Spectral Synchronization of Multiple Views in SE(3)

SIAM J. IMAGING SCIENCES
Vol. xx, pp. x–x

Spectral Synchronization of Multiple Views in SE(3)

Federica Arrigoni[†], Beatrice Rossi[‡], and Andrea Fusiello[†]

Abstract. This paper addresses the problem of rigid-motion synchronization (a.k.a. motion averaging) in the Special Euclidean Group SE(3), which finds application in structure-from-motion and registration of multiple 3D point-sets. After relaxing the geometric constraints of rigid motions, we derive a simple closed-form solution based on a spectral decomposition, which is then projected onto SE(3). Our formulation is extremely efficient, as rigid-motion synchronization is cast to an eigenvalue decomposition problem. Robustness to outliers is gained through Iteratively Reweighted Least Squares. Besides providing a theoretically appealing solution, since our method recovers at the same time both rotations and translations, we demonstrate through experimental results that our approach is significantly faster than the state of the art, while providing accurate estimates of rigid motions.

Key words. rigid-motion synchronization, motion averaging, motion registration, spectral decomposition, iteratively reweighted least squares, structure from motion, multiple point-set registration

AMS subject classifications. 68T45

1. Introduction.

In this paper we address the *rigid-motion synchronization* problem, which consists in recovering the *absolute orientations* (i.e. position and angular attitude expressed in an absolute coordinate system) of a number of three-dimensional reference frames, starting from a redundant set of *relative orientations* (i.e. rigid motions of one frame relative to another). Such relative information is usually corrupted by a diffuse noise, in addition to sparse gross errors (outliers).

This problem, also known as *registration* or *averaging*[1], appears in the context of *structure-from-motion* – where the reference frames are those attached to pinhole cameras, and *multiple point-set registration* – where the goal is to compute rigid motions that bring multiple 3D point-sets into alignment. In both cases reference frames can be also identified with *views*.

1.1. Problem Definition.

Let us consider a finite simple directed graph $\vec{\mathcal{G}} = (\mathcal{V}, \mathcal{E})$, where vertices correspond to reference frames and edges to the available relative measurements. Both vertices and edges are labelled with rigid motions representing absolute and relative orientations[2], respectively. Rigid motions, or *direct isometries*, are elements of the Special Euclidean Group SE(3), which is the semi-direct product of the Special Orthogonal Group SO(3) with $\mathbb{R}^3$.

As a matrix group, SE(3) is a subgroup of the General Linear Group GL(4), thus inverse and composition of rigid motions reduce to matrix operations. Accordingly, each absolute

[†]DPIA - University of Udine, Via delle Scienze, 208 - Udine, Italy (arrigoni.federica@spes.uniud.it, andrea.fusiello@uniud.it).
[‡]AST Lab - STMicroelectronics, Via Olivetti, 2 - Agrate Brianza, Italy (beatrice.rossi@st.com).
[1]The term "averaging" in this paper is used in the Computer Vision acceptation (see e.g. [13]).
[2]As customary in Photogrammetry, the name "orientation" refers to angular attitude *and* position.

orientation is described by a homogeneous transformation

$$M_i = \begin{pmatrix} R_i & \mathbf{t_i} \\ \mathbf{0}^\mathsf{T} & 1 \end{pmatrix} \in \mathrm{SE}(3) \tag{1.1}$$

where $R_i \in \mathrm{SO}(3)$ and $\mathbf{t_i} \in \mathbb{R}^3$ represent the rotation and translation components of the i-th transformation. Similarly, each relative orientation can be expressed as

$$M_{ij} = \begin{pmatrix} R_{ij} & \mathbf{t_{ij}} \\ \mathbf{0}^\mathsf{T} & 1 \end{pmatrix} \in \mathrm{SE}(3) \tag{1.2}$$

where $R_{ij} \in \mathrm{SO}(3)$ and $\mathbf{t_{ij}} \in \mathbb{R}^3$ represent the transformation between frames i and j. The labelling of the edges is such that if $(i,j) \in \mathcal{E}$ then $(j,i) \in \mathcal{E}$ and $M_{ji} = M_{ij}^{-1}$. Hence, w.l.o.g. we will consider $\mathcal{G}$, the undirected version of $\vec{\mathcal{G}}$. The link between absolute and relative orientations is encoded by the *compatibility* constraint

$$M_{ij} = M_i M_j^{-1}. \tag{1.3}$$

The rigid-motion synchronization problem requires to compute an estimate of the vertex labels M_i given enough measurements of the ratios $M_i M_j^{-1}$. This is an instance of the more general *synchronization* problem that consists in finding group elements from noisy measurements of their ratios [20, 50], where the group is that of rigid motions. The solution is defined up to a global direct isometry, in the sense that if $M_i \in \mathrm{SE}(3)$ satisfies (1.3) for all i, then also $M_i N$ satisfies (1.3) for any (fixed) $N \in \mathrm{SE}(3)$.

Clearly, a solution exists only if the graph is connected. The minimum number of relative measurements is $|\mathcal{V}| - 1$, which makes $\mathcal{G}$ a tree. In this case every vertex can be labelled with its absolute orientation by simply propagating Equation (1.3) along the tree, starting from the root labelled with the identity. In this case, however, there is no remedy to error propagation: the error affecting a relative measurement propagates down to the leaves of the tree without compensation. In the synchronization problem, instead, the goal is to exploit *redundant* relative measurements in a global fashion to improve the final estimate.

The rigid-motion synchronization problem can be tackled directly in SE(3), as we propose in this paper, or by breaking the problem into rotation and translation and solving the two sub-problems separately, according to the respective compatibility constraints

$$R_{ij} = R_i R_j^\mathsf{T} \tag{1.4}$$

$$\mathbf{t_{ij}} = -R_i R_j^\mathsf{T} \mathbf{t_j} + \mathbf{t_i}. \tag{1.5}$$

The latter approach is widely adopted in the structure-from-motion literature, since relative translations are only known as *directions*, i.e. the magnitude is unknown, thus relative measurements can not be properly represented as elements of SE(3). A wide overview of existing solutions is provided in the next section.

1.2. Broader Context and Related Work. The rigid-motion synchronization problem has a wide range of applications in Computer Vision, including structure-from-motion and multiple point-set registration. In both cases the relative orientations can be estimated through

standard techniques, thus the problem is to average them in a consistent manner in order to recover the absolute orientations. In the structure-from-motion case, given a set of matching points between two images, the relative motion (up to scale) between the views can be recovered by first estimating the essential matrix and then computing its Singular Value Decomposition [28]. In the case of 3D registration, the relative orientation between two point sets is commonly computed through the Iterative Closest Point Algorithm [11], which alternates the correspondence step – in which each point is matched with its closest neighbour under the current transformation – with the orientation step – in which the rotation and translation that align the points in a common reference frame are computed by solving an absolute orientation problem. Other applications of rigid-motion synchronization include sensor network localization [15], cryo-electron microscopy [51] and simultaneous localization and mapping (SLAM) [46].

In the literature on multiple point-set registration, the origins of rigid-motion synchronization can be traced back to the *frame space* methods [19, 48] that optimize the internal coherence of the network of rotations and translations applied to the local coordinate frames, as opposed to solutions that optimize a cost function depending on the distance between corresponding points (e.g. [43, 9, 44, 55]).

In the structure-from-motion literature, *global* methods, that first solve for the motion by optimizing the network of relative orientations and leave the 3D structure recovery at the end, are fairly recent (e.g. [36]), although the origins of these approaches can be traced back to [22]. The majority of global techniques do not address rigid-motion synchronization as such but break the problem into rotation and translation, and solve the two problems separately.

For what regards *rotation synchronization*, a theoretical analysis of the problem is reported in [27]. The absolute rotations can be recovered by using the quaternion representation of SO(3), as done in [22, 19], or by distributing the error over cycles in the graph of neighbouring views [48]. In [36] the problem is cast as the optimization of an objective function based on the ℓ_2-norm of the compatibility error between relative estimates and unknown absolute rotations, and this approach is extended in [50, 3] where approximate solutions are computed either via spectral decomposition (EIG) or semidefinite programming (SDP). Recent works on rotation synchronization [26, 62, 13, 6] focus on introducing robustness to outliers directly in the cost function. In [26] a cost based on the ℓ_1-norm is used to average relative rotations, where each absolute rotation is updated in turn using the Weiszfeld algorithm. The sum of unsquared deviations is proposed in [62] as a more robust consistency error. Chatterjee et al. in [13] exploit the Lie-group structure of rotations, and develop an L1-IRLS algorithm by combining ℓ_1-averaging in the tangent space with Iteratively Reweighted Least Squares (IRLS). In [6] the rotation synchronization problem is reformulated in terms of "low-rank and sparse" matrix decomposition, and an efficient algorithm – called R-GoDec – is developed to compute such decomposition. Alternatively, robustness can be achieved by removing outliers before performing rotation synchronization [65, 17, 40, 7].

As for *translation recovery* methods, a discriminating factor relevant to our analysis is whether they work in frame-space (e.g. [22, 12, 42, 41, 38, 32]), or they leverage on point correspondences as well (e.g. [35, 33, 14, 64]). Only the former fit the general statement of the rigid-motion synchronization problem, which is agnostic on the data that generated relative orientation measurements. In [22] absolute translations are computed as the least squares

solution of a linear system of equations involving relative orientations, while in [12] a fast spectral solution to translation recovery is proposed by reformulating the problem in terms of graph embedding. The authors of [42] first compute relative directions through a robust subspace estimation and then derive absolute translations using a semidefinite relaxation (SDR). The method presented in [41] is based on a "least unsquared deviations" (LUD) formulation, which gives rise to a convex program robust to outlier directions. In [38] relative directions are computed through an a-contrario trifocal tensor estimation, and then absolute translations are recovered by using an ℓ_∞ formulation. In [32] a linear solution which minimizes a geometric error in triplets of views is presented, and such approach is extended in [16] by considering feature tracks. All these techniques recover absolute translations from pairwise directions $\mathbf{t_{ij}}/||\mathbf{t_{ij}}||$, i.e. they solve a *bearing-only localization* problem [66]. Theoretical conditions under which the solution is unique (up to a global translation and scale) are more complex than the cases of rotation and rigid-motion synchronization– where the measurement graph is simply required to be connected – and they refer to the concept of *parallel rigidity* [42, 63].

A different approach for rigid-motion synchronization is followed in [23, 57, 46] where rotations and translations are *jointly* considered as elements of SE(3). Govindu in [23] exploits the Lie-group structure of SE(3) and uses an iterative scheme in which at each step the absolute orientations are approximated by averaging relative orientations in the tangent space. In [24] robustness is added to the original technique through random sampling in the measurement graph. Originally proposed in the structure-from-motion framework, the same technique was also applied to multiple point-set registration [25] and simultaneous localization and mapping [2]. The authors of [57] represent rigid motions as dual quaternions, and propose a graph diffusion algorithm where each absolute orientation is updated in turn through linear or geodesic averaging. In [46] a geometric error is adopted and the rigid motion constraints are relaxed by considering the convex hull of SE(3), which has a semi-definite representation [47]. This results in a convex cost function which is (globally) minimized through the interior-point method. Such a solution is then improved by (locally) minimizing the objective function over SE(3) through the Levenberg-Marquardt algorithm.

A related approach is adopted in [59, 60] where rotations and translations are initialized separately and then they are jointly refined through Riemannian gradient descent. In [60] the scales of relative translations are enforced to be positive – hence this method is referred to as "constrained least squares" (CLS) – while in [59] non-isotropic noise and incomplete relative orientations are taken into account through the use of covariance matrices.

Other techniques [39, 18, 45] compute at the same time both rotations and translations by minimizing an objective function involving point correspondences. At the intersection between frame-based and point-based methods is the formulation in [49] where 3D points are used to compute a second-order approximation of the cost function, but they are not involved in subsequent computations.

