Supplementary material for
A Certifiably Globally Optimal Solution to the Non-Minimal Relative Pose Problem

Jesus Briales
MAPIR-UMA Group
University of Malaga, Spain
jesusbriales@uma.es

Laurent Kneip
Mobile Perception Lab
SIST ShanghaiTech
lkneip@shanghaitech.edu.cn

Javier Gonzalez-Jimenez
MAPIR-UMA Group
University of Malaga, Spain
javiergonzalez@uma.es

Contents

1. Brief summary

This supplementary material contains several proofs related to the claims and content in the main document. We also included some technical results in Section 4 which are necessary for the actual implementation of the approach. Finally, we show in Section 9 additional experimental results obtained on real data that were omitted in the main document due to space limitations.

2. Compact formulation of the covariance matrix of normals $\boldsymbol{M}(\boldsymbol{R})$

The covariance matrix of all the epipolar normals

$$\boldsymbol{M}(\boldsymbol{R}) = \sum_{i=1}^{N} (\boldsymbol{f}_i \times \boldsymbol{R}\boldsymbol{f}'_i)(\boldsymbol{f}_i \times \boldsymbol{R}\boldsymbol{f}'_i)^\top, \tag{1}$$

where $\boldsymbol{n}_i = \boldsymbol{f}_i \times \boldsymbol{R}\boldsymbol{f}'_i$, is clearly a quadratic expression on the rotation $\boldsymbol{R}$. Thus, each element m_{ij} of this matrix $\boldsymbol{M}(\boldsymbol{R})$ can be written as a quadratic function of the elements $\boldsymbol{r}$ in the rotation matrix $\boldsymbol{R}$:

$$m_{ij} \overset{1}{=} \boldsymbol{e}_i^\top \boldsymbol{M}(\boldsymbol{R}) \boldsymbol{e}_j \tag{2}$$

$$\overset{2}{=} \sum_{k=1}^{n} \boldsymbol{e}_i^\top (\boldsymbol{f}_k \times \boldsymbol{R}\boldsymbol{f}'_k)(\boldsymbol{f}_k \times \boldsymbol{R}\boldsymbol{f}'_k)^\top \boldsymbol{e}_j \tag{3}$$

$$\overset{3}{=} \sum_{k=1}^{n} \boldsymbol{e}_i^\top [\boldsymbol{f}_k]_\times \boldsymbol{R}\boldsymbol{f}'_k (\boldsymbol{f}'_k)^\top \boldsymbol{R}^\top [\boldsymbol{f}_k]_\times^\top \boldsymbol{e}_j \tag{4}$$

$$\overset{4}{=} \sum_{k=1}^{n} \operatorname{tr}(\boldsymbol{e}_i^\top [\boldsymbol{f}_k]_\times \boldsymbol{R}\boldsymbol{f}'_k (\boldsymbol{f}'_k)^\top \boldsymbol{R}^\top [\boldsymbol{f}_k]_\times^\top \boldsymbol{e}_j) \tag{5}$$

$$\overset{5}{=} \sum_{k=1}^{n} \operatorname{tr}(\boldsymbol{f}'_k (\boldsymbol{f}'_k)^\top \boldsymbol{R}^\top [\boldsymbol{f}_k]_\times^\top \boldsymbol{e}_j \boldsymbol{e}_i^\top [\boldsymbol{f}_k]_\times \boldsymbol{R}) \tag{6}$$

$$\overset{6}{=} \sum_{k=1}^{n} \operatorname{tr}(\boldsymbol{f}'_k (\boldsymbol{f}'_k)^\top \boldsymbol{R}^\top (\boldsymbol{f}_k \times \boldsymbol{e}_j)(\boldsymbol{f}_k \times \boldsymbol{e}_i)^\top \boldsymbol{R}) \tag{7}$$

$$\overset{7}{=} \sum_{k=1}^{n} \boldsymbol{r}^\top (\boldsymbol{f}'_k (\boldsymbol{f}'_k)^\top \otimes (\boldsymbol{f}_k \times \boldsymbol{e}_j)(\boldsymbol{f}_k \times \boldsymbol{e}_i)^\top) \boldsymbol{r} \tag{8}$$

$$\overset{8}{=} \boldsymbol{r}^\top \left(\sum_{k=1}^{n} (\boldsymbol{f}'_k \otimes (\boldsymbol{f}_k \times \boldsymbol{e}_j))(\boldsymbol{f}'_k \otimes (\boldsymbol{f}_k \times \boldsymbol{e}_i))^\top \right) \boldsymbol{r}. \tag{9}$$

A brief summary of the steps performed here follows:

1. Write the (i, j)-th element in terms of the complete matrix, multiplied by canonical vectors.
2. Substitute the definition of the covariance matrix (1).
3. Rewrite the cross product as $\boldsymbol{f}_k \times \boldsymbol{R}\boldsymbol{f}'_k = [\boldsymbol{f}_k]_\times \boldsymbol{R}\boldsymbol{f}'_k$, in terms of the corresponding skew-matrix.
4. The trace of a scalar quantity is the identity function.
5. Use the cyclic properties of the trace to rotate the terms.
6. Rewrite the cross-products back to its explicit form.
7. Apply vectorization, according to (123), to extract $\boldsymbol{r} = \operatorname{vec}(\boldsymbol{R})$ from the expression.
8. The sum of quadratic functions is a single quadratic function.

Thus, in view of the result above, given a list of feature correspondences $\{(\boldsymbol{f}_k, \boldsymbol{f}'_k)\}_{k=1}^N$, each element in the covariance matrix $\boldsymbol{M}(\boldsymbol{R})$ can be seen then as a a quadratic function $m_{ij}(\boldsymbol{R}) = \boldsymbol{r}^\top \boldsymbol{C}_{ij} \boldsymbol{r}$, where

$$\boldsymbol{C}_{ij} = \sum_{k=1}^{n} (\boldsymbol{f}'_k \otimes (\boldsymbol{f}_k \times \boldsymbol{e}_j))(\boldsymbol{f}'_k \otimes (\boldsymbol{f}_k \times \boldsymbol{e}_i))^\top. \tag{10}$$

This expression is beneficial, as it means that all the data in the problem can be condensed into a fixed number of matrices of small dimension, in linear time w.r.t. the number N of feature correspondences. The complexity of any subsequent operations should then be independent of N.

3. Data matrix for the quadratic objective in terms of $\text{vec}(\boldsymbol{X})$

In the main document we claim that the joint optimization objective

$$f(\boldsymbol{R}, \boldsymbol{t}) = \boldsymbol{t}^\top \boldsymbol{M}(\boldsymbol{R}) \boldsymbol{t} = \sum_{i,j=1}^{n} t_i (\boldsymbol{r}^\top \boldsymbol{C}_{ij} \boldsymbol{r}) t_j, \tag{11}$$

may be written as

$$f(\boldsymbol{R}, \boldsymbol{t}) = \sum_{i,j=1}^{n} (t_i \boldsymbol{r})^\top \boldsymbol{C}_{ij} (t_j \boldsymbol{r}) = \boldsymbol{x}^\top \boldsymbol{C} \boldsymbol{x}, \quad \text{s.t. } \boldsymbol{x} = \text{vec}(\boldsymbol{X}),\ \boldsymbol{X} = \boldsymbol{r}\boldsymbol{t}^\top, \tag{12}$$

where all the data in the problem has been collected into a single matrix $\boldsymbol{C}$. A proof follows:

$$f(\boldsymbol{R}, \boldsymbol{t}) \stackrel{1}{=} \boldsymbol{t}^\top \boldsymbol{M}(\boldsymbol{R}) \boldsymbol{t} \tag{13}$$

$$\stackrel{2}{=} \sum_{i,j=1}^{n} t_i \boldsymbol{r}^\top \boldsymbol{C}_{ij} \boldsymbol{r} t_j \tag{14}$$

$$\stackrel{3}{=} \sum_{i,j=1}^{n} \boldsymbol{e}_i^\top (\boldsymbol{t}\boldsymbol{r}^\top) \boldsymbol{C}_{ij} (\boldsymbol{r}\boldsymbol{t}^\top) \boldsymbol{e}_j \tag{15}$$

$$\stackrel{4}{=} \sum_{i,j=1}^{n} \text{tr}(\boldsymbol{e}_{ij}^\top \boldsymbol{X}^\top \boldsymbol{C}_{ij} \boldsymbol{X}) \tag{16}$$

$$\stackrel{5}{=} \boldsymbol{x}^\top \underbrace{\left(\sum_{i,j=1}^{n} \boldsymbol{e}_{ij} \otimes \boldsymbol{C}_{ij} \right)}_{\boldsymbol{C}} \boldsymbol{x} \tag{17}$$

The performed steps in the proof are:

1. This is the joint objective, corresponding to the Rayleigh quotient of matrix $\boldsymbol{M}(\boldsymbol{R})$.
2. Simply write the sum form of the quadratic form, substituting $m_{ij}(\boldsymbol{R}) = \boldsymbol{r}^\top \boldsymbol{C}_{ij} \boldsymbol{r}$ as argued in Section 2.
3. Write elements $t_k = \boldsymbol{e}_k^\top \boldsymbol{t}$ and regroup.
4. Identify $\boldsymbol{X} = \boldsymbol{r}\boldsymbol{t}^\top$, introduce trace and apply cyclic property. Also we make use of the definition $\boldsymbol{e}_{ij} = \boldsymbol{e}_i \boldsymbol{e}_j^\top$.
5. Vectorize the trace of a product using (123).

