

Online Variational Bayesian Motion Averaging

Guillaume Bourmaud^(✉)

Toshiba Research Europe, Cambridge, UK
guillaume.bourmaud@crl.toshiba.co.uk

Abstract. In this paper, we propose a novel algorithm dedicated to *online motion averaging for large scale problems*. To this end, we design a filter that continuously approximates the posterior distribution of the estimated transformations. In order to deal with large scale problems, we associate a *variational Bayesian approach* with a *relative parametrization of the absolute transformations*. Such an association allows our algorithm to simultaneously possess two features that are essential for an algorithm dedicated to large scale online motion averaging: (1) a low computational time, (2) the ability to detect wrong loop closure measurements. We extensively demonstrate on several applications (binocular SLAM, monocular SLAM and video mosaicking) that our approach not only exhibits a low computational time and detects wrong loop closures but also significantly outperforms the state of the art algorithm in terms of RMSE.

Keywords: Variational Bayes · Motion averaging · Pose-graph · Lie group · Filtering · Relative parametrization · Large scale · Visual SLAM

1 Introduction

The motion averaging problem, also called “multiple rotation averaging” when dealing with 3D rotations or “pose-graph inference” when applied to camera poses, has been studied for more than fifteen years [6, 14, 16–19, 24, 31, 32] and is still a very active area of research [4, 5, 7, 8, 10, 12, 20, 29]. This generic problem arises in a large number of applications, such as video mosaicking [6, 24], reconstruction of 3D scenes [10, 27] or visual SLAM [13, 14], where *only* the considered group of transformations changes: $SE(3)$ for 3D euclidean motions, $SL(3)$ for homographies, $Sim(3)$ for 3D similarities. In fact, in all these applications, the task consists in estimating absolute transformations, between a “world” coordinate system and local coordinate systems, given noisy measurements corresponding to relative transformations between pairs of local coordinate systems.

The noisy relative transformation measurements are usually obtained by processing a video stream, coming from an RGB or RGB-D camera, with two different modules:

Electronic supplementary material The online version of this chapter (doi:[10.1007/978-3-319-46484-8.8](https://doi.org/10.1007/978-3-319-46484-8.8)) contains supplementary material, which is available to authorized users.

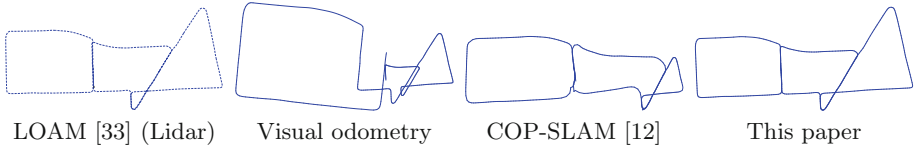


Fig. 1. Results for monocular visual SLAM ($Sim(3)$) on sequence KITTI 13. The ground truth is not available for that sequence. Thus, we reported the best result obtained using a Lidar [33].

- a *visual odometry module* that continuously computes the transformation between the current and the previous local coordinate system of the camera;
- a *loop closure module* that detects when the camera comes back in a previously visited area and computes a relative transformation.

The odometry measurements and loop closure measurements are essentially of the same nature, however, in practice the loop closure module might produce erroneous measurements because of some *perceptual aliasing* (two different places can be very similar), while the visual odometry module usually produces outlier free measurements.

Since the input data is a video stream, most of the applications require an *online estimation* of the absolute transformations. However, the majority of the state of the art approaches do not take that constraint into account in their initial specifications. They usually design a batch algorithm which is applied each time a new measurement is received, using a generic optimization toolbox such as GTSAM [11], g^2o [21] or Google Ceres Solver [2]. These toolboxes are highly optimized and able to provide an online estimation with a reasonable computational time for small or medium sized problems, nevertheless, their computational time becomes prohibitive for large scale problems (see [12]).

The purpose of this paper is to present a novel approach specifically designed to operate *online on large scale problems*.

Requirements and Contributions Besides trying to be as accurate as possible, an algorithm dedicated to online motion averaging for large scale problems should also satisfy the following specifications:

- (1) Computational efficiency: As it was recently pointed out in [12], minimizing a criterion involving all the past measurements each time a new measurement is received is not suitable for the problem we consider. In order to achieve a low computational time, it is compulsory to *perform filtering*, i.e. to process the measurements one by one;
- (2) Memory efficiency: Nevertheless, to perform filtering is not sufficient to obtain an efficient algorithm. For instance, applying a Kalman filter, as proposed in [5], leads to maintaining a covariance matrix whose size grows quadratically with the number of absolute transformations. Hence, such a filter becomes impractical for large scale problems. One way to get a filter able to deal with large scale problems is to seek to continuously approximate the posterior distribution of the estimated transformations, such that

the number of parameters of that distribution grows *at most linearly* with the number of estimated transformations;

- (3) **Robustness:** Finally, dealing with large scale problems increases the risk of *perceptual aliasing* and consequently the number of wrong loop closures. Hence, our approach should also be able to detect and remove wrong loop closures.

As we will see, taking into account the constraints previously described will lead us to considering mathematical tools, such as variational Bayesian approximations, that have not been applied to motion averaging yet. Using these tools, we show that it is possible to obtain a highly efficient and robust online motion averaging algorithm that significantly outperforms the state of the art algorithm [12] (see Fig. 1).

Outline of the paper: The rest of the paper is organized as follows: The mathematical notations and models are presented in Sect. 2. In Sect. 3 we discuss work related to our novel approach. Section 4 deals with the specific case of motion averaging from odometry measurements and a single loop closure. Based on the analysis performed in Sect. 4, we derive a novel motion averaging algorithm in Sect. 5 which is evaluated experimentally in Sect. 6. Finally, a conclusion and future work directions are provided in Sect. 7.

2 Models and Notations

Let us now introduce the notations and mathematical models that are used throughout the paper.

