Calibration of Extrinsic Transformation Using Manifold Optimization

Hu, Xiao; Olesen, Daniel; Knudsen, Per

Published in:
I F A C Workshop Series

Link to article, DOI:
10.1016/j.ifacol.2019.08.059

Publication date:
2019

Document Version
Publisher's PDF, also known as Version of record

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 52-8 (2019) 124–129

Calibration of Extrinsic Transformation Using Manifold Optimization ⋆

Xiao Hu * **Daniel Olesen** * **Knudsen Per** *

* *National Space Institute, Technical University of Denmark, Elektrovej, 2800, Kgs. Lyngby, Denmark (e-mail: xiahaa@space.dtu.dk, danole@space.dtu.dk, pk@space.dtu.dk)*

Abstract: Data fusion with multiple heterogeneous sensors has shown great importance for motion control and navigation filter design of autonomous vehicles. However, data fusion often requires prior knowledge of the extrinsic transformations between sensors. This paper focuses on the usage of manifold optimization to calibrate the extrinsic transformation for a pair of sensors from a batch of measurements. Instead of reparameterization the transformation matrix in other forms, we formulate an objective function directly with the special Euclidean group. Then this manifold optimization problem is solved iteratively in a Gauss-Newton fashion. The usage of manifold optimization guarantees the obtained result being strictly within the special Euclidean group without the need for further operations like normalization and orthogonalization. The experimental results on both synthetic and real data show the superiority and robustness of the proposed method. Considering the performance and time consumption, the proposed approach is a good option for real applications.

Keywords: Calibration, Manifold Optimisation, Sensor Fusion, Multisensor Integration

1. INTRODUCTION

Thanks to the developments of sensor technology, data fusion with heterogeneous sensors have witnessed increasing popularity and shown great importance for motion control and navigation filter design of autonomous vehicles. For the purpose of efficiently fusing data from multiple sensors, prior extrinsic calibration is crucial in order to determine the rigid transformation between a pair of sensors. In general, the extrinsic relationship between two rigidly mounted sensors, shown in Fig. 1, can be modeled as $\mathbf{AX} = \mathbf{XB}$ where $\mathbf{A}$, $\mathbf{B}$, $\mathbf{X}$ are homogeneous matrices in $\mathbb{SE}(3)$. Closely related to the hand-eye calibration problem, the extrinsic calibration problem aims to estimate a $\mathbf{X}$ from multiple ego-motion pairs of A_i, B_i, $i = 1, 2, \cdots, n$. Previous work by Tsai and Lenz (1989) has proved that a minimum number of two motion pairs with nonparallel rotation axes are necessary to solve this problem. In practice, a batch of measurements ($n > 2$) is favored to handle the influence of noise by formulating the problem in an overdetermined fashion.

In this paper, we present a novel calibration algorithm using manifold optimization for quickly determining the extrinsic transformation between a pair of sensors by using a batch of egomotion data. Unlike previous methods which reparameterize the transformation matrix in other forms, the objective function is established directly with the special Euclidean group, which forms an unconstrained manifold optimization problem. This manifold optimization problem is then solved iteratively in a Gauss-Newton fashion. Furthermore, the analytical solution of the Jacobian matrix is explicitly derived in this paper. The usage of manifold optimization guarantees the obtained result being strictly within the special Euclidean group, so no more operations like normalization or orthogonalization are required. The proposed method, which is able to the rotational and translational parts simultaneously, provides a feasible and accurate estimation of the extrinsic transformation matrix. Evaluations on synthetic data and real datasets have been carried out to validate the performance of the proposed method. The experimental results show the superiority and robustness of the proposed method against another five state-of-the-art counterparts. The

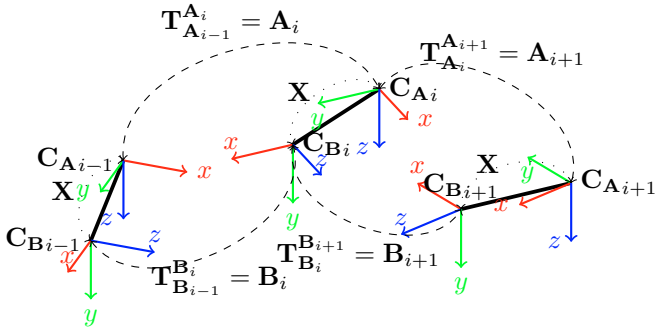

Fig. 1. Illustration diagram of extrinsic calibration problem: $\mathbf{C_A}$ represents the frame of sensor A, $\mathbf{C_B}$ defines the coordinate frame of B, A and B are rigidly mounted and the corresponding transformation matrix is denoted with $\mathbf{X}$. Dashed connections denote transformation matrices directly measured, while dotted links denote transformation matrices to be estimated.

⋆ This work was financially supported by the Innovation Fund Denmark through the project Unmanned Aerial Vehicle for high-Quality Magnetic Surveying (UAV-QMS).