The data matrix $\boldsymbol{C}$ in the last expression gathers the 9×9 data matrices $\boldsymbol{C}_{ij}$ standing for every component m_{ij} into a single 27×27 matrix

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{C}_{11} & \boldsymbol{C}_{12} & \boldsymbol{C}_{13} \\ \boldsymbol{C}_{21} & \boldsymbol{C}_{22} & \boldsymbol{C}_{23} \\ \boldsymbol{C}_{31} & \boldsymbol{C}_{32} & \boldsymbol{C}_{33} \end{bmatrix} \in \text{Sym}_{27}. \tag{18}$$

4. Details on the standard QCQP formulation

As stated in the main document, any Quadratically Constrained Quadratic Programming (QCQP) problem instance (with equality constraints only), and in particular any QCQP characterization of our relative pose problem, may be written in the following generic form:

$$\min_{\tilde{\boldsymbol{z}}} = \tilde{\boldsymbol{z}}^\top \tilde{\boldsymbol{Q}}_0 \tilde{\boldsymbol{z}}, \quad \tilde{\boldsymbol{z}} = \left[1, \boldsymbol{z}^\top\right]^\top, \tag{19}$$

$$\text{s.t. } \tilde{\boldsymbol{z}}^\top \tilde{\boldsymbol{Q}}_i \tilde{\boldsymbol{z}} = 0, \quad i = 1, \dots, m, \tag{20}$$

where $\boldsymbol{z}$ is a vector stacking all unknowns involved in the problem, $\tilde{\boldsymbol{Q}}_0$ is the *homogeneized* data matrix and $\tilde{\boldsymbol{Q}}_i$, $i = 1, \dots, m$, are the *homogeneized* constraint matrices. *Homogeneization* here refers to the common trick of putting together all the terms in a quadratic function by homogeneizing the variable vector $\boldsymbol{z}$ with an additional unit element[1], so that at the end we can

[1]Common conventions are either to append [1] or prepend [3] the unit element. For convenience in the expressions we are prepending here.

regard it as if it was a plain *quadratic form* $q_{\tilde{\boldsymbol{Q}}_i}(\tilde{\boldsymbol{z}})$:

$$\boldsymbol{z}^\top \boldsymbol{Q}_i \boldsymbol{z} + 2\boldsymbol{b}_i^\top \boldsymbol{z} + c_i = \begin{bmatrix} 1 & \boldsymbol{z}^\top \end{bmatrix} \begin{bmatrix} c_i & \boldsymbol{b}_i^\top \\ \boldsymbol{b}_i & \boldsymbol{Q}_i \end{bmatrix} \begin{bmatrix} 1 \\ \boldsymbol{z} \end{bmatrix} = \tilde{\boldsymbol{z}}^\top \tilde{\boldsymbol{Q}}_i \tilde{\boldsymbol{z}} \equiv q_{\tilde{\boldsymbol{Q}}_i}(\tilde{\boldsymbol{z}}). \tag{21}$$

Since $\boldsymbol{z}$ stacks all the unknowns in the quadratic formulation, $\boldsymbol{R} \in \mathrm{SO}(3)$, $\boldsymbol{t} \in \mathrm{S}^2$, and $\boldsymbol{X} = \boldsymbol{r}\boldsymbol{t}^\top$, with the chosen representations this vector will have $\#(\boldsymbol{z}) = \#(\boldsymbol{R}) + \#(\boldsymbol{t}) + \#(\boldsymbol{X}) = 9 + 3 + 27 = 39$ elements, and thus $\#(\tilde{\boldsymbol{z}}) = 40$ elements. The corresponding 40×40 *homogeneized* matrices in the QCQP formulation may be regarded as divided into blocks. We will refer to the $(\boldsymbol{a}, \boldsymbol{b})$-block of a matrix $\tilde{\boldsymbol{Q}}$, noted as $\tilde{\boldsymbol{Q}}_{\boldsymbol{ab}}$ or $\tilde{\boldsymbol{Q}}[\boldsymbol{a}, \boldsymbol{b}]$, as the submatrix formed by the elements indexed by $\boldsymbol{a}$ and $\boldsymbol{b}$ (not necessarily contiguous):

$$\tilde{\boldsymbol{Q}}_{\boldsymbol{ab}} = \tilde{\boldsymbol{Q}}[\boldsymbol{a}, \boldsymbol{b}] = \left[Q_{ij}\right]_{i \in \boldsymbol{a}, j \in \boldsymbol{b}}. \tag{22}$$

If we consider *e.g*. the variable ordering $\boldsymbol{z} = \left[1, \boldsymbol{x}^\top, \boldsymbol{t}^\top, \boldsymbol{r}^\top\right]^\top$ (the concrete chosen order is irrelevant, and it just results in permutations on the $\tilde{\boldsymbol{Q}}_i$ matrices), a particular matrix $\tilde{\boldsymbol{Q}}$ can be seen then as

$$\begin{array}{c} \\ 1 \\ \boldsymbol{x} \\ \boldsymbol{t} \\ \boldsymbol{r} \end{array} \begin{array}{c} \begin{array}{cccc} 1 & \boldsymbol{x} & \boldsymbol{t} & \boldsymbol{r} \end{array} \\ \begin{pmatrix} c & \boldsymbol{b}_{\boldsymbol{x}}^\top & \boldsymbol{b}_{\boldsymbol{t}}^\top & \boldsymbol{b}_{\boldsymbol{r}}^\top \\ \boldsymbol{b}_{\boldsymbol{x}} & \boldsymbol{Q}_{\boldsymbol{xx}} & \boldsymbol{Q}_{\boldsymbol{xt}} & \boldsymbol{Q}_{\boldsymbol{xr}} \\ \boldsymbol{b}_{\boldsymbol{t}} & \boldsymbol{Q}_{\boldsymbol{tx}} & \boldsymbol{Q}_{\boldsymbol{tt}} & \boldsymbol{Q}_{\boldsymbol{tr}} \\ \boldsymbol{b}_{\boldsymbol{r}} & \boldsymbol{Q}_{\boldsymbol{rx}} & \boldsymbol{Q}_{\boldsymbol{rt}} & \boldsymbol{Q}_{\boldsymbol{rr}} \end{pmatrix} \end{array} \equiv \tilde{\boldsymbol{Q}}. \tag{23}$$

Next, we will obtain the concrete expression for the matrices $\tilde{\boldsymbol{Q}}_i$ in the QCQP problem corresponding to the optimization objective and the specific constraints considered in our problem, which in the most general, redundantly constrained case, come from the extended sets $\{\hat{\mathcal{C}}_{\boldsymbol{R}}, \hat{\mathcal{C}}_{\boldsymbol{t}}, \hat{\mathcal{C}}_{\boldsymbol{X}}\}$ characterized in the main document.

For the subsequent steps, it will be particularly convenient to introduce the *homogeneized* auxiliary variable

$$\tilde{\boldsymbol{X}} = \tilde{\boldsymbol{r}}\tilde{\boldsymbol{t}}^\top = \begin{bmatrix} 1 & \boldsymbol{t}^\top \\ \boldsymbol{r} & \boldsymbol{r}\boldsymbol{t}^\top \end{bmatrix} = \begin{bmatrix} 1 & \boldsymbol{t}^\top \\ \boldsymbol{r} & \boldsymbol{X} \end{bmatrix}, \; \tilde{\boldsymbol{r}} = \begin{bmatrix} 1 \\ \mathrm{vec}(\boldsymbol{R}) \end{bmatrix}, \; \tilde{\boldsymbol{t}} = \begin{bmatrix} 1 \\ \boldsymbol{t} \end{bmatrix}, \tag{24}$$

and choose the variable ordering $\tilde{\boldsymbol{z}}$ induced by the vectorization of $\tilde{\boldsymbol{X}}$, so that

$$\tilde{\boldsymbol{z}} = \mathrm{vec}(\tilde{\boldsymbol{X}}) = \tilde{\boldsymbol{t}} \otimes \tilde{\boldsymbol{r}}. \tag{25}$$

We will also make extensive use of some basic vector algebra, canonical vectors and vectorization that are documented in Appendices A and B.

4.1. Objective matrix: $\tilde{\boldsymbol{Q}}_0$

This is straightforward, as the objective $\boldsymbol{x}^\top \boldsymbol{C} \boldsymbol{x} = \tilde{\boldsymbol{z}}^\top \tilde{\boldsymbol{Q}}_0 \tilde{\boldsymbol{z}}$ clearly corresponds to a matrix $\tilde{\boldsymbol{Q}}_0$ which is zero everywhere except for the $(\boldsymbol{x}, \boldsymbol{x})$-block, whose value is $\tilde{\boldsymbol{Q}}_0[\boldsymbol{x}, \boldsymbol{x}] = \boldsymbol{C}$.

4.2. Constraint matrices for all sphere constraints: $\hat{\mathcal{C}}_{\boldsymbol{t}}$

Consider to begin with the simple quadratic constraint characterizing the sphere constraint

$$\boldsymbol{t}^\top \boldsymbol{t} = 1 \Rightarrow q_{\tilde{\boldsymbol{P}}}(\tilde{\boldsymbol{t}}) = \tilde{\boldsymbol{t}}^\top \underbrace{\begin{bmatrix} -1 & \boldsymbol{0}^\top \\ \boldsymbol{0} & \boldsymbol{I}_3 \end{bmatrix}}_{\tilde{\boldsymbol{P}}} \tilde{\boldsymbol{t}} = 0, \tag{26}$$

where $\tilde{\boldsymbol{P}}$ is the 4×4 matrix corresponding to this constraint when seen as a quadratic form of $\tilde{\boldsymbol{t}}$.

The lifting trick proposed in the main document is to multiply the original constraint $q_{\tilde{\boldsymbol{P}}}(\tilde{\boldsymbol{t}}) = 0$ by either linear factors r_i for $i = 1, \ldots, 9$ or quadratic factors $r_i r_j$ for $i, j = 1, \ldots, 9$ to obtain new valid redundant constraints. By using the convenient homogeneized counterparts $\tilde{\boldsymbol{r}}$, $\tilde{\boldsymbol{t}}$ and $\tilde{\boldsymbol{X}}$ proposed in (24), all of these constraints may be conveniently characterized as the set

$$\tilde{r}_i \cdot \tilde{r}_j \cdot q_{\tilde{\boldsymbol{P}}}(\tilde{\boldsymbol{t}}) = 0, \quad \forall i, j = 1, \ldots, 10. \tag{27}$$

Note there are 10 indexes, as the homogeneous component of $\tilde{\boldsymbol{r}}$ is also featured in the products above. Indeed, when $i = j = 1$ we get the original quadratic constraint $q_{\tilde{\boldsymbol{P}}}(\tilde{\boldsymbol{t}}) = 0$, if $i = 1$, $j \neq 1$ we get the cubic lifts $\tilde{r}_j \cdot q_{\tilde{\boldsymbol{P}}}(\tilde{\boldsymbol{t}}) = 0$, and finally $i \neq 1$, $j \neq 1$ leads to the quartic lifts. Next we characterize the constraint matrices ${}^{\mathrm{S}^2}\tilde{\boldsymbol{Q}}_{ij}$ for *all* these possible variations of the sphere constraint (27), under the QCQP formulation (20) and with the natural ordering $\tilde{\boldsymbol{z}} = \mathrm{vec}(\tilde{\boldsymbol{X}})$ (25):

$$\begin{aligned}
\tilde{r}_i \cdot \tilde{r}_j \cdot q_{\tilde{\boldsymbol{P}}}(\tilde{\boldsymbol{t}}) &= (\tilde{r}_i \tilde{r}_j) \tilde{\boldsymbol{t}}^\top \tilde{\boldsymbol{P}} \tilde{\boldsymbol{t}} && (28)\\
&= (\tilde{r}_i \tilde{\boldsymbol{t}}^\top) \tilde{\boldsymbol{P}} (\tilde{\boldsymbol{t}} \tilde{r}_j) && (29)\\
&= \boldsymbol{e}_i^\top (\tilde{\boldsymbol{r}} \tilde{\boldsymbol{t}}^\top) \tilde{\boldsymbol{P}} (\tilde{\boldsymbol{t}} \tilde{\boldsymbol{r}}^\top) \boldsymbol{e}_j && (30)\\
&= \boldsymbol{e}_i^\top \tilde{\boldsymbol{X}} \tilde{\boldsymbol{P}} \tilde{\boldsymbol{X}}^\top \boldsymbol{e}_j && (31)\\
&= \mathrm{tr}(\tilde{\boldsymbol{P}} \tilde{\boldsymbol{X}}^\top \boldsymbol{e}_j \boldsymbol{e}_i^\top \tilde{\boldsymbol{X}}) && (32)\\
&= \mathrm{vec}(\tilde{\boldsymbol{X}})^\top (\tilde{\boldsymbol{P}} \otimes \boldsymbol{e}_{ji}) \, \mathrm{vec}(\tilde{\boldsymbol{X}}) && (33)\\
&= \tilde{\boldsymbol{z}}^\top \underbrace{(\tilde{\boldsymbol{P}} \otimes \boldsymbol{e}_{ji})}_{{}^{\mathrm{S}^2}\tilde{\boldsymbol{Q}}_{ij}} \tilde{\boldsymbol{z}} = 0. && (34)
\end{aligned}$$

