On the quotient representation for the essential manifold

Roberto Tron and Kostas Daniilidis*

Abstract

The essential matrix, which encodes the epipolar constraint between projected points in two views, is a corner stone of modern computer vision. Previous works have proposed different characterizations of the space of essential matrices as a Riemannian manifold. However, these works either do not consider the symmetric role played by the two views, or do not fully take into account the geometric peculiarities of the epipolar constraint. We address these limitations and give a characterization as a quotient manifold which preserves the geometrical interpretation in terms of camera poses. While our main focus in on theoretical aspects, we include experiments in pose averaging, and show that the proposed formulation produces a meaningful distance between essential matrices.

1. Introduction

The essential matrix and the epipolar constraint have been introduced more than thirty years ago [**?**]. Since then, they have been a major mainstay of compute vision and the basic building block in any Structure from Motion (SfM) system. Its robust estimation from image data is now textbook material [7, 10]. In practical terms, the space of essential matrices is a subset of $\mathbb{R}^{3\times3}$, but the algebraic relations imposed by the epipolar constraint render its geometry far from trivial. There have been a few attempts to interpret this space as a Riemannian manifold. The earliest works in this aspect are [11, 13], which use the relative pose between the two cameras (with a normalized translation) to parametrize the space of essential matrices (*i.e.*, each essential matrix is given as the product of a skew-symmetric matrix with a rotation). This implies a preferential treatment of one of the two cameras, whose local reference frame is chosen as a global reference frame, which breaks the natural symmetry of the epipolar constraint. A different representation, based on the Singular Value Decomposition (SVD) of the essential matrix, was used in [**?**, 8, 15, 16]. While this representation has a natural symmetry, previous works do not provide an intuitive geometric interpretation of its parameters. Also, they do not properly take into account the well-known twisted-pair ambiguity (the fact that four different pairs of poses correspond to the same essential matrix), and the algorithm used for the computation of the logarithm map (which is related to the notion of geodesics in this space) is neither efficient nor rigorously motivated.

In this paper, we propose a characterization which is closely related to the aforementioned SVD representation and we show how:

1. Our approach naturally arises from a particular choice of the global reference frame, and that the parameters have a clear geometric meaning.
2. The cheirality constraint (*i.e.*, the constraint that all the points lie in front of both cameras) impacts our representation and how it can be used to simplify the structure of the space.
3. To endow the space with a Riemannian manifold structure, and how to naturally obtain geodesics in this space from those in the space of camera rotations.
4. To efficiently compute the logarithm map and distance function, and how these can be used in a two-view SfM problem using the Weiszfeld algorithm.

Some material in this paper might appear quite basic for any reader versed in computer vision. However, it is necessary to revisit it and place it in the context of our work.

2. Definitions and notation

In this section we recall several definitions from Riemannian geometry and group theory. We mention just the minimum necessary to follow the paper, and we refer the reader to the literature for the complete and rigorous definitions [2].

At a high level, a *manifold* $\mathcal{M}$ is defined by a topological space together with a set of overlapping local coordinate charts, which allow to locally parametrize the space and smoothly pass from one chart to the other. The *tangent space* at a point $x \in \mathcal{M}$, denoted as $T_x\mathcal{M}$, can be defined as the linear space containing all the *tangent vectors* corresponding to the curves passing through x. We use the notation $v^\vee$ to

*The authors are with the Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104, {tron,kostas}@seas.upenn.edu. This work was supported by grants NSF-IIP-0742304, NSF-OIA-1028009, ARL MAST CTA W911NF-08-2-0004, ARL Robotics CTA W911NF-10-2-0016, NSF-DGE-0966142, and NSF-IIS-1317788.

denote the vector of coordinates of v in some reference frame for $T_x\mathcal{M}$. A *vector field* X assigns a tangent vector to each x in $\mathcal{M}$ or a subset of it. A Riemannian manifold $(\mathcal{M}, \langle\cdot,\cdot\rangle)$ is a manifold equipped with a metric, that is, a collection of inner products $\langle\cdot,\cdot\rangle_x$ over $T_x\mathcal{M}$ which varies smoothly with x. The metric is used to define the length of a curve $\gamma : \mathbb{R} \supset [a, b] \to \mathcal{M}$. A curve is a *geodesic* if $\nabla_{\dot\gamma}\dot\gamma \equiv 0$. The *exponential map* $\exp_x : T_x\mathcal{M} \to \mathcal{M}$ maps each tangent vector v to the endpoint of the unit-speed geodesic starting at x and having v as its tangent at x. The *logarithm map* $\log_x$ is the inverse of the $\exp_x$ and is defined (in general) only on a neighborhood of x. We use the shorthand notation $\mathrm{Log} = \log^\vee$. The negative logarithm is the gradient of the squared distance function, *i.e.*, for any point x and any curve $\gamma(t)$ in $\mathcal{M}$ sufficiently close together we have

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}d^2\big(x,\gamma(t)\big) = -\operatorname{Log}_x\big(\gamma(t)\big)^T\dot\gamma(t)^\vee. \tag{1}$$

Given a map between manifolds $f : \mathcal{M} \to \tilde{\mathcal{M}}$, we define $\mathrm{D}f$ as the *differential* of the map, *i.e.*, the linear operator (the Jacobian, in local coordinates) that maps $T_x\mathcal{M}$ to $T_{f(x)}\tilde{\mathcal{M}}$ for any $x \in \mathcal{M}$ and satisfies, for any curve $\gamma(t) \in \mathcal{M}$, the expression

$$\left(\frac{\mathrm{d}}{\mathrm{d}t}f\big(\gamma(t)\big)\right)^\vee = \mathrm{D}f\big(\gamma(t)\big)\dot\gamma^\vee. \tag{2}$$