10.1016/j.ifacol.2019.08.059

main contributions of our work are: 1) a novel formulation of the calibration problem directly by the special Euclidean group; 2) explicit derivation of the Jacobian matrix which makes the Gauss-Newton iteration fast and feasible; 3) extensive comparisons of state-of-the-art hand-eye calibration algorithms by numerical simulations and experiments on real datasets. The remaining of the paper is organized as follows: Section 2 presents related work. Section 3 introduces notations and mathematical preliminaries used in this paper. Section 4 explains the details of the proposed approach. Section 5 describes the experiments and results. Finally, the conclusion is drawn in Section 6.

2. RELATED WORK

The hand-eye calibration problem was first studied by Shiu and Ahmad (1989), followed by a considerable number of works on hand-eye calibration so far and new ones are still appearing continuously. In this section, we will review a few hand-eye calibration algorithms.

Traditional methods solve the hand-eye calibration in a decoupled fashion, i.e. firstly the rotation estimation and followed by the estimation of the translational part. The main difference among those traditional methods is the parameterization of the rotational part: 1) using rotation axis and angle by Tsai and Lenz (1989) Shiu and Ahmad (1989); 2) using unit quaternion by Chou and Kamel (1991) Horaud and Dornaika (1995) Malti and Barreto (2010); 3) with the special orthogonal group by Park and Martin (1994). Decouple rotation from translation yields uncomplex, fast solutions, but at a cost that the rotation estimation errors would propagate to the translational part.

Modern methods tend to solve the hand-eye calibration jointly, which means the rotation and translation are estimated simultaneously, see Daniilidis and Bayro-Corrochano (1996) Daniilidis (1999) Andreff et al. (1999) Horaud and Dornaika (1995) Steger et al. (2018). The introduction of the screw theory in the hand-eye calibration by Chen (1991) provides a geometric view to analyze the restrictions of the hand-eye problem, which later motivated the invention of the dual quaternion approach by Daniilidis and Bayro-Corrochano (1996) Daniilidis (1999) that simultaneously recover rotation and translation using the singular value decomposition (SVD). A similar linear formulation was proposed by Andreff et al. (1999) but using the Kronecker product. However, due to the measurements contaminated by noise, neither the constraint of the unit quaternion ($||\mathbf{q}|| = 1$) nor the orthogonality constraint of $\mathbb{SO}(3)$ can be maintained, which requires normalization or orthogonalization. Considering the influence of random noise, optimization methods are favored in recent years. Horaud and Dornaika (1995) solve this problem with iterative nonlinear minimization. A new metric on the rigid transformations $\mathbb{SE}(3)$ for nonlinear optimization was proposed by Strobl and Hirzinger (2006) with the ability for automatic weighting. A recent iterative approach with superior performance was proposed by Pachtrachai et al. (2018). Aside from iterative solutions, global optimization approaches arose nowadays which perform calibration using convex relaxation and convex optimization. A Second Order Cone Programming (SOCP) method was proposed by Zhao (2011). Heller et al. (2014) presented several formulations of hand-eye calibration as multivariate polynomial optimization problems over semi-algebraic sets. Convex linear matrix inequality relaxations were used to set up a series of Semi-Definite Programming (SDP) problems, which can be effectively solved to obtain globally optimal solutions.

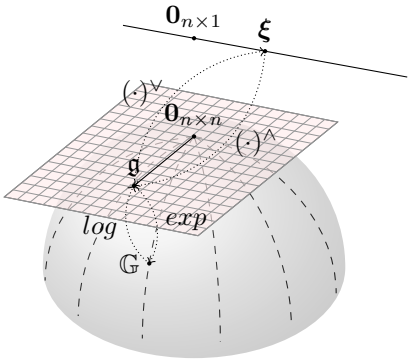

Fig. 2. Mapping relationship among matrix Lie Group, Lie Algebra, and the vector space.

3. NOTATION & PRELIMINARIES

In this section, we describe the notation and relevant mathematical preliminaries used to formulate the optimization problem addressed in this paper.

3.1 Notation

In the following sections, scalars are indicated with small letters (e.g. f). Vector are indicated as small bold letters (e.g. $\mathbf{r}$). Matrices are represented by bold capital letters (e.g. $\mathbf{K}$). The reference systems addressed in this paper are shown in Fig. 1 and defined as $\mathbf{C}_{(\cdot)}$, e.g. $\mathbf{C_A}$ is the local coordinate system (right-hand) of sensor A. Furthermore, the local coordinate system of i^{th} measurement is written with a subscript i. $\mathbf{X}$ is reserved for describing the transformation matrix between sensor A and sensor B. Without losing generality, $\mathbf{X}$ represents the transformation from $\mathbf{C_B}$ to $\mathbf{C_A}$.

3.2 Preliminaries

In this paper, optimization processes are completed on $\mathbb{SE}(3)$ for sakes of avoiding singularities and maintaining smoothness. Consequently, it is necessary to recall some useful concepts of $\mathbb{SE}(3)$. More details can be found in references by Barfoot and Furgale (2014) Hu et al. (2018). For motion data, we mean a 3D rigid transformation matrix $\mathbf{T} \in \mathbb{SE}(3)$ which is composed by a rotation matrix $\mathbf{R}$ and a translation $\mathbf{t}$ as:

$$\mathbb{SE}(3) : \mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \in \mathbb{R}^{4\times 4} | \mathbf{R} \in \mathbb{SO}(3), \mathbf{t} \in \mathbb{R}^3 \quad (1)$$

$$\mathbb{SO}(3) : \mathbf{R} \in \mathbb{R}^{3\times 3} | \mathbf{R}\mathbf{R}^T = \mathbf{I}_{3\times 3},\ det(\mathbf{R}) = 1 \quad (2)$$

$\mathbb{SO}(3)$ and $\mathbb{SE}(3)$ are examples of matrix Lie Group which has a topological structure of a smooth manifold rather than a vector space, see Hauberg et al. (2013).