Note that since we are dealing with quadratic functions, there are infinitely many representations for the quadratic form. In particular, we choose the usual symmetric one, and the extended set of all constraint matrices stemming from the sphere constraint for the QCQP problem is given by the 40×40 matrices

$$ {}^{\mathrm{S}^2}\tilde{\boldsymbol{Q}}_{ij} = \underbrace{\begin{bmatrix} -1 & \boldsymbol{0}^\top \\ \boldsymbol{0} & \boldsymbol{I}_3 \end{bmatrix}}_{4\times4} \otimes \underbrace{\left(\frac{\boldsymbol{e}_{ij} + \boldsymbol{e}_{ji}}{2} \right)}_{10\times10}, \quad \forall i = 1, \ldots, 10, j = i, \ldots, 10. \tag{35}$$

Due to the symmetry above, this results overall in $\binom{10}{2} = 55$ independent constraints. Otherwise stated, the family of lifted sphere constraints may be written in terms of the canonical vectors for the space Sym_{10} (see App. A):

$$ {}^{\mathrm{S}^2}\tilde{\boldsymbol{Q}}_{ij} = \begin{bmatrix} -1 & \boldsymbol{0}^\top \\ \boldsymbol{0} & \boldsymbol{I}_3 \end{bmatrix} \otimes ({}^{+}_{10}\boldsymbol{e}_{ij}), \quad \forall ({}^{+}_{10}\boldsymbol{e}_{ij}) \in \mathcal{B}(\mathrm{Sym}_{10}). \tag{36}$$

4.3. Constraint matrices for all rotation constraints: $\hat{\mathcal{C}}_{\boldsymbol{R}}$

The set of redundant constraints chosen to represent $\mathrm{SO}(3)$ in this work is the same as for [1],

$$ \mathcal{C}_{\boldsymbol{R}} \equiv \begin{cases} \boldsymbol{R}^\top \boldsymbol{R} = \boldsymbol{I}_3, \quad \boldsymbol{R}\boldsymbol{R}^\top = \boldsymbol{I}_3, \\ (\boldsymbol{R}\boldsymbol{e}_i) \times (\boldsymbol{R}\boldsymbol{e}_j) = (\boldsymbol{R}\boldsymbol{e}_k), \\ \quad \forall (i,j,k) = \{(1,2,3),(2,3,1),(3,1,2)\}, \end{cases} \tag{37}$$

which features only quadratic constraints. Thus, every rotation constraint can be written as

$$ q_{\tilde{\boldsymbol{P}}_k}(\tilde{\boldsymbol{r}}) = \tilde{\boldsymbol{r}}^\top \tilde{\boldsymbol{P}}_k \tilde{\boldsymbol{r}} = 0, \quad k \in \mathcal{C}_{\boldsymbol{R}}. \tag{38}$$

The particular expression for each 10×10 constraint matrix $\tilde{\boldsymbol{P}}_k$ is featured in the supplementary material for [1], and we revisit them next for completeness.

Orthonormality of rotation columns The common matrix constraint enforcing orthonormal columns,

$$ \boldsymbol{R}^\top \boldsymbol{R} = \boldsymbol{I}_3 \to \boldsymbol{R}^\top \boldsymbol{R} - \boldsymbol{I}_3 = \boldsymbol{0}_3, $$

may be seen as 9 scalar constraints indexed by $(i,j) \in \{1,2,3\} \times \{1,2,3\}$:

$$\begin{aligned}
q_{\tilde{\boldsymbol{P}}^c_{ij}}(\tilde{\boldsymbol{r}}) &= \boldsymbol{e}_i^\top (\boldsymbol{R}^\top \boldsymbol{R} - \boldsymbol{I}_3) \boldsymbol{e}_j && (39)\\
&= \tilde{\boldsymbol{r}}^\top \underbrace{\begin{bmatrix} -\delta_{ij} & \boldsymbol{0}^\top \\ \boldsymbol{0} & \boldsymbol{e}_{ij} \otimes \boldsymbol{I}_3 \end{bmatrix}}_{\tilde{\boldsymbol{P}}^c_{ij}} \tilde{\boldsymbol{r}} = 0. && (40)
\end{aligned}$$

Here, δ_{ij} stands for the Kronecker delta whose value is 1 only if $i = j$, and 0 otherwise.

Part of these constraints are actually equivalent, and after symmetrization of the corresponding quadratic forms we see the set of matrix constraints can be characterized as

$$\tilde{\boldsymbol{P}}^c_{ij} = \begin{bmatrix} -\delta_{ij} & \boldsymbol{0}^\top \\ \boldsymbol{0} & ({}^+_3\boldsymbol{e}_{ij}) \otimes \boldsymbol{I}_3 \end{bmatrix}, \quad \forall({}^+_3\boldsymbol{e}_{ij}) \in \mathcal{B}(\mathrm{Sym}_3), \tag{41}$$

which corresponds to $\binom{3}{2} = 6$ linearly independent matrices.

Orthonormality of rotation rows Akin to the rotation matrix columns, the rotation matrix rows must be orthonormal,

$$\boldsymbol{R}\boldsymbol{R}^\top = \boldsymbol{I}_3 \rightarrow \boldsymbol{R}\boldsymbol{R}^\top - \boldsymbol{I}_3 = \boldsymbol{0}_3,$$

which again amounts to 9 scalar constraints

$$q_{\tilde{\boldsymbol{P}}^r_{ij}}(\tilde{\boldsymbol{r}}) = \boldsymbol{e}_i^\top(\boldsymbol{R}\boldsymbol{R}^\top - \boldsymbol{I}_3)\boldsymbol{e}_j \tag{42}$$

$$= \tilde{\boldsymbol{r}}^\top \underbrace{\begin{bmatrix} -\delta_{ij} & \boldsymbol{0}^\top \\ \boldsymbol{0} & \boldsymbol{I}_3 \otimes \boldsymbol{e}_{ij} \end{bmatrix}}_{\tilde{\boldsymbol{P}}^r_{ij}} \tilde{\boldsymbol{r}} = 0, \tag{43}$$

of which, after symmetrization, only $\binom{3}{2} = 6$ linearly independent ones remain:

$$\tilde{\boldsymbol{P}}^r_{ij} = \begin{bmatrix} -\delta_{ij} & \boldsymbol{0}^\top \\ \boldsymbol{0} & \boldsymbol{I}_3 \otimes ({}^+_3\boldsymbol{e}_{ij}) \end{bmatrix}, \quad \forall({}^+_3\boldsymbol{e}_{ij}) \in \mathcal{B}(\mathrm{Sym}_3). \tag{44}$$

Right-hand rule on rotation columns The well-known right-hand rule, which features the chirality constraint on the rotation columns[2]

$$(\boldsymbol{R}\boldsymbol{e}_i) \times (\boldsymbol{R}\boldsymbol{e}_j) = \boldsymbol{R}\boldsymbol{e}_k, \quad (i,j,k) \in \{(1,2,3),(2,3,1),(3,1,2)\}, \tag{45}$$

provides 3 scalar constraints for each ijk triplet. This amounts to $3 \cdot 3 = 9$ scalar scalar constraints, each of which is writable as an homogeneized quadratic form

$$q_{\tilde{\boldsymbol{P}}^d_{ijk\alpha}}(\tilde{\boldsymbol{r}}) = \boldsymbol{e}_\alpha^\top\left(-(\boldsymbol{R}\boldsymbol{e}_i) \times (\boldsymbol{R}\boldsymbol{e}_j) + \boldsymbol{R}\boldsymbol{e}_k\right) \tag{46}$$

$$= \tilde{\boldsymbol{r}}^\top \underbrace{\begin{bmatrix} 0 & (\boldsymbol{e}_k \otimes \boldsymbol{e}_\alpha)^\top \\ \boldsymbol{0} & \boldsymbol{e}_{ij} \otimes [\boldsymbol{e}_\alpha]_\times \end{bmatrix}}_{\tilde{\boldsymbol{P}}^d_{ijk\alpha}} \tilde{\boldsymbol{r}} = 0, \tag{47}$$

or, after symmetrization, the constraint matrices

$$\tilde{\boldsymbol{P}}^d_{ijk\alpha} = \begin{bmatrix} 0 & \frac{1}{2}(\boldsymbol{e}_k \otimes \boldsymbol{e}_\alpha)^\top \\ \frac{1}{2}(\boldsymbol{e}_k \otimes \boldsymbol{e}_\alpha) & ({}^-_3\boldsymbol{e}_{ij}) \otimes [\boldsymbol{e}_\alpha]_\times \end{bmatrix}, \tag{48}$$

$$\forall(i,j,k) \in \{(1,2,3),(2,3,1),(3,1,2)\},\ \alpha = 1,2,3. \tag{49}$$

Note that with the defined canonical basis elements for Skew_3, ${}^-_3\boldsymbol{e}_{ij} = \frac{1}{2}({}_3\boldsymbol{e}_{ij} - {}_3\boldsymbol{e}_{ji})$, we have the equivalence

Right-hand rule on rotation rows For rotation matrix rows, the right hand rule states

$$(\boldsymbol{R}^\top\boldsymbol{e}_i) \times (\boldsymbol{R}^\top\boldsymbol{e}_j) = \boldsymbol{R}^\top\boldsymbol{e}_k, \quad (i,j,k) \in \{(1,2,3),(2,3,1),(3,1,2)\}, \tag{50}$$

[2] It may be also applied to rotation rows, but the corresponding constraints are linearly related to those provided by column relations only [1].

and similarly we get the quadratic constraints

$$q_{\tilde{\boldsymbol{P}}^d_{ijk\alpha}}(\tilde{\boldsymbol{r}}) = \boldsymbol{e}_\alpha^\top \left(-(\boldsymbol{R}^\top \boldsymbol{e}_i) \times (\boldsymbol{R}^\top \boldsymbol{e}_j) + \boldsymbol{R}^\top \boldsymbol{e}_k \right) \tag{51}$$

$$= \tilde{\boldsymbol{r}}^\top \underbrace{\begin{bmatrix} 0 & (\boldsymbol{e}_\alpha \otimes \boldsymbol{e}_k)^\top \\ \boldsymbol{0} & [\boldsymbol{e}_\alpha]_\times \otimes \boldsymbol{e}_{ij} \end{bmatrix}}_{\tilde{\boldsymbol{P}}^d_{ijk\alpha}} \tilde{\boldsymbol{r}} = 0, \tag{52}$$

or, after symmetrization, the constraint matrices

$$\tilde{\boldsymbol{P}}^d_{ijk\alpha} = \begin{bmatrix} 0 & \frac{1}{2}(\boldsymbol{e}_\alpha \otimes \boldsymbol{e}_k)^\top \\ \frac{1}{2}(\boldsymbol{e}_\alpha \otimes \boldsymbol{e}_k) & [\boldsymbol{e}_\alpha]_\times \otimes ({}^{-}_{3}\boldsymbol{e}_{ij}) \end{bmatrix}, \tag{53}$$

$$\forall (i,j,k) \in \{(1,2,3),(2,3,1),(3,1,2)\},\ \alpha = 1,2,3. \tag{54}$$

Comparing this expression to that obtained for cross-product of rotation columns it may be shown both sets of constraints are exactly the same, so we only keep *e.g.* those from rotation columns.