A *group* is a set G together with an operation $\circ : G\times G \to G$ which satisfy the four axioms of closure, associativity, identity element (denoted as $e \in G$) and inverse element. A *group action* "$\cdot$" on a set M is a mapping $\cdot : G \times M \to M$ which satisfies the properties $g \cdot (h \cdot x) = (g \circ h) \cdot x$ and $e \cdot x = x$ for all $g, h \in G$, $x \in M$. The group action induces an *equivalence relation* between the points in M, and we say that x is equivalent to y, *i.e.*, $x \sim y$ if there exist $g \in G$ such that $g \cdot x = y$. We denote all the elements equivalent to $x \in M$ as the *equivalence class* $[x]$. The *quotient space* M/G is the space of all equivalence classes. The *canonical projection* $\pi : G \to M/G$ maps each point $x \in \mathcal{M}$ to $[x]$.

In this paper, we will heavily use the space of 3-D rotations $SO(3) = \{R \in \mathbb{R}^{3\times3} : R^TR = I, \det(R) = 1\}$, and, to a less extent, the space of rigid body transformations $SE(3) = SO(3) \ltimes \mathbb{R}^3$. We will also use $SO(3)^2$, the cartesian product of $SO(3)$ with itself. The space $SO(3)$ is a *Lie group*, *i.e.*, it is at the same time a group (with the matrix multiplication) and a manifold. The tangent space at $R \in SO(3)$ is $T_RSO(3) = \{RV : V \in \mathfrak{so}(n)\}$, where $\mathfrak{so}(3)$ is the space of 3×3 skew-symmetric matrices. We can identify a tangent vector $v \in T_RSO(3)$ with a vector of local coordinates $w \in \mathbb{R}^3$ using the usual *hat* $(\cdot)^\wedge$ and *vee* $(\cdot)^\vee$ operators, given by the relations

$$w = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \overset{(\cdot)^\wedge}{\underset{(\cdot)^\vee}{\rightleftarrows}} v = R\begin{bmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{bmatrix}. \tag{3}$$

With this notation, the standard metric for $SO(3)$, with $v_1, v_2 \in T_RSO(3)$, is given by

$$\langle v_1, v_2\rangle = (v_1^\vee)^T v_2^\vee. \tag{4}$$

The exponential and logarithm maps for $SO(3)$ can be computed in closed form by using the Rodrigues' formula [10].

We denote as $R_x(\theta)$, $R_y(\theta)$, $R_z(\theta)$, the rotations around the x, y and z axes, respectively, with angle $\theta \in [-\pi, \pi)$, and as e_z the unit vector aligned with the z axis. We denote as $I \in \mathbb{R}^{3\times3}$ the identity matrix and as $P_z = \mathrm{diag}(1, 1, 0)$ the standard projector on the xy-plane. We denote as $(A)_{ij}$ the element in row i, column j of the matrix A. With some abuse of notation, for standard vectors $a \in \mathbb{R}^3$, $a^\wedge$ is given by (1) with $R = I$ and is the matrix representation of the cross product operator, *i.e.*, $\hat a b = a \times b$ for all $a, b \in \mathbb{R}^3$.

3. Derivation of the essential matrix

As customary, we model the pose of camera i as $g_i = (R_i', T_i') \in SE(3)$, where g_i represents the transformation from camera to world coordinates. Given an image x_i in homogeneous coordinates and the corresponding depth λ_i, the 3-D point in world coordinates is given by

$$X = \lambda_i R_i' x_i + T_i'. \tag{5}$$

Note that a change of world coordinates represented by $g = (R_0, T_0)$, *i.e.*, $X \mapsto R_0X + T_0$ induces a transformation of the camera representation equivalent to multiplying g_i by g on the left, *i.e.*, $(R_i', T_i') \mapsto (R_0R_i', R_0T_i' + T_0)$.

We now derive the essential matrix from two camera poses (R_i', T_i') and the two images x_i, $i = 1, 2$, of a same 3-D point X. We follow a general approach [1] as opposed to the traditional one which uses the first camera as global reference frame. From (3), and using the cross product's properties $\hat a a = 0$ and $b^T\hat a b = 0$ for all $a, b \in \mathbb{R}^3$, we have:

$$\lambda_1R_1'x_1 + T_1' = \lambda_2R_2'x_2 + T_2' \tag{6}$$

$$\lambda_1R_1'x_1 = \lambda_2R_2'x_2 + (T_2' - T_1') \tag{7}$$

$$\lambda_1(T_2' - T_1')^\wedge R_1'x_1 = \lambda_2(T_2' - T_1')^\wedge R_2'x_2 \tag{8}$$

$$x_1^T {R_1'}^T(T_2' - T_1')^\wedge R_2'x_2 = 0 \tag{9}$$

The essential matrix is then defined as

$$E = {R_1'}^T(T_2' - T_1')^\wedge R_2'. \tag{10}$$

4. The normalized essential space

In this section, we define a canonical decomposition of the essential matrix in terms of two rotations by choosing a global reference frame aligned with the baseline between the two cameras. Then, we define the *normalized essential space*, and analyze its structure as a quotient space (which includes the twisted pair ambiguity). We also give an interpretation in terms of vector transformations.

4.1. The normalized canonical decomposition

Since (7) is a homogeneous equation, we cannot determine the scale of E from image data alone. Also, while E does not depend on the choice of global reference frame, this is not true for its decomposition (8). To remove most of the degrees of freedom, we use the following.