The Lie algebra $\mathfrak{g}$ associated to a n-dimensional matrix Lie group is a n-dimensional tangent space. The mapping

relationships are shown in Fig. 2. The exponential $exp(\cdot)$ and logarithm $log(\cdot)$ operations establish a local diffeomorphism between a neighborhood of $\mathbf{0}_{n\times n}$ in the tangent space to a local neighborhood of the identity on the manifold. $\mathfrak{g}$ associates to its vector space $\mathbb{R}^n$ by $(\cdot)^\vee : \mathfrak{g} \to \mathbb{R}^n$ and $(\cdot)^\wedge : \mathbb{R}^n \to \mathfrak{g}$. This property allows to express the differential calculus, as needed for optimization, in vector space. The Lie algebra of $\mathfrak{se}(3)$ is given as follows,

$$\boldsymbol{\xi}^\wedge = \begin{bmatrix}\boldsymbol{\phi}\\ \boldsymbol{\rho}\end{bmatrix}^\wedge = \begin{bmatrix}\boldsymbol{\phi}^\wedge & \boldsymbol{\rho}\\ \mathbf{0}^T & 0\end{bmatrix} \in \mathfrak{se}(3) \tag{3}$$

$$\boldsymbol{\phi}^\wedge = \begin{bmatrix}\phi_x\\ \phi_y\\ \phi_z\end{bmatrix}^\wedge = \begin{bmatrix}0 & -\phi_z & \phi_y\\ \phi_z & 0 & -\phi_x\\ -\phi_y & \phi_x & 0\end{bmatrix} \in \mathfrak{so}(3) \tag{4}$$

where $\boldsymbol{\phi}, \boldsymbol{\rho} \in \mathbb{R}^3$ and $\boldsymbol{\xi} \in \mathbb{R}^6$ being the coefficients in corresponding vector spaces.

The adjoint $\boldsymbol{\mathcal{T}}$ of a transformation matrix $\mathbf{T}$ is defined as:

$$\boldsymbol{\mathcal{T}} = \begin{bmatrix}\mathbf{R} & \mathbf{t}^\wedge\mathbf{R}\\ \mathbf{0} & \mathbf{R}\end{bmatrix} \tag{5}$$

An additive increment in the vector space will associate to a multiplication increment on the manifold, which follows the below approximation

$$exp((\boldsymbol{\xi} + \delta\boldsymbol{\xi})^\wedge) \approx exp\left((\boldsymbol{\mathcal{J}}_l^{-1}\delta\boldsymbol{\xi})^\wedge\right) exp(\boldsymbol{\xi}^\wedge) \tag{6}$$

where $\boldsymbol{\mathcal{J}}_l$ is the left Jacobian of $\mathbb{SE}(3)$ defined as (7):

$$\boldsymbol{\mathcal{J}}_l = \sum_{n=0}^{\infty} \frac{1}{(n+1)} (\boldsymbol{\xi}^\curlywedge)^n \tag{7}$$

$$\boldsymbol{\xi}^\curlywedge = \begin{bmatrix}\boldsymbol{\phi}\\ \boldsymbol{\rho}\end{bmatrix}^\curlywedge = \begin{bmatrix}\boldsymbol{\phi}^\wedge & \boldsymbol{\rho}^\wedge\\ \mathbf{0}^T & \boldsymbol{\phi}^\wedge\end{bmatrix} \tag{8}$$

Practically, a finite number of terms can sufficiently approximate $\boldsymbol{\mathcal{J}}$, while a more complex analytical solution is detailed in Barfoot and Furgale (2014).

The Lie bracket for matrix Lie group is defined as:

$$[\mathbf{A}, \mathbf{B}] = \mathbf{AB} - \mathbf{BA} \tag{9}$$

More specifically, for $\boldsymbol{\xi}_1^\wedge,\ \boldsymbol{\xi}_2^\wedge\ \in \mathfrak{se}(3)$, we have

$$[\boldsymbol{\xi}_1^\wedge,\ \boldsymbol{\xi}_2^\wedge] = \boldsymbol{\xi}_1^\wedge\boldsymbol{\xi}_2^\wedge - \boldsymbol{\xi}_2^\wedge\boldsymbol{\xi}_1^\wedge = \left(\boldsymbol{\xi}_1^\curlywedge\boldsymbol{\xi}_2\right)^\wedge \tag{10}$$

The *Baker-Campbell-Hausdorff (BCH)* formula gives the following result:

$$log(exp(\mathbf{A})exp(\mathbf{B})) = \mathbf{A} + \mathbf{B} + \frac{1}{2}[\mathbf{A},\ \mathbf{B}] + h.o.t \tag{11}$$

where $h.o.t$ stands for the 2nd and higher order terms, see Gilmore (1974).