2.1 Lie Group Notations

The theory we develop in the paper can be applied to any matrix Lie group (typically $SE(3)$, $SL(3)$, $Sim(3)$, etc.), which turns out to be very convenient in practice since it allows to apply our algorithm to various applications (see Sect. 6). For a detailed description of Lie groups the reader is referred to [9]. Throughout the paper, we will use the following notations: $G \subset \mathbb{R}^{n \times n}$ is a matrix Lie group of intrinsic dimension p (i.e. $p = 6$ if $G = SE(3) \subset \mathbb{R}^{4 \times 4}$, $p = 8$ if $G = SL(3) \subset \mathbb{R}^{3 \times 3}$, etc.); $\exp_G^\wedge(\cdot) : \mathbb{R}^p \rightarrow G$ and $\log_G^\vee(\cdot) : G \rightarrow \mathbb{R}^p$ correspond to the exponential and logarithm maps of G respectively; $T_{ij} \in G$ is a matrix representing the transformation from the coordinate system j to the coordinate system i , thus in our notations $T_{ij}T_{jk} = T_{ik}$ and $T_{ij}^{-1} = T_{ji}$.

Another important operator that we will employ is the adjoint representation of G , $\text{Ad}_G(\cdot) : G \rightarrow \mathbb{R}^{p \times p}$, which allows to transport an element $\delta_{ij} \in \mathbb{R}^p$, acting initially on T_{ij} through left multiplication, onto the right side of T_{ij} such that $\exp_G^\wedge(\delta_{ij})T_{ij} = T_{ij}\exp_G^\wedge(\text{Ad}_G(T_{ji})\delta_{ij})$. Finally, we introduce the notation for a Gaussian distribution on G :

$$\mathcal{N}_G(T_{ij}; \bar{T}_{ij}, P_{ij}) \propto e^{-\frac{1}{2} \left\| \log_G^\vee(T_{ij} \bar{T}_{ij}^{-1}) \right\|_{P_{ij}}^2} \iff \begin{array}{l} T_{ij} = \exp_G^\wedge(\epsilon_{ij}) \bar{T}_{ij} \\ \text{where } \epsilon_{ij} \sim \mathcal{N}_{\mathbb{R}^p}(\epsilon_{ij}; \mathbf{0}, P_{ij}) \end{array} \quad (1)$$

where $\|\cdot\|^2$ stands for the squared Mahalanobis distance while \bar{T}_{ij} and P_{ij} are the mean and the covariance of the random variable T_{ij} respectively.

2.2 Measurement Models

In order to tackle the motion averaging problem, two different parametrizations of the absolute transformations are commonly used: the relative parametrization and the absolute parametrization. Of course, different parametrizations lead to different measurement models and consequently to algorithms having different computational complexities and posterior distributions having different shapes. In this paper, we employ the relative parametrization. This choice is motivated in Sect. 4. Here we simply introduce the notations and the measurement models for this parametrization.

Table 1. Odometry measurement model and loop closure measurement model using a relative parametrization of the absolute transformations

Odometry likelihood	Loop closure likelihood
$ \begin{aligned} & p(Z_{n(n+1)} T_{n(n+1)}) \\ & = \mathcal{N}_G(Z_{n(n+1)}; T_{n(n+1)}, \Sigma_{n(n+1)}) \end{aligned} \tag{2} $	$ \begin{aligned} & p(Z_{mn} \{T_{i(i+1)}\}_{i=m, \dots, n-1}) \\ & = \mathcal{N}_G(Z_{mn}; \prod_{i=m}^{n-1} T_{i(i+1)}, \Sigma_{mn}) \end{aligned} \tag{3} $

Let us first define our notations for the measurements. An odometry measurement, which we denote $Z_{n(n+1)} \in G$, is a noisy transformation between two temporally *consecutive* local coordinate systems. A loop closure measurement, which we denote $Z_{mn} \in G$ where $n \neq m + 1$, is a noisy transformation between two temporally *nonconsecutive* local coordinate systems. Moreover, in this work we assume the noises on the measurements to be *mutually independent*.

The relative parametrization consists in estimating transformations of the form $T_{(k-1)k}$ where k is the local coordinate system of the camera at time instant k . Thus, at time instant k , the set of estimated transformations is $\{T_{i(i+1)}\}_{i=1, \dots, k-1}$. Let us note that the absolute transformation T_{1k} can be obtained simply by composing the estimated relative transformations i.e. $T_{1k} = T_{12}T_{23} \cdots T_{(k-1)k} = \prod_{i=1}^{k-1} T_{i(i+1)}$. The likelihood for an odometry measurement and a loop closure are assumed to be Gaussian and are given in Table 1, Eq. (2) and Eq. (3), respectively.

3 Related Work

In this section, we describe the most recent state of the art approaches and how they are related to the novel method we propose in this paper.

The current workhorse for motion averaging is the Gauss-Newton (GN) algorithm. In fact, this algorithm has been employed for more than a decade to tackle the motion averaging problem (the seminal work of [17] was already proposing

to use it). In this context, both relative and absolute parametrizations of the absolute transformations have been employed.

The most widely used is the *absolute parametrization* [13]. The main reason why people tend to use this parametrization is that it leads to solving, at each iteration of the GN, a *sparse* linear system. Even if the size of this linear system is proportional to the number of absolute transformations, its sparsity is usually exploited in solvers such as g^2o , resulting in an algorithm with a reasonable computational time for a small or medium sized problem. Using this formalism, several algorithms have been recently proposed to perform robust motion averaging (i.e. when loop closures contain erroneous measurements): [1, 8] proposed re-weighted schemes; [10, 28] introduced auxiliary variables that in fact correspond to using a robust kernel as it was recently shown in [30] in the context of bundle adjustment; [22] proposed a consensus based algorithm which optimizes clusters of loop closures with a GN and checks their consistency with statistical tests. None of these approaches fulfill our first two requirements (Computational efficiency and Memory efficiency). However, in order to demonstrate the ability of our novel algorithm to detect wrong loop closures, we will compare its results against the Dynamic Covariance Scaling (DCS) [1].

The *relative parametrization* was initially used in [14] for the specific case of planar motions and was then extended to general matrix Lie groups in [25]. Each iteration of these algorithms corresponds to a GN step, even if it is not presented as such. Employing the relative parametrization leads to solving, at each iteration of the GN, a *dense* linear system whose size is proportional to the number of loop closure measurements. Consequently, the approach proposed in [25] is highly efficient when the number of loop closures is small but impractical for large scale problems. At first sight, the relative parametrization does not seem very attractive for our problem since we are mostly interested in large scale problems. However, we will see that, using this parametrization, the posterior distribution of the relative transformations has a specific shape that can be approximated with few parameters.

