Essential Matrix Estimation using Convex Relaxations in Orthogonal Space

Arman Karimian and Roberto Tron
Boston University
{armandok,tron}@bu.edu

Abstract

We introduce a novel method to estimate the essential matrix for two-view Structure from Motion (SfM). We show that every 3×3 essential matrix can be embedded in a 4×4 rotation having its bottom right entry fixed to zero; we call the latter the quintessential matrix. *This embedding leads to rich relations with the space of 4-D rotations, quaternions, and the classical twisted-pair ambiguity in two-view SfM. We use this structure to derive a succession of semidefinite relaxations that require fewer parameters than the existing non-minimal solvers and yield faster convergence with certifiable optimality. We then exploit the low-rank geometry of these relaxations to reduce them to an equivalent optimization on a Riemannian manifold and solve them via the Riemannian Staircase method. The experimental evaluation confirms that our algorithm always finds the globally optimal solution and outperforms the existing non-minimal methods. We make our implementations open source.*[1].

1. Introduction

A cornerstone of geometric perception is finding the relative pose between two images using 2D-2D point correspondences. It constitutes the basic building block in many structure-from-motion (SfM), visual odometry, and simultaneous localization and mapping (SLAM) systems. Relative pose estimation is a complex problem since the relation between the coordinates of the correspondences is intrinsically nonlinear and has ambiguous solutions. Also, the most natural cost function, the reprojection error, is known to have many local minima. The traditional approach to estimating the relative pose between calibrated cameras passes through the essential matrix, a rank-two matrix with two identical singular values that encode the *epipolar* constraint between corresponding images in the two views [32].

Without information about the translation scale, the relative pose problem has five degrees of freedom, and finding a solution requires at least five correspondences. Most existing solvers can only handle a fixed number of correspondences as input. These so-called *minimal solvers* are usually paired with a hypothesize-and-test framework such as RANSAC to find the maximum consensus set of inlier matches. When inliers are found, a *non-minimal* solver is needed to fine-tune the estimated solution and reduce the influence of noise. Therefore, solvers' real-time performance and low processing and memory requirements are essential.

This paper introduces an intuitive algebraic characterization of the space of essential matrices. We show that the space of 4-D rotations with the bottom right entry set to zero double covers the space of essential matrices and uniquely encode epipolar configurations up to scale. We call such rotations *quintessential* matrices. We show that they share an intuitive connection with the well-known twisted-pair ambiguity, i.e., four different epipolar configurations exist for the same (unsigned) essential matrix. We also explore the relationship between quintessential matrices and unit quaternions and show that the space of two orthonormal quaternions double covers the space of quintessential matrices. We use these findings to present two semidefinite relaxations of the essential matrix estimation problem and show that a Burer-Monteiro factorization of these spaces lives on Riemannian manifolds. We then design a Riemannian truncated-Newton trust-region method to solve this reduction efficiently.

2. Related Work

Two-view Structure from Motion (SfM), epipolar geometry, and the essential matrix are a fundamental textbook topic in computer vision geometry [11, 19, 22]. The typical formulation aims to find the essential matrix given pairs of image points that are matched across the two images. Initial algorithms were based on linear relaxations (the Direct Linear Transform, [19]) or minimal algorithms (which use a minimum of point correspondences [16, 25]). These solution have typically been paired with RANSAC [12] to improve the robustness to outliers. Successive solutions followed different instances of the problem, e.g., using optimization on manifolds (to minimize more meaningful objective functions, which, however, lead to many local minima, [20, 23]),

[1] https://github.com/armandok/QME

or focusing on iteratively identifying and removing outlier correspondences [18, 26]. Despite the many proposed solutions, to this day there is no algorithm that has clearly emerged as the most robust and versatile.

Some of the optimization techniques used in this paper for two-view SfM have found success in closely related areas. A first problem is point registration, where relaxation to convex sets has been applied (by relaxing the geometry of the space of rotations to their convex hulls, [21, 30]). Another related problem is the one of *pose averaging* or *pose synchronization*, where the goal is to combine multiple two-view pose estimations into a single localization for all the camera frames. The rich structure of this problem has led to many different approaches, such as averaging in the space of rotations [8,15,17,35], or the use of linear [24] or spectral [3] relaxations. The latter has lead to techniques [9, 29] that provide guarantees of *global* optimality despite the non-convexity of the problem. These guarantees are mainly given by the use of dual certificates [10,13,29] and the *Riemannian staircase* algorithm [4].

3. Preliminaries

General Notation. We use lowercase characters to denote scalars (e.g., s), bold lowercase characters to denote real vectors (e.g., $\mathbf{v}$), and bold uppercase characters for real matrices (e.g., $\mathbf{M}$). We denote the i-th row and j-th column entry of $\mathbf{M}$ by $\mathbf{M}_{ij}$, and the identity matrix as $\mathbf{I}_d \in \mathbb{R}^{d\times d}$. The trace, determinant, and vectorization of $\mathbf{M}$ are given by $\mathrm{tr}(\mathbf{M})$, $\det(\mathbf{M})$, and $\mathrm{vec}(\mathbf{M})$, and $\otimes$ denotes the Kronecker product. The inner product between two matrices is given by

$$\langle \mathbf{Z}_1, \mathbf{Z}_2 \rangle \doteq \mathrm{tr}(\mathbf{Z}_1^\mathsf{T}\mathbf{Z}_2) = \mathrm{vec}(\mathbf{Z}_1)^\mathsf{T}\mathrm{vec}(\mathbf{Z}_2). \tag{1}$$

The norm induced by this inner product is the *Frobenius norm* given by $\|\mathbf{Z}\|_\mathrm{F} \doteq \sqrt{\mathrm{tr}(\mathbf{Z}^\mathsf{T}\mathbf{Z})}$. The cross-product between two vectors $\mathbf{t}, \mathbf{v} \in \mathbb{R}^3$ is given by $\mathbf{t}\times\mathbf{v} = [\mathbf{t}]_\times \mathbf{v}$, where for $\mathbf{t} = [t_x, t_y, t_z]^\mathsf{T}$ we have $[\mathbf{t}]_\times$ as

$$[\mathbf{t}]_\times = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}. \tag{2}$$

The *Riemannian gradient* and *Hessian* of a function $f : \mathcal{M} \mapsto \mathbb{R}$ acting on a Riemannian manifold $\mathcal{M}$ are denoted as $\mathrm{grad}\, f(\mathbf{X})$ and $\mathrm{Hess}\, f(\mathbf{X})[\dot{\mathbf{X}}]$. Projection on the tangent space of $\mathcal{M}$ at $\mathbf{X}$ is given by $\mathrm{Proj}_\mathbf{X}(\cdot)$. A *double cover* is a two-to-one mapping from one topological space to another.

Sets. The d-dimensional unit sphere is denoted as $\mathbb{S}^{d-1} \doteq \{\mathbf{u} \in \mathbb{R}^d : \|\mathbf{u}\| = 1\}$. The sets of $d \times d$ symmetric and symmetric *positive semidefinite* matrices are denoted as $\mathcal{S}^d$ and $\mathcal{S}^d_+ = \{\mathbf{M} \in \mathcal{S}^d : \mathbf{M} \succeq \mathbf{0}\}$. The set of orthonormal k-frames in $\mathbb{R}^d$ is shown by $\mathrm{St}(k,d) \doteq \{\mathbf{Y} \in \mathbb{R}^{d\times k} \mid \mathbf{Y}^\mathsf{T}\mathbf{Y} = \mathbf{I}_k\}$ for $k \leq d$. This set represents a smooth and compact matrix manifold known as the *Stiefel manifold* [1]. We consider the Riemannian metric induced by its embedding in $\mathbb{R}^{d\times k}$ given in (1). The orthogonal and special orthogonal groups are denoted as $\mathrm{O}(d) = \{\mathbf{O} \in \mathbb{R}^{d\times d} : \mathbf{O}^\mathsf{T}\mathbf{O} = \mathbf{I}_d\}$ and $\mathrm{SO}(d) = \{\mathbf{R} \in \mathrm{O}(d) : \det(\mathbf{R}) = +1\}$. Every non-trivial rotation in $\mathrm{SO}(3)$ is about an axis, or equivalently about an axis-plane normal to the rotation axis. Non-trivial rotations in $\mathrm{SO}(4)$ are about two axis-planes that are completely orthogonal to each other. If the rotation angle about one of the axis-planes is zero, it is called a *simple rotation*, otherwise called a *double rotation*. If the angle of rotation in both planes are equal, then the double rotation is called an *isoclinic rotation*. If the two angles have the same sign, it is called a *left* isoclinic rotation, otherwise it is a *right* isoclinic rotation.

