On Hand-Eye Calibration via On-Manifold Gauss-Newton Optimization

Marta Čolaković-Bencerić, Juraj Peršić, Ivan Marković, and Ivan Petrović

University of Zagreb Faculty of Electrical Engineering and Computing Laboratory for Autonomous Systems and Mobile Robotics (LAMOR),
{marta.colakovic, juraj.persic, ivan.markovic, ivan.petrovic}@fer.hr

Abstract. Perception of autonomous robotic systems is highly dependent on accurate data fusion of multiple heterogeneous sensors. However, to maximally exploit the advantages of such setups, sensor data fusion necessitates accurate extrinsic calibration. In this paper, we propose a novel derivation of the Gauss-Newton based iterative on-manifold batch solution to the hand-eye calibration problem. By adopting a special Euclidean group formulation of the objective function, we derive exact and approximate solutions and validate them via synthetic and real-world experiments. The results show that the accuracy of the proposed approximate solutions is on par with the exact solution and alternative on-manifold iterative solutions. Moreover, due to the near commutativity of the hand-eye problem in low noise scenarios, the proposed 0th order approximation achieves up to 4 times faster execution time, thus opening up practical possibilities of utilization in more complex optimization techniques.

Keywords: hand-eye calibration, Lie groups, odometry-based calibration, on-manifold optimization

1 Introduction

In order to successfully navigate the environment and complete varied and complex tasks, autonomous robotic systems are often equipped with multiple heterogeneous sensors. Fusing information gathered by such sensors relies on accurate intrinsic, extrinsic, and temporal calibrations. Intrinsic calibration involves building a sensor model and determining the internal parameters, while temporal calibration estimates the time delays between measurements made by different sensors of a robotic system. Extrinsic calibration, on the other hand, is the process of determining the rigid transformation between different sensors, or a sensor and a part of the robotic system to which it is rigidly attached. This is usually achieved by finding correspondences in data coming from multiple sensors. The source of data correspondences can be a manufactured target or the environment itself, resulting in the division of calibration approaches into target-based and targetless. Target-based extrinsic calibration methods use the manufactured target's features to find correspondences in sensor data [25, 24, 15], which are then used to evaluate the closed-form solution or to iteratively optimize the calibration parameters. Conversely, targetless methods rely on information gathered from the environment.

A special class of tragetless methods relies on ego-motion for correspondence registration. These methods can be (i) SLAM-based [5, 18, 22], with correspondence registration relying on features, or (ii) odometry-based, leveraging incremental motion of each sensor [6, 17]. In this paper, we focus on the odometry-based methods, for which a common mathematical formulation is called the hand-eye problem. One highly studied case of the hand-eye problem can be modeled by the equation $\mathbf{AX} = \mathbf{XB}$, and we can group related methods into decoupled and simultaneous. Decoupled methods first solve for the rotational component of the rigid transform, making the translational part of the hand-eye problem linear. To obtain closed-form solutions, in [26, 29], the authors used the angle-axis representation and in [8, 13] rotation was represented as unit quaternions, while in [23] the closed-form solution was found through $\mathfrak{so}(3)$ Lie algebra representation of rotation. In [19], the hand-eye problem was formulated using dual-quaternions reaching a least squares problem subject to a linear constraint. However, unlike simultaneous approaches, these solutions ignore the non-linear coupling of the translational and rotational components, thus propagating rotational error to the translational component of the calibration. Additionally, quaternion and dual-quaternion representation of rotation necessitates renormalization of the state vector. Such attributes of the rigid transform motivated further research into various possible parametrizations of the transformation components.