1.3. Our Contribution. In this paper we propose a novel method for rigid-motion synchronization in SE(3). After relaxing the geometric constraints of rigid motions, we derive a closed-form solution – based on a spectral decomposition – which is then projected onto SE(3). This can be regarded as an "extrinsic calculation", for the rigid motion constraints are relaxed to compute the solution, as in [61, 46]. This approach is extremely efficient, as

rigid-motion synchronization is cast to an eigenvalue decomposition of a $4|\mathcal{V}| \times 4|\mathcal{V}|$ matrix.

Our method can be seen as the extension to SE(3) of the spectral synchronization proposed in [50] for SO(2) and generalized to SO(3) in [51, 3]. Thanks to the introduction of weights, this simple matrix formulation naturally copes with missing measures, and it can be embedded into an IRLS scheme in order to handle rogue measurements. The same idea appeared also in [10] in the context of shape alignment, though without considering missing relative orientations and outliers. Further theoretical analysis is reported in [54] where the authors address the synchronization problem over the General Linear Group GL(N). They describe a (theoretical) drawback of our approach applied to GL(N) and propose a new formulation that overcomes it. That counterexample, however, does not work in SE(3) and simulations reported in [54] show that in a realistic scenario (low dimension and many reference frames) the differences between the two methods are negligible.

An exhaustive set of experiments on synthetic and real data show that our method compares favourably with the state of the art in terms of accuracy, and it is the fastest solution among all the analysed techniques.

The paper is organized as follows. Section 2 describes our spectral solution to rigid-motion synchronization in SE(3). In Section 3 our method is compared to state-of-the-art algorithms developed both in the context of structure-from-motion and 3D registration. The conclusions are presented in Section 4.

2. Our Method.

In this section we introduce our approach for solving the rigid-motion synchronization problem in SE(3). In Section 2.1 we describe properties that hold when all the relative information is exact, necessary to define our technique. Then we derive our spectral solution to rigid-motion synchronization (Section 2.2). In Section 2.3 our method is embedded into an IRLS framework in order to handle outliers among relative orientations. Finally, Section 2.4 briefly presents the extension of our method to SE(N).

2.1. The Exact Case.

The absolute orientations can be recovered from (1.3) – up to a global rigid motion – if we express it in a useful equivalent way that takes into account all the relative information at once. For simplicity of exposition, we first consider the case where all the pairwise measures are available.

Let $X \in \mathbb{R}^{4n\times 4n}$ denote the block-matrix containing the ideal (noise free) relative orientations and let $M \in \mathbb{R}^{4n\times 4}$ be the stack of the absolute orientations, namely

$$M = \begin{bmatrix} M_1 \\ M_2 \\ \dots \\ M_n \end{bmatrix}, \quad X = \begin{pmatrix} I_4 & M_{12} & \dots & M_{1n} \\ M_{21} & I_4 & \dots & M_{2n} \\ \dots & & & \dots \\ M_{n1} & M_{n2} & \dots & I_4 \end{pmatrix} \tag{2.1}$$

where I_4 indicates the 4×4 identity matrix and $n = |\mathcal{V}|$. If $M^{-\flat} \in \mathbb{R}^{4\times 4n}$ denotes the concatenation by rows of the inverse of absolute orientations, i.e. $M^{-\flat} = \begin{bmatrix} M_1^{-1} & M_2^{-1} & \dots & M_n^{-1} \end{bmatrix}$, then the compatibility constraint turns into

$$X = MM^{-\flat} \tag{2.2}$$

and hence $\operatorname{rank}(X) = 4$. Note that here X is not symmetric positive semidefinite, in contrast to the case of SO(3). Since $M^{-\flat}M = nI_4$, we obtain

$$XM = nM \tag{2.3}$$

which means that – in the absence of noise – the columns of M are 4 (independent) eigenvectors of X associated to the eigenvalue n. Note that, since X has rank 4, all the other eigenvalues are zero, thus n is also the largest eigenvalue of X. Equation (2.3) is equivalent to

$$(nI_{4n} - X)M = 0, \tag{2.4}$$

thus the columns of M form a basis for the 4-dimensional null-space of $L = (nI_{4n} - X)$.

Conversely, any basis U for $\operatorname{null}(L)$ will not coincide with M in general, since it will not be composed of rigid motions. Specifically, it will not coincide with [0 0 0 1] in every fourth row. In order to recover M from U it is sufficient to choose a different basis for $\operatorname{null}(L)$ that satisfies such constraint, which can be found by taking a suitable linear combination of the columns of U. More precisely, let $B \in \mathbb{R}^{n\times 4n}$ be the 0/1-matrix such that $BU \in \mathbb{R}^{n\times 4}$ consists of the rows of U with indices multiple of four. The coefficients $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathbb{R}^4$ of the linear combination are solution of

$$BU\boldsymbol{\alpha} = \mathbf{0}, \quad BU\boldsymbol{\beta} = \mathbf{1} \tag{2.5}$$

where the first equation has a three-dimensional solution space. Let $\boldsymbol{\alpha_1}, \boldsymbol{\alpha_2}, \boldsymbol{\alpha_3}$ be a basis for the null-space of BU. Thus the columns of M corresponding to rotations coincide (up to a permutation) with $[U\boldsymbol{\alpha_1}, U\boldsymbol{\alpha_2}, U\boldsymbol{\alpha_3}]$ and M is recovered as $M = U[\boldsymbol{\alpha_1}, \boldsymbol{\alpha_2}, \boldsymbol{\alpha_3}, \boldsymbol{\beta}]$. Note that this post-processing on the eigenvectors is not required for the spectral method in SO(3) [3], since any orthogonal basis for the null-space of L coincides (up to a permutation) with the stack of the absolute rotations.

Dealing with missing data. We now consider the case of missing data, in which the graph $\mathcal{G}$ is connected but not complete. Let $A \in \mathbb{R}^{n\times n}$ be the adjacency matrix of $\mathcal{G}$, i.e. the symmetric matrix such that $A_{ij} = 1$ if $(i, j) \in \mathcal{E}$ and $A_{ij} = 0$ otherwise, and let $D \in \mathbb{R}^{n\times n}$ be the degree matrix of $\mathcal{G}$, i.e. the diagonal matrix such that $D_{ii} = \sum_j A_{ij}$ (D_{ii} contains the degree of node i).

In this situation missing pairwise measures correspond to zero blocks in X, i.e. the available relative information is represented by $(A \otimes \mathbb{1}_{4\times 4}) \circ X$, where $\otimes$ denotes the Kronecker product and $\circ$ denotes the Hadamard product. The adjacency matrix A gets "inflated" to a 4×4-block structure by the Kronecker product with $\mathbb{1}_{4\times 4}$ (a matrix filled by ones), to match the block structure of X. Being a matrix of 0/1, the effect of its entry-wise product with X is to zero the unspecified blocks of X and leave the others unchanged.

It can be seen that Equation (2.3) generalizes to

$$((A \otimes \mathbb{1}_{4\times 4}) \circ X)M = (D \otimes I_4)M. \tag{2.6}$$

Indeed, the i-th block-row in (2.6) is

$$\sum_{j \text{ s.t. } (i,j)\in\mathcal{E}} M_{ij}M_j = D_{ii}M_i \tag{2.7}$$

which is true since $M_{ij} = M_i M_j^{-1}$. Therefore the columns of M are 4 (independent) eigenvectors of

$$P = (D \otimes I_4)^{-1}((A \otimes \mathbb{1}_{4\times 4}) \circ X) \tag{2.8}$$

associated to the eigenvalue 1. Note that the incomplete data matrix $(A \otimes \mathbb{1}_{4\times 4}) \circ X$ will have full rank in general, thus 1 is not the unique non-zero eigenvalue of P, in contrast to the case of Equation (2.3). However, it can be shown that 1 is the largest eigenvalue of P.

Proposition 2.1. *The matrix P defined in (2.8) has real eigenvalues. The largest eigenvalue is 1 and it has multiplicity 4.*

Proof. By substituting the compatibility constraint (2.2) into the definition of P we get:

$$P = (D \otimes I_4)^{-1}((A \otimes \mathbb{1}_{4\times 4}) \circ (MM^{-\flat})). \tag{2.9}$$

It easy to observe that

$$P = (D \otimes I_4)^{-1}\mathrm{bdiag}(M)(A \otimes I_4)\mathrm{bdiag}(M)^{-1} \tag{2.10}$$

where $\mathrm{bdiag}(M)$ produces a $4n \times 4n$ block-diagonal matrix with 4×4 blocks $M_1, \ldots, M_n$ along the diagonal. Indeed, the (i,j)-th block in $(A \otimes \mathbb{1}_{4\times 4}) \circ (MM^{-\flat})$ is

$$\begin{cases} \mathbb{1}_{4\times 4} \circ (M_i M_j^{-1}) & \text{if } (i,j) \in \mathcal{E} \\ 0 \quad \text{if } (i,j) \notin \mathcal{E} \end{cases} \tag{2.11}$$

while the (i,j)-th block in $\mathrm{bdiag}(M)(A \otimes I_4)\mathrm{bdiag}(M)^{-1}$ is

$$\begin{cases} M_i I_4 M_j^{-1} & \text{if } (i,j) \in \mathcal{E} \\ 0 \quad \text{if } (i,j) \notin \mathcal{E}. \end{cases} \tag{2.12}$$

Moreover we observe that the diagonal matrix $(D \otimes I_4)^{-1}$ commutes with $\mathrm{bdiag}(M)$, since each 4×4 block along its diagonal is multiple of the identity matrix. Thus we obtain

$$P = \mathrm{bdiag}(M)(D \otimes I_4)^{-1}(A \otimes I_4)\mathrm{bdiag}(M)^{-1} \tag{2.13}$$

$$= \mathrm{bdiag}(M)((D^{-1}A) \otimes I_4)\mathrm{bdiag}(M)^{-1} \tag{2.14}$$

which means that the matrix P defined in (2.8) is similar to the matrix $(D^{-1}A) \otimes I_4$, i.e. they have the same eigenvalues. Thus the thesis reduces to analyse the eigenvalues of $(D^{-1}A) \otimes I_4$. The matrix $D^{-1}A$, a.k.a. the *transition matrix* of the graph $\mathcal{G}$, has real eigenvalues since it is similar to the symmetric matrix $N = D^{-1/2}AD^{1/2}$. It follows from the Perron-Frobenius theorem [37] that the largest eigenvalue of $D^{-1}A$ is 1 and it has multiplicity 1, assuming that $\mathcal{G}$ is connected. Since the eigenvalues of the Kronecker product of two matrices are the product of the eigenvalues of the matrices, we conclude that the largest eigenvalue of $(D^{-1}A) \otimes I_4$ is 1 and it has multiplicity 4. ■

The proof of Proposition 2.1 has pointed out that – provided that X is decomposable as $X = MM^{-\flat}$ – the matrix P has a particular structure that yields real eigenvalues, although

it is not symmetric. In particular, the eigenvalues of P do not depend on the measured data, i.e. the relative orientations, but they depend *only* on the structure of the measurement graph $\mathcal{G}$ (through the matrices D and A).

We now show that Equation (2.6) can also be expressed as a null-space problem. Let us rewrite it as

$$((D \otimes I_4) - (A \otimes \mathbb{1}_{4\times 4}) \circ X)M = 0. \tag{2.15}$$

which generalizes Equation (2.4). In fact, if $\mathcal{G}$ is complete then $D = (n-1)I_n$ and $A = \mathbb{1}_{n\times n} - I_n$, hence we get Equation (2.4). Thus the columns of M belong to the null-space of

$$L = (D \otimes I_4) - (A \otimes \mathbb{1}_{4\times 4}) \circ X. \tag{2.16}$$

It can be shown that such null-space is 4-dimensional, as happens in the case of a complete graph. Before doing that, let us observe that the matrix $D \otimes I_4$ coincides with $(D \otimes \mathbb{1}_{4\times 4}) \circ X$, since X has identity blocks along its diagonal and $D \otimes \mathbb{1}_{4\times 4}$ is block-diagonal. By using the distributive property of the involved products we obtain an equivalent expression for L

$$L = ((D - A) \otimes \mathbb{1}_{4\times 4}) \circ X. \tag{2.17}$$

where the matrix $(D - A)$ (a.k.a. the *Laplacian* matrix of the graph $\mathcal{G}$) gets "inflated" to a 4×4-block structure by the Kronecker product with $\mathbb{1}_{4\times 4}$, to match the block structure of X.