Lifted constraints Here, we proceed similarly to the sphere constraint case: For each constraint in (38), we may build lifted versions by multiplying either by $\tilde{t}_i$ or $\tilde{t}_i\tilde{t}_j$. Again, all the possible cases are characterized by

$$\tilde{t}_i \cdot \tilde{t}_j \cdot q_{\tilde{\boldsymbol{P}}_k}(\tilde{\boldsymbol{r}}) = 0, \quad \forall i,j = 1,\dots,4;\ k \in \mathcal{C}_{\boldsymbol{R}}. \tag{55}$$

The corresponding constraint matrices ${}^{\mathrm{SO}(3)}\tilde{\boldsymbol{Q}}_{ijk}$ in our standard QCQP framework (20), with the natural ordering $\tilde{\boldsymbol{z}} = \mathrm{vec}(\tilde{\boldsymbol{X}})$ (25), would read

$$\tilde{t}_i \cdot \tilde{t}_j \cdot q_{\tilde{\boldsymbol{P}}_k}(\tilde{\boldsymbol{r}}) = (\tilde{t}_i\tilde{t}_j)\tilde{\boldsymbol{r}}^\top \tilde{\boldsymbol{P}}_k \tilde{\boldsymbol{r}} \tag{56}$$

$$= (\tilde{t}_i\tilde{\boldsymbol{r}}^\top)\tilde{\boldsymbol{P}}_k(\tilde{\boldsymbol{r}}\tilde{t}_j) \tag{57}$$

$$= \boldsymbol{e}_i^\top(\boldsymbol{t}\tilde{\boldsymbol{r}}^\top)\tilde{\boldsymbol{P}}_k(\tilde{\boldsymbol{r}}\tilde{\boldsymbol{t}}^\top)\boldsymbol{e}_j \tag{58}$$

$$= \boldsymbol{e}_i^\top \tilde{\boldsymbol{X}}^\top \tilde{\boldsymbol{P}}_k \tilde{\boldsymbol{X}} \boldsymbol{e}_j \tag{59}$$

$$= \mathrm{tr}(\boldsymbol{e}_{ij}^\top \tilde{\boldsymbol{X}}^\top \tilde{\boldsymbol{P}}_k \tilde{\boldsymbol{X}}) \tag{60}$$

$$= \mathrm{vec}(\tilde{\boldsymbol{X}})^\top (\boldsymbol{e}_{ij} \otimes \tilde{\boldsymbol{P}}_k)\,\mathrm{vec}(\tilde{\boldsymbol{X}}) \tag{61}$$

$$= \tilde{\boldsymbol{z}}^\top \underbrace{(\boldsymbol{e}_{ij} \otimes \tilde{\boldsymbol{P}}_k)}_{{}^{\mathrm{SO}(3)}\tilde{\boldsymbol{Q}}_{ijk}} \tilde{\boldsymbol{z}} = 0. \tag{62}$$

Again, after symmetrization, this results in the 40×40 constraint matrices

$${}^{\mathrm{SO}(3)}\tilde{\boldsymbol{Q}}_{ijk} = \underbrace{\left(\frac{\boldsymbol{e}_{ij} + \boldsymbol{e}_{ji}}{2} \right)}_{4\times 4} \otimes \underbrace{\tilde{\boldsymbol{P}}_k}_{10\times 10}, \quad \forall i = 1,\dots,4, j = i,\dots,4, k \in \mathcal{C}_{\boldsymbol{R}}, \tag{63}$$

resulting in $\binom{4}{2} = 10$ independent constraints for each $k \in \mathcal{C}_{\boldsymbol{R}}$. Otherwise stated, the family of lifted rotation constraints may be written in terms of the canonical vectors for the space Sym_4 (see App. A):

$${}^{\mathrm{SO}(3)}\tilde{\boldsymbol{Q}}_{ijk} = ({}^{+}_{4}\boldsymbol{e}_{ij}) \otimes \tilde{\boldsymbol{P}}_k, \quad \forall ({}^{+}_{4}\boldsymbol{e}_{ij}) \in \mathcal{B}(\mathrm{Sym}_4). \tag{64}$$

There are 20 independent constraints in the rotation constraint set proposed in [1], so this lifting leads to an overall of $10 \times 20 = 200$ independent constraints.

4.4. Constraint matrices for all constraints on auxiliary variable: $\hat{\mathcal{C}}_{\boldsymbol{X}}$

Finally, in the main document it has been proposed that besides the quadratic definition constraint $\boldsymbol{X} = \boldsymbol{r}\boldsymbol{t}^\top$, we also may feature the constraint $\mathrm{rank}(\boldsymbol{X}) = 1$ through the condition that every 2×2 minor in $\boldsymbol{X}$ has zero determinant:

$$\begin{vmatrix} X_{ij} & X_{ij'} \\ X_{i'j} & X_{i'j'} \end{vmatrix} = X_{ij}X_{i'j'} - X_{ij'}X_{i'j} = 0, \tag{65}$$

$$\text{s.t. } i = 1, \dots, 9;\ i' = i, \dots, 9; \tag{66}$$

$$j = 1, \dots, 3;\ j' = j, \dots, 3. \tag{67}$$

In fact, we may characterize the definition and $\mathrm{rank}(\boldsymbol{X}) = 1$ constraints jointly through the constraint $\mathrm{rank}(\tilde{\boldsymbol{X}}) = 1$. This way, all the constraints in the set $\hat{\mathcal{C}}_{\boldsymbol{X}}$ may be written as

$$\begin{vmatrix} \tilde{X}_{ij} & \tilde{X}_{ij'} \\ \tilde{X}_{i'j} & \tilde{X}_{i'j'} \end{vmatrix} = \tilde{X}_{ij}\tilde{X}_{i'j'} - \tilde{X}_{ij'}\tilde{X}_{i'j} = 0, \tag{68}$$

$$\text{s.t. } i = 1, \dots, 10;\ i' = i, \dots, 10; \tag{69}$$

$$j = 1, \dots, 4;\ j' = j, \dots, 4, \tag{70}$$

and the constraint $\boldsymbol{X} = \boldsymbol{r}\boldsymbol{t}^\top$ corresponds to the particular $3 \cdot 9 = 27$ cases with $i = j = 1$.

Building upon this convenient general formulation we obtain the corresponding constraint matrices ${}^{X}\tilde{\boldsymbol{Q}}_{jj'ii'}$ for our general QCQP framework:

$$\begin{aligned}
\begin{vmatrix} \tilde{X}_{ij} & \tilde{X}_{ij'} \\ \tilde{X}_{i'j} & \tilde{X}_{i'j'} \end{vmatrix} &= \tilde{X}_{ij}\tilde{X}_{i'j'} - \tilde{X}_{ij'}\tilde{X}_{i'j} && (71)\\
&= (\boldsymbol{e}_i^\top \tilde{\boldsymbol{X}} \boldsymbol{e}_j)(\boldsymbol{e}_{i'}^\top \tilde{\boldsymbol{X}} \boldsymbol{e}_{j'}) - (\boldsymbol{e}_i^\top \tilde{\boldsymbol{X}} \boldsymbol{e}_{j'})(\boldsymbol{e}_{i'}^\top \tilde{\boldsymbol{X}} \boldsymbol{e}_j) && (72)\\
&= (\boldsymbol{e}_j^\top \tilde{\boldsymbol{X}}^\top \boldsymbol{e}_i)(\boldsymbol{e}_{i'}^\top \tilde{\boldsymbol{X}} \boldsymbol{e}_{j'}) - (\boldsymbol{e}_{j'}^\top \tilde{\boldsymbol{X}}^\top \boldsymbol{e}_i)(\boldsymbol{e}_{i'}^\top \tilde{\boldsymbol{X}} \boldsymbol{e}_j) && (73)\\
&= \mathrm{tr}(\boldsymbol{e}_{jj'}^\top \tilde{\boldsymbol{X}}^\top \boldsymbol{e}_{ii'} \tilde{\boldsymbol{X}} - \boldsymbol{e}_{jj'} \tilde{\boldsymbol{X}}^\top \boldsymbol{e}_{ii'} \tilde{\boldsymbol{X}}) && (74)\\
&= \mathrm{tr}((\boldsymbol{e}_{jj'}^\top - \boldsymbol{e}_{jj'}) \tilde{\boldsymbol{X}}^\top \boldsymbol{e}_{ii'} \tilde{\boldsymbol{X}}) && (75)\\
&= \mathrm{tr}((\boldsymbol{e}_{jj'} - \boldsymbol{e}_{jj'}^\top)^\top \tilde{\boldsymbol{X}}^\top \boldsymbol{e}_{ii'} \tilde{\boldsymbol{X}}) && (76)\\
&= \tilde{\boldsymbol{z}}^\top \underbrace{\left((\boldsymbol{e}_{jj'} - \boldsymbol{e}_{jj'}^\top) \otimes \boldsymbol{e}_{ii'}\right)}_{{}^{X}\tilde{\boldsymbol{Q}}_{jj'ii'}} \tilde{\boldsymbol{z}} = 0 && (77)
\end{aligned}$$

