that cover different scenarios to build SLAM graphs of appropriate size to evaluate the redundancy detection strategies:

- **MH1+2**: Sequences "MH_01_easy" and "MH_02_easy". 5:32min total flight time, 150m total trajectory length. Industrial environment, good texture, bright scene. Multiple visits at the same location, no fast movements or viewpoint changes.

- **MH4+5**: Sequences 'MH_04_difficult" and "MH_05_difficult". 3:40min total flight time, 190m total trajectory length. Industrial environment, dark scene, fast motion. Exploratory trajectories.

- **V11+12**: Sequences "V1_01_easy" and "V1_02_medium". 3:48min total flight time, 135m total trajectory length. Fast motion and camera rotation, bright scene. Small office environment (8m ×8m), repeated observations of the same scene.

---

[2]http://ceres-solver.org

We evaluate the Root-Mean-Square trajectory Error (RMSE) for the following algorithms for redundancy detection on the SLAM graph:

- **MI-cond**: Classification of redundant KFs based on the measure from Eq. (10), conditioning the covariance matrices on the Markov Blanket before calculation Mutual Information (MI).

- **MI-orig**: Classification of redundant KFs based on "standard" mutual information (Eq. (3)), without conditioning the covariance matrices on the Markov Blanket.

- **Struct**: Classification of redundant KFs using our structure-based heuristic introduced in Sec. 3.3.

- **ORB**: Original classification of redundant KFs based on the algorithm proposed in [22].

- **No-red**: No redundancy detection and KF removal.

For our evaluation, "no-red" gives the baseline to quantify the decrease in accuracy by the removed data. For each experiment, we specify a maximum number of KFs Θ that are allowed in the SLAM graph, and enforce this limit using the different redundancy detection algorithms. As explained in Sec. 3, the methods "Struct" and "ORB" are able to run in real-time and therefore become active online during the SLAM estimate, as soon as the number of KFs in the SLAM graph exceeds Θ. "MI-cond" and "MI-orig" rely on costly optimization algorithms, and therefore compress the final pose graph produced by the SLAM estimate, after the SLAM estimation process itself is finished.
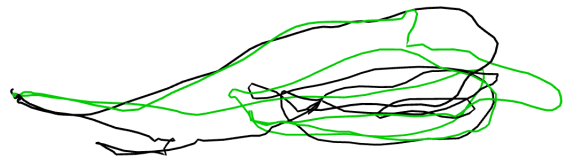
4.2.1 Evaluation on MH1+2

Figure 5: Collaborative trajectory estimate from CVI-SLAM on MH1+2 (top view, color encodes different agents).

Using the sequence pair "MH1+2", CVI-SLAM generates a SLAM graph with 585 KFs. The trajectory estimated by CVI-SLAM is depicted in Fig. 5. In a first experiment, we compress this graph by approximately 50% to $\Theta = 300$ KFs. In addition to the redundancy classification methods described above, we compare in this experiments also to random removal of KFs until the desired number of KFs in the graph is reached. The results in terms of RMSE reported in Fig. 6 show that the RMSE of the estimate increases only marginally in terms of absolute values, and all algorithms for redundancy detection perform more or less similar. This supports our assumption that for robust camera pose tracking, the SLAM front-end generates much more KFs than are

actually needed for accurate estimation of the SLAM graph in the back-end. To increase the effect of the redundancy detection algorithms and better determine the quality of their classification of redundant KFs, we compress the graph further to only 200 KFs. The results are again reported in Fig. 6. While the increase in RMSE is now more significant for all algorithms, we observe the expected behavior of the methods: The information-theoretic approaches lead to a lower decrease of accuracy, with "MI-cond" performing better than the "original" MI-approach. However, both outperform the structure-based heuristics, for which the one proposed in this paper outperforms the approach from ORB-SLAM. Furthermore, the results show that all algorithms perform better than naive random removal of KFs from the graph.
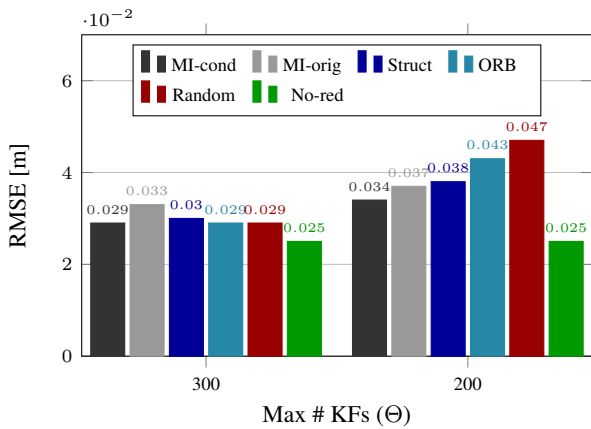