Finally, the distance of two poses $\mathbf{T}_1$ and $\mathbf{T}_2$ is defined with (12), which will be used in the following section.

$$dist_{\mathbf{T}_1,\mathbf{T}_2} = \| log(\mathbf{T}_1^{-1}\mathbf{T}_2)^\vee \| \tag{12}$$

It is easy to derive when $\mathbf{T}_1^{-1}\mathbf{T}_2 \to \mathbf{I}$, the distance will approach zero.

4. PROPOSED METHOD

In this section, details of the proposed method are described. We begin with the formulation with the matrix Lie group and derive a solution of the formulated problem in a Gauss-Newton fashion. Finally, we summarize the proposed approach in an algorithm for clarity.

4.1 Formulation & Derivation

With a batch of measurements, the calibration problem is equivalent to find the optimal $\mathbf{X}$ by minimizing an objective function:

$$\underset{\mathbf{X}\in\mathbb{SE}(3)}{\text{minimize:}}\ \ f_0(\mathbf{X}) \tag{13}$$

The explicit definition of f_0 varies with different approaches, e.g. the L_2-norm in Heller et al. (2014), the L_∞-norm in Zhao (2011). Here, we choose to formulate the objective function directly on $\mathbb{SE}(3)$ by minimizing the sum of pose distances defined as:

$$\begin{aligned} f_0(\mathbf{X}) &= \sum_{i=0}^{n} (dist_{\mathbf{T}_{A_i}^{-1}, \mathbf{X}\mathbf{T}_{B_i}\mathbf{X}^{-1}})^2 \\ &= \sum_{i=0}^{n} ||log\left(\mathbf{T}_{A_i}^{-1}\mathbf{X}\mathbf{T}_{B_i}\mathbf{X}^{-1}\right)^\vee||_2^2 \end{aligned} \tag{14}$$

The expansion is derived with the help of (12). Next we derive our solution to solve (14) iteratively. In the following derivation, the subscript i is dropped and $log\left(\mathbf{T}_A^{-1}\mathbf{X}\mathbf{T}_B\mathbf{X}^{-1}\right)^\vee$ is used for brevity.

Using the Lie algebra, we will have:

$$\begin{aligned} &log\left(\mathbf{T}_A^{-1}\mathbf{X}\mathbf{T}_B\mathbf{X}^{-1}\right)^\vee \\ &\quad = log\left(exp(-\boldsymbol{\xi}_A{}^\wedge)\mathbf{X}exp(\boldsymbol{\xi}_B{}^\wedge)\mathbf{X}^{-1}\right)^\vee \\ &\quad = log\left(exp\left(-\boldsymbol{\xi}_A{}^\wedge\right) exp\left((\boldsymbol{\mathcal{T}}_X\boldsymbol{\xi}_B)^\wedge\right)\right)^\vee \end{aligned} \tag{15}$$

where $\boldsymbol{\mathcal{T}}_X$ denotes the adjoint of $\mathbf{X}$, $\boldsymbol{\xi}_A^\wedge$, $\boldsymbol{\xi}_B^\wedge$ denote the corresponding $\mathfrak{se}(3)$ of $\mathbf{T}_A$, $\mathbf{T}_B$, respectively. For brevity, $\boldsymbol{\mathcal{T}}_X\boldsymbol{\xi}_B$ is represented with a new variable $\boldsymbol{\xi}_{XB}$. Then by adding a small amount of perturbation $\delta\mathbf{X}$ on the manifold, we will have:

$$\begin{aligned} f_1 &= log\left(\mathbf{T}_A^{-1}\delta\mathbf{X}\mathbf{X}\mathbf{T}_B\mathbf{X}^{-1}\delta\mathbf{X}^{-1}\right)^\vee \\ &= log(exp\left(-\boldsymbol{\xi}_A{}^\wedge\right) \underbrace{exp(\delta\boldsymbol{\xi}_X^\wedge)exp\left(\boldsymbol{\xi}_{XB}^\wedge\right)exp(-\delta\boldsymbol{\xi}_X^\wedge)}_{f_2})^\vee \end{aligned} \tag{16}$$

An approximation of the formula f_2 can be obtained using (6) and the relationship between the left and right Jacobians:

$$f_1 \approx exp((\underbrace{\boldsymbol{\xi}_{XB} + \boldsymbol{\mathcal{J}}_{\boldsymbol{l}}(\boldsymbol{\xi}_{XB})^{-1}(\delta\boldsymbol{\xi}_X - \boldsymbol{\mathcal{T}}'\delta\boldsymbol{\xi}_X)}_{\boldsymbol{\xi}'_{XB}})^\wedge) \tag{17}$$

where $\boldsymbol{\mathcal{T}}'$ and $\boldsymbol{\mathcal{J}}_{\boldsymbol{l}}\left(\boldsymbol{\xi}_{XB}\right)^{-1}$ denote the adjoint and the inverse of the left Jacobian matrix of the matrix formed by $exp\left(\boldsymbol{\xi}_{XB}^\wedge\right)$, respectively. With the simplified symbol $\boldsymbol{\xi}'_{XB}$, (16) can be represented as

$$f_1 = log\left(exp\left(-\boldsymbol{\xi}_A{}^\wedge\right) exp(\boldsymbol{\xi}'^\wedge_{XB})\right)^\vee \tag{18}$$