Scaling the constraint matrix above we may write ${}^{X}\tilde{\boldsymbol{Q}}_{jj'ii'} \sim ({}^{-}_{4}\boldsymbol{e}_{jj'}) \otimes ({}_{10}\boldsymbol{e}_{ii'})$ (see App. A for notation), and this in turn may be equivalently symmetrized into

$$\begin{aligned}
{}^{X}\tilde{\boldsymbol{Q}}_{jj'ii'} &= \mathrm{sym}({}^{-}_{4}\boldsymbol{e}_{jj'} \otimes {}_{10}\boldsymbol{e}_{ii'}) && (78)\\
&= \frac{1}{2}\left({}^{-}_{4}\boldsymbol{e}_{jj'} \otimes {}_{10}\boldsymbol{e}_{ii'} + {}^{-}_{4}\boldsymbol{e}_{jj'}^\top \otimes {}_{10}\boldsymbol{e}_{ii'}^\top\right) && (79)\\
&= \frac{1}{2}\left({}^{-}_{4}\boldsymbol{e}_{jj'} \otimes {}_{10}\boldsymbol{e}_{ii'} - {}^{-}_{4}\boldsymbol{e}_{jj'} \otimes {}_{10}\boldsymbol{e}_{ii'}^\top\right) && (80)\\
&= \frac{1}{2}\left({}^{-}_{4}\boldsymbol{e}_{jj'} \otimes ({}_{10}\boldsymbol{e}_{ii'} - {}_{10}\boldsymbol{e}_{ii'}^\top)\right) && (81)\\
&= {}^{-}_{4}\boldsymbol{e}_{jj'} \otimes \frac{1}{2}({}_{10}\boldsymbol{e}_{ii'} - {}_{10}\boldsymbol{e}_{ii'}^\top) && (82)\\
&= ({}^{-}_{4}\boldsymbol{e}_{jj'}) \otimes ({}^{-}_{10}\boldsymbol{e}_{ii'}). && (83)
\end{aligned}$$

The matrix above is clearly symmetric:

$${}^{X}\tilde{\boldsymbol{Q}}_{jj'ii'}^\top = {}^{-}_{4}\boldsymbol{e}_{jj'}^\top \otimes {}^{-}_{10}\boldsymbol{e}_{ii'}^\top = (-\,{}^{-}_{4}\boldsymbol{e}_{jj'}) \otimes (-\,{}^{-}_{10}\boldsymbol{e}_{ii'}) = {}^{X}\tilde{\boldsymbol{Q}}_{jj'ii'}. \tag{84}$$

Once again, the whole family of lifted constraints may be written in terms of the canonical vectors of a suitable space, in this case the Cartesian product $\text{Skew}_4 \times \text{Skew}_{10}$ (see App. A for details):

$$ {}^{X}\tilde{\boldsymbol{Q}}_{jj'ii'} = ({}^{-}_{4}\boldsymbol{e}_{jj'}) \otimes ({}^{-}_{10}\boldsymbol{e}_{ii'}), \quad \forall ({}^{-}_{4}\boldsymbol{e}_{jj'}) \in \mathcal{B}(\text{Skew}_4), ({}^{-}_{10}\boldsymbol{e}_{ii'}) \in \mathcal{B}(\text{Skew}_{10}). \tag{85} $$

From this characterization in terms of the canonical vectors for Skew_4 and Skew_{10}, it becomes fairly clear that the number of linearly independent constraints that get generated by these expressions is $\dim(\text{Skew}_4) \cdot \dim(\text{Skew}_{10}) = 6 \cdot 45 = 270$.

4.5. Summary

Collecting all results in the sections above, we conclude that the extended set of quadratic constraints corresponding to $\{\hat{\mathcal{C}}_{\boldsymbol{R}}, \hat{\mathcal{C}}_{\boldsymbol{t}}, \hat{\mathcal{C}}_{\boldsymbol{X}}\}$ results in the following specific sets of quadratic matrices for the QCQP problem (20) in its standard form:

$$ \hat{\mathcal{C}}_{\boldsymbol{t}} \Rightarrow {}^{\mathrm{S}^2}\tilde{\boldsymbol{Q}}_{ij} = \begin{bmatrix} -1 & \boldsymbol{0}^\top \\ \boldsymbol{0} & \boldsymbol{I}_3 \end{bmatrix} \otimes {}^{+}_{10}\boldsymbol{e}_{ij}, \quad i = 1, \dots, 10; \; j = i, \dots, 10, \tag{86} $$

$$ \hat{\mathcal{C}}_{\boldsymbol{R}} \Rightarrow {}^{\mathrm{SO}(3)}\tilde{\boldsymbol{Q}}_{ijk} = {}^{+}_{4}\boldsymbol{e}_{ij} \otimes \tilde{\boldsymbol{P}}_k, \quad i = 1, \dots, 4; \; j = i, \dots, 4; \; \tilde{\boldsymbol{P}}_k \in \mathcal{C}_{\boldsymbol{R}}, \tag{87} $$

$$ \hat{\mathcal{C}}_{\boldsymbol{X}} \Rightarrow {}^{X}\tilde{\boldsymbol{Q}}_{jj'ii'} = {}^{-}_{4}\boldsymbol{e}_{jj'} \otimes {}^{-}_{10}\boldsymbol{e}_{ii'}, \quad i = 1, \dots, 10; \; i' = i, \dots, 10; \; j = 1, \dots, 4; \; j' = j, \dots, 4. \tag{88} $$

5. Symmetries in the algebraic error

Consider the squared algebraic error used in our main document. In the objective to optimize (11), each pair $(\boldsymbol{f}_i, \boldsymbol{f}'_i)$ produces a squared algebraic error term given by

$$ e_i^2 = \boldsymbol{t}^\top (\boldsymbol{n}_i \boldsymbol{n}_i^\top) \boldsymbol{t} = (\boldsymbol{t}^\top \boldsymbol{n}_i)^2 = (\boldsymbol{t} \cdot (\boldsymbol{f}_i \times (\boldsymbol{R}\boldsymbol{f}'_i)))^2. \tag{89} $$

This, in turn, is just the triple product of three vectors

$$ e_i^2 = \det([\boldsymbol{t} \mid \boldsymbol{f}_i \mid \boldsymbol{R}\boldsymbol{f}'_i])^2. \tag{90} $$

The error term $e_i^2 = e_i^2(\boldsymbol{R}, \boldsymbol{t})$ presents two symmetries w.r.t. the parameters $\boldsymbol{t}$ and $\boldsymbol{R}$:

$$ e_i^2(\boldsymbol{R}, -\boldsymbol{t}) = e_i^2(\boldsymbol{R}, \boldsymbol{t}), \tag{91} $$

$$ e_i^2(\boldsymbol{P}_{\boldsymbol{t}}\boldsymbol{R}, \boldsymbol{t}) = e_i^2(\boldsymbol{R}, \boldsymbol{t}). \tag{92} $$

The first symmetry (91) is straightforward as

$$ e_i^2(\boldsymbol{R}, -\boldsymbol{t}) = \det([-\boldsymbol{t} \mid \boldsymbol{f}_i \mid \boldsymbol{R}\boldsymbol{f}'_i])^2 \tag{93} $$

$$ = (-1)^2 e_i^2(\boldsymbol{R}, \boldsymbol{t}) = e_i^2(\boldsymbol{R}, \boldsymbol{t}). \tag{94} $$

For the second symmetry (92), substituting the reflection matrix $\boldsymbol{P}_{\boldsymbol{t}} = 2\boldsymbol{t}\boldsymbol{t}^\top - \boldsymbol{I}_3$ we get

$$ e_i^2(\boldsymbol{P}_{\boldsymbol{t}}\boldsymbol{R}, \boldsymbol{t}) = \det([\boldsymbol{t} \mid \boldsymbol{f}_i \mid \boldsymbol{P}_{\boldsymbol{t}}\boldsymbol{R}\boldsymbol{f}'_i])^2 \tag{95} $$

$$ = \det([\boldsymbol{t} \mid \boldsymbol{f}_i \mid (2\boldsymbol{t}\boldsymbol{t}^\top - \boldsymbol{I}_3)\boldsymbol{f}'_i])^2 \tag{96} $$

$$ = \left(\det([\boldsymbol{t} \mid \boldsymbol{f}_i \mid (2(\boldsymbol{t} \cdot \boldsymbol{f}'_i)\boldsymbol{t}]) + \det([\boldsymbol{t} \mid \boldsymbol{f}_i \mid -\boldsymbol{R}\boldsymbol{f}'_i])\right)^2 \tag{97} $$

$$ = \left(0 - \det([\boldsymbol{t} \mid \boldsymbol{f}_i \mid \boldsymbol{R}\boldsymbol{f}'_i])\right)^2 = e_i^2(\boldsymbol{R}, \boldsymbol{t}), \tag{98} $$

where the left determinant in (97) cancels because it has parallel columns.

These symmetries are illustrated in Fig. 1. In conclusion, the algebraic error is invariant to swaps of the translation direction, as well as reflections of the vector $\boldsymbol{R}\boldsymbol{f}'_i$ w.r.t. the axis of direction $\boldsymbol{t}$.

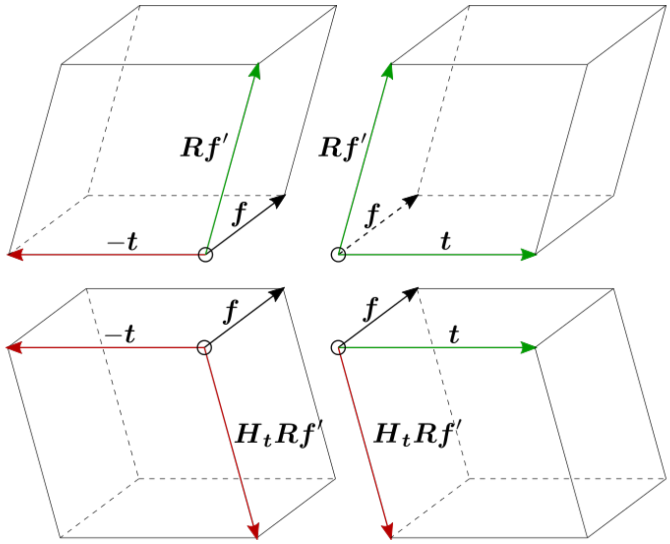

Figure 1. The *unsquared* algebraic error is the absolute value of the triple product of $\boldsymbol{t}$, $\boldsymbol{f}_i$ and $\boldsymbol{R}\boldsymbol{f}'_i$, or equivalently, the unsigned volume of a parallelepiped formed with the three vectors. The vectors $\boldsymbol{t}$ and $\boldsymbol{R}\boldsymbol{f}'$ in this figure lie in the paper plane. The symmetries are: (1) In the horizontal axis, swapping the translation sign. (2) In the vertical axis, reflecting $\boldsymbol{R}\boldsymbol{f}'_i$ w.r.t. $\boldsymbol{t}$. Both of these modifications keep the parallelepiped's volume the same.