Quaternions. We denote a unit quaternion as a unit-norm vector $\mathbf{q} = [\mathbf{v}^\mathsf{T}\ s]^\mathsf{T} \in \mathbb{S}^3$, where s is the scalar part and $\mathbf{v} = [v_x\ v_y\ v_z]^\mathsf{T}$ is the vector part. The inverse of a quaternion is given by negating its vector part, i.e., $\mathbf{q}^{-1} = [-\mathbf{v}^\mathsf{T}\ s]^\mathsf{T}$. The left and right product by a quaternion $\mathbf{q}$ is given by matrices

$$\mathbf{Q}_\mathrm{L}(\mathbf{q}) \doteq \begin{bmatrix} s & -v_z & v_y & v_x \\ v_z & s & -v_x & v_y \\ -v_y & v_x & s & v_z \\ -v_x & -v_y & -v_z & s \end{bmatrix} = \begin{bmatrix} +[\mathbf{v}]_\times & \mathbf{v} \\ -\mathbf{v}^\mathsf{T} & 0 \end{bmatrix} + s\mathbf{I}_4, \tag{3}$$

$$\mathbf{Q}_\mathrm{R}(\mathbf{q}) \doteq \begin{bmatrix} s & v_z & -v_y & v_x \\ -v_z & s & v_x & v_y \\ v_y & -v_x & s & v_z \\ -v_x & -v_y & -v_z & s \end{bmatrix} = \begin{bmatrix} -[\mathbf{v}]_\times & \mathbf{v} \\ -\mathbf{v}^\mathsf{T} & 0 \end{bmatrix} + s\mathbf{I}_4, \tag{4}$$

such that $\mathbf{q} \circ \mathbf{p} = \mathbf{Q}_\mathrm{L}(\mathbf{q})\,\mathbf{p} = \mathbf{Q}_\mathrm{R}(\mathbf{p})\,\mathbf{q}$, where $\circ$ denotes the quaternion product operation. The two matrix operators $\mathbf{Q}_\mathrm{L}(\cdot)$, $\mathbf{Q}_\mathrm{R}(\cdot)$ represent left and right isoclinic rotations.
Every rotation matrix $\mathbf{R}$ in $\mathrm{SO}(4)$ can be written as the product of a pair of left and right isoclinic rotations, e.g., $\mathbf{R} = \mathbf{Q}_\mathrm{L}(\mathbf{p})\mathbf{Q}_\mathrm{R}(\mathbf{q})$ for $\mathbf{p}, \mathbf{q} \in \mathbb{S}^3$. The quaternion pair $(\mathbf{p}, \mathbf{q})$ is unique up to a change of sign, i.e., $(-\mathbf{p}, -\mathbf{q})$ produces the same matrix $\mathbf{R}$. In other words, $\mathbb{S}^3 \times \mathbb{S}^3$ is the (unique) double cover of $\mathrm{SO}(4)$. For any quaternion $\mathbf{q}$, the pair $(\mathbf{q}, \mathbf{q}^{-1})$ yields a matrix of the form

$$\mathbf{Q}_\mathrm{L}(\mathbf{q})\mathbf{Q}_\mathrm{R}(\mathbf{q}^{-1}) = \begin{bmatrix} \mathbf{R} & \mathbf{0}_3 \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix},\ \mathbf{R} \in \mathrm{SO}(3), \tag{5}$$

meaning that $\mathbb{S}^3$ is a double cover of $\mathrm{SO}(3)$.

4. Essential Matrix as an Orthogonal Matrix

A *normalized essential matrix* is typically defined as the product of a rotation matrix $\mathbf{R}$ and a cross product matrix $[\mathbf{t}]_\times$ of a unit vector $\mathbf{t}$ such that

$$\mathbf{E} \doteq [\mathbf{t}]_\times \mathbf{R}. \tag{6}$$

This matrix encodes the relative pose between two cameras, and the objective of the *non-minimal* essential matrix estimation problem is to estimate $\mathbf{E}$ by minimizing the squared *algebraic error*

$$\mathbf{E}^\star = \underset{\mathbf{E}\in\mathbb{E}}{\operatorname{argmin}} \sum_{k=1}^{N} (\mathbf{f}_{i,k}^\top \mathbf{E} \mathbf{f}_{j,k})^2, \tag{7}$$

given $N \geq 5$ correspondences of bearing vectors $(\mathbf{f}_{i,k}, \mathbf{f}_{j,k})$ observed by cameras i, j. The optimal solution $\mathbf{E}^\star$ is then decomposed into matrix $\mathbf{R}^\star$ and the vector $\mathbf{t}^\star$, which are the orientation and normalized position of camera j represented in the coordinate frame of camera i. Finally, $\mathbb{E}$ is the set of all normalized essential matrices.

In the literature, several endeavors have been undertaken to characterize $\mathbb{E}$ as a Riemannian product manifold. The most direct definition arises from (6), where $\mathbb{E}$ is defined as

$$\mathbb{E} = \{\mathbf{E} = [\mathbf{t}]_\times \mathbf{R} \ : \ \mathbf{R} \in \mathrm{SO}(3), \mathbf{t} \in \mathbb{S}^2\}, \tag{8}$$

expressed in terms of the $\mathrm{SO}(3) \times \mathbb{S}^2$ manifold. An alternative characterization based on the Singular Value Decomposition (SVD) is provided by

$$\mathbb{E} = \{\mathbf{E} \in \mathbb{R}^{3\times 3} \ : \ \sigma(\mathbf{E}) = \{1, 1, 0\}\}, \tag{9}$$

which is equivalent to a quotient of the $\mathrm{O}(3) \times \mathrm{O}(3)$ manifold as demonstrated in [14]. More recently, a different characterization involving a quotient of $\mathrm{SO}(3) \times \mathrm{SO}(3)$ has been presented in [32]. These approaches depend on the parametrization of $\mathbb{E}$, primarily due to the challenges in enforcing the constraints on the singular values as given in (9). The derivatives of the singular values depend on the singular vectors, which are not explicitly expressed in terms of the entries of $\mathbf{E}$.

In this section, we provide a novel characterization of $\mathbb{E}$ as a Riemannian submanifold defined by differentiable algebraic constraints. This achievement is realized by embedding essential matrices into 4×4 rotation matrices within $\mathrm{SO}(4)$. Furthermore, we demonstrate that this embedding successfully resolves the twisted-pair ambiguity, thereby disambiguating the two possible rotation and unit translation decompositions for each normalized essential matrix.

4.1. Orthogonal Embedding

To accomplish this embedding, we start by replacing the definition given in (8) with a canonical representation proposed in [3,32]. This alternative representation preserves the inherent symmetry of the epipolar constraint. Leveraging the explicit poses of the two cameras and their relative bearing, denoted as $(\mathbf{R}_i, \mathbf{t}_i)$, $(\mathbf{R}_j, \mathbf{t}_j)$, and $\mathbf{t}_{ij} \doteq (\mathbf{t}_j - \mathbf{t}_i)\|\mathbf{t}_j - \mathbf{t}_i\|^{-1}$, the expression in (6) takes the form $[\mathbf{R}_i^\top \mathbf{t}_{ij}]_\times \mathbf{R}_i^\top \mathbf{R}_j$. Utilizing the identity $\mathbf{R}[\mathbf{t}]_\times \mathbf{R}^\top = [\mathbf{R}\mathbf{t}]_\times$, we rewrite (6) as

$$\mathbf{E}_{ij} = \mathbf{R}_i^\top [\mathbf{t}_{ij}]_\times \mathbf{R}_j. \tag{10}$$

In the next step, we proceed to embed $\mathbf{E}_{ij}$ in $\mathbb{R}^{4\times 4}$, ensuring that the resulting matrix achieves full rank. The key idea here is that $\mathbf{E}_{ij}$ has a rank of two, and concatenating it with its left null vector horizontally yields a matrix with full column rank. Similarly, vertical concatenation of $\mathbf{E}_{ij}$ with its right null vector gives a matrix with full row rank. These left and right null vectors are the epipoles $\mathbf{R}_i^\top \mathbf{t}_{ij}$ and $\mathbf{R}_j^\top \mathbf{t}_{ji}$. Importantly, negating these vectors does not alter this property.

Definition 1 (Signed Quintessential Matrix). *Given an essential matrix $\mathbf{E}_{ij} \in \mathbb{E}$, the* signed quintessential matrix *$\mathbf{Q}_{ij} \in \mathbb{R}^{4\times 4}$ is defined as*

$$\mathbf{Q}_{ij} = \begin{bmatrix} \mathbf{E}_{ij} & \delta_i \mathbf{R}_i^\top \mathbf{t}_{ij} \\ \delta_j (\mathbf{R}_j^\top \mathbf{t}_{ji})^\top & 0 \end{bmatrix}, \tag{11}$$

for $\delta_i, \delta_j \in \{-1, +1\}$.