Proposition 2.2. *The matrix L defined in (2.17) has a 4-dimensional null-space.*

Proof. By using the compatibility constraint (2.2), we can express L as

$$L = ((D - A) \otimes \mathbb{1}_{4\times 4}) \circ (MM^{-\flat}). \tag{2.18}$$

It can be easily verified that

$$L = \mathrm{bdiag}(M)((D - A) \otimes I_4)\mathrm{bdiag}(M)^{-1} \tag{2.19}$$

which means that the matrix L defined in (2.17) is similar to the matrix $(D - A) \otimes I_4$, thus they have the same rank. The matrix $D - A$, a.k.a. the *Laplacian matrix* of the graph $\mathcal{G}$, has rank $n - 1$ assuming that $\mathcal{G}$ is connected. Since the rank of the Kronecker product of two matrices is the product of the rank of the matrices, we obtain

$$\mathrm{rank}(L) = \mathrm{rank}(D - A)\ \mathrm{rank}(I_4) = 4n - 4 \tag{2.20}$$

which implies that the null-space of L is 4-dimensional. ■

Please note that in practice one cannot measure L from Equation (2.17), because the full X is not available; in fact only the product $(A \otimes \mathbb{1}_{4\times 4}) \circ X$ is available. Therefore in practice L will be derived from Equation (2.16).

Weighted graph. In some applications we are given non-negative weights w_{ij} that reflect the reliability of the pairwise measurements. In other words, $\mathcal{G}$ is a weighted graph with real weights, stored in the symmetric adjacency matrix $A = [w_{ij}]$. Accordingly, the degree matrix D of the weighted graph is defined as $D_{ii} = \sum_j w_{ij}$. Equations (2.6) and (2.15) still hold with these definitions, thus our spectral method extends straightforwardly to *weighted* rigid-motion synchronization.

2.2. Dealing with Noise. We now consider the case where the relative measures are corrupted by noise, hence they do not satisfy equations (1.3) and (2.15) exactly. Thus the goal is to recover the absolute orientations such that they are "maximally compatible" with the available relative information.

Let $\widehat{M}_{ij} \in \mathrm{SE}(3)$ denote an estimate of the theoretical relative orientation $M_{ij} \in \mathrm{SE}(3)$. Hereafter we will consistently use the hat accent to denote noisy measurements. Rigid-motion synchronization can be formalized as the problem of minimizing the (weighted) residuals of the compatibility constraint (1.3), namely

$$\min_{M_i \in \mathrm{SE}(3)} \sum_{(i,j)} w_{ij} \left\| \widehat{M}_{ij} - M_i M_j^{-1} \right\|_F^2 \tag{2.21}$$

where the Frobenius norm $||\cdot||$ defines a left-invariant metric on SE(3), or, more generally, as the problem of minimizing the following geometric error

$$\min_{R_i \in \mathrm{SO}(3), \mathbf{t_i} \in \mathbb{R}^3} \sum_{(i,j)} \alpha_{ij} \left\| \widehat{R}_{ij} - R_i R_j^{\mathsf{T}} \right\|_F^2 + \beta_{ij} \left\| \widehat{\mathbf{t}}_{\mathbf{ij}} - \mathbf{t_i} + R_i R_j^{\mathsf{T}} \mathbf{t_j} \right\|_2^2 \tag{2.22}$$

where $\alpha_{ij}, \beta_{ij} \geq 0$. Note that this is a non-convex optimization problem that needs to be solved iteratively through e.g. the Levenberg-Marquardt algorithm. Since convergence to a global minimum is not guaranteed, current research puts efforts in formulating tractable approaches that solve the problem approximately but accurately, so as to provide a good starting point for a local refinement.

In this spirit, we consider an algebraic cost function that measures the residuals (in the Frobenius norm sense) of Equation (2.15), namely

$$\min_{M \in \mathrm{SE}(3)^n} \left\| \widehat{L} M \right\|_F^2 \tag{2.23}$$

where $\widehat{L} = (D \otimes I_4) - (A \otimes \mathbb{1}_{4\times 4}) \circ \widehat{X}$, and $\widehat{X}$ denotes a noisy version of the ideal matrix X, which contains the measured relative orientations $\widehat{M}_{ij} \in \mathrm{SE}(3)$. By using Equation (2.7) and expressing the squared Frobenius norm in (2.23) as the sum of the squared Frobenius norm over each 4×4 block, we get that such a problem coincides with

$$\begin{aligned} &\min_{M_i \in \mathrm{SE}(3)} \sum_{i=1}^{n} \left\| \sum_{j=1}^{n} w_{ij} \left(\widehat{M}_{ij} M_j - M_i \right) \right\|_F^2 = \\ &\min_{\substack{R_i \in \mathrm{SO}(3) \\ \mathbf{t_i} \in \mathbb{R}^3}} \sum_{i=1}^{n} \left\| \sum_{j=1}^{n} w_{ij} \left(\widehat{R}_{ij} R_j - R_i \right) \right\|_F^2 + \sum_{i=1}^{n} \left\| \sum_{j=1}^{n} w_{ij} \left(\widehat{\mathbf{t}}_{\mathbf{ij}} - \mathbf{t_i} + \widehat{R}_{ij} \mathbf{t_j} \right) \right\|_2^2 . \end{aligned} \tag{2.24}$$

For each node, this cost function considers the edges incident to that node, it sums the (weighted) residuals of the compatibility constraints, and takes the squared Frobenius norm. These norms are then summed up over all the nodes.

Problem (2.23) is difficult to solve since the feasible set $\mathrm{SE}(3)^n = \mathrm{SE}(3)\times\cdots\times\mathrm{SE}(3)$ is non-convex. In order to make the computation tractable, we do not solve Problem (2.23) directly, but we proceed as follows. First, we look for an orthogonal basis for the (approximated) 4-dimensional null-space of $\widehat{L}$, by solving the following optimization problem

$$\min_{U^\mathsf{T}U=nI_4} \left\|\widehat{L}U\right\|_F^2 \tag{2.25}$$

where we use U instead of M to underline that we are looking for the subspace spanned by the columns of M, instead of M itself. In other words, we solve the homogeneous system of equations $\widehat{L}U = \mathbf{0}$ in the least-squares sense. Within this space, we find the solution that is closest to have every fourth row equal to [0 0 0 1] by solving system (2.5) in the least-squares sense. Then, such a solution is projected onto $\mathrm{SE}(3)^n$ – as in [8] – by forcing every fourth row to [0 0 0 1] and projecting 3×3 rotation blocks onto SO(3) through Singular Value Decomposition (SVD) [61].

Proposition 2.3. *Problem (2.25) admits a closed-form solution, which is given by the 4 eigenvectors of $\widehat{L}^\mathsf{T}\widehat{L}$ associated to the 4 smallest eigenvalues.*

Proof. We first observe that Problem (2.25) coincides with

$$\min_{U^\mathsf{T}U=nI_4} \mathrm{trace}(U^\mathsf{T}(\widehat{L}^\mathsf{T}\widehat{L})U). \tag{2.26}$$

Let $\mathcal{F}$ be the unconstrained cost function corresponding to this problem, namely

$$\mathcal{F}(U) = \mathrm{trace}(U^\mathsf{T}(\widehat{L}^\mathsf{T}\widehat{L})U) + \mathrm{trace}(\Lambda(U^\mathsf{T}U - nI_4)) \tag{2.27}$$

where $\Lambda \in \mathbb{R}^{4\times 4}$ is a symmetric matrix of unknown Lagrange multipliers. Setting to zero the partial derivatives of $\mathcal{F}$ with respect to U we obtain

$$\frac{\partial\mathcal{F}}{\partial U} = 2(\widehat{L}^\mathsf{T}\widehat{L})U + 2U\Lambda = 0 \Rightarrow (\widehat{L}^\mathsf{T}\widehat{L})U = -U\Lambda. \tag{2.28}$$

Let $\mathbf{u_i}$ be any four eigenvectors of $\widehat{L}^\mathsf{T}\widehat{L}$ (normalized so that $\|\mathbf{u_i}\| = \sqrt{n}$) and let λ_i be the corresponding eigenvalues. Then $U = [\mathbf{u_1}|\mathbf{u_2}|\mathbf{u_3}|\mathbf{u_4}]$ satisfies both (2.28) and the constraint $U^\mathsf{T}U = nI_4$, with $\Lambda = -\mathrm{diag}(\lambda_1,\lambda_2,\lambda_3,\lambda_4)$ (indeed $\widehat{L}^\mathsf{T}\widehat{L}$ admits an orthonormal basis of real eigenvectors since it is symmetric). In other words, any quadruple of eigenvectors is a stationary point for the objective function $\mathcal{F}$. The minimum is attained in (2.26) if $\mathbf{u_i}$ are the 4 least eigenvectors of $\widehat{L}^\mathsf{T}\widehat{L}$. ■

It follows from Proposition 2.3 that Problem (2.25) can be equivalently solved by computing the least 4 right singular vectors of $\widehat{L}$. It should be noted that in all practical scenarios the measurements graph $\mathcal{G}$ is sparse, for pairwise measurements are typically available only for a small number of neighbouring views, hence the number of edges is linear in the number of nodes. However, while $\widehat{L}$ inherits the sparsity of $\mathcal{G}$, this does not necessarily imply that $\widehat{L}^\mathsf{T}\widehat{L}$ will be comparably sparse. Indeed, if A has a full row, then $\widehat{L}^\mathsf{T}\widehat{L}$ is full. Hence the singular vector solution has to be preferred if a sparsity-aware algorithm is used.

Note that our method produces an *extrinsic estimate*, for it provides a closed-form solution to Problem (2.25), which is a relaxed version of Problem (2.23), and this solution is then

projected onto the manifold of rigid motions. The idea of relaxing rigid-motion constraints is also present in [46] where the feasible set $\mathrm{SE}(3)^n$ is relaxed to its convex hull. The authors of [46] minimize a geometric error tightly related to (2.22), resulting in an iterative scheme. This leads to a more accurate solution than our approach, which is based on an algebraic error, but it requires a significant amount of time, while our solution is fast and direct – as confirmed by the simulations.

Alternatively, we can consider the equivalent formulation (2.6) and compute the 4 top eigenvectors of

$$\widehat{P} = (D \otimes I_4)^{-1}((A \otimes \mathbb{1}_{4\times 4}) \circ \widehat{X}) \tag{2.29}$$

where the matrix $D \otimes I_4$ is easy to invert since it is diagonal. Note that $\widehat{P}$ inherits the sparsity of $\mathcal{G}$. In the presence of noise that cripples the structure of P (i.e. $\widehat{X} \neq MM^{-\flat}$) the eigenvalues and the eigenvectors will be complex in general. As a consequence, after computing the linear combination that yields [0 0 0 1] in every fourth row, we remove the imaginary part of the eigenvectors, and subsequently project 3×3 rotation blocks onto $\mathrm{SO}(3)$.