Figure 6: Final trajectory RMSE for a SLAM estimate on MH1+2 with 585 KFs, compressed to 300 resp. 200 KFs. Results averaged over 5 runs.
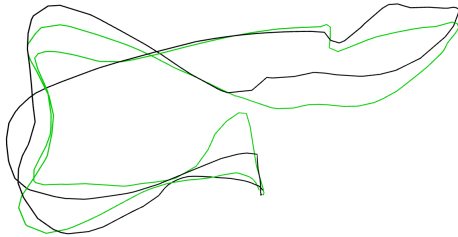
4.2.2 Evaluation on MH4+5

Figure 7: Collaborative trajectory estimate from CVI-SLAM on MH4+5 (top view, color encodes different agents).

We perform the same graph compression as for "MH1+2" on the more difficult dataset pair "MH4+5", generating a SLAM graph with 426 KFs (Fig. 7). Since the previous experiment confirms our assumption that "MI-cond" shows better performance in terms of accuracy compared to "MI-orig", we focus on "MI-cond" in the following experiments. The results are reported in Fig. 8. The RMSE of the estimate is generally higher, because of the more difficult trajectory with less texture, faster motion and less possibilities for loop closures due to te exploratory nature of the tra-

jectories. The information-theoretic metric again performs good, with almost no decrease in accuracy. Both heuristics perform significantly worse in this experiment, with our proposed heuristic and the one from ORB-SLAM showing similar error for a compression to 300 KFs, while for $\Theta = 200$, "Struct" performs again better.
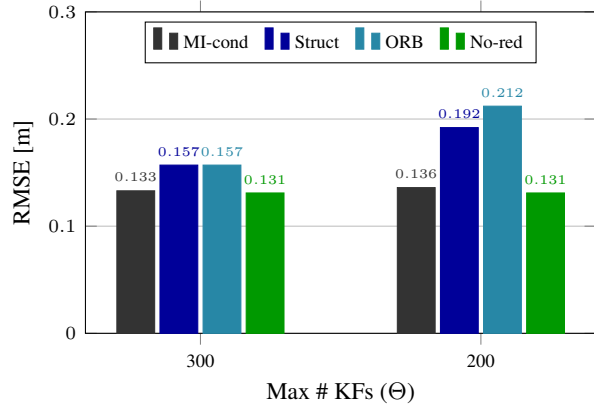

Figure 8: Final trajectory RMSE for a SLAM estimate on MH4+5 with 426 KFs, compressed to 300 resp. 200 KFs. Results averaged over 5 runs.
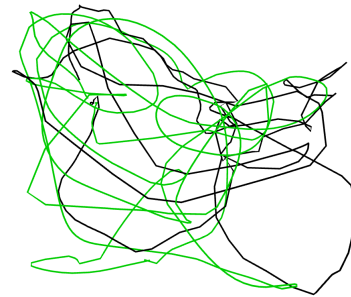
4.2.3 Evaluation on V11+12

Figure 9: Collaborative trajectory estimate from CVI-SLAM on V11+12 (top view, color encodes different agents).

For the dataset pair on "V11+12", CVI-SLAM generates an estimate with 776 KFs (Fig. 9). We again compress this graph by approximately 50% to $\Theta = 400$ KFs. As for the previous experiments, the results in Fig. 10 show that the information-theoretic metric is able to perform a compression of roughly 50% with almost no loss in accuracy. Also, both heuristic exhibit again a more significant drop in accuracy, with the structure-based heuristic proposed by this paper performing better than the one from [22].