This definition entails four matrices per essential matrix. Interestingly, signed quintessential matrices are orthogonal and the space of all signed quintessential matrices is given by the set $\mathbb{Q}$ defined as

$$\mathbb{Q} = \{\mathbf{O} \in \mathrm{O}(4) : \mathbf{O}_{44} = 0\}. \tag{12}$$

Lemma 1. *Given a unit vector $\mathbf{t} \in \mathbb{S}^2$, we have $[\mathbf{t}]_\times^2 = \mathbf{t}\mathbf{t}^\top - \mathbf{I}_3$.*

Proof. For $\mathbf{U} = [\mathbf{t}]_\times^2$, since $\|\mathbf{t}\| = 1$ we have $u_{ii} = -t_j^2 + -t_k^2 = t_i^2 - 1$ and $u_{ij} = t_i t_j$ for $\{i, j, k\} = \{1, 2, 3\}$. □

Theorem 1. *$\mathbb{Q}$ is the set of all signed quintessential matrices.*

Proof. It suffices to show that $\mathbf{Q}_{ij}^\top \mathbf{Q}_{ij} = \mathbf{I}_4$. We have

$$\begin{aligned} \mathbf{Q}_{ij}^\top \mathbf{Q}_{ij} &= \begin{bmatrix} \mathbf{E}_{ij}^\top \mathbf{E}_{ij} + \delta_j^2 \mathbf{R}_j^\top \mathbf{t}_{ji} \mathbf{t}_{ji}^\top \mathbf{R}_j & \delta_i \mathbf{E}_{ij}^\top \mathbf{R}_i^\top \mathbf{t}_{ij} \\ \delta_i \mathbf{t}_{ij}^\top \mathbf{R}_i \mathbf{E}_{ij} & \delta_i^2 \mathbf{t}_{ij}^\top \mathbf{R}_i \mathbf{R}_i^\top \mathbf{t}_{ij} \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{R}_j^\top (-[\mathbf{t}_{ij}]_\times^2 + \mathbf{t}_{ij} \mathbf{t}_{ij}^\top) \boldsymbol{R}_j & \delta_i \mathbf{R}_j^\top [\mathbf{t}_{ij}]_\times^\top \mathbf{t}_{ij} \\ \delta_i \mathbf{t}_{ij}^\top [\mathbf{t}_{ij}]_\times \mathbf{R}_j & \mathbf{t}_{ij}^\top \mathbf{t}_{ij} \end{bmatrix} \end{aligned}$$

Using Lemma 1, $\mathbf{t}_{ji} = -\mathbf{t}_{ij}$, and $[\mathbf{t}_{ij}]_\times^\top = -[\mathbf{t}_{ij}]_\times$, the top left block simplifies to $\mathbf{I}_3$. Since $\delta_i^2 = \delta_j^2 = 1$ and $\mathbf{t}_{ij}$ has unit norm, the bottom right element equals to one. The top right and bottom left 3-dimensional vectors are zero, since $\mathbf{t}_{ij}^\top [\mathbf{t}_{ij}]_\times = \mathbf{0}^\top$,$[\mathbf{t}_{ij}]_\times \mathbf{t}_{ij} = \mathbf{0}$. Conversely, for a matrix $\mathbf{Q} = \left[\begin{smallmatrix} \mathbf{E} & \mathbf{t}_l \\ \mathbf{t}_r^\top & 0 \end{smallmatrix}\right] \in \mathbb{Q}$, we have $\|\mathbf{E}\|_\mathrm{F}^2 = \|\mathbf{Q}\|_\mathrm{F}^2 - \|\mathbf{t}_r\|_\mathrm{F}^2 - \|\mathbf{t}_r\|_\mathrm{F}^2 = 4 - 2 = 2$ and also $\mathbf{E}^\top \mathbf{t}_l = \mathbf{E}\mathbf{t}_r = \mathbf{0}$ due to orthonormality of the rows and columns. This means that $\mathbf{E}$ must have at least one zero singular value ($\sigma_3 = 0$) which yields $\|\mathbf{E}\|_\mathrm{F}^2 = \sigma_1^2 + \sigma_2^2 = 2$. Since the singular values of $\mathbf{Q}$ are all equal to one, the interlacing property dictates that $\sigma_1, \sigma_2 \leq 1$. Thus $\sigma(\mathbf{E}) = \{1, 1, 0\}$, hence $\mathbf{E} \in \mathbb{E}$. □

We can determine the condition on δ_i and δ_j under which $\mathbf{Q}_{ij}$ is a rotation matrix.

Theorem 2. *A signed quintessential matrix is a rotation if $\delta_i \delta_j = +1$.*

Proof. The matrix $\mathbf{Q}_{ij}$ can be decomposed as

$$\mathbf{Q}_{ij} = \begin{bmatrix} \mathbf{R}_i^\top & \mathbf{0}_3 \\ \mathbf{0}_3^\top & 1 \end{bmatrix} \begin{bmatrix} [\mathbf{t}_{ij}]_\times & \delta_i \mathbf{t}_{ij} \\ \delta_j \mathbf{t}_{ji}^\top & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R}_j & \mathbf{0}_3 \\ \mathbf{0}_3^\top & 1 \end{bmatrix}.$$

The left and right matrices are in $\mathrm{SO}(4)$. Take the unit quaternion $\mathbf{q}_{ij} = [\mathbf{t}_{ij}^\top \ 0]$. The middle matrix is equal to matrices $\mathbf{Q}_{\mathtt{L}}(\mathbf{q}_{ij}), \mathbf{Q}_{\mathtt{R}}(\mathbf{q}_{ij})^\top$ for the values $(+1,+1)$ and $(-1,-1)$ of (δ_i, δ_j), both of which are in $\mathrm{SO}(4)$. □

Following the results of Theorem 2, we denote the set of (rotational) quintessential matrices $\mathbb{Q}_+ \subset \mathbb{Q}$ as

$$\mathbb{Q}_+ = \{\mathbf{R} \in \mathrm{SO}(4) : \mathbf{R}_{44} = 0\}. \tag{13}$$

For a given essential matrix $\mathbf{E}_{ij}$, $\mathbb{Q}_+$ contains two matrices per the two configurations $(\delta_i, \delta_j) \in \{(+1,+1),(-1,-1)\}$. As we will show in the next section, these two matrices correspond to two different decompositions of $\mathbf{E}_{ij}$ into a rotation matrix and a baseline vector. This means that this redundancy from embedding essential matrices in $\mathrm{SO}(4)$ serves to disambiguate the mapping from an essential matrix to its two relative pose decompositions. In simpler terms, quintessential matrices uniquely encode epipolar configurations up to scale.

4.2. Twisted-pair Ambiguity

Since the objective of problem (7) is quadratic, an estimated solution is recovered up to an arbitrary choice of sign. In addition, there are two relative pose decompositions per an essential matrix. These two facts indicate four different epipolar configurations corresponding to an estimated essential matrix, known as the twisted-pair ambiguity.

As discussed, $\mathbb{Q}_+$ contains two members per essential matrix. Considering both $\mathbf{E}_{ij}$ and $-\mathbf{E}_{ij}$, there are four members of interest in $\mathbb{Q}_+$. We prefer the configuration $\delta_i = \delta_j = +1$ for $\mathbf{Q}_{ij}$ representing the actual epipolar configuration, leading to

$$\mathbf{Q}_{ij} = \begin{bmatrix} \mathbf{E}_{ij} & \mathbf{R}_i^\top \mathbf{t}_{ij} \\ \mathbf{t}_{ji}^\top \mathbf{R}_j & 0 \end{bmatrix}, \ \rotatebox{180}{$\mathbf{Q}$}_{ij} = \begin{bmatrix} \mathbf{E}_{ij} & -\mathbf{R}_i^\top \mathbf{t}_{ij} \\ -\mathbf{t}_{ji}^\top \mathbf{R}_j & 0 \end{bmatrix}. \tag{14}$$

While these two correspond to $\mathbf{E}_{ij}$, we have $-\mathbf{Q}_{ij}, -\rotatebox{180}{$\mathbf{Q}$}_{ij}$ for $-\mathbf{E}_{ij}$, thanks to the central inversion $-\mathbf{I}_4 \in \mathrm{SO}(4)$. These configurations are depicted in Figure 1. The relation between $\mathbf{Q}_{ij}, \rotatebox{180}{$\mathbf{Q}$}_{ij}$ is given by $\mathbf{H}_{ij} \in \mathrm{SO}(4)$ such that

$$\mathbf{Q}_{ij} = \mathbf{H}_{ij} \rotatebox{180}{$\mathbf{Q}$}_{ij}, \ \mathbf{H}_{ij} = \begin{bmatrix} \mathbf{R}_i^\top (\mathbf{I}_3 - 2\mathbf{t}_{ij}\mathbf{t}_{ij}^\top)\mathbf{R}_i & \mathbf{0}_{3\times 1} \\ \mathbf{0}_{1\times 3} & -1 \end{bmatrix},$$

where the top left 3×3 block of $\mathbf{H}_{ij}$ contains a Householder reflection.

4.3. Relations to Quaternions

Another useful insight is the connection between essential matrices and quaternions. Here we show that the set of all orthonormal quaternion pairs covers $\mathbb{Q}_+$ twice.

Corollary 1. *The set* $\mathrm{St}(2,4)$ *is a double cover of* $\mathbb{Q}_+$.