To the best of our knowledge, the most closely related approaches to the one we propose in this paper are the filters proposed in [5, 12]. The algorithm proposed in [5] uses a Kalman filter to estimate absolute transformations using an absolute parametrization and validation gating to detect wrong loop closures. However, their approximation of the posterior distribution is a multivariate Gaussian distribution whose covariance matrix grows quadratically with the number of absolute transformations. Consequently, this filter is impractical for large scale problems. On the contrary, the authors of [12] use a relative parametrization and propose a novel closed-form way to process each loop closure using the concept of trajectory bending. This leads to a highly efficient filter which does not explicitly try to approximate the posterior distribution of the relative transformations but estimates the uncertainty of each transformation with a single scalar. Consequently, this filter also fulfills our “memory efficiency” requirement. However, this approach assumes that the loop closure measurements do not contain outliers.

Table 2. Comparison of state of the art approaches dedicated to motion averaging

	Ours	[12]	[5]	[1]
Computational efficiency	✓	✓	✓	×
Memory efficiency	✓	✓	×	×
Robustness	✓	×	✓	✓

Contrary to these methods, in this paper, we propose a novel filter based on a variational Bayesian approximation of the posterior distribution of the relative transformations which allows to fulfill our three requirements (see Table 2) while being almost as accurate as a batch approach.

4 The Case of a Single Loop

In this paper, we are interested in designing a Bayesian filter which, by definition, has to process the measurements sequentially in order to approximate the posterior distribution of the estimated transformations. However, as we have already seen, two parametrizations of the absolute transformations are possible. In this section, we motivate our choice of employing the relative parametrization on the simpler problem of motion averaging from odometry measurements and a single loop closure (see Fig. 2(a)).

In fact, we consider a loop of length N_L , where we are given $N_L - 1$ odometry measurements $\{Z_{i(i+1)}\}_{i=1, \dots, N_L-1}$ and a single loop closure Z_{1N_L} between the local coordinate systems 1 and N_L .

Using the likelihoods defined in Eqs. 2 and 3, we wish to minimize the following criterion w.r.t the relative transformations $\{T_{i(i+1)}\}_{i=1, \dots, N_L-1}$:

$$\begin{aligned}
 & -2 \ln \left(p \left(\{T_{i(i+1)}\}_{i=1, \dots, N_L-1} \mid Z_{1N_L}, \{Z_{i(i+1)}\}_{i=1, \dots, N_L-1} \right) \right) \\
 & = \left\| \log_G^\vee \left(Z_{1N_L} \left(\prod_{i=1}^{N_L-1} T_{i(i+1)} \right)^{-1} \right) \right\|_{\Sigma_{1N_L}}^2 + \sum_{i=1}^{N_L-1} \left\| \log_G^\vee \left(T_{i(i+1)} Z_{i(i+1)}^{-1} \right) \right\|_{\Sigma_{i(i+1)}}^2 + \text{cst} \quad (4)
 \end{aligned}$$

One way to minimize this criterion is to apply a Gauss-Newton algorithm where the relative transformations are jointly refined iteratively as follows (the superscript stands for the iteration):

$$T_{i(i+1)}^{(l)} = \exp_G^\wedge \left(\delta_{i(i+1)}^{(l/l-1)} \right) T_{i(i+1)}^{(l-1)} \quad \text{for } i = 1 \dots N_L - 1. \quad (5)$$

The increments $\left\{ \delta_{i(i+1)}^{(l/l-1)} \right\}_{i=1, \dots, N_L-1}$ are obtained at each iteration by solving the following (dense) linear system of size $p \times N_L$:

$$\begin{bmatrix} \delta_{12}^{(l/l-1)} \\ \vdots \\ \delta_{(N_L-1)N_L}^{(l/l-1)} \end{bmatrix} = \left(\left(J_{rel}^{(l)} \right)^T \Lambda J_{rel}^{(l)} \right)^{-1} \left(J_{rel}^{(l)} \right)^T \Lambda \begin{bmatrix} r_{12}^{(l-1)} \\ \vdots \\ r_{(N_L-1)N_L}^{(l-1)} \\ r_{1N_L}^{(l-1)} \end{bmatrix} \quad (6)$$

where $J_{rel}^{(l)}$ is the Jacobian matrix of the system (see Fig. 2(b)), Λ is a block diagonal matrix concatenating the inverse of the covariance matrices of the measurements, $r_{1N_L}^{(l-1)} = \log_G^\vee \left(Z_{1N_L} \left(\prod_{i=1}^{N_L-1} T_{i(i+1)}^{(l-1)} \right)^{-1} \right)$ and $r_{i(i+1)}^{(l-1)} = \log_G^\vee \left(T_{i(i+1)}^{(l-1)} Z_{i(i+1)}^{-1} \right)$.

At first sight, the relative parametrization does not seem very interesting compared to the absolute parametrization since it requires to solve a dense linear system (in Fig. 2(b) the pseudo-Hessian $J_{rel}^T \Lambda J_{rel}$ is completely dense) instead of a sparse one in the absolute parametrization case¹ (in Fig. 2(b) the pseudo-Hessian $J_{abs}^T \Lambda J_{abs}$ is extremely sparse). However, as proven in the supplementary material, by initializing $T_{i(i+1)}^{(0)} = Z_{i(i+1)}$ for $i = 1 \dots N_L - 1$, using the Woodbury formula and exploiting the structure of the problem, it is possible to show that

$$\begin{aligned} \begin{bmatrix} \delta_{12}^{(l/l-1)} \\ \vdots \\ \delta_{(N_L-1)N_L}^{(l/l-1)} \end{bmatrix} &\simeq \begin{bmatrix} \Sigma_{12} & \mathbf{0} \\ & \ddots \\ \mathbf{0} & \Sigma_{(N_L-1)N_L} \end{bmatrix} \left(J_{LC}^{(l)} \right)^T \left(\Sigma_{1N_L} + \sum_{i=1}^{N_L-1} J_{1i}^{(l)} \Sigma_{i(i+1)} \left(J_{1i}^{(l)} \right)^T \right)^{-1} \\ &\cdot \left(r_{1N_L}^{(l-1)} + J_{LC}^{(l)} \sum_{n=1}^{l-1} \begin{bmatrix} \delta_{12}^{(n/n-1)} \\ \vdots \\ \delta_{(N_L-1)N_L}^{(n/n-1)} \end{bmatrix} \right) - \sum_{n=1}^{l-1} \begin{bmatrix} \delta_{12}^{(n/n-1)} \\ \vdots \\ \delta_{(N_L-1)N_L}^{(n/n-1)} \end{bmatrix} \end{aligned} \quad (7)$$