4.2.4 Results for monocular SLAM

In addition to the evaluation on a visual-inertial collaborative SLAM system [16], we applied the methods for redundancy detection to a monocular SLAM system [28] to test the approaches under a different sensor modality, and without the time constraint for IMU integration in the visual-inertial system. We repeated selected experiments already
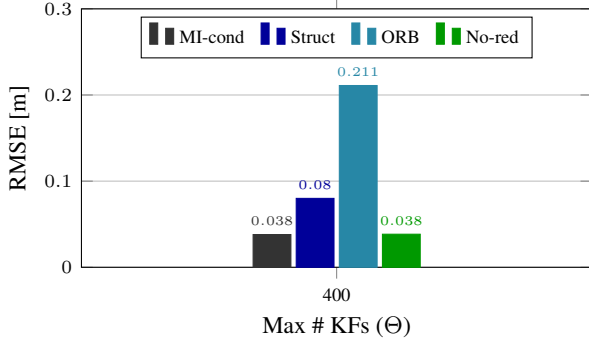
Figure 10: Final trajectory RMSE for a SLAM estimate on V11+12 with 776 KFs, compressed to 400 KFs. Results averaged over 5 runs.

performed with CVI-SLAM: In a first experiment, we compress a graph of 542 KFs generated from "MH1+2" to Θ = 200 KFs, and in a second experiments, a graph with 550 KFs generated from "MH4+5" also to Θ = 200 KFs. The results reported in Fig. 11 show similar results to those from the visual-inertial system: "MI-cond" is able to compress the graph with minimal loss in accuracy. On the easier "MH1+2" dataset, our structure-based heuristic gets close to the results of the information-theoretic approach, while for the more difficult dataset "MH4+5", the information-theoretic method clearly outperform it. Both proposed approaches outperform the original method from ORB-SLAM in both experiments.
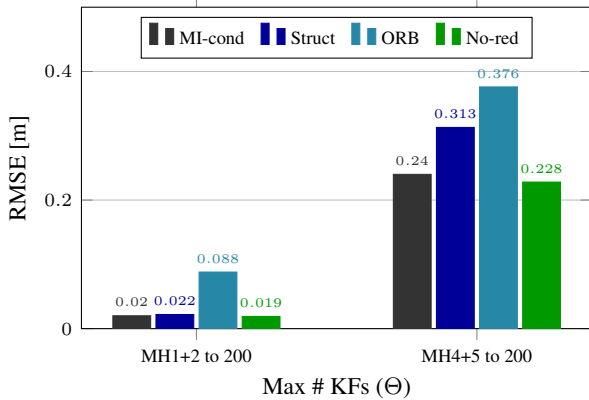

Figure 11: Final trajectory RMSE for a SLAM estimate using monocular SLAM [28] on MH1+2 with 542 KFs and MH4+5 with 550 KFs, both compressed to 200 KFs. Results averaged over 3 runs.

4.3. Discussion

The experimental results show that the proposed approach based in "conditional" MI for redundancy detection is able to compress SLAM graphs from visual-inertial as well as monocular SLAM by > 50% with minimal decrease in accuracy. In our experiments, the "original" MI-measure (without conditioning the covariance on the Markov blanket of the KF) exhibits a higher increase in RMSE compared to the "conditional" MI, attesting to the effectiveness of the proposed approach. For easier datasets, the difference be-

tween structure-based heuristics and information-theoretic approaches becomes only apparent for higher compression rates. On more difficult datasets, the choice which data should be removed affects the accuracy of the estimate considerably more. However, the structure-based heuristic proposed in this article exhibits a good trade-off between the loss of accuracy from the approximation (i.e. following the removal) of KFs deemed to be redundant and computation time, enabling online functionality in parallel to the SLAM estimation process. The minimal increase of the RMSE after the reduction of the graph supports the assumption that while for robust and accurate tracking by the SLAM front-end, a generous KF-creations strategy is beneficial, many of these KFs are dispensable later on, during global map optimization in the back-end without any negative impact on the accuracy.