A first order approximation is obtained by applying (11) and (10) as

$$f_1 \approx -\boldsymbol{\xi}_A + \boldsymbol{\xi}'_{XB} - \frac{1}{2}\boldsymbol{\xi}_A{}^\curlywedge\boldsymbol{\xi}'_{XB} \tag{19}$$

Expanding (19) with the definition of $\boldsymbol{\xi}'_{XB}$, we will have the following:

$$
\begin{aligned}
f_1 \approx & -\boldsymbol{\xi}_A + \boldsymbol{\xi}'_{XB} - \frac{1}{2}\boldsymbol{\xi}_A{}^{\curlywedge}\boldsymbol{\xi}'_{XB} \\
= & \underbrace{-\boldsymbol{\xi}_A + \boldsymbol{\xi}_{XB} - \frac{1}{2}\boldsymbol{\xi}_A{}^{\curlywedge}\boldsymbol{\xi}_{XB}}_{\boldsymbol{e}} \\
& + \underbrace{\left(\mathbf{I} - \frac{1}{2}\boldsymbol{\xi}_A{}^{\curlywedge}\right)\left(\boldsymbol{\mathcal{J}}_{\boldsymbol{l}}(\boldsymbol{\xi}_{XB})^{-1}\left(\mathbf{I} - \boldsymbol{\mathcal{T}}'\right)\right)}_{\mathbf{G}} \delta\boldsymbol{\xi}_X
\end{aligned} \tag{20}
$$

where $\boldsymbol{e}$ and $\mathbf{G}$ are $\mathbf{X}$ dependent. By assuming $\mathbf{X}^*$ is the current estimation of $\mathbf{X}$, the objective function of (14) is expanded with first order approximation around $\mathbf{X}^*$ as

$$
f_0 \approx \sum_{i=0}^{n} ||\mathbf{e}(\mathbf{X}^*) + \mathbf{G}(\mathbf{X}^*)\delta\boldsymbol{\xi}_X||_2^2 \tag{21}
$$

Thus, the minimization of f_0 with respect to $\mathbf{X}$ is locally equivalent to minimize the approximated function with respect to $\delta\boldsymbol{\xi}_X$. Since the perturbation is applied directly on the manifold, there is no need to enforce any constraints on the optimized variables during the optimization procedure, which leaves an unconstrained quadratic minimization problem. The optimum solution $\delta\boldsymbol{\xi}_X^*$ can be easily obtained by taking the derivative with respect to $\delta\boldsymbol{\xi}_X$ and setting to zero, which equals to solve the following normal equation:

$$
\mathbf{G}^T(\mathbf{X}^*)\mathbf{G}(\mathbf{X}^*)\delta\boldsymbol{\xi}_X{}^* = -\mathbf{G}^T(\mathbf{X}^*)\mathbf{e}(\mathbf{X}^*) \tag{22}
$$

Once we have the optimal perturbation $\delta\boldsymbol{\xi}_X{}^*$, the current $\mathbf{X}^*$ is going to be updated as:

$$
\mathbf{X}^* = exp\left((\delta\boldsymbol{\xi}_X^*)^{\wedge}\right)\mathbf{X}^* \tag{23}
$$

which automatically ensures $\mathbf{X}^*$ remains in $\mathbb{SE}(3)$. This process is to be iterated until a maximum iteration achieves or the cost is lower than a given threshold ϵ. With this Gauss-Newton iterative scheme, $\delta\boldsymbol{\xi}_X{}^*$ will quickly converge to zero, which means $\mathbf{X}^*$ converges to a minimum quickly, see relevant experiments in Section 5.

We finally summarize the proposed approach in Algorithm 1 for clarity.

Algorithm 1 Proposed calibration approach

Input: Paired motion data $(\mathbf{T}_{A_i},\ \mathbf{T}_{B_i}),\ i = 1, \cdots, n$
Output: Extrinsic transformation matrix $\mathbf{X}$
  Initialization: $\mathbf{X}$ and $\boldsymbol{\xi}_{A_i}, \boldsymbol{\xi}_{B_i}, i = 1, \cdots, n$
  Compute $\boldsymbol{\xi}_{A_i}^{\curlywedge}, i = 1, \cdots, n$
  while not converged **do**
    Compute $\boldsymbol{\mathcal{T}}_X$ by (5)
    for $i = 1, \cdots, n$ **do**
      Compute $\boldsymbol{\xi}_{XB_i} = \boldsymbol{\mathcal{T}}_X\boldsymbol{\xi}_{B_i}$
      Compute $\boldsymbol{\mathcal{T}}'$ by (5) and $\boldsymbol{\mathcal{J}}_{\boldsymbol{l}}$ by (7) for $\boldsymbol{\xi}_{XB_i}$.
      Compute $\boldsymbol{e}_i$ and $\boldsymbol{G}_i$ by (20)
      $\boldsymbol{G} \leftarrow \boldsymbol{G} + \boldsymbol{G}_i$, $\boldsymbol{e} \leftarrow \boldsymbol{e} + \boldsymbol{e}_i$
    end for
    Solve (22) and update $\mathbf{X}$ by (23)
  end while
  return X