Proof. Since a matrix $\mathbf{R} \in \mathbb{Q}_+$ lies in $\mathrm{SO}(4)$, it can be decomposed uniquely (up to a sign change) by a unit quaternion pair $(\mathbf{p}, \mathbf{q}) = ([\mathbf{v}_p^\top \ s_p], [\mathbf{v}_q^\top \ s_q])$ such that $\mathbf{R} = \mathbf{Q}_{\mathtt{L}}(\mathbf{p})\mathbf{Q}_{\mathtt{R}}(\mathbf{q})$. We also have $\mathbf{R}_{44} = -\mathbf{v}_p^\top \mathbf{v}_q + s_p s_q = 0$, which is equivalent to $\mathbf{p}^\top \mathbf{q}^{-1} = 0$. The matrix $[\mathbf{p} \ \mathbf{q}^{-1}]$ hence lives in $\mathrm{St}(2,4)$, same as $[-\mathbf{p} \ \ -\mathbf{q}^{-1}]$. Alternatively, for two perpendicular unit quaternions $\mathbf{p}, \mathbf{q}$ we have $\mathbf{R} = \mathbf{Q}_{\mathtt{L}}(\mathbf{p})\mathbf{Q}_{\mathtt{R}}(\mathbf{q}^{-1}) \in \mathbb{Q}_+$. □

Based on Corollary 1, the two pairs $(\mathbf{q}_1, \mathbf{q}_2), (-\mathbf{q}_1, -\mathbf{q}_2)$ both yield a quintessential matrix $\mathbf{Q}_{ij} = \mathbf{Q}_{\mathtt{L}}(\mathbf{q}_1)\mathbf{Q}_{\mathtt{R}}(\mathbf{q}_2^{-1})$ if they are orthonormal. We saw four possible quintessential matrix solutions for the relative pose estimation problem in the previous section. This indicates that there are eight possible orthogonal quaternion pair solutions. If we denote these pairs as $\{\mathbf{q}_k = [\mathbf{v}_k^\top \ s_k]^\top \in \mathbb{S}^3\}_{k=1}^2$, then we have

$$\mathbf{Q}_{ij} = \begin{bmatrix} \begin{matrix} [\mathbf{v}_1]_\times [\mathbf{v}_2]_\times + s_1 [\mathbf{v}_2]_\times \\ + s_2 [\mathbf{v}_1]_\times + \mathbf{v}_1 \mathbf{v}_2^\top + s_1 s_2 \mathbf{I}_3 \end{matrix} & -[\mathbf{v}_1]_\times \mathbf{v}_2 - s_1 \mathbf{v}_2 + s_2 \mathbf{v}_1 \\ -\mathbf{v}_1^\top [\mathbf{v}_2]_\times - s_2 \mathbf{v}_1^\top + s_1 \mathbf{v}_2^\top & \mathbf{v}_1^\top \mathbf{v}_2 + s_1 s_2 \end{bmatrix}.$$

The bottom right element is zero since $\mathbf{q}_1 \perp \mathbf{q}_2$. Using the identity $[\mathbf{v}_1]_\times [\mathbf{v}_2]_\times = \mathbf{v}_2 \mathbf{v}_1^\top - \mathbf{v}_1^\top \mathbf{v}_2 \mathbf{I}_3$, we can simplify this further to

$$\mathbf{E}_{ij} = \mathbf{v}_2 \mathbf{v}_1^\top + \mathbf{v}_1 \mathbf{v}_2^\top + s_1 [\mathbf{v}_2]_\times + s_2 [\mathbf{v}_1]_\times + 2 s_1 s_2 \mathbf{I}_3, \tag{15}$$

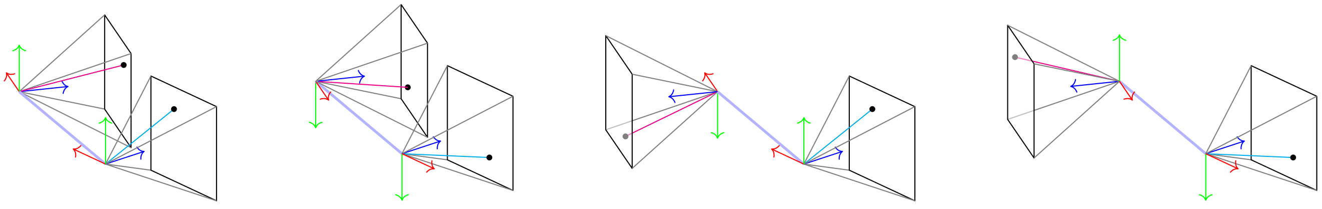
$$\mathbf{R}_i^\top \mathbf{t}_{ij} = -[\mathbf{v}_1]_\times \mathbf{v}_2 - s_1 \mathbf{v}_2 + s_2 \mathbf{v}_1, \tag{16}$$

$$\mathbf{R}_j^\top \mathbf{t}_{ji} = [\mathbf{v}_2]_\times \mathbf{v}_1 + s_1 \mathbf{v}_2 - s_2 \mathbf{v}_1. \tag{17}$$

One immediate observation is that if we multiply both $\mathbf{q}_1, \mathbf{q}_2$ by -1, the three terms above remain the same, but if we multiply only one of them, the sign of all three terms changes. Another interesting observation is that if we swap $\mathbf{q}_1$ and $\mathbf{q}_2$, the essential matrix $\mathbf{E}_{ij}$ remains the same, but the sign of epipoles is flipped, indicating that $\rotatebox{180}{$\mathbf{Q}$}_{ij} = \mathbf{Q}_{\mathtt{L}}(\mathbf{q}_2)\mathbf{Q}_{\mathtt{R}}(\mathbf{q}_1^{-1})$. These relations are summarized in Table 1.

4.4. Reduced Quintessential Manifold

We've established that the quintessential manifold is embedded in $\mathrm{SO}(4)$ with an additional constraint: the bottom right element must be zero. This characterization relies on 15 parameters and 10 quadratic constraints to ensure the orthonormality of columns. This section will introduce an alternative but equivalent representation for $\mathbb{Q}_+$ that employs



(a) $\mathbf{Q}_{ij}$ (b) $\rotatebox{180}{$\mathbf{Q}$}_{ij}$ (c) $-\rotatebox{180}{$\mathbf{Q}$}_{ij}$ (d) $-\mathbf{Q}_{ij}$

Figure 1: Twisted-pair ambiguity in two-view SfM and the four corresponding quintessential matrices.

Matrix	1st pair	2nd pair
$\mathbf{Q}_{ij}$	$\mathbf{q}_1, \mathbf{q}_2$	$-\mathbf{q}_1, -\mathbf{q}_2$
$-\mathbf{Q}_{ij}$	$-\mathbf{q}_1, \mathbf{q}_2$	$\mathbf{q}_1, -\mathbf{q}_2$
$\rotatebox{180}{$\mathbf{Q}$}_{ij}$	$\mathbf{q}_2, \mathbf{q}_1$	$-\mathbf{q}_2, -\mathbf{q}_1$
$-\rotatebox{180}{$\mathbf{Q}$}_{ij}$	$-\mathbf{q}_2, \mathbf{q}_1$	$\mathbf{q}_2, -\mathbf{q}_1$

Table 1: All possible quaternion-pair decompositions of quintessential matrices corresponding to the four solutions of the essential matrix estimation problem.

fewer parameters by embedding it in a lower-dimensional ambient space.

The core concept here revolves around the fact that any rotation matrix with a removed row or column can be fully recovered. This can be achieved, for instance, by performing a QR decomposition and selecting the sign of the removed row or column to maintain a determinant of one. Technically speaking, $\mathrm{St}(n-1, n)$ is diffeomorphic to $\mathrm{SO}(n)$.

Using this concept, we propose removing either the last row or the last column of $\mathbf{Q}_{ij}$ to preserve the essential matrix $\mathbf{E}_{ij}$. The resulting matrices from eliminating the last row and column of $\mathbf{Q}_{ij}$ are given by:

$$\mathbf{V}_{\mathtt{r}} = \begin{bmatrix} \mathbf{E}_{ij}^{\mathsf{T}} \\ \mathbf{t}_{ij}^{\mathsf{T}} \mathbf{R}_i \end{bmatrix}, \quad \mathbf{V}_{\mathtt{c}} = \begin{bmatrix} \mathbf{E}_{ij} \\ \mathbf{t}_{ji}^{\mathsf{T}} \mathbf{R}_j \end{bmatrix}. \tag{18}$$

These matrices are in $\mathbb{R}^{4\times 3}$ and satisfy $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}_3$. Moreover, their last row has a norm of one. We then define the *Reduced Quintessential Manifold* as:

$$\mathfrak{Q} \doteq \{\mathbf{V} \in \mathrm{St}(3,4) : \|\mathbf{e}_4^{\mathsf{T}}\mathbf{V}\| = 1\}, \tag{19}$$

where $\mathbf{e}_4 = [0,0,0,1]^{\mathsf{T}}$. This representation needs 12 parameters and enforces 7 constraints. In the following section, this definition will be utilized to solve the non-minimal essential matrix estimation problem.

5. Fast Global Solver using Burer-Monteiro Factorization

The algebraic error in (7) is quadratic in $\mathrm{vec}(\mathbf{E}) \in \mathbb{R}^9$, or equivalently in $\mathrm{vec}(\mathbf{V}) \in \mathbb{R}^{12}$ for $\mathbf{V} \in \mathfrak{Q}$. Furthermore, members of $\mathfrak{Q}$ can be characterized using seven quadratic constraints. This enables the formulation of the algebraic error minimization problem as a Quadratically Constrained Quadratic Program (QCQP), which can subsequently be transformed into a convex semidefinite programming (SDP) problem using Shor's relaxation. The relaxed SDP problem is defined as

$$\min_{\mathbf{X}\in\mathcal{S}_+^{12}} \langle \mathbf{C}, \mathbf{X} \rangle \quad \text{s.t.} \;\; \mathcal{A}(\mathbf{X}) = \mathbf{b}. \tag{20}$$

In this formulation, $\mathbf{X}$ replaces $\mathrm{vec}(\mathbf{V})\mathrm{vec}(\mathbf{V})^{\mathsf{T}}$ while relaxing the rank-1 constraint. The linear operator $\mathcal{A}(\cdot)$ enforces the three columns and the last row of $\mathbf{V}$ to be of norm one and also forces its columns to be orthogonal to each other. By adopting the convention of $\mathbf{V}_{\mathtt{r}}$ from (18) and column-wise vectorization, the objective matrix $\mathbf{C} \in \mathbb{R}^{12\times 12}$ takes the form

$$\mathbf{C} = \sum_{k=1}^{N} (\mathbf{f}_{i,k} \otimes \check{\mathbf{f}}_{j,k})(\mathbf{f}_{i,k} \otimes \check{\mathbf{f}}_{j,k})^{\mathsf{T}}, \tag{21}$$

for $\check{\mathbf{f}}_{j,k} \doteq [\mathbf{f}_{j,k}^{\mathsf{T}} \;\, 0]^{\mathsf{T}}$. Finally, this problem can be handed off to a standard SDP solver to obtain the optimal solution $\mathbf{X}^\star$, from which the optimal essential matrix can be extracted through a *rounding* procedure.