where $J_{LC}^{(l)} = \left[J_{11}^{(l)} \dots J_{1(N_L-1)}^{(l)} \right]$ is the Jacobian of the loop closure error and $J_{1n}^{(l)} \simeq \text{Ad}_G \left(\prod_{i=1}^{n-1} T_{i(i+1)}^{(l-1)} \right)$. In this case, only a linear system of size p (i.e. independent of the length of the loop) has to be solved, making the algorithm highly efficient to close a single loop (in practice, $p = 6$ for $G = SE(3)$, $p = 8$ for $G = SL(3)$, etc.). Moreover, the inverse of the pseudo-Hessian $\left(J_{rel}^T \Lambda J_{rel} \right)^{-1}$ (see Fig. 2(b)), which represents (once the algorithm has reached convergence) the covariance matrix of the posterior distribution under a linear approximation, exhibits very small correlations between the transformations. Therefore, a block diagonal approximation of that covariance matrix seems to be a reasonable approximation that would allow us to derive a filter being able to deal with large

¹ Inference in the case of a single loop using the absolute parametrization is detailed in the supplementary material.

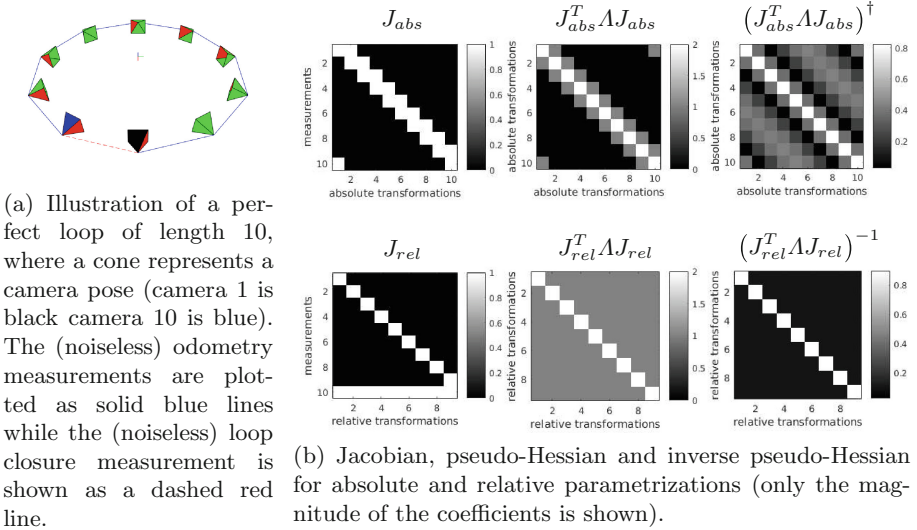


Fig. 2. Illustration of the motion averaging problem on $SE(3)$ for a single loop. Using an absolute parametrization, the inverse pseudo-Hessian exhibits very strong correlations between the absolute transformations. On the contrary, using a relative parametrization, the inverse pseudo-Hessian has very small correlations (not null but close to zero) between the relative transformations, motivating our variational Bayesian approximation of the posterior distribution which assumes independent relative transformations (see text for details). (a) Illustration of a perfect loop of length 10, where a cone represents a camera pose (camera 1 is black camera 10 is blue). The (noiseless) odometry measurements are plotted as solid blue lines while the (noiseless) loop closure measurement is shown as a dashed red line. (b) Jacobian, pseudo-Hessian and inverse pseudo-Hessian for absolute and relative parametrizations (only the magnitude of the coefficients is shown). (Color figure online)

scale problems very efficiently. On the contrary, when using the absolute parametrization, the (pseudo-)inverse of the pseudo-Hessian $(J_{abs}^T \Lambda J_{abs})^\dagger$ manifests very strong correlations, making any approximation of that matrix difficult.

From this point of view, the relative parametrization seems to be much more attractive than the absolute parametrization, at least for online inference.

Consequently, when designing our filter, we will employ a relative parametrization. Loop closure measurements will be processed sequentially using this highly efficient GN which only requires to solve a linear system of size p at each iteration. After having processed a loop closure, the covariance matrix of the posterior distribution will be approximated with a block diagonal covariance matrix using a variational Bayesian approximation. All these steps are detailed in Sect. 5.

Let us note that, since our approach employs a GN to process loop closures sequentially, it is optimal for *any* problem (i.e. any matrix Lie group G with

anisotropic noises on the measurements) *containing loops that do not interact with each other*. On the contrary, for the same problems, COP-SLAM [12] is only optimal when the noise is isotropic and the logarithm map of G is related to a bi-invariant metric which in practice is usually not true, except for $SO(3)$.

5 Online Variational Bayesian Motion Averaging

In the previous Section, we showed that the relative parametrization was appealing for the online motion averaging problem. We now derive our novel filter using this parametrization.

5.1 Estimated State

At time instant $k-1$ (where $k > 2$), the estimated state consists in all the relative transformations $\mathcal{X}_{k-1} = \{T_{i(i+1)}\}_{i=1, \dots, k-2}$. More specifically, at time instant $k-1$, the posterior distribution of the state is assumed to have the following factorized form:

$$p(\mathcal{X}_{k-1} | \mathcal{D}_{odo, k-1}, \mathcal{D}_{LC, k-1}) = Q_{k-1}(\mathcal{X}_{k-1}) = \prod_{i=1}^{k-2} \mathcal{N}_G(T_{i(i+1)}; \bar{T}_{i(i+1)}, P_{i(i+1)}) \quad (8)$$

where $\mathcal{D}_{odo, k-1} = \{Z_{i(i+1)}\}_{i=1, \dots, k-2}$ and $\mathcal{D}_{LC, k-1} = \{Z_{ij}\}_{1 \leq i < j-1 < k-1}$.