In summary, an approximate solution to rigid-motion synchronization in $\mathrm{SE}(3)$ can be found by either computing the 4 least right singular vectors of $\widehat{L}$ or the 4 top eigenvectors of $\widehat{P}$. An empirical comparison between the former – called NULL-SE(3)– and the latter – called EIG-SE(3)– is provided in Section 3. In both cases the matrices are sparse, so in the implementation we exploited the sparse eigen-solver of MATLAB (`eigs`) in EIG-SE(3), and the sparse SVD routine (`svds`) in NULL-SE(3). As a matter of fact, `svds(A)` calls `eigs([0 A; A' 0])`, as reported in the function documentation and consequently – as confirmed by our simulations – it runs slower, for the dimension of the matrix is double. From the computational complexity point of view, the Lanczos method (implemented by `eigs`) is "nearly linear", meaning that, if the matrix is sparse, every iteration is linear in n [21], but the number of iterations cannot be bounded by a constant.

2.3. Dealing with Outliers. The fact that our spectral method easily copes with weights on individual relative orientations, by way of the weighted adjacency matrix A, allows a straightforward extension to gain resilience to rogue input measures via Iteratively Reweighted Least Squares (IRLS) [29]. Such outliers are ubiquitous in real scenarios. In the structure-from-motion context, for example, repetitive structures in the images cause epipolar geometries which are not compatible with the 3D geometry of the scene. In global registration of multiple point-sets, outliers are caused by faulty pairwise registration, which can be originated by insufficient overlap and/or bad initialization.

First, we obtain an estimate for M with given weights[3] as explained in the previous section. Then, we update the weights w_{ij} using the current estimate of absolute orientations, and these steps are iterated until convergence or a maximum number of iterations is reached. In our experiments we used the Cauchy weight function

$$w_{ij} = \frac{1}{1 + \left(\frac{r_{ij}}{c}\right)^2} \tag{2.30}$$

[3]The initial weights are all 1 by default, but they can be initialized from any reliability information coming from the relative orientation estimation procedure.

with $r_{ij} = \|\widehat{R}_{ij} - R_i R_j^{\intercal}\|_F$. It is more convenient to define the residual in terms of the rotational component only, because the translation is affected by an arbitrary scale in the structure-from-motion scenario. On the other hand it is reasonable to assume that if a relative orientation is outlier, then both the rotation and translation components are wrong.

The factor c has been computed, as customary, with

$$c = 1.482\,\mathrm{med}(|\mathbf{r} - \mathrm{med}(\mathbf{r})|) \cdot \theta \tag{2.31}$$

where med( ) is the median operator, $\mathbf{r}$ is the vectorization of the residuals r_{ij}, and θ is a tuning constant set to $\theta = 2$ in our experiments.

2.4. Generalization to SE(N). In this paper we focus on SE(3) because this group arises in several applications. However, it is straightforward to see that our analysis and the derived spectral method apply equally well to any dimension.

Suppose that we are given a redundant number of pairwise ratios $M_{ij} \in \mathrm{SE(N)}$, and we want to estimate the associated group elements $M_i \in \mathrm{SE(N)}$, which represent rigid displacements in $\mathbb{R}^N$. If the graph is complete then – in the absence of noise – the block-matrix $X \in \mathbb{R}^{(N+1)n \times (N+1)n}$ has rank $N+1$, and the columns of M are $N+1$ eigenvectors of X with eigenvalue n, or, equivalently, they form a basis for the $(N+1)$-dimensional null-space of L. If the graph is not complete then Equations (2.6) and (2.15) still hold, thus we can generalize our spectral method to synchronize elements of SE(N), by computing the $N+1$ least right singular vectors of $\widehat{L}$, or by computing the $N+1$ top eigenvectors of $\widehat{P}$.

Mutatis mutandis, the method applies as well to the group of Euclidean motions $E(N)$.

3. Experiments. We evaluated our spectral solutions on both simulated and real data in terms of accuracy, execution time and robustness to outliers. We compared EIG-SE(3) and NULL-SE(3) to several techniques from the state-of-the-art. All the experiments were performed on a MacBook Air with i5 dual-core @ 1.3 GHz. The MATLAB code is available on the web at www.diegm.uniud.it/fusiello/demo/gmf.

In order to compare estimated and ground-truth absolute orientations, we found the optimal isometry that aligns them by applying single averaging for the rotation term and least-squares for the translation (and scale). Specifically, if $\widehat{M}_1, \ldots, \widehat{M}_n$ are estimates of the theoretical absolute orientations $M_1, \ldots, M_n$ then the optimal $N \in \mathrm{SE(3)}$ that aligns them into a common reference system solves $M_i = \widehat{M}_i N$, which is equivalent to $R_i = \widehat{R}_i R$ and $\mathbf{t_i} = \widehat{R}_i \mathbf{t} + \widehat{\mathbf{t}}_\mathbf{i}$ by considering separately the rotation and translation term. Thus the optimal $R \in \mathrm{SO(3)}$ is the single mean of the set $\{R_i \widehat{R}_i^{\intercal}, i = 1, \ldots, n\}$, which can be estimated by applying ℓ_1-geodesic averaging [26], while the optimal $\mathbf{t} \in \mathbb{R}^3$ is computed in the least-squares sense. If a bearing-based method is considered, a scale $s \in \mathbb{R}$ has to be estimated in addition to the translation $\mathbf{t} \in \mathbb{R}^3$, resulting in a linear system of the form $\mathbf{t_i} = \widehat{R}_i \mathbf{t} + s\widehat{\mathbf{t}}_\mathbf{i}$.

We used the angular distance $d_\angle(R_i, \widehat{R}_i)$ and Euclidean norm $||\mathbf{t_i} - \widehat{\mathbf{t}}_\mathbf{i}||_2$ to measure the accuracy of absolute rotations and translations respectively, where the angular (or geodesic) distance between two rotations $R, S \in \mathrm{SO(3)}$ is the angle (in the angle-axis space) of the rotation $SR^{\intercal}$ so chosen to lie in the range $[0, 180°]$, namely $d_\angle(S, R) = d_\angle(RS^{\mathsf{T}}, I) = 1/\sqrt{2}\,\left\|\log(RS^{\mathsf{T}})\right\|_2$. Note that the angular distance is equivalent to the Frobenius norm in the sense that they are related by a positive continuous strictly increasing function [31], namely $||R - S||_F = 2\sqrt{2}\sin\left(1/2\; d_\angle(S, R)\right)$.

Experimental results on simulated data are presented in Section 3.1, while in Sections 3.2 and 3.3 we report experiments on real data in the context of structure-from-motion and multiple point-set registration.

3.1. Simulated Data. In these experiments we considered n absolute orientations in which rotations are sampled from random Euler angles and translation components follow a standard Gaussian distribution. The measurement graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a random graph drawn from the Erdős-Rényi model with parameters (n, p), i.e. given a vertex set $\mathcal{V} = \{1, 2, \dots, n\}$ each edge (i, j) is in the set $\mathcal{E}$ with probability p, independently of all other edges. The true relative orientations were corrupted by a multiplicative noise $\widehat{M}_{ij} = M_{ij}E_{ij}$, with $E_{ij} \in \mathrm{SE}(3)$ representing a small perturbation of the identity matrix. The rotation component of E_{ij} has axis uniformly distributed over the unit sphere and angle following a Gaussian distribution with zero mean and standard deviation $\sigma_R \in [1°, 10°]$, and the translation components were sampled from a Gaussian distribution with zero mean and standard deviation $\sigma_T \in [0.01, 0.1]$. In this way we perturbed both direction and magnitude of pairwise translations. All the results were averaged over 50 trials.

We evaluated the effect of noise on rotations and translations both separately and together, by considering $n = 100$ absolute orientations, in the cases $p = 0.05$ and $p = 0.3$, which correspond to about 95% and 70% of missing pairs, respectively. Higher values of p correspond to better conditioned problems, with the same qualitative behaviour as $p = 0.3$. Please note that in the real cases reported in Table 3 the percentage of missing pairs ranges from 30% to 90%.

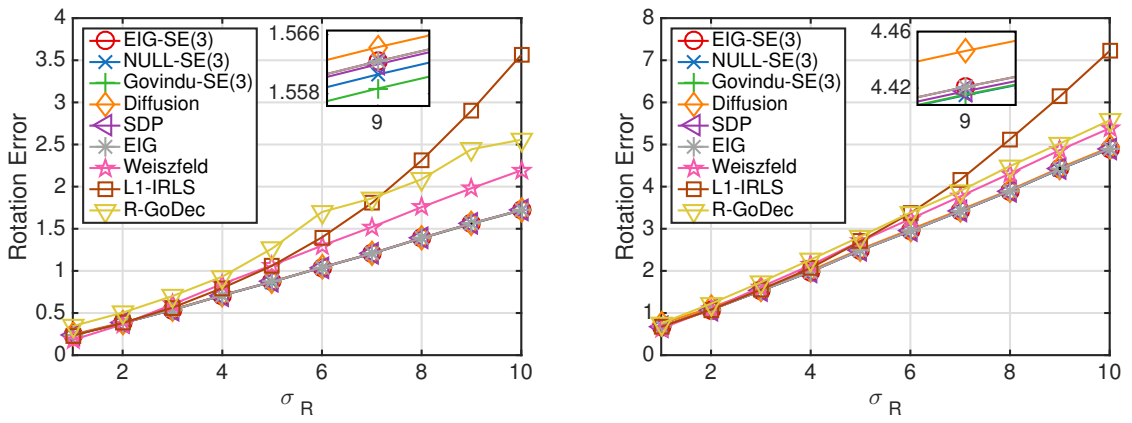


Figure 1: Mean angular errors (degrees) on the absolute rotations with $p = 0.3$ (left) and $p = 0.05$ (right). The value of σ_T does not affect rotations. The inset shows a magnification at $\sigma_R = 9°$.

Rotation. Besides the SE(3)-averaging technique developed by Govindu (Govindu-SE(3)) [23] and the Diffusion method by Torsello et al. [57], we considered general synchronization techniques such as the Weiszfeld algorithm [26], spectral relaxation [3] (EIG), semidefinite programming [3] (SDP), the L1-IRLS algorithm [13], and the R-GoDec algorithm [6]. Meth-

ods based on quaternions (such as [22]) were already proved inferior to the other methods in [36]. The codes of L1-IRLS and Diffusion are available on-line, the one of Govindu-SE(3) was provided by the author, while in the other cases we used our implementation. The parameter that determines the transition from quadratic to fixed loss in L1-IRLS was set equal to 10 degrees in these simulations.

Figure 1 reports the mean angular errors on the absolute rotations as a function of σ_R, obtained by running the rotation synchronization techniques mentioned above. The noise on relative translations does not have any influence on absolute rotations, hence the value of σ_T is meaningless in this experiment. The best accuracy is obtained by our methods EIG-SE(3) and NULL-SE(3) together with EIG, SDP, Govindu-SE(3) and Diffusion. On the contrary, the robust approaches R-GoDec, L1-IRLS and Weiszfeld yield worse results, to different extents, because they inherently trade robustness for statistical efficiency. By manual inspection it can be seen that EIG-SE(3) coincides with EIG, according to the fact that the former generalises the latter.

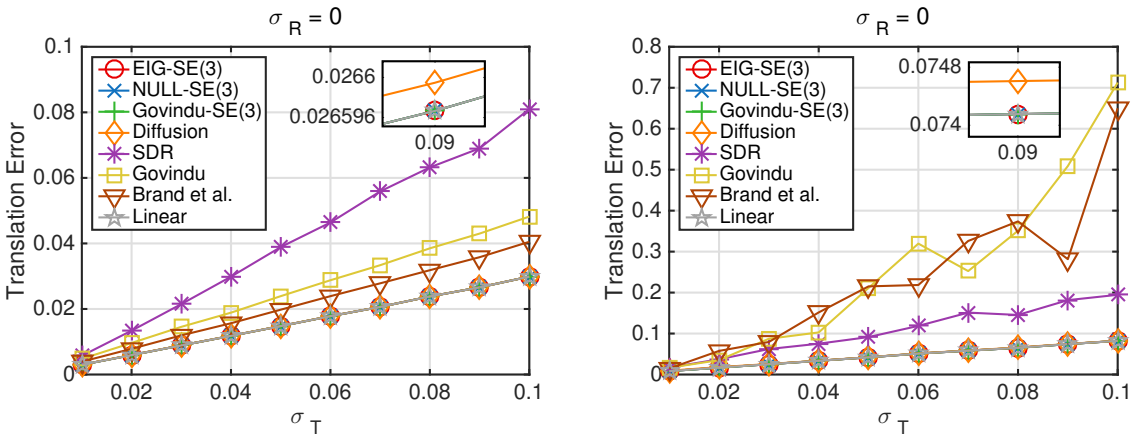


Figure 2: Mean errors on the absolute translations with $p = 0.3$ (left) and $p = 0.05$ (right). The inset shows a magnification at $\sigma_T = 0.09$.