Proposition 4.1. *Any essential matrix E admits, up to scale, the following* normalized canonical decomposition*:*

$$E = R_1^T \hat{e}_z R_2. \tag{11}$$

Proof. Starting from (8), choose a global scale such that $\|T_2' - T_1'\| = 1$ and let $R_0 \in SO(3)$ be such that $R_0(T_2' - T_1') = e_3$. There are infinite candidates for such rotation (we pick one using Householder transformations). Then, by applying the transformation $g_0 = (R_0, 0)$ and using the property $R\hat{a}R^T = (R\hat{a})^\wedge$ for all $R \in SO(3)$, we can write

$$E = (R_0 R_1')^T \big(R_0(T_2' - T_1')\big)^\wedge R_0 R_2' = (R_0 R_1')^T \hat{e}_z (R_0 R_2'), \tag{12}$$

which is of the form (9). □

Intuitively, the change of world coordinates performed in the proof above aligns the vector $T_2' - T_1'$ with the z-axis. In this way, the translation direction is known, and we are left with only the information about the two rotations.

Remark 1. *Notice that $\hat{e}_z R_z\left(\frac{\pi}{2}\right) = P_z = \mathrm{diag}(1, 1, 0)$. Hence, $E = R_1^T P_z \left(R_z\left(\frac{\pi}{2}\right) R_2\right)$ is a valid SVD of E.*

The value of Remark 1 is twofold. First, it provides a practical way to compute the decomposition (9). Second, it relates our representation with the one of [15], giving a geometric meaning to the SVD of E.

We define the *normalized essential space* $\mathcal{M}_E$ as the image of $SO(3)^2$ under the map (9). Since, according to Proposition 4.1, this map is surjective, $\mathcal{M}_E$ corresponds to the space of all the essential matrices.

4.2. Ambiguities of the canonical form

While the map (9) is surjective, it cannot be also injective, because it is known that the space of essential matrices is five-dimensional, while $SO(3)^2$ is six-dimensional. In fact, the extra degree of freedom corresponds the choice of a rotation R_0 in the proof of Proposition 4.1, *i.e.*, to a rotation of the global reference frame around the baseline between the two cameras. However, it turns out that this is not the only ambiguity. To be more precise, consider any two points $Q_a, Q_b \in SO(3)^2$ which, through (9), correspond to the essential matrices E_a, E_b. We define an equivalence relation "$\sim$" between points in $SO(3)^2$ as

$$Q_a \sim Q_b \iff E_a = E_b, \tag{13}$$

where, again, equality is intended up to scale (since E_a and E_b are normalized, this reduces to a "up to a sign flip").

Proposition 4.2. *Define the groups*

$$H_z = \{(R_z(\theta), R_z(\theta)) : \theta \in [-\pi, \pi)\}, \tag{14}$$

$$H_\pi = \big\{(I, I), \big(R_x(\pi), R_x(\pi)\big), \big(I, R_z(\pi)\big), \big(R_x(\pi), R_y(\pi)\big)\big\} \tag{15}$$

acting on the left on $SO(3)^2$ by simple component-wise left multiplication. Then, the equivalence class of a point Q with respect to "$\sim$" is exactly given by

$$[Q] = \{S_z S_\pi Q : S_z \in H_z, S_\pi \in H_\pi\}. \tag{16}$$

The proof involves first showing that H_z and H_π are subgroup of $SO(3)^2$, and then showing that the only matrices satisfying (14) are those in the equivalence class $[Q]$. The details can be found in the additional material. In the following we will use $S_z = (S_{z1}, S_{z2})$ and $S_\pi = (S_{\pi 1}, S_{\pi 2})$ to denote points in H_z and in H_π, respectively.

Intuitively, $[Q]$ has four components, each one isomorphic to $SO(2)$. In view of Proposition 4.2, the space $\mathcal{M}_E$ can be identified with the quotient space

$$\mathcal{M}_E = \big(SO(3) \times SO(3)\big)/(H_z \times H_\pi), \tag{17}$$

where the actions of H_z and H_π are defined above.

Since $SO(3)^2$ has dimension six, and H_z has dimension one, we get the well known fact that the normalized essential space has dimension five (being discrete, H_π does not change the intrinsic dimension of the space).

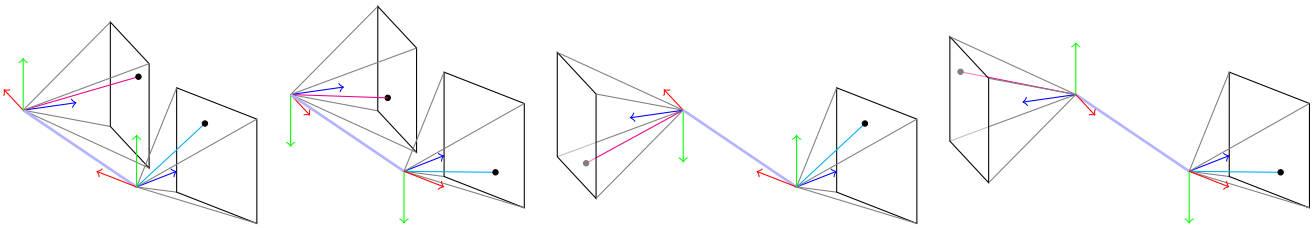
4.3. Geometric interpretation

From the proof of Proposition 4.1, we know that (R_1, R_2) can be interpreted geometrically as the rotations of the cameras after an appropriate change of coordinates. We now show that also the epipolar constraint $x_1^T E x_2 = 0$ has a geometrical interpretation. Given an essential matrix E with decomposition $E = R_1^T \hat{e}_z R_2$, from Proposition 4.2 and the equivalence $\hat{e}_z = P_z^T R_z\left(\frac{\pi}{2}\right) P_z$, we have

$$x_1^T E x_2 = (P_z S_z S_{\pi 1} R_1 x_1)^T R_z\left(\frac{\pi}{2}\right) (P_z S_z S_{\pi 2} R_2 x_2) = 0. \tag{18}$$