5.2 Processing of a New Odometry Measurement

At time instant k , when the new odometry measurement $Z_{(k-1)k}$ (with known covariance $\Sigma_{(k-1)k}$) is available, the estimated state simply augments, i.e. $\mathcal{X}_k = \{T_{i(i+1)}\}_{i=1, \dots, k-1}$. Consequently, the posterior distribution of the state remains factorized and has the following form:

$$p(\mathcal{X}_k | \mathcal{D}_{odo, k}, \mathcal{D}_{LC, k-1}) = Q_k^{odo}(\mathcal{X}_k) = \prod_{i=1}^{k-1} \mathcal{N}_G(T_{i(i+1)}; \bar{T}_{i(i+1)}, P_{i(i+1)}) \quad (9)$$

where $\mathcal{D}_{odo, k} = \{Z_{i(i+1)}\}_{i=1, \dots, k-1}$, $\bar{T}_{(k-1)k} = Z_{(k-1)k}$ and $P_{(k-1)k} = \Sigma_{(k-1)k}$.

5.3 Processing of a New Loop Closure Measurement

At time instant k , after having received the odometry measurement $Z_{(k-1)k}$, a new loop closure measurement Z_{lk} (with known covariance Σ_{lk}) may be available (where $l < k$). In fact, multiple loop closures may be available, however, in order to keep the notations uncluttered, we only describe how to deal with one loop closure. In practice, the processing is applied sequentially to each loop closure as it is described in the pseudo-code presented in the supplementary material.

When a new loop closure measurement Z_{lk} is available, we would like to take into account the information coming from that observation in order to refine our current estimate of the state. However, the observation model Eq. 3 creates dependencies between all the relative transformations involved in the loop, and, therefore, the posterior distribution $p(\mathcal{X}_k | \mathcal{D}_{odo,k}, \mathcal{D}_{LC,k-1}, Z_{lk})$ is not factorized anymore. Thus, the number of parameters required to describe that non-factorized posterior distribution becomes huge (typically in $O(k^2)$ using a linear approximation, see [5]), especially for large scale problems.

In order for our filter to be able to operate online on large scale problems, we propose to approximate that posterior distribution with a factorized distribution, whose number of parameters will be in $O(k)$. Such an approximation is motivated by our analysis of the single loop case (see Fig. 2). One way to find a factorized distribution “similar” to the true posterior distribution is to minimize the Kullback-Leibler divergence

$$D_{KL}(Q_{VB}(\mathcal{X}_k) || p(\mathcal{X}_k | \mathcal{D}_{odo,k}, \mathcal{D}_{LC,k-1}, Z_{lk})) \text{ with } Q_{VB}(\mathcal{X}_k) = \prod_{i=1}^{k-1} q_{VB}(T_{i(i+1)}). \quad (10)$$

This approach is usually called “variational Bayesian approximation” in the literature [3] and sometimes “Structured Mean Field” since we do not assume a fully factorized distribution but only the relative transformations to be mutually independent.

Variational Bayesian Approximation. Minimizing the KL divergence in (10) w.r.t \mathcal{X}_k corresponds to maximizing the lower bound

$$\mathcal{L}(Q_{VB}(\mathcal{X}_k)) = \int Q_{VB}(\mathcal{X}_k) \ln \left(\frac{p(\mathcal{X}_k, \mathcal{D}_{odo,k}, \mathcal{D}_{LC,k-1}, Z_{lk})}{Q_{VB}(\mathcal{X}_k)} \right) d\mathcal{X}_k. \quad (11)$$

In our case the (log)-joint distribution of all the random variables has the form:

$$\begin{aligned} \ln(p(\mathcal{X}_k, \mathcal{D}_{odo,k}, \mathcal{D}_{LC,k-1}, Z_{lk})) &= \ln \left(p(Z_{lk} | \{T_{i(i+1)}\}_{i=1, \dots, k-1}) Q_k^{odo}(\mathcal{X}_k) \right) \\ &= -\frac{1}{2} \left\| \log_G^{\vee} \left(Z_{lk} \left(\prod_{i=1}^{k-1} T_{i(i+1)} \right)^{-1} \right) \right\|_{\Sigma_{lk}}^2 - \frac{1}{2} \sum_{i=1}^{k-1} \left\| \log_G^{\vee} \left(T_{i(i+1)} \bar{T}_{i(i+1)}^{-1} \right) \right\|_{P_{i(i+1)}}^2 + \text{cst} \end{aligned} \quad (12)$$

However, because of the curvature of the Lie group, the terms inside the norms in (12) are not linear in the transformations. One way to apply a variational Bayesian approach in this case is to linearize the terms inside the norms.

At this point, we find it convenient to define the variables involved in the linearization step:

$$T_{i(i+1)} = \exp_G^{\wedge}(\epsilon_{i(i+1)}) \check{T}_{i(i+1)} \quad \text{for } i = 1 \dots k-1 \quad (13)$$

where $\check{T}_{i(i+1)}$ is the (fixed) linearization point for the relative transformation $T_{i(i+1)}$ and $\epsilon_{i(i+1)}$ is a random variable. This linearization point is quite important and its value is discussed at the end of the section. After linearization, the

log-joint distribution becomes:

$$\begin{aligned} & \ln \left(p \left(\{ \epsilon_{i(i+1)} \}_{i=1, \dots, k-1}, \mathcal{D}_{odo,k}, \mathcal{D}_{LC,k-1}, Z_{lk} \right) \right) \\ &= -\frac{1}{2} \left\| r_{lk} - \sum_{i=l}^{k-1} J_{li} \epsilon_{i(i+1)} \right\|_{\Sigma_{lk}}^2 - \frac{1}{2} \sum_{i=1}^{k-1} \| r_{i(i+1)} + \epsilon_{i(i+1)} \|_{P_{i(i+1)}}^2 + \text{cst} \end{aligned} \quad (14)$$

where we approximated the Jacobian of \log_G^\vee by the identity, $J_{li} \simeq \text{Ad}_G \left(\prod_{j=l}^{i-1} \check{T}_{j(j+1)} \right)$ for $i > l$, $J_{ll} \simeq Id$, $r_{lk} = \log_G^\vee \left(Z_{lk} \left(\prod_{i=l}^{k-1} \check{T}_{i(i+1)} \right)^{-1} \right)$ and $r_{i(i+1)} = \log_G^\vee \left(\check{T}_{i(i+1)} \overline{T}_{i(i+1)}^{-1} \right)$.