6. Equivalence to algebraic objective of 8-point algorithm

The classical 8-point algorithm [2] models the epipolar constraint $\boldsymbol{f}_i^\top \boldsymbol{E}\boldsymbol{f}'_i = 0$, where $\boldsymbol{E} = [\boldsymbol{t}]_\times \boldsymbol{R}$ is the *essential matrix*, and proposes to minimize the sum of squared algebraic residues

$$f(\boldsymbol{E}) = \sum_{i=1}^{N} (\boldsymbol{f}_i^\top \boldsymbol{E}\boldsymbol{f}'_i)^2 = \sum_{i=1}^{N} e_i^2. \tag{99}$$

We may well rewrite each squared algebraic error above as

$$\begin{aligned} e_i^2 &= \left(\boldsymbol{f}_i^\top \boldsymbol{E}\boldsymbol{f}'_i\right)^2 && (100)\\ &= \left(\boldsymbol{f}_i^\top ([\boldsymbol{t}]_\times \boldsymbol{R})\boldsymbol{f}'_i)\right)^2 && (101)\\ &= \left(\boldsymbol{f}_i^\top [\boldsymbol{t}]_\times (\boldsymbol{R}\boldsymbol{f}'_i)\right)^2 && (102)\\ &= \left(\boldsymbol{f}_i^\top (\boldsymbol{t} \times (\boldsymbol{R}\boldsymbol{f}'_i))\right)^2 && (103)\\ &= \left(\det([\boldsymbol{f}_i \mid \boldsymbol{t} \mid \boldsymbol{R}\boldsymbol{f}'_i])\right)^2. && (104) \end{aligned}$$

Using the properties of the determinant, we may simply swap the columns to certify this expression is exactly the same squared algebraic error we optimize in the formulation presented in our main document.

7. Convex set of optimal SDP solutions

It is argued in the main document that in a well-defined instance of the relative pose problem there exist 4 possible solutions $\{\tilde{z}_k\}_{k=1}^4$, which in the tight case drive to 4 independent rank-1 solutions $\{\tilde{\boldsymbol{Z}}_k = \tilde{z}_k^\star(\tilde{z}_k^\star)^\top\}_{k=1}^4$ in the SDP relaxed problem.
Let us consider the convex hull of these rank-1 solutions:

$$\tilde{\boldsymbol{Z}}^\star = \sum_{k=1}^{4} a_k \tilde{\boldsymbol{Z}}_k^\star, \quad \text{s.t. } a_k \geq 0, \; \sum_{k=1}^{4} a_k = 1. \tag{105}$$

The following theorem will be necessary for the subsequent results.

Theorem 1. *The convex combination (105) is equivalent to an orthogonal eigenvalue decomposition*

$$\tilde{\boldsymbol{Z}}^\star = \sum_{k=1}^{4} a_k \tilde{\boldsymbol{z}}_k^\star (\tilde{\boldsymbol{z}}_k^\star)^\top = \sum_{k=1}^{4} \lambda_k \boldsymbol{u}_k \boldsymbol{u}_k^\top, \tag{106}$$

where $\lambda_k = 8a_k$ and $\boldsymbol{u}_k = \tilde{\boldsymbol{z}}_k^\star/\sqrt{8}$ stand for the eigenvalues and (orthogonal) eigenvectors of $\tilde{\boldsymbol{Z}}^\star$, respectively. Since $\tilde{\boldsymbol{Z}}^\star$ is at most rank-4, we consider only the 4 leading eigenvalues, as the rest will be zero by definition.

Proof. The convex combination already has the same structure as the eigendecomposition. Besides, the different solutions $\tilde{\boldsymbol{z}}_k^\star$ to from the relative pose problem turn out to be *orthogonal*, so that $\tilde{\boldsymbol{z}}_i^\star \cdot \tilde{\boldsymbol{z}}_j^\star = 0$ if $i \neq j$ (the proof can be found in Appendix C). So we identify each solution $\tilde{\boldsymbol{z}}_k^\star$ with an eigenvector $\boldsymbol{u}_k$, and it only remains to rescale the solutions and barycentric coefficients to fulfill the unitary constraint $\|\boldsymbol{u}_k\|_2 = 1$:

$$\sum_{k=1}^{4} a_k \tilde{\boldsymbol{z}}_k^\star (\tilde{\boldsymbol{z}}_k^\star)^\top = \sum_{k=1}^{4} (a_k \|\tilde{\boldsymbol{z}}_k^\star\|_2^2)(\frac{\tilde{\boldsymbol{z}}_k^\star}{\|\tilde{\boldsymbol{z}}_k^\star\|_2})(\frac{\tilde{\boldsymbol{z}}_k^\star}{\|\tilde{\boldsymbol{z}}_k^\star\|_2})^\top. \tag{107}$$

From here, identifying terms and using that $\|\tilde{\boldsymbol{z}}_k^\star\|_2^2 = 8$ holds as shown in (130) the results of the theorem follow. □

Now we are in a position to prove the core claim of this section.

Theorem 2 (Convex set of optimal SDP solutions)**.** *Given the finite set of SDP solutions $\{\tilde{\boldsymbol{Z}}_k^\star\}_{k=1}^r$ with $\operatorname{rank}(\tilde{\boldsymbol{Z}}_k^\star) = 1$, all points contained in the convex hull of this set, defined by*

$$\tilde{\boldsymbol{Z}}^\star = \sum_{k=1}^{r} a_k \tilde{\boldsymbol{Z}}_k^\star, \quad \text{s.t. } a_k \geq 0, \ \sum_{k=1}^{4} a_k = 1, \tag{108}$$

are also optimal SDP solutions.

Proof. By linearity of the SDP objective we have

$$\operatorname{tr}(\tilde{\boldsymbol{Q}}_0 \tilde{\boldsymbol{Z}}^\star) = \operatorname{tr}(\tilde{\boldsymbol{Q}}_0 \sum_{k=1}^{r} a_k \tilde{\boldsymbol{Z}}_k^\star) = \sum_{k=1}^{r} a_k \underbrace{\operatorname{tr}(\tilde{\boldsymbol{Q}}_0 \tilde{\boldsymbol{Z}}_k^\star)}_{f^\star} = f^\star (\sum_{k=1}^{r} a_k) = f^\star, \tag{109}$$

so the points in the convex set reach the same optimal objective. Given the close relationship between the barycentric coordinates a_k and the eigenvalues λ_k of a solution $\tilde{\boldsymbol{Z}}^\star$, shown in Theorem 1, it is straightforward that the non-negativity of a_k is required to fulfill the Positive Semidefinite (PSD) constraint on $\tilde{\boldsymbol{Z}}^\star$. □

8. Practical recovery of original solution from SDP solution

This section provides further insight onto the practical recovery procedure referred in the main document. The question that arises in practice is, given a particular optimal solution $\tilde{\boldsymbol{Z}}_0^\star$ to the SDP problem returned by a particular SDP solver, how do we recover the optimal solutions $(\boldsymbol{R}_k^\star, \boldsymbol{t}_k^\star)$ to the original problem?

An appealing approach, considering the result presented in Theorem 1, is to perform an eigenvalue decomposition of the IPM solution $\tilde{\boldsymbol{Z}}_0^\star = \sum_{k=1}^4 \lambda_k \boldsymbol{u}_k \boldsymbol{u}_k^\top$, and identify the sought solutions $\tilde{\boldsymbol{z}}_k^\star$ through proper scaling of the eigenvectors $\boldsymbol{u}_k$: $\tilde{\boldsymbol{z}}_k^\star \leftarrow 2\sqrt{2}\boldsymbol{u}_k$. As appealing as this might appear, we will see next this approach breaks when we encounter eigenvalue multiplicity though.

First, let us provide some insight on the behavior of the solution provided by off-the-shelf Primal-Dual Interior Point Method (IPM) solvers, such as SeDuMi [6] or SDPT3 [7]. By the inherent way IPM solvers work, leveraging log-barrier terms associated with the constraint cone, they return a solution which lies strictly inside the convex set of solutions $\boldsymbol{Z}^\star$. So, assuming the relaxation is tight and the problem is well-defined (no degeneracies or multiple physically feasible solutions), a chosen IPM is expected to return a rank-4 solution $\tilde{\boldsymbol{Z}}_0^\star$, never one of the rank-1 solutions $\tilde{\boldsymbol{Z}}_k^\star$ (vertices of the convex set) we are actually interested in. In fact, the iterative solution of the IPM tends to the barycenter of the convex set of solutions, leading to the solution $\tilde{\boldsymbol{Z}}_0^\star = \sum_{k=1}^4 \frac{1}{4} \tilde{\boldsymbol{Z}}_k^\star$ with $a_k = \frac{1}{4}$, which has the single eigenvalue $\lambda_1 = 2$ with multiplicity 4.

In the extreme case of eigenvalue multiplicity 4, we know the solutions $\{\tilde{\boldsymbol{z}}_k^\star\}_{k=1}^4$ form a basis $\left[\tilde{\boldsymbol{z}}_k^\star\right]$ for the non-null eigenspace of $\tilde{\boldsymbol{Z}}_0^\star$, but there are infinitely many possible bases $\boldsymbol{U}$ for this eigenspace that the eigenvalue decomposition might return, related to our basis of interest by an unknown orthogonal transformation $\boldsymbol{O} \in \mathrm{O}(4)$:

$$\tilde{\boldsymbol{Z}}_0^\star = \frac{1}{4}\left[\tilde{\boldsymbol{z}}^\star\right]\left[\tilde{\boldsymbol{z}}^\star\right]^\top = 2\boldsymbol{U}\boldsymbol{U}^\top = 2(\boldsymbol{U}\boldsymbol{O})(\boldsymbol{U}\boldsymbol{O})^\top \tag{110}$$

Thus, finding the desired solutions $\left[\tilde{\boldsymbol{z}}^\star\right]$ given any eigenspace basis $\boldsymbol{U}$ may not be straightforward.

In practice, though, the limit situation with multiplicity 4 would happen only after infinite iterations. Instead, we observed that the solution $\tilde{\boldsymbol{Z}}_0^\star$ returned by the IPM features two eigenvalues λ_1, λ_2 with multiplicity 2, which approach each other as further iterations proceed, but numerically their associated eigenvectors remain well-differentiated. Despite this observation being fully empirical so far, this property held for every evaluated case and we are confident this may be a consequence of the underlying problem structure and the inner workings of the IPM solver.

This means that the convex (or eigenvalue) decomposition of the returned solution $\tilde{\boldsymbol{Z}}_0^\star$ may be regarded as

$$\tilde{\boldsymbol{Z}}_0^\star = a_1\left[\tilde{\boldsymbol{z}}^\star\right]_1\left[\tilde{\boldsymbol{z}}^\star\right]_1^\top + a_2\left[\tilde{\boldsymbol{z}}^\star\right]_2\left[\tilde{\boldsymbol{z}}^\star\right]_2^\top \tag{111}$$

$$= \lambda_1\boldsymbol{U}_1\boldsymbol{U}_1^\top + \lambda_2\boldsymbol{U}_2\boldsymbol{U}_2^\top \tag{112}$$

$$= \lambda_1(\boldsymbol{U}_1\boldsymbol{O}_1)(\boldsymbol{U}_1\boldsymbol{O}_1)^\top + \lambda_2(\boldsymbol{U}_2\boldsymbol{O}_2)(\boldsymbol{U}_2\boldsymbol{O}_2)^\top, \tag{113}$$

where all matrices have width 2, in particular, $\boldsymbol{O}_1, \boldsymbol{O}_2 \in \mathrm{O}(2)$. Even more relevant to us, from empirical observation, the pairs of solutions grouped under a common eigenvalue λ_k correspond to those with a common rotation value and swapped translation signs.