This can be interpreted as the following procedure:

- Take the images x_i and rotate them as $R_i x_i$, $i = 1, 2$. This is equivalent to expressing in global coordinates the vectors corresponding to the images and centering them at the origin. Notice that, by construction, the transformed vectors and the z-axis e_z all lie in the same plane passing through the origin.
- Apply the action of an element of $S_\pi = H_\pi$ (see Figure 1). If $S_\pi = (I, I)$, no changes are made. If $S_\pi = (R_x(\pi), R_x(\pi))$ (and considering also H_z), the direction of the baseline is reversed. If $S_\pi = (I, R_z(\pi))$, one of the cameras is rotated from front-facing to rear-facing. Finally, if $S_\pi = (R_x(\pi), R_y(\pi))$, the last two



(a) $S_\pi = (I, I)$ (b) $S_\pi = (R_x(\pi), R_x(\pi))$ (c) $S_\pi = (I, R_z(\pi))$ (d) $S_\pi = (R_x(\pi), R_y(\pi))$

Figure 1: Diagram depicting the geometric twisted pair ambiguity given by the four elements of H_π

cases are combined. Note that the coplanarity condition of the transformed vectors with e_z is preserved.

- Apply the action of an element of H_z, *i.e.*, rotate the two vectors around the z-axis by an arbitrary amount. This is equivalent to a rotation around the baseline, and does not change the coplanarity condition.
- Project the transformed vectors onto the xy-plane. In practice, this sets the last coordinate to zero. Since before the projection the vectors belonged to the same plane, after the projection they will have the same direction (but, in general, different lengths).
- Rotate one of the projected vectors by $R_z\left(\frac{\pi}{2}\right)$, *e.g.*, $R_z\left(\frac{\pi}{2}\right)(P_z R_2 x_2)$ and take the inner product. Since the vectors were collinear, they are now orthogonal and the result is zero, as in (16).

Although this interpretation of the epipolar constraint is probably not new, in our context it shows that the action of H_π corresponds exactly to the well-known twisted-pair ambiguity in the decomposition of the essential matrix.

5. The signed normalized essential manifold

In this section we review how the cheirality constraint can be used to resolve the twisted pair ambiguity (*i.e.*, to choose an element of the group H_π), and how this simplifies the quotient structure of the normalized essential space into what we call the *signed normalized essential space*. We show that this space is a manifold, and that a metric and the corresponding geodesics can be naturally induced from $SO(3)^2$. Finally, we give a Newton-based algorithm for computing the logarithm map and the Riemannian distance.

5.1. Depth triangulation

Since the canonical form has a simple geometrical interpretation, it can be used to estimates the depths of the 3-D points from their image projections. This, in turn, can be used to enforce the cheirality constraint, *i.e.*, the fact that all the 3-D points need to be in front of both cameras.

From the discussion in Section 4.2, the baseline $T'_2 - T'_1$ in the canonical form is equal to e_z. Therefore, taking into account the actions of H_z and H_π, and assuming noiseless image points, (5) becomes

$$\lambda_1 S_{z1} S_{\pi 1} R_1 x_1 = \lambda_2 S_{z1} S_{\pi 2} R_2 x_2 + e_z. \tag{19}$$

Note that $e_z = S_{z1} e_z = S_{z2} e_z$, hence we can cancel S_z from (17). We then have the following:

Proposition 5.1. *There is only one choice of S_π for which the solution of (17) is positive,* i.e., $\lambda_1, \lambda_2 > 0$.

The proof, which can be found in the additional material, is similar to the one used to solve the twisted pair ambiguity, *i.e.*, to decide, given an essential matrix, which of the four possible pose estimate to use [10].

5.2. The signed normalized essential manifold

In our context, Proposition 5.1 allows us to pick one of four components in the equivalent class $[Q]$, *i.e.*, we can dispense with the group H_π and consider a new quotient space using H_z alone, which we call the signed essential space. Just for fun, we use the symbol $\mathcal{M}_\mathcal{I}$ (because it differs from "$\mathcal{M}_E$" by a 180 degrees rotation). Formally, we have

$$\mathcal{M}_\mathcal{I} = \big(SO(3) \times SO(3)\big)/H_z. \tag{20}$$

In general, a quotient space of a Riemannian manifold is not a Riemannian manifold (because its topology might fail to satisfy some necessary conditions or there might not be an obvious way to choose the metric). However, in our case the action of H_z satisfies some "nice" properties, which lead to the following.

Proposition 5.2. *The space $\mathcal{M}_\mathcal{I}$ can be given a Riemannian manifold structure for which the natural projection $\pi_{\mathcal{M}_\mathcal{I}} : SO(3)^2 \to \mathcal{M}_\mathcal{I}$ introduces a local isometry between a subspace of $T_Q SO(3)^2$ and $T_{[Q]}\mathcal{M}_\mathcal{I}$.*