An equivalent SDP formulation of (20) was previously proposed in [37], in which the quadratic constraints used is $\mathbf{E}\mathbf{E}^{\mathsf{T}} = [\mathbf{t}]_\times [\mathbf{t}]_\times^{\mathsf{T}}$. The author proves this relaxation's tightness and local stability and utilizes an off-the-shelf interior-point-based SDP solver to find the optimal solution. Despite the efficient performance and certifiable global optimality, this method is not yet suitable for real-time applications due to the following reasons: *1)* The domain of the corresponding SDP problem is $\mathcal{S}_+^{12}$, which requires 78 parameters if we neglect the repeated ones due to the symmetry. This relaxation is largely over-parameterized, considering that the actual domain of interest $\mathbb{E}$ is five-dimensional. *2)* Standard interior-point algorithms used for semidefinite optimization have a high computational cost. In real-time applications with spurious feature matches, fast global solvers are needed to solve the essential matrix estimation problem multiple times in an iteratively reweighted least squares [27] or graduated non-convexity [34] routine. More computationally efficient solvers are hence needed to accommodate such needs.

In this section, we will introduce a Burer-Monteiro factorization [7] of (20) and analyze the feasible set of the factorized problem. Then we derive the machinery needed for applying a second-order Riemannian solver, which yields super-linear convergence and is robust to poor numerical conditioning. Later we present the expression for the certificate of optimality matrix and employ these derivations to solve the factorized problem with the *Riemannian staircase* algorithm [4]. The proposed algorithm exploits the fact that this formulation admits a low-rank solution, and its rank-restricted version is a smooth optimization problem on a Riemannian manifold. This approach needs as low as 24 parameters, and we will show that it outperforms state-of-the-art SDP solvers in runtime when the noise is mild.

5.1. Factorized SDP Problem

The Burer-Monteiro method proposes to factorize $\mathbf{X}$ from (20) as $\mathbf{X} = \mathbf{Y}\mathbf{Y}^\mathsf{T}$ and solve for $\mathbf{Y}$ by finding

$$\min_{\mathbf{Y}\in\mathbb{R}^{12\times r}} \langle \mathbf{C}, \mathbf{Y}\mathbf{Y}^\mathsf{T}\rangle \quad \text{s.t.} \quad \mathcal{A}(\mathbf{Y}\mathbf{Y}^\mathsf{T}) = \mathbf{b}. \tag{22}$$

This yields a *rank-restricted* version of (20) as its solution has $\operatorname{rank}(\mathbf{Y}\mathbf{Y}^\mathsf{T}) \leq r$. This reformulation reduces the search space dimension from 78 to $12r$ and eliminates the positive semidefiniteness constraint, albeit at the expense of convexity. However, a second-order optimal solution $\mathbf{Y}$ of (22) for a given r is not necessarily such that $\mathbf{Y}\mathbf{Y}^\mathsf{T}$ is the global optimal solution of (20). A pivotal aspect in this approach is the availability of a *dual certificate matrix* $\mathbf{S}(\mathbf{X})$, which aids in detecting if a solution from (22) is globally optimal and provides a direction for improvement if it is not.

Theorem 3 ([4, 28])**.** *If $\mathbf{Y}$ is a feasible point and a local minimizer of (22), then there exists a matrix $\mathbf{S} \in \mathcal{S}^{12}$ such that $\mathbf{S}\mathbf{Y} = \mathbf{0}$. In addition, if $\mathbf{S} \succeq \mathbf{0}$ then $\mathbf{X} = \mathbf{Y}\mathbf{Y}^\mathsf{T}$ is a global minimizer of (20). Otherwise, there exists $\mathbf{v} \in \mathbb{R}^{12}$ with $\mathbf{v}^\mathsf{T}\mathbf{S}\mathbf{v} < 0$ such that $\dot{\mathbf{Y}}_+ = [\mathbf{0}_{12\times r}\ \ \mathbf{v}]$ is a feasible second-order direction of descent from the lifted point $\mathbf{Y}_+ = [\mathbf{Y}\ \mathbf{0}_{12\times 1}] \in \mathbb{R}^{12\times(r+1)}$.*

Theorem 3 provides a strategy to achieve a globally optimal solution for (20). If the domain of (22) happens to be a Riemannian manifold, this theorem allows us to apply the Riemannian Staircase Algorithm [4, Alg. 1], outlined in Algorithm 1, to find such a solution. Instead of using $\mathbf{Y}$ directly in this algorithm, we rearrange its entries into another matrix $\mathbf{V}$, explaining this rearrangement in the subsequent section.

Algorithm 1 Riemannian Staircase

```
Require: An initial point $\mathbf{V} \in \mathfrak{Q}(r_0)$, $r_0 \geq 2$
  for $r = r_0, \ldots, 12$ do
      Starting from $\mathbf{V}$, find a second-order optimal solution
        $\mathbf{V}^\star \in \mathfrak{Q}(r)$ by applying a Riemannian optimizer
      $\mathbf{S} \leftarrow \mathrm{CertificateMatrix}(\mathbf{V}^\star)$
      if $\mathbf{S} \succeq \mathbf{0}$ then
          return $\mathbf{V}^\star$
      else
          $\mathbf{V} \leftarrow \begin{bmatrix} \mathbf{V}^\star \\ \mathbf{0}_{4\times 3} \end{bmatrix}$
          Escape saddle point by finding a direction of
            descent $\dot{\mathbf{V}}_+$ from $\mathbf{S}$ and update $\mathbf{V}$ ▷ Theorem 3
      end if
  end for
```

5.2. Factorized Manifold

This section finds the manifold structure of $\mathbf{Y}$, which is imposed by the seven linear constraints in $\mathcal{A}(\mathbf{X}) = \mathbf{b}$ as

$$\begin{aligned}
&\sum_{i=1}^{4} x_{k+i,k+i} = 1, && k \in \{0,4,8\},\\
&\sum_{i=1}^{4} x_{k_1+i,k_2+i} = 0, && k_1, k_2 \in \{0,4,8\}, k_1 < k_2,\\
&x_{4,4} + x_{8,8} + x_{12,12} = 1.
\end{aligned} \tag{23}$$

If we denote the three $4 \times r$ blocks of $\mathbf{Y}$ as

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \mathbf{Y}_3 \end{bmatrix}, \quad \mathbf{Y}_i = \begin{bmatrix} \mathbf{e}_i^{(1)} & \cdots & \mathbf{e}_i^{(r)} \\ t_i^{(1)} & \cdots & t_i^{(r)} \end{bmatrix} \in \mathbb{R}^{4\times r}, \tag{24}$$

then we can rewrite the constraints in (23) for $\mathbf{Y}$ as

$$\begin{aligned}
&\|\mathbf{Y}_i\|_\mathsf{F} = 1 && i \in \{1,2,3\}\\
&\langle \mathbf{Y}_i, \mathbf{Y}_j \rangle = 0, && i \neq j\\
&\|\mathbf{e}_4^\mathsf{T}\mathbf{Y}_1\|_\mathsf{F}^2 + \|\mathbf{e}_4^\mathsf{T}\mathbf{Y}_2\|_\mathsf{F}^2 + \|\mathbf{e}_4^\mathsf{T}\mathbf{Y}_3\|_\mathsf{F}^2 = 1.
\end{aligned} \tag{25}$$

The first three constraints in (25) force the $\operatorname{vec}(\mathbf{Y}_i)$s to be unit norm vectors, and the next three enforce that $\operatorname{vec}(\mathbf{Y}_i) \perp \operatorname{vec}(\mathbf{Y}_j)$. Concatenating these three orthonormal vectors into a matrix $\mathbf{V} \in \mathbb{R}^{4r\times 3}$ gives

$$\mathbf{V} = \begin{bmatrix} \operatorname{vec}(\mathbf{Y}_1) & \operatorname{vec}(\mathbf{Y}_2) & \operatorname{vec}(\mathbf{Y}_3) \end{bmatrix} \in \mathrm{St}(3, 4r). \tag{26}$$

The last constraint of (25) requires that $\|\mathbf{V}_t\|_\mathsf{F} = 1$, where $\mathbf{V}_t \in \mathbb{R}^{r\times 3}$ is a submatrix of $\mathbf{V}$ containing the rows with epipole variables $t_i^{(k)}$, given as $\mathbf{V}_t \doteq (\mathbf{I}_r \otimes \mathbf{e}_4^\mathsf{T})\mathbf{V}$.