Let us consider then, without loss of generality, that the pair of eigenvectors in $\boldsymbol{U}_1$ correspond to the pair of solutions $(\boldsymbol{R}^\star, +\boldsymbol{t}^\star) \to \tilde{\boldsymbol{z}}_1^\star$ and $(\boldsymbol{R}^\star, -\boldsymbol{t}^\star) \to \tilde{\boldsymbol{z}}_2^\star$. We know by the arguments above that there exists an orthogonal transformation $\boldsymbol{O}_1$ so that

$$\boldsymbol{U}_1 \propto \left[\tilde{\boldsymbol{z}}^\star\right]_1\boldsymbol{O}_1. \tag{114}$$

Now let us consider a column $\boldsymbol{u}$ in $\boldsymbol{U}_1$ (an eigenvector) and the corresponding column $\boldsymbol{o} = \left[o_1, o_2\right]^\top$ in the orthogonal transformation $\boldsymbol{O}_1$. In view of the expression (114), the blocks $\boldsymbol{r}$ and $\boldsymbol{t}$ in the eigenvector $\boldsymbol{u}$ fulfill the condition

$$\boldsymbol{u}_{\boldsymbol{r}} = o_1\boldsymbol{r}^\star + o_2\boldsymbol{r}^\star = (o_1 + o_2)\boldsymbol{r}^\star, \tag{115}$$

$$\boldsymbol{u}_{\boldsymbol{t}} = o_1(+\boldsymbol{t}^\star) + o_2(-\boldsymbol{t}^\star) = (o_1 - o_2)\boldsymbol{t}^\star. \tag{116}$$

Since $\|\boldsymbol{o}\|_2 = 1$, it is clear that the two blocks can never be simultaneously zero, that is, it cannot happen that $(o_1 + o_2) = 0$ and $(o_1 - o_2) = 0$ at the same time. As a result, we are always able to recover the optimal values $\boldsymbol{r}^\star$ and $\boldsymbol{t}^\star$ from one eigenvector (or both eigenvectors) in $\boldsymbol{U}_1$, by simply scaling properly the corresponding $\boldsymbol{r}$- or $\boldsymbol{t}$-block in any eigenvector, as long as this block is not zero.

The exact same procedure may be applied on the other pair of eigenvectors contained in $\boldsymbol{U}_2$ to obtain the remaining two solutions $(\boldsymbol{P}_{\boldsymbol{t}^\star}\boldsymbol{R}^\star, +\boldsymbol{t}^\star) \to \tilde{\boldsymbol{z}}_3^\star$ and $(\boldsymbol{P}_{\boldsymbol{t}^\star}\boldsymbol{R}^\star, -\boldsymbol{t}^\star) \to \tilde{\boldsymbol{z}}_4^\star$, or we might simply build these solutions from the already recovered solution by reflecting the rotation solution with $\boldsymbol{P}_{\boldsymbol{t}^\star}$.

9. Experimental results on real data

In this section we provide some additional results obtained from the evaluation of the proposed method (and reference ones) on relative pose problem instances that originate from real images in the TUM benchmark datasets [5]. Details on how we generated the instances are provided in the main document.

Note the datasets in [5] feature a varied set of camera pair configurations, varying between $o(100)$ and $o(2000)$ pairs depending on the dataset, and spanning from generic 3D hand-held trajectories (such as the `desk` and `room` sequences) to specific near-degenerate configurations with almost-zero translation or almost-zero rotation, recorded with the main purpose of debugging algorithms (that is the case of the `xyz` or `rpy` sequences). Despite this wide generality in the geometric configurations of the problem, as well as some challenging factors such as the presence of real noise stemming from the actual feature detection step, small FOV, or non-perfect calibration, the resulting optimality gaps depicted in Fig. 2 certify that the proposed approach through the SDP relaxation is able to retrieve (and certify) the optimal solution to the formulated problem in *all* cases. This is a remarkable result, that supports the promising claim that the proposed SDP relaxation remains tight in relative pose problems encountered in practice.

The results observed on the real data reinforces the already mentioned aspect that even though the optimality gap corresponding to other suboptimal methods, such as `8pt` or `8pt+eig`, may be deceptively small in many occasions (see Fig. 2), the corresponding error committed in the estimated orientation is much higher, as observed in Fig. 3. As we argued in the main document, this fact may be justified both by the nature of the optimization objective, since even erroneous residues tend not to have a very high value, and also by the existence of numerous suboptimal local minima whose objective value is deceptively close to the optimal one, yet the local solution may be far from the optimum.

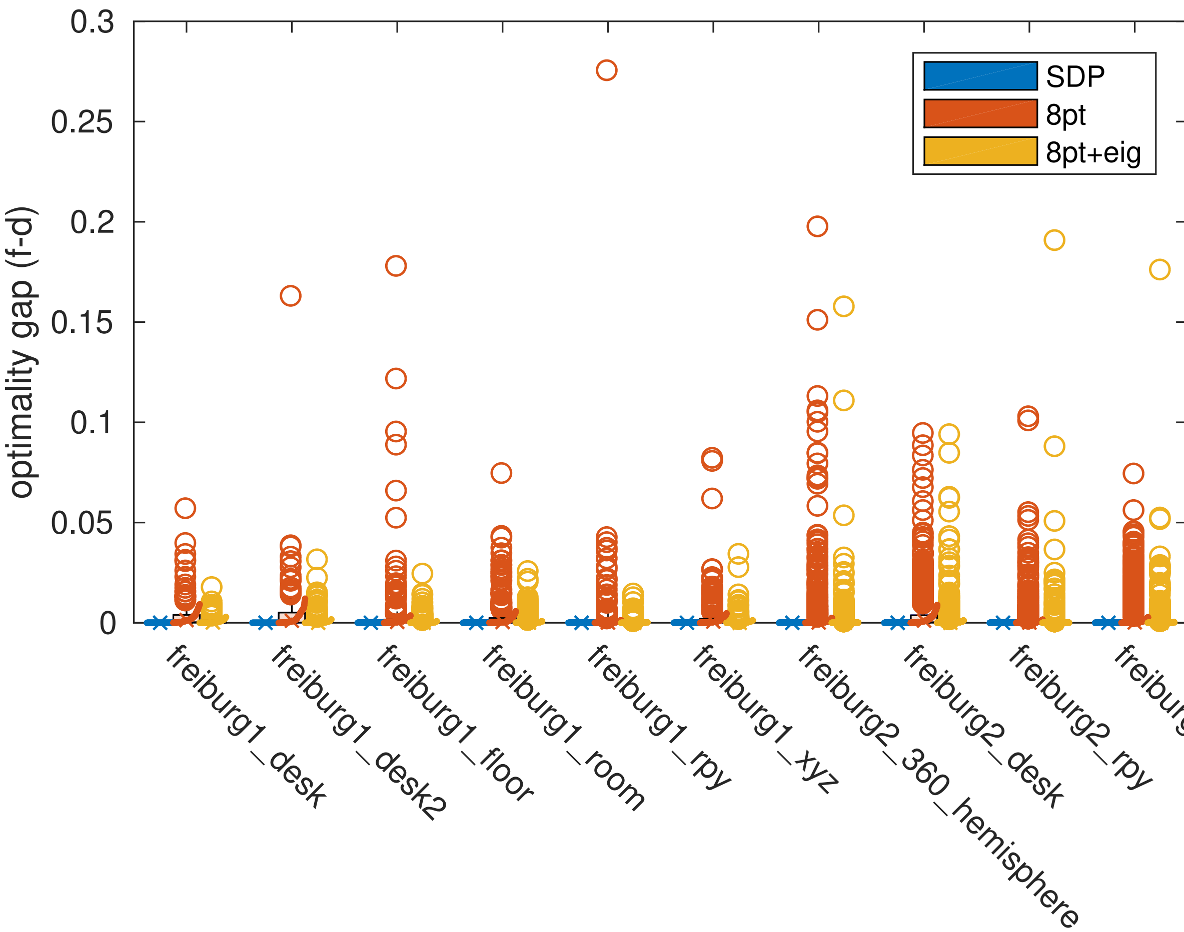

Figure 2. Optimality gap w.r.t. certified optimal objective $f^\star$ for all tested relative pose instances. The instances consist of overlapping image pairs extracted from each considered TUM dataset [5]. Our method recovered and certified the optimal solution in all cases featuring an optimality gap that is numerically zero ($o(10^{-10})$).

A. Vector spaces, canonical vectors and basis

Consider a usual vector space $\mathbb{R}^n$. The canonical vector ${}_n\boldsymbol{e}_i$ for this space is the $n \times 1$ vector that has a 1-element at position i and is zero everywhere else. The set of all canonical vectors $\{{}_n\boldsymbol{e}_i\}_{i=1}^n$ forms the canonical basis $\mathcal{B}(\mathbb{R}^n)$ of $\mathbb{R}^n$. We will often drop the dimension value if this can be inferred from the context, writing only $\boldsymbol{e}_i$.

Similarly, for the space $\mathbb{R}^{m\times n}$ of $m \times n$ matrices, the canonical matrix ${}_{mn}\boldsymbol{e}_{ij} = {}_m\boldsymbol{e}_i\, {}_n\boldsymbol{e}_j^\top$ stands for a matrix that has a 1-element at position (i,j) and is zero everywhere else. Once again, the set of all canonical matrices $\{{}_{mn}\boldsymbol{e}_i\}_{\substack{i=1,\dots,m\\ j=1,\dots,n}}$