This proposition (for which the proof is somewhat technical, see the additional material) means that not only we can endow $\mathcal{M}_\mathcal{I}$ with a Riemannian structure, but also that the differential (*i.e.*, the Jacobian, in local coordinates) of the natural projection $\pi_{\mathcal{M}_\mathcal{I}}$ induces an orthogonal decomposition of the tangent space $T_Q SO(3)^2$ as:

$$T_Q SO(3)^2 = T_{VQ} SO(3)^2 \oplus T_{HQ} SO(3)^2, \tag{21}$$

where the *vertical space* $T_{VQ}SO(3)^2$ is the subspace of tangent vectors tangential to the equivalence class $[Q]$ and the *horizontal space* $T_{HQ}SO(3)^2$ is its orthogonal complement. Moreover, the same differential uniquely maps each vector $\tilde{v} \in T_{[Q]}\mathcal{M}_{\mathcal{I}}$ to a vector $v \in T_{HQ}SO(3)^2$, called the *horizontal lift* of $\tilde{v}$. The metric $\langle , \rangle_{[Q]}$ for $\mathcal{M}_{\mathcal{I}}$ is then defined from the metric $\langle , \rangle_Q$ in $SO(3)^2$ as

$$\langle \tilde{u}, \tilde{v} \rangle_{[Q]} = \langle u, v \rangle_Q \tag{22}$$

In our case, the vertical space is one-dimensional, and $T_{VQ}SO(3)^2 = \operatorname{span}(v_V)$, where

$$v_V = (\hat{e}_z R_1, \hat{e}_z R_2) = \big(R_1 (R_1^T e_z)^\wedge, R_2 (R_2^T e_z)^\wedge\big). \tag{23}$$

By definition, then, the horizontal space at Q includes all vectors v_H such that $v_H \perp v_V$, *i.e.*:

$$0 = \langle v_V, v_H \rangle = e_z^T (R_1 v_{H1} + R_2 v_{H2}). \tag{24}$$

where we used the notation

$$(v_H)^\vee = \begin{bmatrix} v_{H1} \\ v_{H2} \end{bmatrix}, \tag{25}$$

We can take this as the condition defining horizontal vectors at Q. Given a vector $v \in T_Q SO(3)^2$, let

$$p_Q(v) = e_z^T (R_1 v_1 + R_2 v_2). \tag{26}$$

We define the projection of v onto $T_{HQ}SO(3)^2$ as

$$\Pi_H(v) = v - \frac{p_Q(v)}{2} \begin{bmatrix} R_1^T e_z \\ R_2^T e_z \end{bmatrix}. \tag{27}$$

5.3. Geodesics and the exponential map

The goal of this section is to show that the natural projection of geodesics in $SO(3)^2$ are geodesics in $\mathcal{M}_{\mathcal{I}}$. The key insight we will use is the following (see the additional material for a proof)

Proposition 5.3. *Let $Q(t) : \mathbb{R} \to SO(3)^2$ be a geodesic curve such that $\dot{Q}(t) \in T_{HQ}SO(3)$ for all t. Then, $\tilde{Q} = \pi_{\mathcal{M}_{\mathcal{I}}}(Q)$ is a geodesic curve in $\mathcal{M}_{\mathcal{I}}$.*

This result tells us that to find geodesics in $\mathcal{M}_{\mathcal{I}}$, we can focus on finding geodesics in $SO(3)^2$ for which the tangent vector is always horizontal. This idea is repeatedly used in [3] to give expressions for the geodesics in the Stiefel and Grassmann manifolds. Here we now show that if a geodesic $Q(t) \in SO(3)^2$ has a horizontal initial tangent vector $\dot{Q}(0)$, then the tangent is horizontal for every t.

Proposition 5.4. *Let $V \in TSO(3)^2$ be a vector field of the form*

$$W(t)^\vee = \begin{bmatrix} R_1(t)^T e_z \\ R_2(t)^T e_z \end{bmatrix} \tag{28}$$

defined along a geodesic $Q(t) \in SO(3)^2$. Then we have

$$\langle \dot{Q}(t), W(t) \rangle = \langle \dot{Q}(0), W(0) \rangle. \tag{29}$$

Proof. Denote the tangent to the geodesic $Q(t)$ as

$$\dot{Q}(t)^\vee = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}, \tag{30}$$

and let

$$m(t) = \langle \dot{Q}(t), W(t) \rangle = v_1^T R_1^T e_z + v_2^T R_2^T e_z. \tag{31}$$

Taking the derivative we have

$$\dot{m}(t) = v_1^T \hat{v}_1^T R_1^T w_1 + v_2^T \hat{v}_2^T R_2^T w_2 \equiv 0. \tag{32}$$

Since the first derivative of $m(t)$ is identically zero, $m(t)$ must be constant, which implies (28). □

Combining Propositions 5.3 and 5.4, we get that the exponential map in $\mathcal{M}_{\mathcal{I}}$, *i.e.*,

$$[Q_b] = \exp_{[Q_a]}(v_a),\ [Q_a] \in \mathcal{M}_{\mathcal{I}},\ v_a \in T_{[Q_a]}\mathcal{M}_{\mathcal{I}}, \tag{33}$$

is obtained by computing

$$Q_b = \exp_{Q_a}(\tilde{v}_a),\ Q_a \in SO(3)^2 \tag{34}$$

where $\tilde{v}_a$ is the horizontal lift of v_a.

5.4. The logarithm map

Let $Q_a = (R_{a1}, R_{a2})$ and $Q_b = (R_{b1}, R_{b2})$ be two points in $SO(3)^2$. We would like to find the distance between $[Q_a]$ and $[Q_b]$ and the logarithm map $\log_{[Q_a]}[Q_b]$. In general, we cannot directly use the distance and logarithm map in $SO(3)^2$, because the tangent of the corresponding geodesic is not horizontal. However, we can "move" Q_b to another representative of the equivalence class $[Q_b]$, so that the geodesic between Q_a and Q_b corresponds to a geodesic between $[Q_a]$ and $[Q_b]$. This is formalized in the following:

Proposition 5.5. *Define the cost*

$$f(t) = \sum_{i=1,2} f_i, \quad f_i = \frac{1}{2}\theta_i^2(t), \quad \theta_i(t) = d(R_{ai}, R_z(t) R_{bi}), \tag{35}$$

and let $t_{opt} = \operatorname{argmin}_t f(t)$. Then, the logarithm

$$\log_{Q_a}\big(S_z(t_{opt}) Q_b\big) = \operatorname{stack}\big(\{\operatorname{Log}(R_{ai}^T R_z(t_{opt}) R_{bi})\}\big)_{i=1,2} \tag{36}$$

is an horizontal vector in $T_{HQ}SO(3)^2$.