Translation. We considered only methods working in frame space, i.e. not requiring point correspondences, such as SDR [42], the graph-embedding approach by Brand et al. [12], the Diffusion method [57] and two works by Govindu [22, 23]. Among these methods, only EIG-SE(3), NULL-SE(3), Govindu-SE(3) [23] and Diffusion are influenced by the noise on the translation norms, for they take as input matrices in SE(3), while this does not influence the remaining algorithms, which take as input relative translation directions. We also included in the comparison the Linear method for translation synchronization derived by solving Equation (1.5) in the least-squares sense, assuming that translations norm are available and absolute rotations have been computed beforehand. The code of SDR is available on-line, while in the other cases we used our implementation. In this simulation we did not perturb relative rotations ($\sigma_R = 0$), thus all the methods were given ground-truth relative/absolute rotations. Noise on rotational components also influences the translation errors with results qualitatively

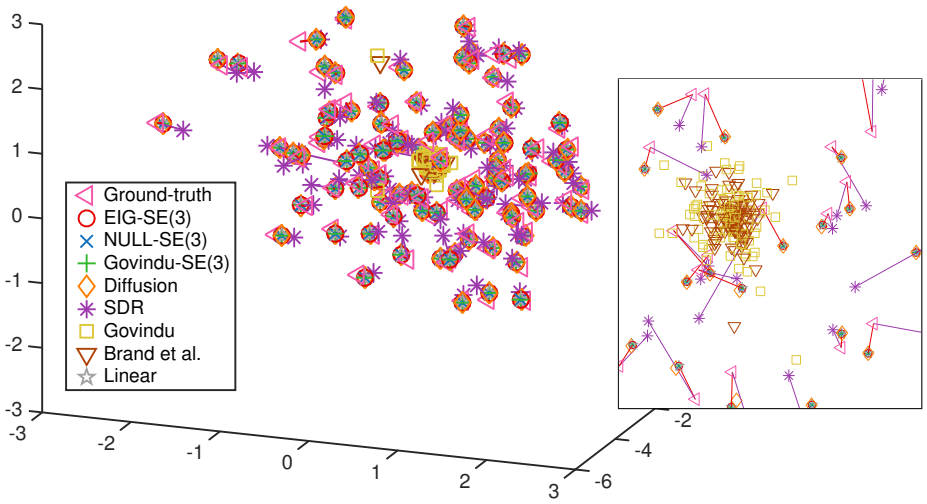


Figure 3: Clustering phenomenon. The inset shows a magnification of the centre of the cloud. The reader will distinguish one cluster of yellow squares and one of reddish triangles near the origin, which correspond to the locations obtained by Govindu [22] and Brand et al. [12] respectively.

similar to those reported here.

Figure 2 shows the mean errors on the absolute translations as a function of σ_T (units are commensurate with the simulated data), obtained after running the techniques mentioned above. Our methods EIG-SE(3) and NULL-SE(3) together with Govindu-SE(3), Diffusion and the Linear solution outperform all the analysed techniques in terms of accuracy.

When the measurement graph is extremely sparse ($p = 0.05$) the methods by Govindu [22] and Brand et al. [12] yield larger errors than usual. By inspecting the solution it is found that this corresponds to wrong solutions concentrated around a few locations. This can be visualized in Figure 3, which shows ground-truth and estimated positions (after alignment) for a single trial when $\sigma_T = 0.1$. Such a behaviour – called "clustering phenomenon" – affects the methods that use only the direction of translation (or bearing) and the cause has been traced back to a lack of constraints on the location distances in [42]. For this reason ad-hoc constraints were introduced in the minimization problem in [42], forcing the differences between translations to be "sufficiently" large. On the other hand, EIG-SE(3), NULL-SE(3), Govindu-SE(3), Diffusion and the Linear solution implicitly enforce such constraints as they take in input the relative translations with their norm.

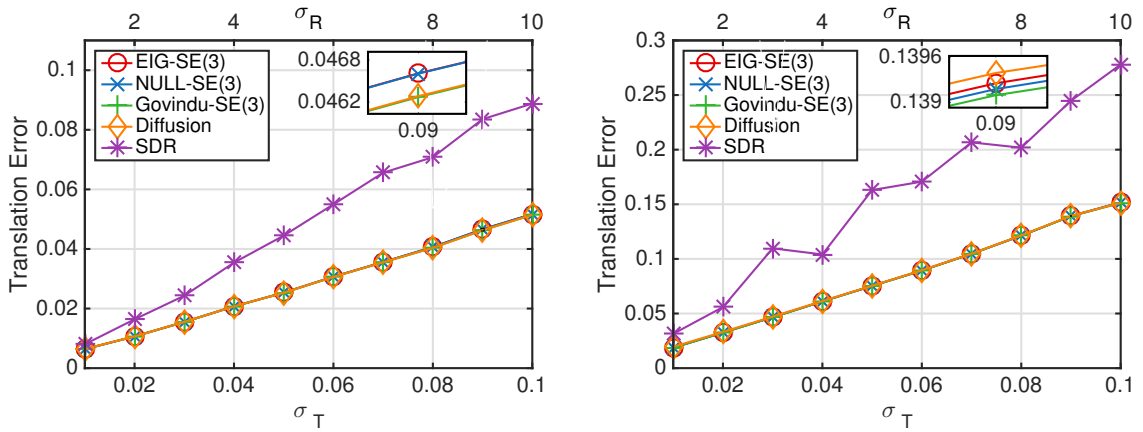


Figure 4: Mean errors on the absolute translations with noise on both relative rotations and translations, with $p = 0.3$ (left) and $p = 0.05$ (right). The inset shows a magnification at $(\sigma_T, \sigma_R) = (0.09, 9°)$.

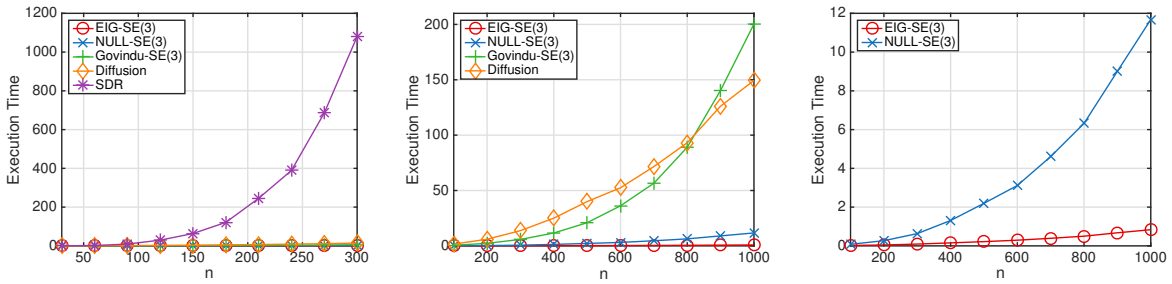


Figure 5: Execution times (seconds) of rigid-motion synchronization versus number of nodes. The SDR algorithm is analysed only with maximum 300 nodes in this experiment due to computational limitations. A magnification is shown on the right to appreciate the difference between EIG-SE(3) and NULL-SE(3).

Rigid-motion synchronization. We considered only methods that cope with rigid-motion synchronization and work in frame space, such as SDR [42], Govindu-SE(3) [23] and Diffusion [57]. Beside SDR – which has an original translation stage and uses EIG to synchronize rotations – all the other methods recover rotations and translations simultaneously.

Figure 4 reports the mean errors on the absolute translations obtained after perturbing both relative rotations and translations. All the methods return good estimates, which are further improved by increasing edge connectivity, and the lowest errors are achieved by EIG-SE(3), NULL-SE(3), Govindu-SE(3) and Diffusion.

We also analysed the execution time of rigid-motion synchronization, by varying the number of absolute orientations from $n = 30$ to $n = 1000$, all the others parameters being fixed. More precisely, we chose the values $p = 0.2$ (about 80% missing data), $\sigma_T = 0.05$ and $\sigma_R = 5°$

to define sparsity and noise. We implemented EIG-SE(3) and NULL-SE(3) in MATLAB using `eigs` with $\widehat{P}$ in input and `svds` with $\widehat{L}$ in input respectively. Govindu-SE(3) and SDR are in MATLAB as well (code from the authors), while Diffusion [57] is implemented in C++ (by the authors).

Figure 5 reports the running times of the analysed algorithms as a function of the number of nodes in the measurement graph, showing that SDR is remarkably slower than the other techniques. In particular, both our methods are faster than the others and EIG-SE(3) is the fastest overall, computing a solution in less than 2 seconds for $n = 1000$.

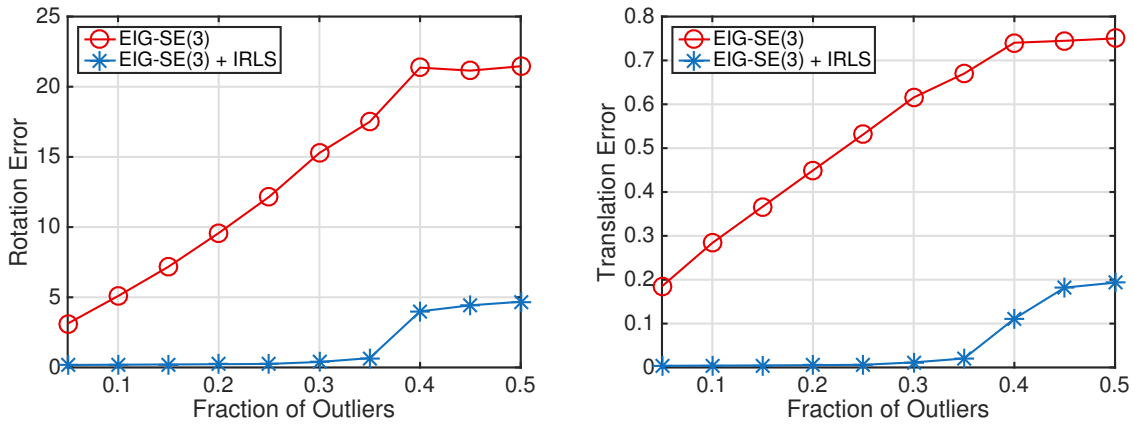


Figure 6: Mean errors on the absolute orientations versus outliers contamination with σ_T=σ_R=0.