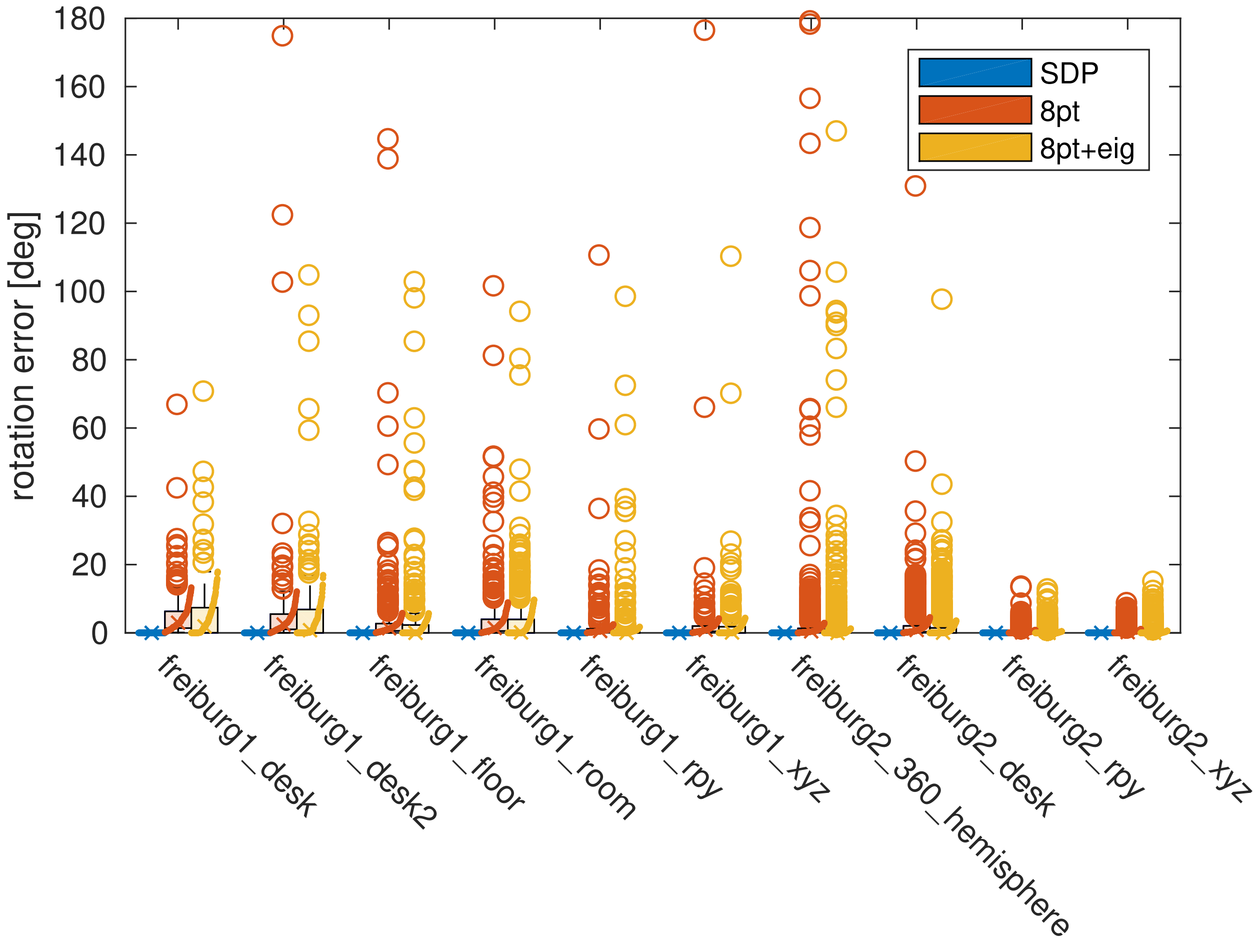

Figure 3. Rotation error w.r.t. the certified optimal rotation $\boldsymbol{R}^\star$ for all tested relative pose instances. The instances consist of overlapping image pairs extracted from each considered TUM dataset [5]. Whereas the alternative (non-optimal) methods `8pt` and `8pt+eig` may quite often return a good solution, it is also true that under the challenging conditions (noisy, small FOV, non-perfect calibration) of the image pairs considered in these datasets these methods may also often fail to return the optimal solution, or even return a completely wrong solution.

constitutes the canonical basis $\mathcal{B}(\mathbb{R}^{m\times n})$ for the whole matrix space $\mathbb{R}^{m\times n}$. We will often write just ${}_n\boldsymbol{e}_{ij}$ if the matrix is square ($m = n$), or again simply drop the dimension index, $\boldsymbol{e}_{ij}$.

Other convenient spaces that will appear in the course of this supplementary material are the sets of $n \times n$ symmetric and skew-symmetric matrices, denoted Sym_n and Skew_n, respectively. Akin to the previous cases, we may define the canonical elements of each space as

$$ {}^{+}_{n}\boldsymbol{e}_{ij} = \frac{1}{2}(\boldsymbol{e}_{ij} + \boldsymbol{e}_{ji}), \tag{117} $$

$$ {}^{-}_{n}\boldsymbol{e}_{ij} = \frac{1}{2}(\boldsymbol{e}_{ij} - \boldsymbol{e}_{ji}). \tag{118} $$

It is straightforward to observe that ${}^{+}_{n}\boldsymbol{e}_{ij} = {}^{+}_{n}\boldsymbol{e}_{ji}$ and ${}^{-}_{n}\boldsymbol{e}_{ij} = -{}^{-}_{n}\boldsymbol{e}_{ji}$. Thus, a canonical basis $\mathcal{B}(\text{Sym}_n)$ for Sym_n is given by the $\binom{n}{2}$ indexes $1 \le i \le n$, $i \le j \le n$, whereas the canonical basis $\mathcal{B}(\text{Skew}_n)$ for Skew_n conforms the $\binom{n-1}{2}$ indexes $1 \le i \le n$, $i < j \le n$. It is a well-known result that the vector space of square matrices is the direct sum of the corresponding symmetric and skew-symmetric matrices: $\mathbb{R}^{n\times n} = \text{Sym}_n \oplus \text{Skew}_n$. In particular, given any square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ we may decompose it as

$$\boldsymbol{A} = \underbrace{\frac{1}{2}(\boldsymbol{A} + \boldsymbol{A}^\top)}_{\text{sym}(\boldsymbol{A})} + \underbrace{\frac{1}{2}(\boldsymbol{A} - \boldsymbol{A}^\top)}_{\text{skew}(\boldsymbol{A})} \tag{119}$$

where $\text{sym}(\cdot)$ and $\text{skew}(\cdot)$ return the symmetric and skew-symmetric component of a matrix, respectively.

B. Vectorization

Along this work, we make heavy use of vectorization tricks that allows us to write the problem of interest in a more convenient form. Some important relations follow [4]:

$$\text{tr}(\boldsymbol{A}^\top \boldsymbol{B}) = \text{vec}(\boldsymbol{A})^\top \text{vec}(\boldsymbol{B}) \tag{120}$$

$$\text{vec}(\boldsymbol{A}\boldsymbol{X}\boldsymbol{B}) = (\boldsymbol{B}^\top \otimes \boldsymbol{A}) \text{vec}(\boldsymbol{X}) \tag{121}$$

$$\text{vec}(\boldsymbol{a}\boldsymbol{b}^\top) = \boldsymbol{b} \otimes \boldsymbol{a} \tag{122}$$

$$\text{tr}(\boldsymbol{A}^\top \boldsymbol{X}^\top \boldsymbol{B}\boldsymbol{Y}) = \text{vec}(\boldsymbol{X})^\top (\boldsymbol{A} \otimes \boldsymbol{B}) \text{vec}(\boldsymbol{Y}) \tag{123}$$

C. Optimal QCQP solutions in the relative pose problem are orthogonal

In order to prove this claim, we are going to choose the following ordering of the QCQP variables:

$$\tilde{\boldsymbol{z}} = \text{vec}(\tilde{\boldsymbol{r}}\tilde{\boldsymbol{t}}^\top) = \tilde{\boldsymbol{t}} \otimes \tilde{\boldsymbol{r}}, \tag{124}$$

where homogeneized variables are appended the unit element as usual:

$$\tilde{\boldsymbol{t}} = \begin{bmatrix} \boldsymbol{t} \\ 1 \end{bmatrix}, \quad \tilde{\boldsymbol{r}} = \begin{bmatrix} \text{vec}(\boldsymbol{R}) \\ 1 \end{bmatrix}. \tag{125}$$

With this ordering, it is almost straightforward to show that orthogonality holds between the 4 symmetric solutions of the problem:

$$\tilde{\boldsymbol{z}}_i^\star \cdot \tilde{\boldsymbol{z}}_j^\star = (\tilde{\boldsymbol{t}}_i^\star \otimes \tilde{\boldsymbol{r}}_i^\star) \cdot (\tilde{\boldsymbol{t}}_j^\star \otimes \tilde{\boldsymbol{r}}_j^\star) \tag{126}$$

$$= (\tilde{\boldsymbol{t}}_i^\star \cdot \tilde{\boldsymbol{t}}_j^\star)(\tilde{\boldsymbol{r}}_i^\star \cdot \tilde{\boldsymbol{r}}_j^\star) \tag{127}$$

$$= (1 + \boldsymbol{t}_i^\star \cdot \boldsymbol{t}_j^\star)(1 + \boldsymbol{r}_i^\star \cdot \boldsymbol{r}_j^\star) \tag{128}$$

$$= (1 + \boldsymbol{t}_i^\star \cdot \boldsymbol{t}_j^\star)(1 + \text{tr}((\boldsymbol{R}_i^\star)^\top \boldsymbol{R}_j^\star) \tag{129}$$

In view of the result above, let us consider the potential scenarios:

- The solutions i and j are symmetric w.r.t. translation. In this case $\boldsymbol{t}_i^\star \cdot \boldsymbol{t}_j^\star = -1$, so the first factor cancels.
- The solutions i and j are symmetric w.r.t. rotation. In this case $(\boldsymbol{R}_i^\star)^\top \boldsymbol{R}_j^\star = \boldsymbol{P}_{\boldsymbol{t}}$, and $\text{tr}(\boldsymbol{P}_{\boldsymbol{t}}) = \sum \lambda(\boldsymbol{P}_{\boldsymbol{t}}) = -1$, so the second factor cancels.

So, in conclusion, as long as $i \neq j$ there exists at least one of these two symmetries and the corresponding dot product between $\tilde{\boldsymbol{z}}_i^\star$ and $\tilde{\boldsymbol{z}}_j^\star$ cancels.

As a corollary of the relation above, each solution has a fixed squared norm

$$\|\tilde{\boldsymbol{z}}_k^\star\|_2^2 = \tilde{\boldsymbol{z}}_k^\star \cdot \tilde{\boldsymbol{z}}_k^\star = (1 + \boldsymbol{t}_k^\star \cdot \boldsymbol{t}_k^\star)(1 + \text{tr}((\boldsymbol{R}_k^\star)^\top \boldsymbol{R}_k^\star) = 2 \cdot 4 = 8. \tag{130}$$

References

[1] J. Briales and J. González-Jiménez. Convex Global 3D Registration with Lagrangian Duality. In *International Conference on Computer Vision and Pattern Recognition*, jul 2017. 3, 5, 6, 7

[2] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003. 10

[3] D. Henrion, J.-B. Lasserre, and J. Löfberg. GloptiPoly 3: moments, optimization and semidefinite programming. *Optimization Methods & Software*, 24(4-5):761–779, 2009. 3

[4] K. Schäcke. On the kronecker product, 2013. 15

[5] J. Sturm and N. Engelhard. A benchmark for the evaluation of RGB-D SLAM systems. *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2012. 12, 13, 14

[6] J. F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization methods and software*, 11(1-4):625–653, 1999. 11

[7] K.-C. Toh, M. J. Todd, and R. H. Tütüncü. SDPT3a MATLAB software package for semidefinite programming, version 1.3. *Optim. methods Softw.*, 11(1-4):545–581, 1999. 11