Proof. We will need the following result

$$\frac{\mathrm{d}}{\mathrm{d}t} R_z(t) R_{bi} = \hat{e}_z R_{bi} = R_{bi} (R_{bi}^T e_z)^\wedge, \quad i = 1, 2 \tag{37}$$

Taking the derivative of each term f_i we have

$$\begin{aligned} \dot{f}_i(t) &= -\langle \log_{R_{ai}}(R_z(t) R_{bi}), R_{bi} (R_{bi}^T e_z)^\wedge \rangle \\ &= -\operatorname{Log}(R_{ai}^T R_z(t) R_{bi})^T R_{bi}^T e_z \\ &= -\operatorname{Log}(R_{ai}^T R_z(t) R_{bi})^T R_{ai}^T R_z(t) R_{bi} R_{bi}^T e_z \\ &= -e_z^T R_{ai} \operatorname{Log}(R_{ai}^T R_z(t) R_{bi}), \end{aligned} \tag{38}$$

where we used the fact that $R^T \operatorname{Log}(R) = \operatorname{Log}(R)$ and, similarly, $R_z(t)e_z = e_z$. For $t = t_{\text{opt}}$ we have $\dot{f}_1(t_{\text{opt}}) + \dot{f}_2(t_{\text{opt}}) = 0$, which, together with (22), implies that the vector is in the horizontal space at Q_a. □

The problem now is to find t_{opt}, the minimum of f. In general, this is a nonlinear optimization problem with multiple local minima (see Figure 2 for an example). However, we can exploit its special structure to reliably and efficiently find the global minimizer t_{opt}. First, consider each function f_i separately. The derivative of f_i can be computed as

$$\dot{f}_i(t) = e_z^T R_{ai} \operatorname{Log}(R_{ai}^T R_z(t) R_{bi}) = \theta_i(t) e_z^T R_{ai} u_i \quad (39)$$

where (using the closed form expression of Log from [10])

$$u_i = \frac{1}{2\sin\theta_i(t)} \left((R_{ai}^T R_z(t) R_{bi}) - (R_{ai}^T R_z(t) R_{bi})^T \right)^\vee, \quad (40)$$

Notice that the derivative of f exists everywhere except at a point t_{di} for which $\sin\big(\theta_i(t_{di})\big) = 0$. The following proposition gives a way to compute the location of this point.

Proposition 5.6. *Let θ_i be defined as in (34), and define*

$$c_{1i} = (R_{bi} R_{ai}^T)_{1,1} + (R_{bi} R_{ai}^T)_{2,2} \quad (41)$$

$$c_{2i} = (R_{bi} R_{ai}^T)_{1,2} - (R_{bi} R_{ai}^T)_{2,1} \quad (42)$$

$$\phi_i = \operatorname{arctan}_2(c_{1i}, c_{2i}). \quad (43)$$

Then, the function $\theta_i(t)$ is continuous, 2π-periodic and

$$\sin(\theta_i(t_{di})) = 0 \text{ for } t_{di} = \frac{3}{2}\pi - \phi_i. \quad (44)$$

Using the definition of DLog and its closed-form expression from [17], the second derivative of f_i is given by:

$$\begin{aligned} \ddot{f}_i(t) &= e_z^T R_{ai} \operatorname{DLog}(R_{ai}^T R_z(t) R_{bi}) R_{ai}^T e_z \\ &= (e_z^T R_{ai} u_i)^2 + \frac{\theta}{2} \cot\left(\frac{\theta}{2}\right)(1 - (e_z^T R_{ai} u_i)^2) \end{aligned} \quad (45)$$

Note that (as a simple plot can confirm)

$$0 \le \frac{\theta}{2} \cot\left(\frac{\theta}{2}\right) \le 1 \quad \text{for } \theta \in [-\pi, \pi]. \quad (46)$$

This implies that $\ddot{f} \ge 0$, and that f is convex between discontinuity points.

In summary, from the results above, we have that he function f is continuous, 2π-periodic and with positive second derivative except at $\{t_{di} + 2k\pi\}$, $k \in \mathbb{Z}$. Assuming (without loss of generality) that $-\frac{\pi}{2} \le t_{d1} \le t_{d2} \le \frac{\pi}{2}$, this suggests an algorithm to find all the global minimizers of f which considering separately the two intervals $[t_{d1}, t_{d2}]$ and $[t_{d2}, t_{d1} + 2\pi]$ (on which the function is convex and differentiable). Since we have a closed form expression for $\ddot{f}$, we can use Newton's method (with the additional projection of the iterates to the interval). In addition, one can easily show (using the intermediate value's theorem on $\dot{f}$) that if $\dot{f}$ has the same sign at the two extremum points of an interval, then that interval does not contain a local minimizer, and it can be skipped. These steps are summarized in Algorithm 1 (see also Figure 2). We use the notation $\dot{f}^+$ and $\dot{f}^-$ to denote the right and left derivatives, respectively. Note that Algorithm 1 is only a basic version. A complete version would also consider degenerate cases, where $m_i = 0$ for some $i \in \{1, 2\}$ or where $t_{d1} = t_{d2}$. In our experiments, we saw that an interval could be skipped about 25% of the time, and that the Newton's iteration took about 5 to 8 iterations to converge to the machine's precision of $2 \cdot 10^{-16}$ (as a comparison, the method suggested in [15] only achieves a precision of 10^{-4} after about 5 iterations).