Given the log-joint distribution (14), the objective is now to maximize the lower bound (11) where \mathcal{X}_k is now replaced with $\{ \epsilon_{i(i+1)} \}_{i=1, \dots, k-1}$ because of the linearization step. Here:

$$Q_{VB} \left(\{ \epsilon_{i(i+1)} \}_{i=1, \dots, k-1} \right) = \prod_{i=1}^{k-1} q_{VB} \left(\epsilon_{i(i+1)} \right). \quad (15)$$

In fact, it is possible to show that (see [3] pp. 446), for each variable $\epsilon_{i(i+1)}$, the best approximated distribution is given by the following expression:

$$\begin{aligned} & \ln \left(q_{VB}^* \left(\epsilon_{i(i+1)} \right) \right) \\ &= \mathbb{E}_{Q_{VB} \setminus q_{VB}^* \left(\epsilon_{i(i+1)} \right)} \left[\ln \left(p \left(\{ \epsilon_{i(i+1)} \}_{i=1, \dots, k-1}, \mathcal{D}_{odo,k}, \mathcal{D}_{LC,k-1}, Z_{lk} \right) \right) \right] + \text{cst} \end{aligned} \quad (16)$$

where $\mathbb{E}_{Q_{VB} \setminus q_{VB}^* \left(\epsilon_{i(i+1)} \right)}$ stands for the conditional expectation w.r.t all the variables except $\epsilon_{i(i+1)}$. Thus, from (16) and (14), we obtain

$$\ln \left(q_{VB}^* \left(\epsilon_{i(i+1)} \right) \right) = -\frac{1}{2} \left(\epsilon_{i(i+1)} - \mu_{i(i+1)} \right)^T \Xi_{i(i+1)}^{-1} \left(\epsilon_{i(i+1)} - \mu_{i(i+1)} \right) + \text{cst} \quad (17)$$

where

$$\Xi_{i(i+1)} = \left(J_{li}^T \Sigma_{lk}^{-1} J_{li} + P_{i(i+1)}^{-1} \right)^{-1} \quad (18)$$

and

$$\mu_{i(i+1)} = \Xi_{i(i+1)}^{-1} \left(J_{li}^T \Sigma_{lk}^{-1} e_{lk,i} - P_{i(i+1)}^{-1} r_{i(i+1)} \right) \quad (19)$$

with $e_{lk,i} = r_{lk} - \left(\sum_{j=l, j \neq i}^{k-1} J_{lj} \mu_{j(j+1)} \right)$. Therefore, for each random variable $\epsilon_{i(i+1)}$ ($i = 1, \dots, k-1$), the best approximated distribution is a Gaussian of the form:

$$q_{VB}^* \left(\epsilon_{i(i+1)} \right) = \mathcal{N}_{\mathbb{R}^p} \left(\epsilon_{i(i+1)}; \mu_{i(i+1)}, \Xi_{i(i+1)} \right) \quad \text{for } i = 1, \dots, k-1 \quad (20)$$

Let us note that if $i < l$, i.e. if the relative transformation $T_{i(i+1)}$ is not involved in the loop closure Z_{lk} , then

$$q_{VB}^*(\epsilon_{i(i+1)}) = \mathcal{N}_{\mathbb{R}^p}(\epsilon_{i(i+1)}; \mathbf{0}, \Xi_{i(i+1)} = P_{i(i+1)}) \quad \text{for } i < l \quad (21)$$

making our algorithm very efficient since a loop closure will only modify the relative transformations involved in that loop.

In theory, in order to obtain the values of $\{\mu_{i(i+1)}\}_{i=l, \dots, k-1}$ we should cycle through (19) for each relative transformation involved in the loop until convergence. However, if the linearization step (see Eq. (13)) is performed around the maximizer of (12), then $\mu_{i(i+1)} = \mathbf{0}$ for $i = l, \dots, k-1$. Thus in practice, for each new loop closure measurement, we first apply, the Gauss-Newton algorithm described in Sect. 4² in order to find the maximizer of (12) very efficiently. Then we only have to compute the covariances $\Xi_{i(i+1)}$ (see Eq. (18)) for $i = l, \dots, k-1$.

Finally, for each relative transformation, $q_{VB}^*(\epsilon_{i(i+1)})$ is a Gaussian with zero mean. Therefore, from Eq. (13), one can see that (up to a linear approximation) $q_{VB}^*(T_{i(i+1)})$ is a Gaussian distribution on Lie group (see Eq. (1)) of the form $\mathcal{N}_G(T_{i(i+1)}; \check{T}_{i(i+1)}, \Xi_{i(i+1)})$. Consequently, after having processed a new loop closure, our factorized approximation of the posterior has the following form:

$$p(\mathcal{X}_k | \mathcal{D}_{odo,k}, \mathcal{D}_{LC,k-1}, Z_{lk}) \approx Q_{VB}(\mathcal{X}_k) = \prod_{i=l}^{k-1} \mathcal{N}_G(T_{i(i+1)}; \check{T}_{i(i+1)}, \Xi_{i(i+1)}) \quad (22)$$

Detection of Outlier Loop Closure Through Validation Gating. So far, we have proposed an efficient way to process a new loop closure measurement assuming it was following the generative model 3. However, in practice, two places being perceived as the same usually produce a wrong loop closure. Consequently, detecting and removing these wrong loop closure measurements is crucial in order to perform motion averaging, especially for large scale problems where wrong loop closures are very likely to occur.

Since we continuously maintain an approximation of the posterior distribution, it is possible to detect wrong loop closure measurements through validation gating [26]. This approach consists in first computing the mean \bar{Z}_{lk} and covariance $\bar{\Sigma}_{lk}$ parameters of the following distribution:

$$p(Z_{lk} | \mathcal{D}_{odo,k}, \mathcal{D}_{LC,k-1}) \approx \mathcal{N}_G\left(Z_{lk}; \bar{Z}_{lk} = \prod_{i=l}^{k-1} \bar{T}_{i(i+1)}, \bar{\Sigma}_{lk} = \Sigma_{lk} + \sum_{i=l}^{k-1} J_{i(i+1)} P_{i(i+1)} J_{i(i+1)}^T\right) \quad (23)$$

and then testing w.r.t a threshold t whether or not the received measurement is likely to be an inlier:

$$\left\| \log_G^V \left(Z_{lk} \bar{Z}_{lk}^{-1} \right) \right\|_{\bar{\Sigma}_{lk}}^2 < t \quad (24)$$

² Equation (12) has the same form (up to a sign) as the cost function (4). Consequently, the highly efficient GN described in Sect. 4 can be applied to maximize (12).