5. Conclusion

In this article, we present a study on redundancy detection in keyframe-based SLAM. We evaluate different information-theoretic and heuristic measures, with Mutual Information using covariance matrices that are conditioned on the Markov blanket of a node in the SLAM graph performing best. Our experiments show that this approach is able to compress graphs estimated by state-of-the-art collaborative SLAM systems by more than 50% with no or minimal loss in accuracy. However, the information-theoretic measures rely on computationally expensive optimization methods, that might not be feasible for systems with limited processing power. Therefore, we propose as an alternative a new structure-based heuristic for classifying and removing redundant nodes from the graph, that is able to run online in parallel to the SLAM estimation process. This heuristic performs almost equally accurate to the information-theoretic approach for easier trajectories or smaller compression rates, while exhibiting a higher but still acceptable loss in accuracy even for large compression rates or more difficult environments, yet being computationally much more lightweight. We put the approaches to the test in two state-of-the-art centralized collaborative SLAM systems, attesting to their practicality. With global bundle adjustment being the most time-consuming part of the SLAM back-end, with cubic complexity in the number of nodes in the graph, these sparsification methods have great potential to significantly boost the scalability of SLAM systems and considerable increase in the workspace of the robotic team when applied to individual and collaborative SLAM. Future directions will focus on improving the efficiency of the MI-based approach, in order to use it online during missions. Furthermore, it would be desirable to drop the time-constraint in the visual-inertial SLAM system. Non-linear factor recovery for visual-inertial systems [32] might be an interesting direction to explore for this purpose.

References

[1] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart. The EuRoC micro aerial vehicle datasets. *International Journal of Robotics Research (IJRR)*, 35(10):1157–1163, 2016. 6

[2] N. Carlevaris-Bianco, M. Kaess, and R. M. Eustice. Generic node removal for factor-graph SLAM. *IEEE Transactions on Robotics (T-RO)*, 30(6):1371–1385, 2014. 2, 5

[3] L. Carlone, A. Censi, and F. Dellaert. Selecting good measurements via l_1 relaxation: A convex approach for robust estimation over graphs. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, pages 2667–2674. IEEE, 2014. 2

[4] M. Chli and A. J. Davison. Active matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 72–85. Springer, 2008. 2

[5] S. Choudhary, V. Indelman, H. I. Christensen, and F. Dellaert. Information-based reduced landmark slam. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4620–4627. IEEE, 2015. 2

[6] W. Churchill and P. Newman. Experience-based navigation for long-term localisation. *International Journal of Robotics Research (IJRR)*, (14):1645–1661, 2013. 2

[7] A. J. Davison. Active search for real-time vision. In *Proceedings of the International Conference on Computer Vision (ICCV)*, volume 1, pages 66–73. IEEE, 2005. 4

[8] I. Deutsch, M. Liu, and R. Siegwart. A Framework for Multi-Robot Pose Graph SLAM. In *IEEE International Conference on Real-time Computing and Robotics (RCAR)*, 2016. 3

[9] E. Eade, P. Fong, and M. E. Munich. Monocular graph SLAM with complexity reduction. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, pages 3017–3024. IEEE, 2010. 2

[10] C. Forster, S. Lynen, L. Kneip, and D. Scaramuzza. Collaborative monocular SLAM with multiple micro aerial vehicles. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, 2013. 3

[11] B. Hepp, M. Nießner, and O. Hilliges. Plan3d: Viewpoint and trajectory optimization for aerial multi-view stereo reconstruction. *ACM Transactions on Graphics (TOG)*, 38(1):4, 2018. 2

[12] J. Hsiung, M. Hsiao, E. Westman, R. Valencia, and M. Kaess. Information sparsification in visual-inertial odometry. 2018. 2

[13] G. Huang, M. Kaess, and J. J. Leonard. Consistent sparsification for graph optimization. In *European Conference on Mobile Robots (ECMR)*, pages 150–157. IEEE, 2013. 2