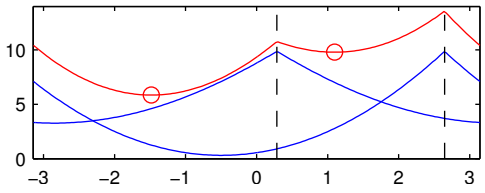


Figure 2: An example realization of the cost $f(t)$ from (34). Blue and red lines: value of each term f_i and of f, respectively. Black dashed line: location of the discontinuity points $\{t_{di}\}$ computed using Proposition 5.6. Red circles: local minimizers $\{t_{\text{opt},i}\}$ computed in Algorithm 1.

Algorithm 1 Global minimization of $f(t)$

1: Compute the points t_{di}, $i = 1, 2$ (assume $t_{d1} < t_{d2}$).
2: Define intervals $S_1 = [t_{d1}, t_{d2}]$ and $S_2 = [t_{d2}, t_{d1} + 2\pi]$.
3: **for** $i \in 1, 2$ **do**
4: **if** $\operatorname{sgn}\big(\dot{f}^+\big(\min(S_i)\big)\big) \neq \operatorname{sgn}\big(\dot{f}^-\big(\max(S_i)\big)\big)$ **then**
5: Compute $t_{\text{opt},i} = \operatorname{argmin}_{t \in S_i} f(t)$ using the projected Newton's method.
6: **end if**
7: **end for**
8: Select t_{opt} as the point $t_{\text{opt},i}$ for which f is minimum.

5.5. Comparison with previous formulations

Among the papers that use the relative pose between cameras to parametrize the essential space, the definition of normalized essential space used in [11] is compatible with $\mathcal{M}_E$, while the definition used in [13] (which includes the cheirality constraints explicitly) is compatible $\mathcal{M}_{\mathcal{I}}$. For the papers using the parametrization derived from the SVD [**?**, 8, 15, 16], the definition used is the same as $\mathcal{M}_{\mathcal{I}}$. However, these papers (mistakenly) do not consider the action of the group H_π and the cheirality constraint. In particular, Proposition 4.2

shows that the claim made in [15] that an essential matrix E corresponds uniquely to a point in $\mathcal{M}_{\mathcal{I}}$ is false.

6. The Weiszfeld algorithm and pose averaging

While the main focus of this paper is on the theoretical analysis of the essential manifold, in this section we show that the resulting Riemannian distance can be naturally and meaningfully used in a proof-of-concept application to the two-view structure from motion problem.

In a standard pipeline, the relative pose (R, T) between two calibrated views is computed using RANSAC (see [7]):

- Extract pairs of matching image points $\{x_1^i, x_2^j\} \in \mathbb{R}^2$.
- For $i \in \{1, \dots, N\}$, select a random subset S_i of point pairs $\{x_1^i, x_2^j\}_{j \in S_i}$, estimate the essential matrix E_i and compute its support (*i.e.*, the number of points which approximatively satisfy the epipolar constraint).
- Compute the pose (R, T) from the matrix E_i with the largest support.

In [5] an alternative approach is suggested, where instead of using RANSAC, each sample E_i is decomposed into a pose estimate (R_i, T_i) and then all the rotations $\{R_i\}$ are averaged. For this purpose, they propose to minimize the cost

$$\varphi(R) = \sum_i d(R, R_i)^p, \tag{47}$$

where $p = 1$ (L1 averaging) or $p = 2$ (L2 averaging), by using the Weiszfeld algorithm, which we report in Algorithm 2 for points lying in a general Riemannian manifold $\mathcal{M}$. Strictly speaking, the traditional Weiszfeld algorithm refers only to the version $p = 1$, but here we give a generalized version for ease of exposition. The set I in (48) is used to take into account the fact that w_i becomes ill-defined when $p = 1$ and the iterate x falls on one of the input points. Intuitively, each iteration of the algorithm maps the input points to the tangent space of the current iterate $x(t)$, take the average (with weights given by the relative distances) and use the resulting vector to obtain the next iterate $x(t+1)$.

Algorithm 2 The Weiszfeld algorithm

Input: Points $x_i \in \mathcal{M}$, $i \in \{1, \dots, N\}$.

1: Initialize $x(0)$

2: **for** $t \in \{0, \dots, N_t\}$ **do**

3: Update x using:

$$w_i(t) = d\big(x(t), x_i\big)^{p-2} \tag{48}$$

$$I(t) = \big\{i \in \{1, \dots, N\} : x(t) \neq x_i\big\} \tag{49}$$

$$x(t+1) = \exp_x\left(\frac{\sum_{i \in I} w_i(t) \log_x(x_i)}{\sum_{i \in I} w_i(t)}\right) \tag{50}$$

4: **end for**

In this section, we follow the same approach proposed by [5], but we average essential matrices instead of rotations. In practice, the only difference is the use of the definition of exp, log and Riemannian distance for $\mathcal{M}_{\mathcal{I}}$ in Algorithm 2. Note that the approach proposed here has the immediate advantage of naturally considering both rotation and translation components together, while the approach of [5] considers only rotations. We compare the two approaches against standard RANSAC on the `fountain-P11` dataset from [14], which includes the ground-truth pose for the cameras.