In theory, t should be based on the p -value of the Chi-squared distribution. However, as we will see in the experiments, such a theoretical value is sometimes too restrictive, especially when processing real data where the covariance of the odometry and loop closure measurements are not very accurate and the assumption of mutually independent noises might be violated.

6 Experiments

We now evaluate experimentally, both on synthetic and real datasets, our novel online variational Bayesian motion averaging algorithm (a pseudo-code is proposed in supplementary material) against the state of the art algorithms LG-IEKF [5], COP-SLAM [12] and DCS [1] (which uses g^2o). To do so, we first compare both the accuracy and the computational time of these approaches on datasets which do not contain wrong loop closures, since COP-SLAM is not able to detect and remove wrong loop closures. The robustness of the different approaches is evaluated separately on datasets specifically dedicated to this task (see [23]). We finally present qualitative results on monocular visual SLAM and video mosaicking applications. In all these experiments, when dealing with synthetic datasets, the threshold t of the validation gating stage of our algorithm has been set to the χ^2 value with p degrees of freedom given by a p -value of 0.001. Otherwise, when dealing with real data, we empirically defined $t = 900$, which is much higher than the theoretical χ^2 value since the covariance of the odometry and loop closure measurements are usually not very accurate in this case and the assumption of mutually independent noises might be violated.

6.1 Evaluation of the Accuracy and the Computational Time

In this experiment, we consider a binocular 6D SLAM application (Lie group $SE(3)$) and use one synthetic sequence (Sphere) and two real sequences (originally from the KITTI dataset [15]) provided by the authors of [12]. The results for this experiment are given in Table 3 where we reported both the Root Mean Squared Error (RMSE) for the absolute positions as well as the computational time for our approach, COP-SLAM, LG-IEKF and g^2o . We provide the computational time both in C++ and Matlab because only a Matlab implementation of LG-IEKF is available.

Let us first note that we should not expect the RMSE of COP-SLAM and our approach to be as low as the RMSE of LG-IEKF and g^2o because these approaches do not try to summarize the past information with a small number of parameters at each time instant but keep all the past information (g^2o keeps all the past measurements while LG-IEKF maintains a full covariance matrix). However, one can see that our approach remains very accurate. Indeed, on the KITTI 02 sequence, our approach even obtains the same RMSE as LG-IEKF. On the contrary, COP-SLAM obtains much higher RMSE than our approach on every sequence. From the computational time point of view, as expected, both our approach and COP-SLAM are orders of magnitude faster than LG-IEKF

Table 3. Results for binocular 6D SLAM ($SE(3)$): In terms of RMSE (for the position), our approach is much closer to the solutions of both g^2o and LG-IEKF [5], compared to COP-SLAM [12]. In terms of computational time, our approach is orders of magnitude faster than both g^2o and LG-IEKF, while being only slightly slower than COP-SLAM. *Remark: for these experiments, wrong loop closures have been removed since COP-SLAM cannot cope with them. The robustness of our method w.r.t wrong loop closures is evaluated against LG-IEKF [5] and DCS [1] (which uses g^2o) in Sect. 6.2.*

	RMSE position (m)			Time (ms): C++ (left) & Matlab (right)					
	Sphere	KITTI 00	KITTI 02	Sphere		KITTI 00		KITTI 02	
Ours	2.1	2.7	13.6	971	136 000	65	8 000	29	4 000
[12]	6.0	3.8	19.7	350	N/a	7	N/a	2	N/a
[5]	0.8	2.0	13.6	N/a	3 340 000	N/a	80 000	N/a	52 000
g^2o	0.2	2.4	13.8	40 000	N/a	1336	N/a	693	N/a

and g^2o . Moreover, our approach is only slightly slower than COP-SLAM which is largely compensated by the fact that our approach has a much lower RMSE.

6.2 Evaluation of the Robustness

In this experiment, we employ the dataset provided by the authors of [23] which allows to evaluate the robustness of an approach to wrong loop closures on a planar visual SLAM application (Lie group $SE(2)$). The results and details regarding this experiment are provided in the supplementary material. Our approach surprisingly achieved exactly the same precision and recall as both LG-IEKF and DCS. This is a remarkable result since these two algorithms are not designed to perform online large scale estimation and are consequently much slower than our approach (see Table 3).

6.3 Additional Experiments

In Fig. 1, we present results for monocular visual SLAM (Lie group $Sim(3)$) on sequence 13 of the KITTI dataset. The details regarding this experiment are provided in the supplementary material. However, one can see that the trajectory estimated with our approach is visually much closer to the result of [33] (which employs a Lidar) than the trajectory estimated with COP-SLAM. Results on sequence 15 of the KITTI dataset as well as results for video mosaicking (Lie group $Sim(3)$) are also provided as supplementary material.

7 Conclusion and Future Work

In this paper, we proposed a novel filter dedicated to online motion averaging for large scale problems. We have shown that using a relative parametrization of the absolute transformations produces a posterior distribution that can be

efficiently approximated assuming independent relative transformations. Based on this representation, we demonstrated that it is possible to obtain an accurate, efficient and robust filter by employing a variational Bayesian approach.

The performances of our novel algorithm were extensively evaluated against the state of the art algorithm COP-SLAM [12]. Actually, our approach achieved a significantly lower RMSE than COP-SLAM while being only slightly slower.

Since COP-SLAM cannot detect wrong loop closures, we also compared the robustness of our filter against LG-IEKF [5] and DCS [1]. In this context, our approach surprisingly achieved the same robustness as these algorithms. This is a remarkable result since our approach is designed to perform online large scale estimation and, consequently, is orders of magnitude faster than both LG-IEKF and DCS.

As future work, we plan to exploit the high efficiency of our filter to build a multi-hypothesis filter. This would prevent failures, such as those described in the supplementary material, to which LG-IEKF, DCS and the approach presented in this paper are prone due to the fact that they are forced to take a decision when a loop closure measurement is available and cannot wait until new evidence is received.