If $r = 1$, we have that $\mathbf{V}$ from (26) lives in $\mathrm{St}(3,4)$ and satisfies $\|\mathbf{e}_4^\mathsf{T}\mathbf{V}\| = 1$, which means that $\mathbf{V}$ lives in the reduced quintessential manifold $\mathfrak{Q}$ defined earlier in (19).

For larger r, we can define a similar set by extending the definition of $\mathfrak{Q}$ using

$$\mathfrak{Q}(r) \doteq \{\mathbf{V} \in \mathrm{St}(3, 4r) : \|(\mathbf{I}_r \otimes \mathbf{e}_4^\top)\mathbf{V}\|_\mathtt{F} = 1\}. \tag{27}$$

Notice how increasing the rank increases the size of the domain

$$\mathfrak{Q}(1) \subset \mathfrak{Q}(2) \subset \cdots \subset \mathfrak{Q}(12). \tag{28}$$

We will use the Stiefel-based arrangement of variables in $\mathbf{V}$ instead of $\mathbf{Y}$ since it allows for a more succinct expression of the derivatives and projections. In the interest of space, we will present the proofs of these derivations in the supplementary material.

5.3. Projection on Tangent Space

A point $\dot{\mathbf{V}}$ on the tangent space of $\mathfrak{Q}(r)$ must satisfy $\mathbf{V}^\top\dot{\mathbf{V}} + \dot{\mathbf{V}}^\top\mathbf{V} = \mathbf{0}$ and also $\langle \mathbf{V}_t, \dot{\mathbf{V}}_t \rangle = 0$. Given a point $\mathbf{Z}$, its projection on the tangent space of $\mathfrak{Q}(r)$ is given by

$$\begin{aligned} \tilde{\mathbf{Z}} &= \mathbf{Z} - \mathbf{V}\mathrm{sym}(\mathbf{V}^\top\mathbf{Z}), \\ \mathrm{Proj}_\mathbf{V}(\mathbf{Z}) &= \tilde{\mathbf{Z}} - \frac{\langle \mathbf{V}_t, \tilde{\mathbf{Z}}_t \rangle}{1 - \|\mathbf{V}_t^\top\mathbf{V}_t\|_\mathtt{F}^2}(\mathbf{I}_{4r} - \mathbf{V}\mathbf{V}^\top)\mathbf{G}\mathbf{V}, \end{aligned} \tag{29}$$

where $\mathbf{G}$ is defined as $\mathbf{G} \doteq \mathbf{I}_r \otimes (\mathbf{e}_4\mathbf{e}_4^\top) \in \mathbb{R}^{4r\times 4r}$. Intuitively, this is a two-step process where first $\mathbf{Z}$ is projected on the tangent space of $\mathrm{St}(3, 4r)$ given by $\tilde{\mathbf{Z}}$, and then the component of $\tilde{\mathbf{Z}}_t$ parallel to $\mathbf{V}_t$ is removed.

5.4. Newton Retraction

A retraction is essentially a smooth mapping from the tangent bundle of a manifold onto itself, approximating the exponential map to the first order. We employ the Newton retraction [36], which provides a second-order approximation of the exponential map by shifting points along manifolds orthogonal to the constraint manifold. To execute this operation, we require a function $\boldsymbol{f}(\cdot) : \mathbb{R}^{4r\times 3} \mapsto \mathbb{R}^7$ that evaluates the values of the seven constraints on $\mathfrak{Q}(r)$—six values for the orthonormality of columns and one for the unit Frobenius norm of rows with epipole variables. Additionally, we need a function $\boldsymbol{J}(\cdot) : \mathbb{R}^{4r\times 3} \mapsto \mathbb{R}^{7\times 12r}$ that computes the Jacobian matrix of $\boldsymbol{f}(\cdot)$. With these components in place, we can outline the retraction process as depicted in Algorithm 2. The $\mathrm{mat}(\cdot)$ represents the inverse of $\mathrm{vec}(\cdot)$ function. We set the convergence threshold to $c_0 = 10^{-4}$.

5.5. Gradient and Hessian

Since $\mathbf{V}$ comes from rearranging the entries of $\mathbf{Y}$, there is a direct relationship between the gradient and the hessian of the objective function $g(\mathbf{Y}) = h(\mathbf{V}) \doteq \langle \mathbf{C}, \mathbf{Y}\mathbf{Y}^\top \rangle$. If we denote the mapping $\mathbf{Y} \to \mathbf{V}$ by $\ell(\cdot) : \mathbb{R}^{12\times r} \mapsto \mathbb{R}^{4r\times 3}$ introduced in (26), then we obtain the Euclidean gradient and hessian of h as

$$\begin{aligned} \nabla h(\mathbf{V}) &= \ell(\nabla g(\mathbf{Y})) = \ell(2\mathbf{C}\mathbf{Y}), \\ \nabla^2 h(\mathbf{V})[\dot{\mathbf{V}}] &= \ell(\nabla^2 g(\mathbf{Y})[\dot{\mathbf{Y}}]) = \ell(2\mathbf{C}\dot{\mathbf{Y}}). \end{aligned} \tag{30}$$

Algorithm 2 Newton Retraction

```
Require: point & tangent $(\mathbf{V}, \dot{\mathbf{V}})$, convergence thresh. $c_0$
  $\mathbf{V} \leftarrow \mathbf{V} + \dot{\mathbf{V}}$
  repeat
      $\mathbf{J} \leftarrow \boldsymbol{J}(\mathbf{V})$          ▷ Update the Jacobian using $\mathbf{V}$
      solve $(\mathbf{J}\mathbf{J}^\top)\mathbf{x} = \boldsymbol{f}(\mathbf{V})$
      $\boldsymbol{\delta} \leftarrow -\mathbf{J}^\top\mathbf{x}$
      $\mathbf{V} \leftarrow \mathbf{V} + \mathrm{mat}(\boldsymbol{\delta})$
  until $\|\boldsymbol{\delta}\| < c_0$
  return $\mathbf{V}$
```

From there, we get the Riemannian gradient and Hessian as

$$\begin{aligned} \mathrm{grad}\, h(\mathbf{V}) &= \mathrm{Proj}_\mathbf{V}(\nabla h(\mathbf{V})) \\ \mathrm{Hess}\, g(\mathbf{V})[\dot{\mathbf{V}}] &= \mathrm{Proj}_\mathbf{V}\Big(\mathrm{D}\big(\mathrm{grad}\, h(\mathbf{V})\big)(\mathbf{V})[\dot{\mathbf{V}}]\Big) \end{aligned} \tag{31}$$

The gradient is thus given using the projection in (29), and the full expression of the Hessian is available in the supplementary material.

5.6. Certificate Matrix

We present a closed-form expression for the certificate matrix $\mathbf{S}$ introduced in Theorem 3. For a KKT point $\mathbf{Y}$ of (22), we have $\mathrm{grad}\, g(\mathbf{Y}) = \mathrm{Proj}_{\ell(\mathbf{Y})}\big(\ell(2\mathbf{C}\mathbf{Y})\big) = \mathbf{0}$. Simplifying this leads to $\mathbf{S}\mathbf{Y} = \mathbf{0}$, and $\mathbf{S}$ is given by

$$\mathbf{S} = \mathbf{C} - \mathrm{sym}(\mathbf{M}) \otimes \mathbf{I}_4 \tag{32}$$

where $\mathbf{M}$ is a 3×3 matrix whose entries are the trace of the corresponding 4×4 blocks of $\mathbf{C}\mathbf{X}$, from $\mathbf{X} = \mathbf{Y}\mathbf{Y}^\top$. More formally, $m_{ij} = \langle \mathbf{C}\mathbf{X}, (\mathbf{e}_i\mathbf{e}_j^\top) \otimes \mathbf{I}_4 \rangle$ where $\mathbf{e}_k \in \mathbb{R}^3$ is the unit vector with its kth entry equal to one.

We emphasize that this certificate matrix can be used to determine the global optimality of the relaxed problem (20) for a solution obtained from other solvers, e.g., a minimal solver like the 5-point method.

5.7. Random Sampling on $\mathfrak{Q}(r)$

To obtain a random point on $\mathfrak{Q}(r)$, we first sample a random $4r \times 3$ matrix $\tilde{\mathbf{V}}$. Then from $\tilde{\mathbf{V}}$, we obtain $\mathbf{V}^\bullet \in \mathfrak{Q}(r)$ by first normalizing rows corresponding to the epipole variables. Later, we update the remaining rows by multiplying them by a symmetric matrix, as

$$\begin{aligned} \mathbf{V}_t^\bullet &= \tilde{\mathbf{V}}_t\|\tilde{\mathbf{V}}_t\|_\mathtt{F}^{-1}, \\ \mathbf{K} &= (\mathbf{I}_3 - \mathbf{V}_t^{\bullet\top}\mathbf{V}_t^\bullet)^{\frac{1}{2}}, \\ \mathbf{V}_E^\bullet &= \tilde{\mathbf{V}}_E\mathbf{K}(\mathbf{K}\tilde{\mathbf{V}}_E^\top\tilde{\mathbf{V}}_E\mathbf{K})^{\frac{\dagger}{2}}\mathbf{K}, \end{aligned} \tag{33}$$

where † is the Moore–Penrose inverse.