[14] V. Ila, J. M. Porta, and J. Andrade-Cetto. Information-based compact Pose SLAM. *IEEE Transactions on Robotics (T-RO)*, 26(1):78–93, 2010. 2

[15] H. Johannsson, M. Kaess, M. Fallon, and J. J. Leonard. Temporally scalable visual SLAM using a reduced pose graph. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 54–61. IEEE, 2013. 2

[16] M. Karrer, P. Schmuck, and M. Chli. CVI-SLAM - Collaborative Visual-Inertial SLAM. *IEEE Robotics and Automation Letters (RA-L)*, 3(4):2762–2769, 2018. 1, 2, 3, 4, 5, 6, 7

[17] K. Konolige and M. Agrawal. FrameSLAM: From bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics (T-RO)*, 24(5):1066–1077, 2008. 2

[18] H. Kretzschmar and C. Stachniss. Information-theoretic compression of pose graphs for laser-based SLAM. *International Journal of Robotics Research (IJRR)*, 31(11):1219–1230, 2012. 2

[19] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale. Keyframe-based visual–inertial odometry using nonlinear optimization. *International Journal of Robotics Research (IJRR)*, 34(3):314–334, 2015. 1

[20] M. Mazuran, W. Burgard, and G. D. Tipaldi. Nonlinear factor recovery for long-term slam. *International Journal of Robotics Research (IJRR)*, 35(1-3):50–72, 2016. 2, 5

[21] B. Mu, L. Paull, A.-A. Agha-Mohammadi, J. J. Leonard, and J. P. How. Two-stage focused inference for resource-constrained minimal collision navigation. *IEEE Transactions on Robotics (T-RO)*, 33(1):124–140, 2017. 2

[22] R. Mur-Artal, J. Montiel, and J. D. Tardós. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics (T-RO)*, 31(5):1147–1163, 2015. 1, 3, 4, 5, 6, 7

[23] R. Mur-Artal and J. D. Tardós. Visual-inertial monocular SLAM with map reuse. *IEEE Robotics and Automation Letters (RA-L)*, 2(2):796–803, 2017. 1

[24] L. Paull, G. Huang, and J. J. Leonard. A unified resource-constrained framework for graph slam. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1346–1353. IEEE, 2016. 3

[25] T. Qin, P. Li, and S. Shen. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics (T-RO)*, 34(4):1004–1020, 2018. 1, 3, 5

[26] L. Riazuelo, J. Civera, and J. Montiel. C2TAM: A cloud framework for cooperative tracking and mapping. *Robotics and Autonomous Systems (RAS)*, 62(4):401–413, 2014. 3

[27] P. Schmuck and M. Chli. Multi-UAV Collaborative Monocular SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2017. 1

[28] P. Schmuck and M. Chli. CCM-SLAM: Robust and efficient centralized collaborative monocular simultaneous localization and mapping for robotic teams. *Journal of Field Robotics (JFR)*, 36(4):763–781, 2019. 1, 2, 3, 5, 7, 8

[29] T. Schneider, M. Li, M. Burri, J. Nieto, R. Siegwart, and I. Gilitschenski. Visual-inertial self-calibration on informative motion segments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 6487–6494. IEEE, 2017. 2

[30] N. Snavely, S. M. Seitz, and R. Szeliski. Skeletal graphs for efficient structure from motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 2, 2008. 2

[31] H. Strasdat, J. M. Montiel, and A. J. Davison. Visual SLAM: why filter? *Image and Vision Computing*, 30(2):65–77, 2012. 1

[32] V. Usenko, N. Demmel, D. Schubert, J. Stückler, and D. Cremers. Visual-inertial mapping with non-linear factor recovery. *arXiv preprint arXiv:1904.06504*, 2019. 8

[33] J. Vallvé, J. Solà, and J. Andrade-Cetto. Graph slam sparsification with populated topologies using factor descent optimization. *IEEE Robotics and Automation Letters (RA-L)*, 3(2):1322–1329, 2018. 2

[34] J. Vial, H. Durrant-Whyte, and T. Bailey. Conservative sparsification for efficient and consistent approximate estimation. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, pages 886–893. IEEE, 2011. 3