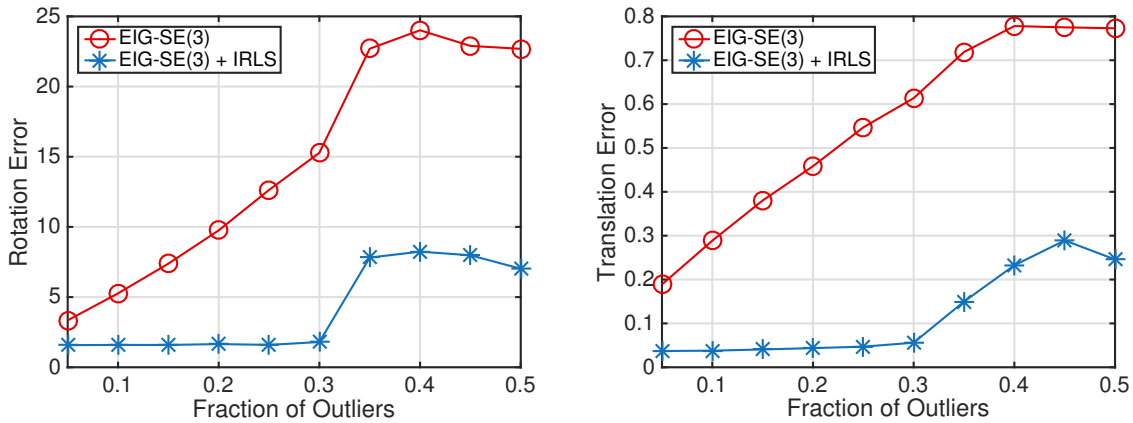


Figure 7: Mean errors on the absolute orientations versus outliers contamination, with noise on both relative rotations and translations.

In the additional material, the method described in [46] is also evaluated, on a smaller

number of views, do to its high computational cost. As expected, [46] slightly improves the results over EIG-SE(3) and NULL-SE(3) (for it minimizes a geometric objective function) but at the cost of a much longer execution time (for instance, it takes around 10 minutes for a single trial with $n = 100$ and $p = 0.3$).

The rundown of these experiments is that EIG-SE(3) achieves the same optimal accuracy of its closest competitors in considerably less time. Therefore we elect EIG-SE(3) as our choice and drop NULL-SE(3) in subsequent comparisons.

Outliers influence. In this experiment we studied the resilience to outliers of EIG-SE(3) coupled with IRLS. We considered $n = 100$ absolute orientations sampled as before, and we fixed $p = 0.2$ to define sparsity. Since we are interested in analysing exact recovery in the presence of outliers, noise was not introduced in this first simulation. The percentage of outliers – randomly generated – varies from 10% to 50%. Such a percentage is referred to the number of available relative orientations, not to the maximum number of pairs (i.e. $n(n-1)/2$). Figure 6 reports the mean errors obtained by EIG-SE(3) and its IRLS modification. It clearly shows the robustness gained by IRLS: exact recovery is achieved up to 35% of outliers, but good performances are still obtained until 50% of outliers.

Next, we analysed the behaviour of our approach in the presence of both noise and outliers. We introduced a fixed level of noise on the relative orientations ($\sigma_T = 0.05, \sigma_R = 5^\circ$), and all the other parameters were chosen as before. Results are reported in Figure 7, showing that the empirical breakdown point of EIG-SE(3) + IRLS is about 30%.

3.2. Real Data: Structure from Motion. We applied our method EIG-SE(3) + IRLS to the structure-from-motion problem, considering both the EPFL benchmark [53] and unstructured, large-scale image sequences from [64]. The latter are available on-line together with the relative orientations, while for the EPFL benchmark we computed them following a standard approach based on the essential matrix with a final bundle adjustment (BA) refinement of camera pairs.

Owing to the depth-speed ambiguity, the magnitude of relative translations (also referred to as *epipolar scales*) is undefined. Therefore, the input relative orientations do not fully specify elements of SE(3), and the unknown scales have to be computed.

A straightforward approach (suggested in [23]) consists in iteratively updating these epipolar scales, i.e. during each iteration the scale of the translation of $\widehat{M}_{ij}$ is set equal to that of $M_i M_j^{-1}$, where M_i and M_j are the current estimates of camera orientations. The starting scales are all equal to 1 and the procedure is iterated until convergence. In our implementation this is combined with IRLS in the same loop: in one step we update the IRLS weights and in the next step we update the epipolar scales.

A different approach is proposed in [4], where a two-stage method is developed for computing the epipolar scales based on the knowledge of two-view geometries only. First, a cycle basis for the measurement graph $\mathcal{G}$ is extracted, then all the scales are recovered simultaneously by solving a homogeneous linear system. This approach is based on the observation that the compatibility constraints associated to cycles can be seen as equations in the unknown scales. In this way all the unknown norms are computed before performing rigid-motion synchronization. In this paper we used the robust version of such approach (named NMCB in [4]) where the cycle basis is a Minimum Cycle Basis composed of null-cycles (i.e. cycles whose relative

rotations compose to the identity), which is computed through a modified version of Horton's algorithm [30]. It is shown in [58] that theoretical conditions under which the epipolar scales can be uniquely (up to a global scale) recovered are the same as the bearing-only localization problem, i.e. the measurement graph is required to be parallel rigid [42, 63].

However, computing the epipolar scales is not part of the synchronization task, strictly speaking. As a matter of fact, this indeterminacy is an idiosyncrasy of the structure-from-motion problem, which is not shared e.g. by multiple point-set registration, where the relative orientations are fully specified. For this reason we are agnostic about the specific method for computing the scales, and we also provide results obtained by using ground-truth scales, in addition to the approaches mentioned above.

EPFL benchmark. The EPFL Benchmark datasets [53] contain from 8 to 30 images, and provide ground-truth absolute orientations. Results are reported in Tables 1 and 2, which show the mean errors of rigid-motion synchronization before and after applying a two-step bundle adjustment, as done in [38], where in the first step rotations are kept fixed.

Table 1: Mean angular errors (degrees) on camera rotations for the EPFL benchmark. Moulon et al. is missing in this table because rotation errors are not reported in [38].

	EIG-SE(3)-GT		EIG-SE(3)-Iter		EIG-SE(3)-NMCB		SDR		R-GoDec+Brand		Samantha
Dataset	pre BA	post BA	pre BA	post BA	pre BA	post BA	pre BA	post BA	pre BA	post BA	post BA
HerzJesuP8	0.04	0.03	0.03	0.03	0.03	0.03	0.06	0.03	0.04	0.03	0.04
HerzJesuP25	0.07	0.03	0.07	0.04	0.06	0.04	0.14	0.04	0.13	0.04	0.03
FountainP11	0.03	0.03	0.03	0.04	0.03	0.03	0.03	0.03	0.03	0.03	0.06
EntryP10	0.05	0.02	0.06	0.02	0.05	0.03	0.56	0.04	0.44	0.03	0.05
CastleP19	0.41	0.06	0.44	0.06	0.41	0.06	3.69	0.05	1.57	0.05	0.09
CastleP30	0.40	0.05	0.48	0.05	0.40	0.05	1.97	0.05	0.78	0.05	0.06

Table 2: Mean errors (meters) on camera translations for the EPFL benchmark.

	EIG-SE(3)-GT		EIG-SE(3)-Iter		EIG-SE(3)-NMCB		SDR		R-GoDec+Brand		Moulon	Samantha
Dataset	pre BA	post BA	pre BA	post BA	pre BA	post BA	pre BA	post BA	pre BA	post BA	post BA	post BA
HerzJesuP8	0.004	0.004	0.512	0.004	0.008	0.004	0.007	0.005	0.009	0.004	0.004	0.007
HerzJesuP25	0.010	0.008	0.931	0.022	0.017	0.008	0.065	0.009	0.038	0.009	0.005	0.031
FountainP11	0.002	0.003	0.257	0.003	0.005	0.003	0.004	0.003	0.006	0.003	0.003	0.006
EntryP10	0.008	0.008	0.307	0.008	0.167	0.009	0.203	0.010	0.433	0.009	0.006	0.022
CastleP19	0.218	0.034	3.747	0.034	2.971	0.035	1.769	0.032	1.493	0.036	0.026	0.046
CastleP30	0.150	0.032	2.308	0.035	0.908	0.034	1.393	0.030	1.123	0.030	0.022	0.033

We considered three versions of EIG-SE(3), which differ for the technique chosen to recover the epipolar scales, namely using ground-truth scales (GT), computing the scales through [4] (NMCB), and updating the scales iteratively (Iter). Our spectral solution is compared with the SDR method [42] and the global structure-from-motion pipeline described by Moulon et al. [38]. We also considered the pipeline obtained by combining the R-GoDec algorithm [6] with the translation recovery method in [12], which is analysed in [5]. As a reference, we

included in the comparison the hierarchical structure-from-motion pipeline SAMANTHA [56], whose binary code is publicly available. With the exception of Moulon et al., for which results are taken from [38], and SAMANTHA, which processes images directly, all the other methods are given the same relative orientations as input.

Both EIG-SE(3) and all the analysed techniques achieve a high precision, obtaining an average rotation error less than 0.1 degrees and an average translation error of the order of millimetres, after the final BA. Our method is able to recover camera parameters efficiently, since the rigid-motion synchronization step takes less than 0.5 seconds for all the sequences.

If we concentrate on the EIG-SE(3)-GT columns, we can see that it achieves the optimum *before* BA in most datasets, confirming the effectiveness of our method for averaging relative orientations, when the latter are fully specified. Without ground-truth scales, good estimates of orientation parameters are still obtained, and precision increases by using NMCB rather than the iterative approach. The errors after BA are always very small and almost equal to the other methods, confirming that EIG-SE(3) provides a good starting point for bundle adjustment.

Large-scale datasets. We tested our technique on irregular large-scale collections of images taken from [64], for which recovering camera orientations is challenging. Following the experiments in [64], we used the output of BUNDLER [52] as reference solution, and we computed the optimal isometry between this solution and our estimate with least median of squares (LMedS), using correspondences between camera centres. Since our MATLAB implementation of [4] is too slow for large datasets, we did not compute the scales through NMCB in this experiment and used the iterative update instead.

Table 3: Median errors (rotation in degrees, translation in metres) on the datasets from [64] *before BA*. Times are in seconds. Some values are missing in this table since some datasets are not analysed in [41] and [16]. The lowest translation errors are highlighted in boldface.

			EIG-SE(3)-Iter			1DSfM		SDR		CLS		LUD		Cui et al.	
Dataset	n	miss %	rot.	tra.	time	tra.	time	tra.	time	tra.	time	tra.	time	tra.	time
Piccadilly	2446	89	5.7	7.1	230	**4.1**	936	–	–	–	–	–	–	–	–
Roman Forum	1102	88	5.5	17.2	77	**6.1**	172	–	–	–	–	–	–	–	–
Vienna Cathedral	898	74	1.6	**3.2**	87	6.6	242	–	–	8.8	313	5.4	522	3.5	242
Union Square	853	93	4.6	5.7	27	**5.6**	92	–	–	–	–	–	–	–	–
Alamo	606	47	1.2	0.7	67	1.1	129	–	–	1.3	143	**0.4**	289	0.6	259
Notre Dame	553	32	0.7	0.5	47	10	112	–	–	0.8	187	**0.3**	382	**0.3**	366
Tower of London	489	80	2.7	5.7	17	11	64	20	376	16	32	4.7	65	**4.4**	100
Montreal Notre Dame	467	52	0.6	0.7	22	2.5	92	–	–	1.1	89	**0.5**	180	0.8	125
Yorkminster	448	72	1.8	6.4	17	3.4	104	5.0	667	6.2	29	**2.7**	70	3.7	45
Madrid Metropolis	370	65	3.9	8.1	14	9.9	35	4.2	208	6.4	33	**1.6**	54	–	–
NYC Library	358	68	1.8	2.3	12	2.5	63	5.0	489	5.0	34	2.0	84	**1.4**	42
Piazza del Popolo	345	58	0.9	**1.1**	17	3.1	49	1.9	393	3.5	44	1.5	70	2.6	51
Ellis Island	240	29	0.7	3.2	11	3.7	25	–	–	–	–	–	–	**3.1**	31

Results are reported in Table 3, which shows the median errors of rigid-motion synchronization before applying bundle adjustment. We also reported the number of views that were

synchronized and the percentage of missing pairs, which refer to the largest parallel rigid subgraph, extracted as explained in [42].