5. EXPERIMENTS

In this section, experiments on numerical data and real datasets are presented to validate the performance of the proposed method. To compare its performance, five state-of-the-art methods are implemented: the joint method **KR** proposed by Andreff et al. (1999), the iterative method **NLQ** by Horaud and Dornaika (1995) and **ATA** by Pachtrachai et al. (2018), the SOCP convex optimization method **SOCP** by Zhao (2011), and the global polynomial optimization method **GPOLY** [1] by Heller et al. (2014). The proposed method is marked with **SE3** throughout this section. Regard to software implementation, we use the original implementation provided by the respective authors for **ATA** and **GPOLY**. While for other approaches, we re-implement and optimize them with our best efforts. All methods are executed in MATLAB R2018A on a workstation with 2.80-GHz Intel Core i7 processor and 32-GB RAM. In regard to the initial value for iterative methods, we use the default initialization routine for **NLQ** and **ATA**, see Horaud and Dornaika (1995) and Pachtrachai et al. (2018). For proposed method **SE3**, we set $\mathbf{X}_0 = \mathbf{I}$ as the initial value.

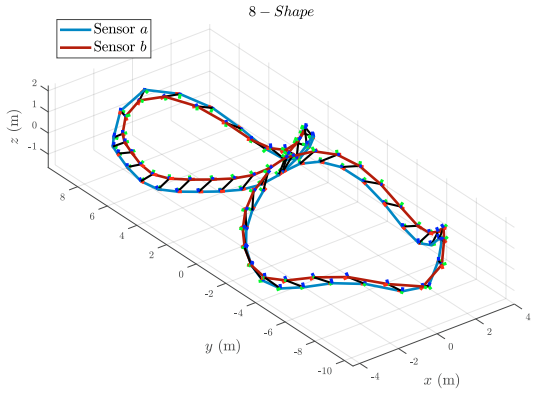

Fig. 3. The simulated path for sampling motion data. Curves in light blue and red show the traveled paths by the centers of two sensors. The black solid line represents the rigid connection.

The following error metrics are used to evaluate the results:

$$
\begin{aligned}
E_{\mathbf{R}} &= ||log(\hat{\mathbf{R}}_X^T\mathbf{R}_{true})^{\vee}|| \\
E_{\mathbf{t}} &= ||\hat{\mathbf{t}}_X - \mathbf{t}_{truth}||
\end{aligned}
$$

where $\mathbf{X}_{truth}$ represents the ground truth, $\hat{\mathbf{X}} = \begin{bmatrix} \hat{\mathbf{R}}_X & \hat{\mathbf{t}}_X \\ \mathbf{0}^T & 1 \end{bmatrix}$ denotes the estimated transformation matrix, and $||\cdot||$ denotes the Euclidean norm of a vector.

5.1 Numerical Data

In order to generate numerical motion data, a simulated trajectory, displayed in Fig. 3, is generated given the ground truth $\mathbf{X}_{true}$. From this trajectory, a batch of motion data $\mathbf{A}_i,\ \mathbf{B}_i,\ i = 1, \cdots, n,\ n = 50$ are equally sampled. Then random noises are generated from a multivariable Gaussian distribution, i.e. $\epsilon \sim \mathcal{N}(\mathbf{0},\ \boldsymbol{\Sigma})$, $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}_{6\times6}$, and added to $\mathbf{A}_i$ and $\mathbf{B}_i$ correspondingly. The units of standard deviation σ are meters for translation and radians for orientation, respectively. The performance of the aforementioned six methods under different levels of noise, (0.05, 0.15, 0.25, 0.35, 0.45, 0.55), are compared in

[1] The dual quaternion method "dqhec" is used for comparison since it is the best approach compared with another two formulations according to the results by Heller et al. (2014).