5.8. Initialization Technique

Albeit our solver can obtain the globally optimal solution, having a good starting point can speed up the process. We propose to initialize the solver using the eigenvector corresponding to the smallest eigenvalue of the bearing matrix, defined as

$$\mathbf{C}_{\mathbb{E}} \doteq \sum_{k=1}^{N} (\mathbf{f}_{i,k} \otimes \mathbf{f}_{j,k})(\mathbf{f}_{i,k} \otimes \mathbf{f}_{j,k})^{\mathsf{T}} \in \mathbb{R}^{9\times 9}. \tag{34}$$

This vector is then placed in a 3×3 matrix and projected on the essential manifold by taking an SVD and replacing the singular values with $\{1, 1, 0\}$. This essential matrix and its corresponding right null vector are then placed into the top 4×3 block of the initial matrix $\mathbf{V} \in \mathfrak{Q}(r_0)$.

5.9. Rounding the Solution

We use the rounding procedure from [37]. First, from a given solution $\mathbf{X}^{\star} \in \mathcal{S}_{+}^{12}$ we construct the matrix $\mathbf{X}_{\mathbb{E}}^{\star} \in \mathcal{S}_{+}^{9}$ containing only entries of the essential matrix. Later we find the eigenvector $\mathbf{v}$ corresponding to the largest eigenvalue of $\mathbf{X}_{\mathbb{E}}^{\star}$ and then project $\mathrm{mat}(\mathbf{v}) \in \mathbb{R}^{3\times 3}$ onto $\mathbb{E}$.

6. Local Solver by Orthogonal Decomposition

After solving (20), whether through an interior point solver or the Riemannian Staircase method, we obtain a solution $\mathbf{X}^{\star}$ with a rank of at least two. This outcome stems from the twisted-pair ambiguity, i.e., all cross-terms between the essential matrix and its epipole in $\mathbf{X}^{\star}$ can be flipped without impacting its optimality. Should $\mathbf{X}^{\star}$ exhibit a rank exceeding two (i.e., if $\mathbf{X}_{\mathbb{E}}^{\star}$ has a rank greater than one), we project the solution onto $\mathbb{E}$. This projection yields a solution close to the higher-rank global solution, yet it may not precisely be the global optimizer of (7). To bridge this gap, we introduce a local solver here to refine this rounded solution further.

Our local solver surfs $\mathbb{Q}$ by decomposing $\mathbf{Q}$ into two orthogonal matrices as

$$\mathbf{Q} = \mathbf{O}_1^{\mathsf{T}} \mathbf{O}_2, \tag{35}$$

where the last columns of $\mathbf{O}_1, \mathbf{O}_2$ are perpendicular due to $q_{44} = 0$. We define the QO^2 manifold to capture these constraints as follows

$$\mathrm{QO}^2 \doteq \{(\mathbf{O}_1, \mathbf{O}_2) \in \mathrm{O}(4)^2 : \langle \mathbf{O}_1^{\mathsf{T}} \mathbf{O}_2, \mathbf{F} \rangle = 0\}, \tag{36}$$

where $\mathbf{F}$ is given by $\mathbf{F} \doteq \mathbf{e}_4 \mathbf{e}_4^{\mathsf{T}} \in \mathbb{R}^{4\times 4}$. Needless to say, we can always fix one of these matrices (e.g., set $\mathbf{O}_1$ to $\mathbf{I}_4$) to reduce the number of parameters in this approach.

6.1. Projection on Tangent Space

The matrices $(\dot{\mathbf{O}}_1, \dot{\mathbf{O}}_2)$ on the tangent space of QO^2 must satisfy $\{\mathbf{O}_i^{\mathsf{T}} \dot{\mathbf{O}}_i + \dot{\mathbf{O}}_i^{\mathsf{T}} \mathbf{O}_i = \mathbf{0}\}_{i=1}^{2}$ and $\langle \dot{\mathbf{O}}_1^{\mathsf{T}} \mathbf{O}_2 + \mathbf{O}_1^{\mathsf{T}} \dot{\mathbf{O}}_2, \mathbf{F} \rangle = 0$. Similar to $\mathfrak{Q}$, we see that projection on the tangent space of QO^2 is a two-step process, in which the first step is the projection on the tangent space of $\mathrm{O}(4)^2$ and the second one makes the last columns perpendicular.

$$\tilde{\mathbf{Z}}_i \doteq \mathbf{Z}_i - \mathbf{O}_i \,\mathrm{sym}(\mathbf{O}_i^{\mathsf{T}} \mathbf{Z}_i),$$
$$\mathrm{Proj}_{\mathbf{O}_i}(\mathbf{Z}_i) \doteq \tilde{\mathbf{Z}}_i - \frac{\sum \langle \mathbf{O}_i^{\mathsf{T}} \tilde{\mathbf{Z}}_j, \mathbf{F} \rangle}{2} \mathbf{O}_i \,\mathrm{skew}(\mathbf{O}_i^{\mathsf{T}} \mathbf{O}_j \mathbf{F}). \tag{37}$$

6.2. Retraction

One popular retraction on $\mathrm{O}(4)$ is given by QR and QL decompositions [1]. These two retractions have the useful property of leaving the direction of the first/last column unchanged. For the QO^2 manifold, the last columns of $\mathbf{O}_1, \mathbf{O}_2$ are orthonormal, i.e., they belong to $\mathrm{St}(2, 4)$. Using this property, we use the SVD retraction [2] for the Stiefel manifold on these two columns and insert the retracted values in the last columns of $\mathbf{O}_i + \dot{\mathbf{O}}_i$. Later, we perform a QL retraction to keep the last columns we got from SVD unchanged. This will ensure that the retracted matrices belong to QO^2.

6.3. Gradient and Hessian

Given an error function $f(\mathbf{Q})$ such as (7), we have the Euclidean gradients as

$$\begin{aligned} \nabla_{\mathbf{O}_1} f(\mathbf{Q}) &= 2\mathbf{O}_2 \nabla_{\mathbf{Q}} f(\mathbf{Q})^{\mathsf{T}} \\ \nabla_{\mathbf{O}_2} f(\mathbf{Q}) &= 2\mathbf{O}_1 \nabla_{\mathbf{Q}} f(\mathbf{Q}) \end{aligned} \tag{38}$$

Using the projection in (37), we find the Riemannian gradient and Hessian, as given in the supplementary material.

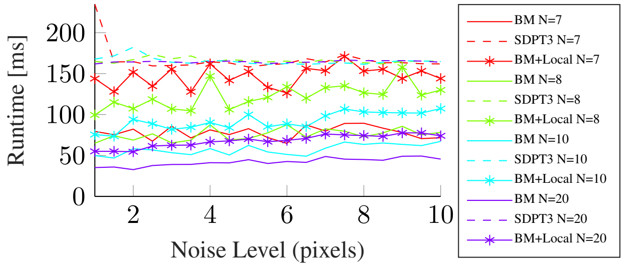
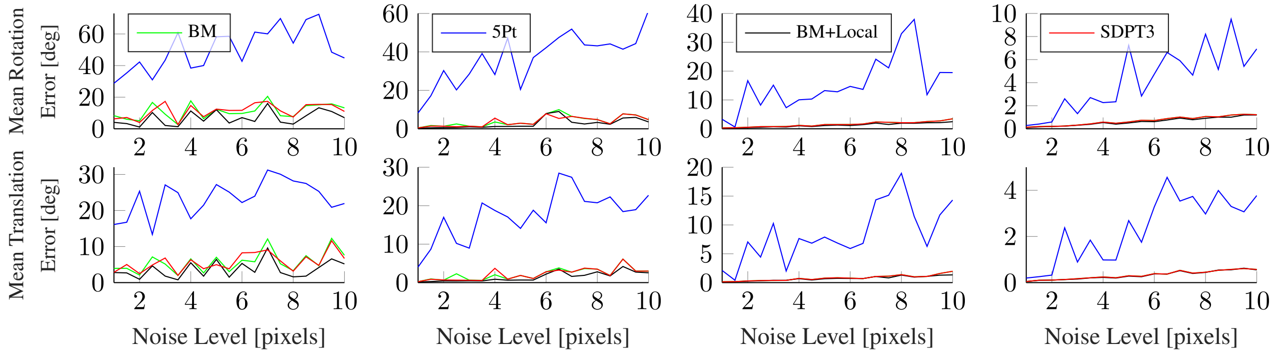


Figure 2: Runtime of our global solver (BM & BM+local) implemented in MATLAB versus the SDPT3 solver.