Acknowledgments. The author would like to thank the reviewers and Cornelia Vacar for their valuable help as well as Christopher Zach for fruitful discussions and feedback.

References

1. Agarwal, P., Tipaldi, G.D., Spinello, L., Stachniss, C., Burgard, W.: Robust map optimization using dynamic covariance scaling. In: 2013 IEEE International Conference on Robotics and Automation (ICRA), pp. 62–69. IEEE (2013)
2. Agarwal, S., Mierle, K., et al.: Ceres solver. <http://ceres-solver.org>
3. Bishop, C.: Pattern Recognition and Machine Learning. Springer, New York (2006)
4. Boumal, N., Singer, A., Absil, P.A.: Robust estimation of rotations from relative measurements by maximum likelihood. In: Proceedings of the 52nd Conference on Decision and Control, CDC (2013)
5. Bourmaud, G., Mégret, R., Giremus, A., Berthoumieu, Y.: Global motion estimation from relative measurements in the presence of outliers. In: Cremers, D., Reid, I., Saito, H., Yang, M.-H. (eds.) ACCV 2014. LNCS, vol. 9007, pp. 366–381. Springer, Heidelberg (2015)
6. Caballero, F., Merino, L., Ferruz, J., Ollero, A.: Homography based Kalman filter for mosaic building. applications to UAV position estimation. In: 2007 IEEE International Conference on Robotics and Automation, pp. 2004–2009. IEEE (2007)
7. Carlone, L., Aragues, R., Castellanos, J.A., Bona, B.: A fast and accurate approximation for planar pose graph optimization. *Int. J. Robot. Res.* (2014)
8. Chatterjee, A., Govindu, V.: Efficient and robust large-scale rotation averaging. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 521–528 (2013)
9. Chirikjian, G.S.: Stochastic Models, Information Theory, and Lie Groups, vol. 2. Springer, Boston (2012)

10. Choi, S., Zhou, Q.Y., Koltun, V.: Robust reconstruction of indoor scenes. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5556–5565. IEEE (2015)
11. Dellaert, F.: GTSAM - The Georgia Tech Smoothing and Mapping Library. <https://collab.cc.gatech.edu/borg/gtsam/>
12. Dubbelman, G., Browning, B.: COP-SLAM: closed-form online pose-chain optimization for visual SLAM. *IEEE Trans. Robot.* **31**(5), 1194–1213 (2015)
13. Engel, J., Schöps, T., Cremers, D.: LSD-SLAM: large-scale direct monocular SLAM. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part II. LNCS, vol. 8690, pp. 834–849. Springer, Heidelberg (2014)
14. Estrada, C., Neira, J., Tardós, J.D.: Hierarchical SLAM: real-time accurate mapping of large environments. *IEEE Trans. Robot.* **21**(4), 588–596 (2005)
15. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: the KITTI dataset. *Int. J. Robot. Res.* (IJRR) (2013)
16. Govindu, V.M.: Combining two-view constraints for motion estimation. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, vol. 2, pp. 218–225. IEEE (2001)
17. Govindu, V.M.: Lie-algebraic averaging for globally consistent motion estimation. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, vol. 1, pp. 684–691. IEEE (2004)
18. Govindu, V.M.: Robustness in motion averaging. In: Narayanan, P.J., Nayar, S.K., Shum, H.-Y. (eds.) ACCV 2006. LNCS, vol. 3852, pp. 457–466. Springer, Heidelberg (2006)
19. Grisetti, G., Stachniss, C., Burgard, W.: Nonlinear constraint network optimization for efficient map learning. *IEEE Trans. Intell. Transp. Syst.* **10**(3), 428–439 (2009)
20. Hartley, R., Trunpf, J., Dai, Y., Li, H.: Rotation averaging. *Int. J. Comput. Vis.* **103**(3), 267–305 (2013)
21. Kümmerle, R., Grisetti, G., Strasdat, H., Konolige, K., Burgard, W.: g2o: a general framework for graph optimization. In: 2011 IEEE International Conference on Robotics and Automation (ICRA), pp. 3607–3613. IEEE (2011)
22. Latif, Y., Cadena, C., Neira, J.: Robust loop closing over time for pose graph SLAM. *Int. J. Robot. Res.* (2013)
23. Latif, Y., Cadena, C., Neira, J.: Robust graph slam back-ends: a comparative analysis. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014), pp. 2683–2690. IEEE (2014)
24. Meidow, J.: Efficient video mosaicking by multiple loop closing. In: Stilla, U., Rottensteiner, F., Mayer, H., Jutzi, B., Butenuth, M. (eds.) PIA 2011. LNCS, vol. 6952, pp. 1–12. Springer, Heidelberg (2011). doi:[10.1007/978-3-642-24393-6_1](https://doi.org/10.1007/978-3-642-24393-6_1)
25. Meidow, J.: Efficient multiple loop adjustment for computer vision tasks. *Photogrammetrie-Fernerkundung-Geoinformation* **2012**(5), 501–510 (2012)
26. Ramachandra, K.: Kalman Filtering Techniques for Radar Tracking. CRC Press, Boca Raton (2000)
27. Roberts, R., Sinha, S.N., Szeliski, R., Steedly, D.: Structure from motion for scenes with large duplicate structures. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3137–3144. IEEE (2011)
28. Sündenhauf, N., Protzel, P.: Switchable constraints for robust pose graph SLAM. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1879–1884. IEEE (2012)

29. Tron, R., Daniilidis, K.: Statistical pose averaging with non-isotropic and incomplete relative measurements. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part V. LNCS, vol. 8693, pp. 804–819. Springer, Heidelberg (2014)
30. Zach, C.: Robust bundle adjustment revisited. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part V. LNCS, vol. 8693, pp. 772–787. Springer, Heidelberg (2014)
31. Zach, C., Irschara, A., Bischof, H.: What can missing correspondences tell us about 3D structure and motion? In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8. IEEE (2008)
32. Zach, C., Klopschitz, M., Pollefeys, M.: Disambiguating visual relations using loop constraints. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1426–1433. IEEE (2010)
33. Zhang, J., Singh, S.: LOAM: lidar odometry and mapping in real-time. In: Robotics: Science and Systems Conference (RSS), Berkeley, CA, July 2014