We used SIFT features extraction and matching [18] to find corresponding points between every possible pair of cameras. We excluded image pairs with less than thirty good matches (as determined using the essential matrix from the ground truth pose). We then use the five point algorithm [6] to generate the RANSAC samples E_i. We compare four versions of the Weiszfeld algorithm corresponding to the four possible combinations of $p = 1, 2$ and $\mathcal{M} = \mathcal{M}_{\mathcal{I}}, SO(3)$ by using between 1 and 50 RANSAC samples. To initialize the algorithm, we evaluate the cost at every input sample, and use the half-way point between the two samples with lower costs. Also, we set the number of iterations N_t to 30 (although, in our preliminary tests, the algorithms usually converged in less than 15 iterations). As baseline, we use the errors of the RANSAC solution after the same number of samples and after 2000 samples. As the error measure, we consider the geodesic distance between estimated and ground-truth rotations. For our approach and the RANSAC-based solutions, we also consider the angle between the estimated translation direction and the ground truth. All the results are averaged across all the image pairs and 30 independent sampling realizations.

We report the results in Figure 3. As one can see, the Weiszfeld algorithm using the proposed distance on $\mathcal{M}_{\mathcal{I}}$ outperforms the corresponding version using the distance on $SO(3)$ for both $p = 1$ and $p = 2$. Moreover, the use of the cost with $p = 1$ produces better results than those using $p = 2$, likely due to the fact that the first cost is more robust to outliers in the samples. This dataset also shows that, while our approach (which does not require setting a threshold) gives reasonably good results, the efficiency of RANSAC with a well-tuned threshold is quite hard to beat.

7. Conclusion

In this paper we considered a Riemannian structure for the essential manifold, and introduced a novel, geometrical interpretation which shed light on the limitations of previous approaches and on the connections with traditional concepts in computer vision. We also proposed efficient algorithms for computing the distance and logarithm map, and considered an application to the problem of two-view

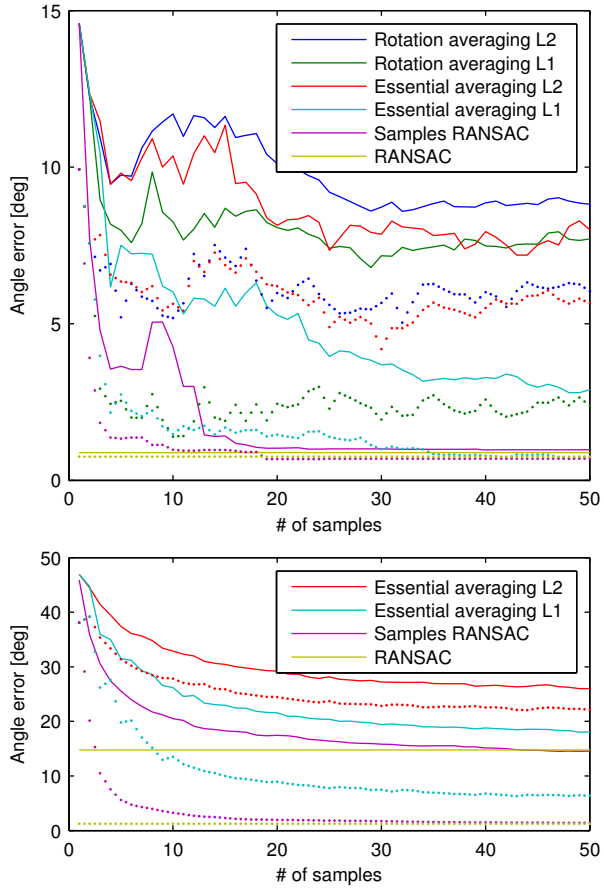


Figure 3: Results for two-view pose estimation. Rotation (top) and translation angle errors (bottom) for the different methods on the `fountain-P11` dataset. Solid and dotted lines represent the mean and median errors, respectively.

pose estimation using averages. In our future work we will investigate relations between three views, and determine if similar ideas can be applied to the space of trifocal tensors and other similar objects.

References

[1] M. Arie-Nachimson, S. 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

[2] M. P. do Carmo. *Riemannian geometry*. Birkhäuser, Boston, MA, 1992. 1

[3] A. Edelman, T. A. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353, 1998. 5

[4] C. Geyer, S. Sastry, and R. Bajcsy. Euclid meets fourier: Applying harmonic analysis to essential matrix estimation in omnidirectional cameras. In *Workshop on omnidirectional vision, camera networks and non-classical cameras*, volume 2, page 3, 2004. 1, 6

[5] R. Hartley, K. Aftab, and J. Trumpf. L1 rotation averaging using the Weiszfeld algorithm. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 7

[6] 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. 7

[7] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. 1, 7

[8] 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, 6

[9] H. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*, pages 61–62, 1987. 1

[10] Y. Ma. *An invitation to 3-D vision: from images to geometric models*, volume 26. Springer, 2004. 1, 2, 4, 6

[11] Y. Ma, J. Košecká, and S. Sastry. Optimization criteria and geometric algorithms for motion and structure estimation. *International Journal of Computer Vision*, 44(3):219–249, 2001. 1, 6

[12] B. O'Neill. The fundamental equations of a submersion. *The Michigan Mathematical Journal*, 13(4):459–469, 1966. 5

[13] S. Soatto, R. Frezza, and P. Perona. Motion estimation via dynamic vision. *IEEE Transactions on Automatic Control*, 41(3):393–413, 1996. 1, 6

[14] C. Strecha, W. von Hansen, L. V. Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 7

[15] R. Subbarao, Y. Genc, and P. Meer. Robust unambiguous parametrization of the essential manifold. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. 1, 3, 6, 7

[16] R. Subbarao and P. Meer. Nonlinear mean shift over riemannian manifolds. *International Journal of Computer Vision*, 84(1):1–20, 2009. 1, 6

[17] R. Tron. *Distributed optimization on manifolds for consensus algorithms and camera network localization*. PhD thesis, The Johns Hopkins University, 2012. 6

[18] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008. 7