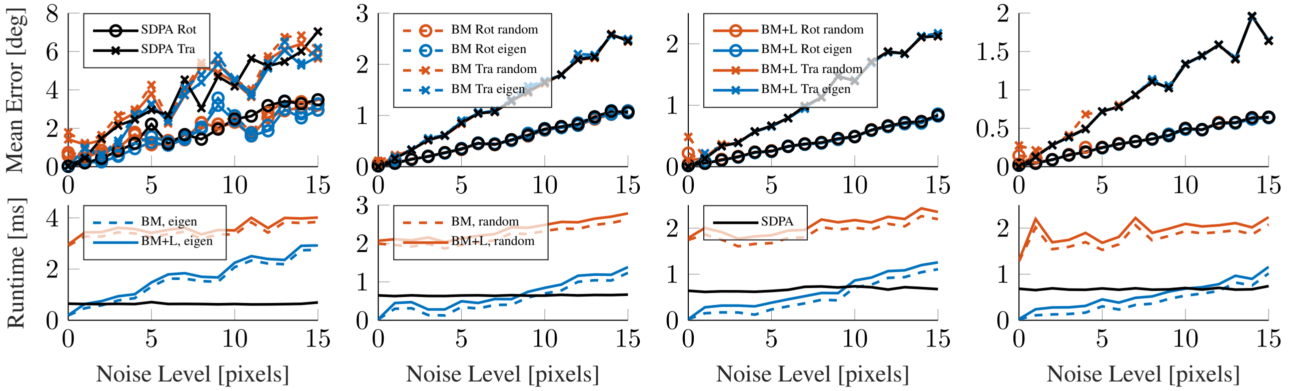
7. Results

We implemented our global and local solvers in MATLAB and C++. For the MATLAB version, we use Manopt [6] and its second-order Riemannian trust-region method [5]. To evaluate the performance of our staircase method, we compare it with its SDP version given in (20) and use SDPT3 [31] to solve it. We use synthetic data with 1900 randomly generated epipolar configurations and feature



(a) $N = 7$ (b) $N = 8$ (c) $N = 10$ (d) $N = 20$

Figure 3: Rotation and translation error between the estimated relative pose and the ground-truth values. The top row shows the rotation error in degrees and the bottom row shows the angle between the estimated and ground-truth baseline vector.



(a) $N = 8$ (b) $N = 12$ (c) $N = 16$ (d) $N = 20$

Figure 4: Performance comparison between our solvers (BM and BM+L) and SDPA. Top row shows the rotation (Rot) and translation (Tra) errors between the estimated relative and the ground truth. For each experiment, the BM solver is initialized both randomly (random) and using the method from Sec. 5.8 (eigen).

points. We add Gaussian noise to feature points in the image plane with a standard deviation between 1 and 10, with increments of 0.5. For a given N and standard deviation, 25 total random configurations are generated such that the relative rotation and translation are randomly generated over a uniform distribution. The rotation angle is uniformly sampled between 0 and $\frac{\pi}{2}$. We also compare our algorithm with MATLAB's 5-point solver. These results are given in Fig. 2 and 3 for random initialization as described in Sec. 5.7.

For the C++ implementation, we compare our approach's runtime and estimation error with SDPA [33], the highly efficient SDP solver used in [37]. We use standard deviations from 0 to 15; for each N and standard deviation, we generated 100 random configurations. In the C++ implementation, once the staircase method (denoted as BM) returns a rounded solution, we carry the local optimization on $\mathfrak{Q}(1)$ instead of QO^2 for faster runtime. These results are given in Fig. 4.

We can see in Fig. 2 and 4 that as N increases, the staircase method finds a solution faster. We speculate this to be due to the improved condition number of $\mathbf{C}$ and, consequently, the Hessian as the Euclidean Hessian is proportional to $\mathbf{C}$ (see (30)). More importantly, we see in Fig. 4 that our global solver with the initialization from Sec. 5.8 outperforms SDPA in most noise levels, with a much better performance in low noise scenarios, whereas SDPA (and SDPT3) yield a nearly constant runtime in all cases.

8. Conclusion

We presented a fast and certifiably correct essential matrix solver that outperforms the best existing method in low to moderate noise levels. We also introduced quintessential matrices and uncovered their relationship with epipolar configurations and orthogonal unit quaternions.

References

[1] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. Optimization algorithms on matrix manifolds. In *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2009. 2, 8

[2] P-A Absil and Jérôme Malick. Projection-like retractions on matrix manifolds. *SIAM Journal on Optimization*, 22(1):135–158, 2012. 8

[3] M. Arie-Nachimson, S.Z. Kovalsky, I. Kemelmacher-Shlizerman, A. Singer, and R. Basri. Global motion estimation from point matches. In *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, pages 81–88, 2012. 2, 3

[4] Nicolas Boumal. A riemannian low-rank method for optimization over semidefinite matrices with block-diagonal constraints. *arXiv preprint arXiv:1506.00575*, 2015. 2, 6

[5] Nicolas Boumal, Pierre-Antoine Absil, and Coralia Cartis. Global rates of convergence for nonconvex optimization on manifolds. *IMA Journal of Numerical Analysis*, 39(1):1–33, 2019. 8

[6] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre. Manopt, a Matlab toolbox for optimization on manifolds. *Journal of Machine Learning Research*, 15(42):1455–1459, 2014. 8

[7] Samuel Burer and Renato DC Monteiro. Local minima and convergence in low-rank semidefinite programming. *Mathematical programming*, 103(3):427–444, 2005. 6

[8] A. Chatterjee and V. M. Govindu. Efficient and robust large-scale rotation averaging. In *IEEE International Conference on Computer Vision*, pages 521–528, 2013. 2

[9] Frank Dellaert, David M Rosen, Jing Wu, Robert Mahony, and Luca Carlone. Shonan rotation averaging: global optimality by surfing $so(p)^n$. pages 292–308. Springer, 2020. 2

[10] Anders Eriksson, Carl Olsson, Fredrik Kahl, and Tat-Jun Chin. Rotation averaging and strong duality. In *IEEE International Conference on Computer Vision*, pages 127–135, 2018. 2

[11] Oliver Faugeras. *Three dimensional computer vision: A geometric viewpoint*. The MIT Press, 1993. 1

[12] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 1

[13] J. Fredriksson and C. Olsson. Simultaneous multiple rotation averaging using lagrangian duality. In *Asian Conference on Computer Vision*, 2012. 2

[14] Geyer and Daniilidis. Mirrors in motion: Epipolar geometry and motion estimation. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 766–773. IEEE, 2003. 3

[15] V. M. Govindu. Robustness in motion averaging. In *Asian Conference on Computer Vision*, volume 3852, pages 457–466. Springer, 2006. 2

[16] R. Hartley and H. Li. An efficient hidden variable approach to minimal-case camera motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012. 1

[17] R. Hartley, J. Trumpf, Y. Dai, and H. Li. Rotation averaging. *International Journal of Computer Vision*, 103(3):267–305, 2013. 2

[18] Richard I Hartley and Fredrik Kahl. Global optimization through rotation space search. *International Journal of Computer Vision*, 82(1):64–79, 2009. 2

[19] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. 1

[20] U. Helmke, K. Hüper, P. Y. Lee, and J. Moore. Essential matrix estimation using gauss-newton iterations on a manifold. *International Journal of Computer Vision*, 74(2):117–136, 2007. 1

[21] Matanya B Horowitz, Nikolai Matni, and Joel W Burdick. Convex relaxations of SE(2) and SE(3) for visual pose estimation. In *IEEE International Conference on Robotics and Automation*, pages 1148–1154. IEEE, 2014. 2

[22] Yi Ma. *An invitation to 3-D vision: from images to geometric models*. Springer, 2004. 1

[23] Y. Ma, J. Kosecka, and S. Sastry. Optimization criteria and geometric algorithms for motion and structure estimation. *International Journal of Computer Vision*, 44(3):219–249, 2001. 1

[24] D. Martinec and T. Pajdla. Robust rotation and translation estimation in multiview reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007. 2

[25] David Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–770, 2004. 1

[26] Carl Olsson, Anders Eriksson, and Richard Hartley. Outlier removal using duality. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1450–1457, 2010. 2

[27] Liangzu Peng, Christian Kümmerle, and René Vidal. Global linear and local superlinear convergence of irls for non-smooth robust regression. *Advances in Neural Information Processing Systems*, 35:28972–28987, 2022. 5

[28] David M Rosen. Scalable low-rank semidefinite programming for certifiably correct machine perception. In *International Workshop on the Algorithmic Foundations of Robotics*, pages 551–566. Springer, 2020. 6

[29] David M Rosen, Luca Carlone, Afonso S Bandeira, and John J Leonard. SE-Sync: A certifiably correct algorithm for synchronization over the special Euclidean group. *The International Journal of Robotics Research*, 38(2-3):95–125, 2019. 2

[30] J. Saunderson, P. A. Parrilo, and A. S. Willsky. Semidefinite relaxations for optimization problems over rotation matrices. In *IEEE International Conference on Decision and Control*, 2014. 2

[31] Kim-Chuan Toh, Michael J Todd, and Reha H Tütüncü. Sdpt3—a matlab software package for semidefinite programming, version 1.3. *Optimization methods and software*, 11(1-4):545–581, 1999. 8

[32] Roberto Tron and Kostas Daniilidis. The space of essential matrices as a riemannian quotient manifold. *SIAM Journal on Imaging Sciences*, 10(3):1416–1445, 2017. 1, 3

[33] Makoto Yamashita, Katsuki Fujisawa, Mituhiro Fukuda, Kazuhiro Kobayashi, Kazuhide Nakata, and Maho Nakata. Latest developments in the sdpa family for solving large-scale sdps. *Handbook on semidefinite, conic and polynomial optimization*, pages 687–713, 2012. 9

[34] Heng Yang, Pasquale Antonante, Vasileios Tzoumas, and Luca Carlone. Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection. *IEEE Robotics and Automation Letters*, 5(2):1127–1134, 2020. 6

[35] D. Yuchao, J. Trumpf, H. Li, N. Barnes, and R. Hartley. Rotation averaging with application to camera-rig calibration. In *Asian Conference on Computer Vision*, pages 335–346, 2010. 2

[36] R Zhang. Newton retraction as approximate geodesics on submanifolds. 2020, 2006. 7

[37] Ji Zhao. An efficient solution to non-minimal case essential matrix estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 5, 8, 9