We compared EIG-SE(3) with a recent technique – called 1DSfM [64] – which first removes outlier directions by solving simpler low-dimensional sub-problems, and then compute absolute translations through the Levenberg-Marquardt algorithm. We also included in the comparison the semi-definite-relaxation (SDR) method [42], the constrained-least-squares (CLS) technique [60], the least-unsquared-deviations (LUD) solver [41], and the method by Cui et al. [16]. The results of 1DSfM are taken from [64], the results of LUD, SDR and CLS are taken from [41], and the results by Cui et al. are taken from [16]. In all these papers rotation errors are not analysed, therefore the comparison concentrates on translation errors. In this respect, the median errors reported in Table 3 show that EIG-SE(3) with iterative scale estimate performs equal or better than 1DSfM in 10 cases out of 13, and it always provides results in the same range of the best methods, namely LUD, and Cui et al. These methods, however, form a class on their own, since they internally refine the pairwise directions, whereas 1DSfM and EIG-SE(3) use as input the relative orientations as they are. In LUD such directions are computed through an IRLS scheme robust to outlier correspondences, while in Cui et al. they are obtained through the method described in [34].

Computation times of 1DSfM, LUD, CLS and SDR reported in Table 3 are obtained by summing the execution costs of rotation and translation recovery. Note that this is an underestimation as the sum does not include the execution cost of intermediate steps which are still part of the rigid-motion synchronization pipeline, namely outlier removal in 1DSfM and robust pairwise direction estimation in LUD, CLS and SDR. Computation times of the method by Cui et al. are obtained by subtracting the cost for bundle adjustment refinement from the total execution cost reported in [16].

Please note that the execution times are taken from different papers [64, 41, 16] and were obtained on disparate computers. Albeit not directly comparable to each other, these machines are all more powerful than ours, however. Notwithstanding this, results in Table 3 demonstrate that our method is the fastest solution among all the analysed techniques.

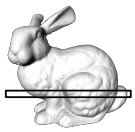
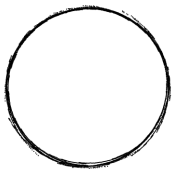
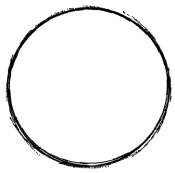
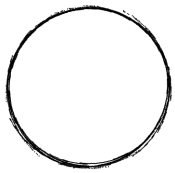
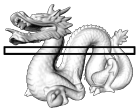
The summary of these experiments with real datasets confirms that, with respect to its competitors, EIG-SE(3) deliver results of comparable accuracy in substantially less time.

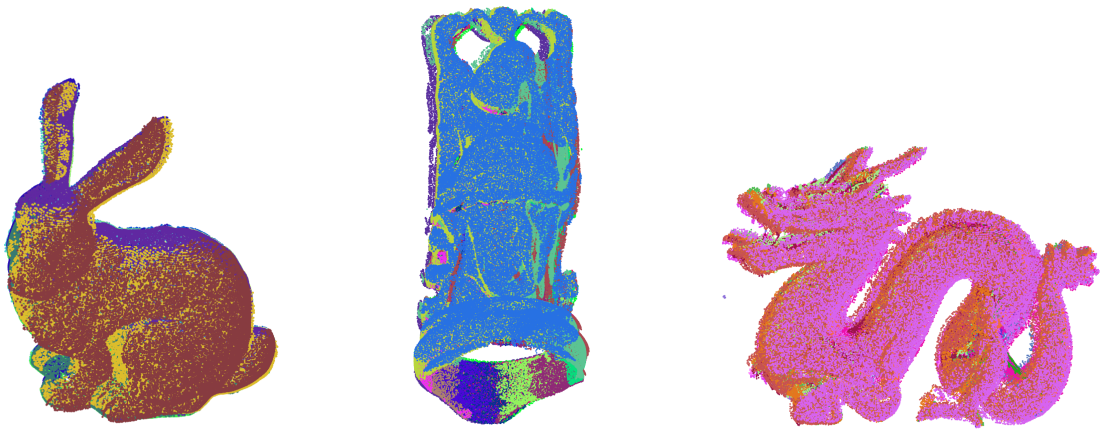
Table 4: Mean angular errors (degrees) on absolute rotations for the Stanford repository.

Dataset	n	% missing	EIG-SE(3)	Govindu-SE(3)	Diffusion	Weiszfeld+Linear
Bunny	10	30	0.91	0.91	0.91	0.89
Buddha	15	20	0.99	1.01	1.00	0.86
Dragon	15	10	1.06	1.06	1.06	0.84

3.3. Real Data: Multiple Point-set Registration. We applied our spectral method to multiple point-set registration, considering both the Stanford 3D Scanning Repository [1] and large-scale datasets used also in [18]. To obtain estimates of relative orientations we applied the Iterative Closest Point Algorithm (ICP) [11] to pairs of 3D point sets, as customary, and

Table 5: Cross-sections of registered point-sets on the Stanford repository.

	Dataset	EIG-SE(3)	Govindu-SE(3)	Diffusion	Weiszfeld+Linear
Bunny					
Buddha					
Dragon					



(a) Bunny (b) Buddha (c) Dragon

Figure 8: 3D models obtained with our method on the Stanford repository. Different colours correspond to different point sets.

we defined the measurement graph by discarding all the pairs with high registration error. With a careful threshold selection, this created a graph with no outliers. Note that, in contrast to the structure-from-motion application, in this case there is not a scale ambiguity in relative translations, for pairwise registration returns elements of SE(3).

Stanford 3D repository. From the Stanford 3D Scanning Repository [1] we used the Bunny, Happy Buddha (standing) and Dragon (standing) datasets, which contain 10, 15 and 15 point sets respectively. As for the initialization of the ICP algorithm, we perturbed the

true relative orientations by a rotation with random axis and angle uniformly distributed over $[0, 2^\circ]$, similarly to the experiments carried out in [25]. In order to evaluate the performances of rigid-motion synchronization in itself, we considered only registration techniques working in frame space, i.e. not requiring point correspondences, and all the methods were given the same relative orientations as input. We compared our method EIG-SE(3) against Govindu-SE(3) [23], Diffusion [57], and we also considered the combination of the Weiszfeld algorithm [26] with the Linear solution for translation synchronization. On account of the fact that such techniques are not robust, in this experiment we used our EIG-SE(3) method without IRLS so as to perform a fair comparison.

Since ground-truth absolute orientations are available for these datasets, we evaluated quantitatively the results by reporting the mean angular errors of rigid-motion synchronization in Table 4 (translation errors are not reported since they are less than 1 millimetre for all the datasets/methods). We also evaluated qualitatively the results in terms of cross-sections in Table 5. Finally, for visualization purposes, we report in Figure 8 the 3D models obtained by aligning the original point clouds with our method. Both EIG-SE(3) and all the analysed techniques achieve a good precision, yielding accurate cross-sections and an average rotation error of the order of 1 degree. Differences in execution time are meaningless for such relatively small datasets, and are not reported. EIG-SE(3) took about 0.02 seconds for all these sequences.

Table 6: Execution times (sec.) of rigid-motion synchronization for the datasets used in [18].

Dataset	n	% missing	EIG-SE(3)	Govindu-SE(3)	Diffusion	Weiszfeld+Linear
Gargoyle	27	62	0.02	0.08	0.22	0.47
Capital	100	76	0.04	0.70	2.11	4.17
Madonna	169	93	0.07	0.59	1.67	4.01

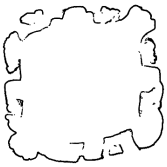
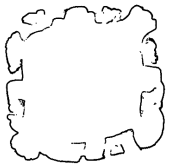
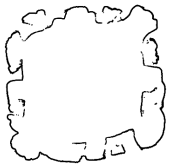
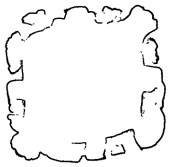
Large-scale datasets. The datasets used in [18] contain 27 (Gargoyle), 103 (Capital) and 196 (Madonna) point sets respectively. In the Madonna dataset, each scan covers a small portion of the entire object, thus making the registration task very challenging. Since there is no information about the scans, we simply initialized the ICP algorithm with identity matrices.

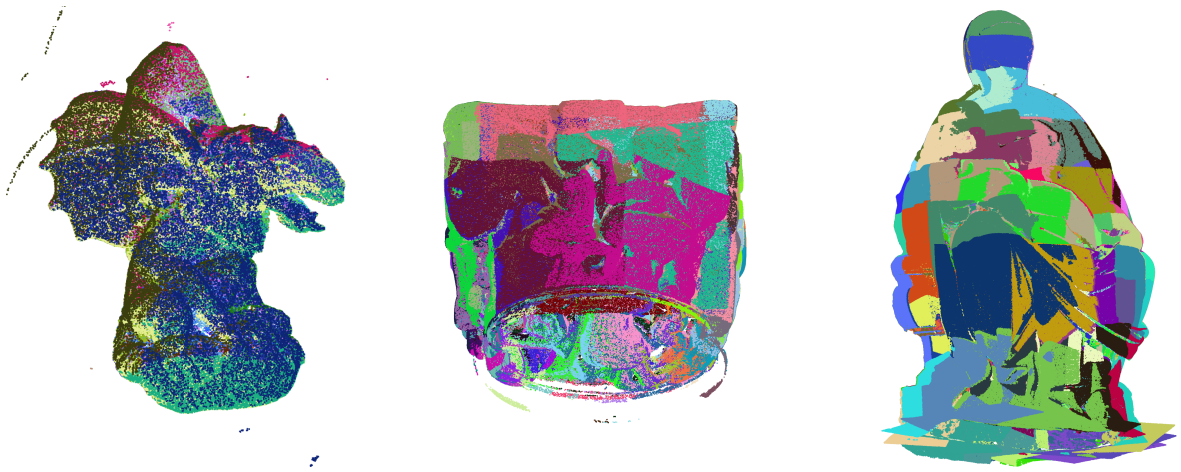
Results are reported in Tables 6 and 7, which show the execution times of rigid-motion synchronization and cross-sections of output 3D models. We also report the number of point sets registered and the percentage of missing pairs, which refer to the largest connected component of the cleaned graph. As before, for visualization purposes, we report the whole 3D models obtained by aligning the original point clouds with our method (Figure 9).

Note that, if needed, the estimates of absolute orientations obtained in this way can be further improved by alternating rigid-motion synchronization and computing relative orientations, as suggested in [25, 57].

These experiments show that our method is the fastest solution within all the analysed techniques, while being comparable in accuracy and providing high-quality reconstructions.

Table 7: Cross-sections of registered point-sets on the datasets used in [18].

Dataset	EIG-SE(3)	Govindu-SE(3)	Diffusion	Weiszfeld+Linear
Gargoyle					
Capital					
Madonna					



(a) Gargoyle (b) Capital (c) Madonna

Figure 9: 3D models obtained with our method on the datasets used in [18]. Different colours correspond to different point sets.

4. Conclusion. We presented a novel approximate solution to rigid-motion synchronization in SE(3). Our method is extremely fast, being based on a spectral decomposition problem, and theoretically relevant, for it computes at the same time both rotations and translations. Experiments on synthetic and real data showed that our method: i) provides accurate estimates of absolute rotations and translations (comparable to the state of the art); ii) it is the fastest among the existing techniques; iii) it can be successfully applied to multiple point-

set registration, returning reliable and accurate 3D models in low execution time; and iv) combined with a method for estimating the unknown translation norms, it can be profitably used in a global structure-from-motion pipeline to obtain a fast/high-quality initialization for bundle adjustment algorithms.

Acknowledgements. The authors would like to thank Simone Fantoni for his support in multiple point-set registration experiments and Massimiliano Corsini for providing the datasets used in [18]. They also thank Avishek Chatterjee and Venu Madhav Govindu for sharing the MATLAB implementation of [23].

REFERENCES

[1] *Stanford University 3D Scanning Repository*. http://graphics.stanford.edu/data/3Dscanrep/.

[2] M. AGRAWAL, *A Lie-algebraic approach for consistent pose registration for general Euclidean motion*, in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 1891–1897.