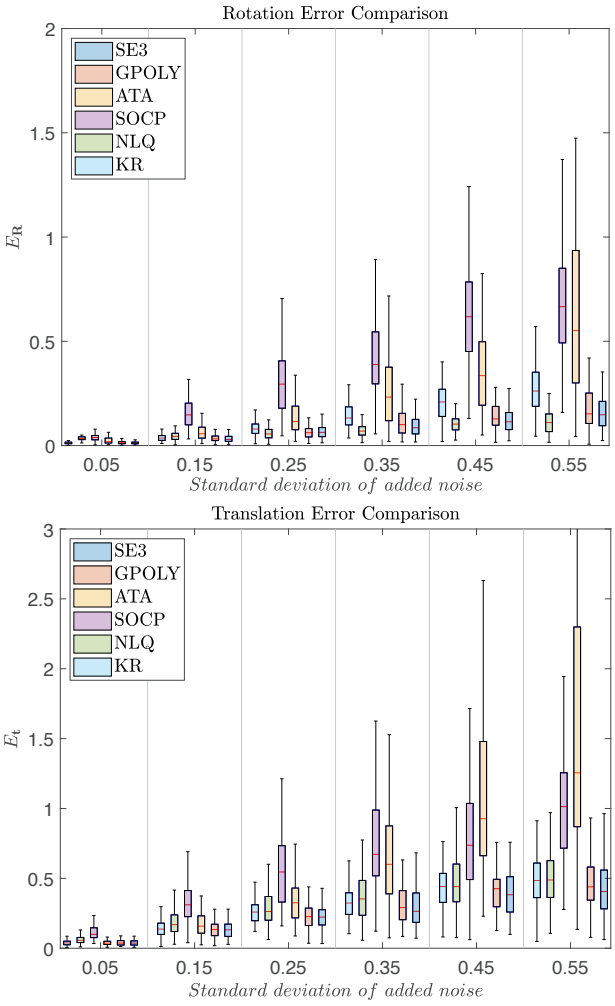

Fig. 4. Performance of rotational (top) and translational error (bottom) of benchmarked approaches under different noise levels. Results of corresponding noise levels are split by thin solid vertical lines.

this experiment. The performance is demonstrated as the box-and-whisker plots in Fig. 4 by performing 100 trials for each noise level.

The experiment shows that all methods behave well provided the noise level is low, e.g. $\sigma = 0.05, 0.15$. However, the rotational and translational errors for all methods start to increase when noises become severe. In terms of orientation, **SE3** demonstrates the best performance with the smallest mean error when noise level is not severe ($\sigma = 0.05, 0.15, 0.25$). In case of severe noise, **NLQ** manifests the best resilient performance to noises, followed by **SE3** and **GPOLY**. With respect to the translational part, **SE3** and **GPOLY** demonstrates superior performance to other methods with slightly different intra-differences. By contrast, **SOCP** and **ATA**, although performs similarly well to other methods under low noise level, are very sensitive to noises with their performance dropping significantly when noise level increases.

5.2 Real Datasets

In this experiment, the real RGB-D data provided by Brookshire and Teller (2012) is used. Aside from the

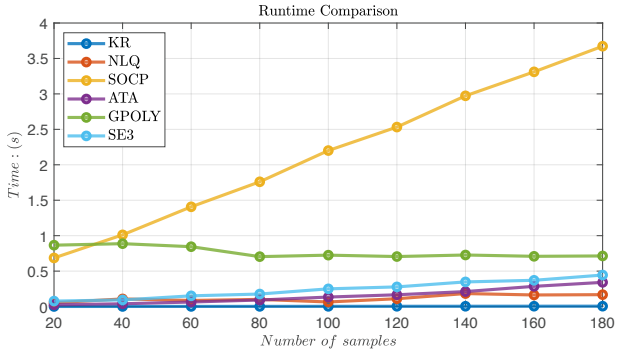

Fig. 5. Evaluations of execution time versus the number of measurements.

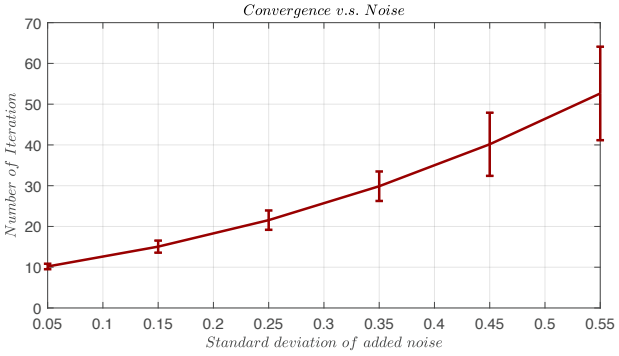

Fig. 6. Evaluations of convergence versus noise.

Table 1. Quantitative result on RGB-D dataset

Method	$E_{\mathbf{R}}$: (rad)	$E_{\mathbf{t}}$: (m)	Runtime: (s)
BL	0.0258	0.0117	3.7979
KR	0.0310	0.0587	0.0205
NLQ	0.0739	0.0233	1.1680
SOCP	0.1005	0.1273	5.8430
ATA	0.0260	0.0127	0.5213
GPOLY	0.0255	0.0180	1.6071
SE3	0.0253	0.0181	0.8191

aforementioned six methods, we also add the method proposed by Brookshire and Teller (2012), marked as **BL**, for comparison. The results are shown in Table. 1. The rotational and translational errors are computed with respect to the ground truth provided in this dataset. It can be seen from Table. 1 that the proposed method **SE3** has the most accurate result for rotational estimation, followed by **GPOLY** which also bypasses **BL**. However, for the translational part, **BL** gives the best performance. The proposed method **SE3** demonstrates similar performance with **GPOLY**, both slightly worse than **BL**. However, **SE3** is significantly faster than **BL**, which serves as an advantage when used in practice.

5.3 Runtime

In order to study the runtime variance with respect to the number of motion measurements, an experiment is carried out by fixing the noise level but varying the number of motion measurements from 20 to 180 with a step of 20. Similarly, time recordings are logged for 100 trials and the mean values are used to draw the curves in Fig. 5. Among those methods, **KR** and **GPOLY** are able to

execute at a constant time regardless of the number of motion pairs. Moreover, **KR** runs with the fastest speed because of its simplicity, while **GPOLY**, being the second slowest method, takes nearly 1 second for one run. It is expected that iterative optimization methods will take more time to converge when the amount of measurements increases, which is also validated in Fig. 5. Nonetheless, they (**NLQ**, **ATA**, **SE3**) are still able to converge with moderate time, bein faster than **GPOLY** and **SOCP**. The proposed method takes more time than **NLQ** and **ATA** because it involves more complex operations on matrix Lie group. However, this flawness is compensated by its superior accuracy and robustness to **NLQ** and **ATA**. **SOCP** is the most time-consuming method because the complexity of its formulation depends on the number of measurements.