[3] M. ARIE-NACHIMSON, S. Z. KOVALSKY, I. KEMELMACHER-SHLIZERMAN, A. SINGER, AND R. BASRI, *Global motion estimation from point matches*, International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, (2012), pp. 81 – 88.

[4] F. ARRIGONI, A. FUSIELLO, AND B. ROSSI, *On computing the translations norm in the epipolar graph*, in International Conference on 3D Vision, 2015, pp. 300 – 308.

[5] F. ARRIGONI, B. ROSSI, AND A. FUSIELLO, *Robust and efficient camera motion synchronization via matrix decomposition*, in Image Analysis and Processing – ICIAP 2015, Vittorio Murino and Enrico Puppo, eds., vol. 9279 of Lecture Notes in Computer Science, Springer International Publishing, 2015, pp. 444–455.

[6] F. ARRIGONI, B. ROSSI, L. MAGRI, P. FRAGNETO, AND A. FUSIELLO, *Robust absolute rotation estimation via low-rank and sparse matrix decomposition*, International Conference on 3D Vision, (2014), pp. 491–498.

[7] F. ARRIGONI, B. ROSSI, F. MALAPELLE, P. FRAGNETO, AND A. FUSIELLO, *Robust global motion estimation with matrix completion*, ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-5 (2014), pp. 63–70.

[8] C. BELTA AND V. KUMAR, *Euclidean metrics for motion generation on SE(3)*, Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 216 (2002), pp. 47–60.

[9] R. BENJEMAA AND F. SCHMITT, *A solution for the registration of multiple 3D point sets using unit quaternions*, in European Conference on Computer Vision, 1998, pp. 34–50.

[10] F. BERNARD, J. THUNBERG, P. GEMMAR, F. HERTEL, A. HUSCH, AND J. GONCALVES, *A solution for multi-alignment by transformation synchronisation*, in IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2161 – 2169.

[11] P. BESL AND N. MCKAY, *A method for registration of 3D shapes*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 14 (1992), pp. 239 – 256.

[12] M. BRAND, M. ANTONE, AND S. TELLER, *Spectral solution of large-scale extrinsic camera calibration as a graph embedding problem*, in European Conference on Computer Vision, 2004, pp. 262–273.

[13] A. CHATTERJEE AND V. M. GOVINDU, *Efficient and robust large-scale rotation averaging*, in International Conference on Computer Vision, 2013, pp. 521 – 528.

[14] D. CRANDALL, A. OWENS, N. SNAVELY, AND D. P. HUTTENLOCHER, *Discrete-continuous optimization for large-scale structure from motion*, in IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 3001 – 3008.

[15] M. CUCURINGU, Y. LIPMAN, AND A. SINGER, *Sensor network localization by eigenvector synchronization over the Euclidean group*, ACM Transactions on Sensor Networks, 8 (2012), pp. 19:1 – 19:42.

[16] Z. CUI, N. JIANG, C. TANG, AND P. TAN, *Linear global translation estimation with feature tracks*, in British Machine Vision Conference, 2015.

[17] O. Enqvist, F. Kahl, and C. Olsson, *Non-sequential structure from motion*, in Eleventh Workshop on Omnidirectional Vision, Camera Networks and Non-classical Camera, 2011.

[18] S. Fantoni, U. Castellani, and A. Fusiello, *Accurate and automatic alignment of range surfaces*, in International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, pp. 73 – 80.

[19] A. Fusiello, U. Castellani, L. Ronchetti, and V. Murino, *Model acquisition by registration of multiple acoustic range views*, in European Conference on Computer Vision, 2002, pp. 805–819.

[20] A. Giridhar and P.R. Kumar, *Distributed clock synchronization over wireless networks: Algorithms and analysis*, Proceedings of the IEEE Conference on Decision and Control, (2006), pp. 4915–4920.

[21] G.H. Golub and C. F. Van Loan, *Matrix computations (3rd ed.)*, Johns Hopkins University Press, 1996.

[22] V. M. Govindu, *Combining two-view constraints for motion estimation*, in IEEE Conference on Computer Vision and Pattern Recognition, 2001.

[23] ———, *Lie-algebraic averaging for globally consistent motion estimation*, in IEEE Conference on Computer Vision and Pattern Recognition, 2004, pp. 684–691.

[24] ———, *Robustness in motion averaging*, in Proceedings of the Asian Conference on Computer Vision, 2006, pp. 457–466.

[25] V. M. Govindu and A. Pooja, *On averaging multiview relations for 3D scan registration*, IEEE Transactions on Image Processing, 23 (2014), pp. 1289–1302.

[26] R. Hartley, K. Aftab, and J. Trumpf, *L1 rotation averaging using the Weiszfeld algorithm*, IEEE Conference on Computer Vision and Pattern Recognition, (2011), pp. 3041–3048.

[27] R. I. Hartley, J. Trumpf, Y. Dai, and H. Li, *Rotation averaging*, International Journal of Computer Vision, 103 (2013).

[28] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, second ed., 2004.

[29] P. W. Holland and R. E. Welsch, *Robust regression using iteratively reweighted least-squares*, Communications in Statistics - Theory and Methods, 6 (1977), pp. 813–827.

[30] J. D. Horton, *A polynomial-time algorithm to find the shortest cycle basis of a graph*, SIAM Journal on Computing, 16 (1987), pp. 358–366.

[31] D. Q. Huynh, *Metrics for 3D rotations: Comparison and analysis*, Journal of Mathematical Imaging and Vision, 35 (2009), pp. 155–164.

[32] N. Jiang, Z. Cui, and P. Tan, *A global linear method for camera pose registration*, in International Conference on Computer Vision, 2013, pp. 481 – 488.

[33] F. Kahl and R. Hartley, *Multiple-view geometry under the l_∞-norm*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 30 (2008), pp. 1603–1617.

[34] L. Kneip, M. Chili, and R. Siegwart, *Robust real-time visual odometry with a single camera and an IMU*, in British Machine Vision Conference, 2011.

[35] S. Krishnan, P. Y. Lee, J. B. Moore, and S. Venkatasubramanian, *Optimisation-on-a-manifold for global registration of multiple 3D point sets.*, International Journal of Intelligent Systems Technologies and Applications, 3 (2007), pp. 319–340.

[36] D. Martinec and T. Pajdla, *Robust rotation and translation estimation in multiview reconstruction*, in IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1 – 8.

[37] C. D. Meyer, *Matrix Analysis and applied linear algebra*, SIAM, 2000.

[38] P. Moulon, P. Monasse, and R. Marlet, *Global fusion of relative motions for robust, accurate and scalable structure from motion*, in International Conference on Computer Vision, 2013, pp. 3248–3255.

[39] K. Nishino and K. Ikeuchi, *Robust simultaneous registration of multiple range images*, in Proceedings of the Asian Conference on Computer Vision, 2002.

[40] C. Olsson and O. Enqvist, *Stable structure from motion for unordered image collections*, in Proceedings of the 17th Scandinavian conference on Image analysis (SCIA11), Springer-Verlag, 2011, pp. 524–535.

[41] O. Ozyesil and A. Singer, *Robust camera location estimation by convex programming*, in IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2674 – 2683.

[42] O. Ozyesil, A. Singer, and R. Basri, *Stable camera motion estimation using convex programming*, SIAM Journal on Imaging Sciences, 8 (2015), pp. 1220 – 1262.

[43] X. Pennec, *Multiple registration and mean rigid shape: Applications to the 3D case*, in 16th Leeds Annual

Statistical Workshop, Leed, UK, 1996, pp. 178–185.

[44] K. Pulli, *Multiview registration for large data sets*, in International Conference on 3D Digital Imaging and Modeling, 1999, pp. 160 – 168.

[45] A. C. Raghuramu, *Robust multiview registration of 3D surfaces via ℓ_1-norm minimization*, in British Machine Vision Conference, 2015.

[46] D. M. Rosen, C. DuHadway, and J. J. Leonard, *A convex relaxation for approximate global optimization in simultaneous localization and mapping*, in IEEE International Conference on Robotics and Automation, 2015, pp. 5822 – 5829.

[47] J. Saunderson, P. A. Parrilo, and A. S. Willsky, *Semidefinite descriptions of the convex hull of rotation matrices*, SIAM Journal on Optimization, 25 (2015), pp. 1314 – 1343.

[48] G. C. Sharp, S. W. Lee, and D. K. Wehe, *Multiview registration of 3D scenes by minimizing error between coordinate frames*, in European Conference on Computer Vision, 2002, pp. 587–597.

[49] S. Shih, Y. Chuang, and T. Yu, *An efficient and accurate method for the relaxation of multiview registration error*, IEEE Transactions on Image Processing, 17 (2008), pp. 968 – 981.

[50] A. Singer, *Angular synchronization by eigenvectors and semidefinite programming*, Applied and Computational Harmonic Analysis, 30 (2011), pp. 20 – 36.

[51] A. Singer and Y. Shkolnisky, *Three-dimensional structure determination from common lines in cryo-em by eigenvectors and semidefinite programming*, SIAM Journal on Imaging Sciences, 4 (2011), pp. 543 – 572.

[52] N. Snavely, S. M. Seitz, and R. Szeliski, *Photo tourism: Exploring photo collections in 3D*, ACM Transactions on Graphics, 25 (2006), pp. 835–846.

[53] C. Strecha, L. J. Van von Hansen, W.and Gool, P. Fua, and U. Thoennessen, *On benchmarking camera calibration and multi-view stereo for high resolution imagery*, in IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1 – 8.

[54] J. Thunberg, F. Bernard, and J. Goncalves, *On transitive consistency for linear invertible transformations between Euclidean coordinate systems*, ArXiv e-prints, 1509.00728 (2015).

[55] R. Toldo, A. Beinat, and F. Crosilla, *Global registration of multiple point clouds embedding the generalized Procrustes analysis into an ICP framework*, in International Symposium on 3D Data Processing, Visualization, and Transmission, 2010.

[56] R. Toldo, R. Gherardi, M. Farenzena, and A. Fusiello, *Hierarchical structure-and-motion recovery from uncalibrated images*, Computer Vision and Image Understanding, 140 (2015), pp. 127 – 143.

[57] A. Torsello, E. Rodola, and A. Albarelli, *Multiview registration via graph diffusion of dual quaternions*, in IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 2441 – 2448.

[58] R. Tron, L. Carlone, F. Dellaert, and K. Daniilidis, *Rigid components identification and rigidity enforcement in bearing-only localization using the graph cycle basis*, in IEEE American Control Conference, 2015, pp. 3911 – 3918.

[59] R. Tron and K. Danilidis, *Statistical pose averaging with varying and non-isotropic covariances*, in European Conference on Computer Vision, 2014, pp. 804–819.

[60] R. Tron and R. Vidal, *Distributed 3-D localization of camera sensor networks from 2-D image measurements*, Automatic Control, IEEE Transactions on, 59 (2014), pp. 3325–3340.

[61] G. Wahba, *A Least Squares Estimate of Satellite Attitude*, SIAM Review, 7 (1965).

[62] L. Wang and A. Singer, *Exact and stable recovery of rotations for robust synchronization*, Information and Inference: a Journal of the IMA, 2 (2013), pp. 145–193.

[63] W. Whiteley, *Matroids from discrete geometry*, in Matroid Theory, J. Bonin, J. Oxley, and B. Servatius, eds., AMS Contemporary Mathematics, 1997, pp. 171–313.

[64] K. Wilson and N. Snavely, *Robust global translations with 1DSfM*, in European Conference on Computer Vision, 2014, pp. 61–75.

[65] C. Zach, M. Klopschitz, and M. Pollefeys, *Disambiguating visual relations using loop constraints*, in IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 1426 – 1433.

[66] S. Zhao and D. Zelazo, *Bearing-only network localization: Localizability, sensitivity, and distributed protocols*, Automatica, 69 (2016), pp. 334–341.