5.4 Convergence

Fig. 6 evaluate the convergence speed with respect to noise level. It is clear that the proposed method needs more iterations to converge when noise becomes extreme. However, it only needs less than 30 iterations for convergence when the noise level is moderate. Considering the initial value as **I**, the convergence speed is admissible.

6. CONCLUSION

In conclusion, this paper focus on the estimation of extrinsic transformation between a pair of sensors from a batch of egomotion data, which closely relates to the hand-eye calibration problem. A novel objective function is proposed which minimizes the sum of the two-pose distance. Instead of reparameterizing the transformation matrix, we apply the optimization directly with the matrix Lie group in a Gauss-Newton style. The analytical solution of the Jacobian matrix is derived with manifold optimization techniques. The usage of manifold optimization guarantees the obtained result being strictly in the special Euclidean group, thus no further normalization or orthogonalization is needed. Numerical simulations and experiments on real datasets show that the proposed method can estimate the extrinsic transformation matrix efficiently and accurately. The performance is benchmarked with another five state-of-the-art methods and the results demonstrate the superiority and robustness of the proposed method.

It is expected that the proposed method will be used for extrinsic alignment among heterogeneous sensors including cameras, IMU/GNSS and LIDAR which will be used later for high-quality surveying tasks.

REFERENCES

Andreff, N., Horaud, R., and Espiau, B. (1999). On-line hand-eye calibration. In *3-D Digital Imaging and Modeling, 1999. Proceedings. Second International Conference on*, 430–436. IEEE.

Barfoot, T.D. and Furgale, P.T. (2014). Associating uncertainty with three-dimensional poses for use in estimation problems. *IEEE Transactions on Robotics*, 30(3), 679–693.

Brookshire, J. and Teller, S. (2012). Extrinsic calibration from per-sensor egomotion. In *Proceedings of Robotics: Science and Systems*. Sydney, Australia.

Chen, H.H. (1991). A screw motion approach to uniqueness analysis of head-eye geometry. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, 145–151. IEEE.

Chou, J.C. and Kamel, M. (1991). Finding the position and orientation of a sensor on a robot manipulator using quaternions. *The international journal of robotics research*, 10(3), 240–254.

Daniilidis, K. (1999). Hand-eye calibration using dual quaternions. *The International Journal of Robotics Research*, 18(3), 286–298.

Daniilidis, K. and Bayro-Corrochano, E. (1996). The dual quaternion approach to hand-eye calibration. In *icpr*, 318. IEEE.

Gilmore, R. (1974). Baker-campbell-hausdorff formulas. *Journal of Mathematical Physics*, 15(12), 2090–2092.

Hauberg, S., Lauze, F., and Pedersen, K.S. (2013). Unscented kalman filtering on riemannian manifolds. *Journal of mathematical imaging and vision*, 46(1), 103–120.

Heller, J., Henrion, D., and Pajdla, T. (2014). Hand-eye and robot-world calibration by global polynomial optimization. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, 3157–3164. IEEE.

Horaud, R. and Dornaika, F. (1995). Hand-eye calibration. *The international journal of robotics research*, 14(3), 195–210.

Hu, X., Jakob, J., Per, K., and Jiang, W. (2018). Accurate fiducial mapping for pose estimation using manifold optimization. In *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 206–212. IEEE.

Malti, A. and Barreto, J.P. (2010). Robust hand-eye calibration for computer aided medical endoscopy. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 5543–5549. IEEE.

Pachtrachai, K., Vasconcelos, F., Chadebecq, F., Allan, M., Hailes, S., Pawar, V., and Stoyanov, D. (2018). Adjoint transformation algorithm for hand–eye calibration with applications in robotic assisted surgery. *Annals of biomedical engineering*, 46(10), 1606–1620.

Park, F.C. and Martin, B.J. (1994). Robot sensor calibration: solving ax= xb on the euclidean group. *IEEE Transactions on Robotics and Automation*, 10(5), 717–721.

Shiu, Y.C. and Ahmad, S. (1989). Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form ax= xb. *IEEE Transactions on robotics and automation*, 5(1), 16–29.

Steger, C., Ulrich, M., and Wiedemann, C. (2018). *Machine vision algorithms and applications*. John Wiley & Sons.

Strobl, K.H. and Hirzinger, G. (2006). Optimal hand-eye calibration. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, 4647–4653. IEEE.

Tsai, R.Y. and Lenz, R.K. (1989). A new technique for fully autonomous and efficient 3d robotics hand/eye calibration. *IEEE Transactions on robotics and automation*, 5(3), 345–358.

Zhao, Z. (2011). Hand-eye calibration using convex optimization. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, 2947–2952. IEEE.