The first formulation of a simultaneous solution to the hand-eye problem was presented in [7] by using screw-motion representation of the rigid transformation. Subsequently, in [13], the authors proposed a quaternion-based simultaneous iterative solution optimized with the Levenberg-Marquardt algorithm. Other simultaneous solutions used the dual-quaternion representation of rigid motion [9, 11, 4]. In [9], calibration parameters are solved for using the singular value decomposition (SVD) method. Similarly, in [11], the authors offered a closed-form linear solution along with an iterative solution based on the Levenberg-Marquardt algorithm. Finally, in [4], the authors have formulated the dual-quaternion-based hand-eye problem as a maximum likelihood optimization problem, again, solving it with the Levenberg-Marquardt algorithm. An alternative approach was proposed in [1], where the Kronecker matrix product was used to reach a linear solution to the hand-eye problem. The proposed solution does not guarantee orthogonality of the estimated rotation matrix when simultaneously estimating rotational and translational components of the rigid transform. Recently, a multi-step iterative quaternion and Lie algebra based solution was proposed in [21] using the Levenberg-Marquardt algorithm for finding the optimal parameters. In [12], the authors formulated the hand-eye problem as a bound-and-branch algorithm using epipolar constraints. Optimization was carried out by linear programming reaching a globally optimal solution with respect to the L_∞ norm. Approaching the hand-eye problem with the errors-in-variables paradigm, in [16], the hand-eye problem was formulated on the *SO(3)*×$\mathbb{R}^3$ group and based the optimization algorithm on the Gauss-Helmert model. The authors of [20] framed the hand-eye problem as a non-linear least squares problem on the *SO(3)* × $\mathbb{R}^3$ group. Similarly, a novel error model for non-linear optimization on the *SO(3)* × $\mathbb{R}^3$ group was proposed in [28]. Finally, the objective function of the hand-eye problem was formulated directly on the *SE(3)* group in [14], where an approximate, yet analytical Jacobian of the objective function was introduced, allowing for on-manifold optimization by the Gauss-Newton algorithm. Rigid transformations were represented as members of the $\mathbb{R}^6$ vector space isomorphic to the $\mathfrak{se}(3)$ Lie algebra.

In this paper, we derive an iterative on-manifold batch optimization algorithm for hand-eye calibration based on the Gauss-Newton method. The hand-eye problem is formulated directly on the *SE(3)* group. That guarantees group closure and removes the need for normalization or orthogonalization during the optimization procedure caused by the presence of noise, while retaining a minimal number of degrees of freedom of the state vector. We derive an exact analytic Jacobian matrix of the objective function and a novel approximate Jacobian matrix necessary for evaluating the Gauss-Newton iterative solution. The proposed solutions are validated and compared against alternative approaches on synthetic and real-world experiments.

2 Mathematical Background

In this section, we introduce the necessary mathematical background of the *SE(3)* group and the associated $\mathfrak{se}(3)$ Lie algebra. Following the notation used in [2], motions, i.e., rigid transforms, are represented by $\mathbf{T} \in$ *SE(3)* – an object composed of a rotation matrix $\mathbf{R} \in$ *SO(3)* and a translation vector $\mathbf{t} \in \mathbb{R}^3$

$$\mathit{SE(3)}: \mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \in \mathit{SE(3)} \subset \mathbb{R}^{4\times 4} \mid \mathbf{R} \in \mathit{SO(3)}, \mathbf{t} \in \mathbb{R}^3. \tag{1}$$

The $SO(3)$ special orthogonal Lie group is the group of all rotations about the origin of 3D Euclidean space, $\mathbb{R}^3$. It is a non-abelian group closed under matrix multiplication. Specifically, that means two members of the group are generally non-commutative. The exceptions being the identity element and the inverse of a particular element

$$\mathbf{R}_1, \mathbf{R}_2 \in SO(3): \quad \mathbf{R}_2\mathbf{R}_1 \neq \mathbf{R}_2\mathbf{R}_1, \quad \mathbf{R}_1\mathbf{R}_1^{-1} = \mathbf{R}_1^{-1}\mathbf{R}_1, \quad \mathbf{R}_1\mathbf{I} = \mathbf{I}\mathbf{R}_1. \tag{2}$$

Transitively, the $SE(3)$ group is non-abelian with equivalent exceptions. Generally, a Lie algebra of a Lie group is a tangent vector space at the identity paired with its associated Lie bracket, with the $SO(3)$ Lie algebra being denoted by $\mathfrak{so}(3)$. Similarly, an $\mathfrak{se}(3)$ Lie algebra element $\boldsymbol{\xi}^\wedge$ is composed of an $\mathfrak{so}(3)$ Lie algebra element $\boldsymbol{\phi}^\wedge$ and a vector $\boldsymbol{\rho} \in \mathbb{R}^3$,

$$\boldsymbol{\xi}^\wedge = \begin{bmatrix} \boldsymbol{\phi}^\wedge & \boldsymbol{\rho} \\ \mathbf{0}^T & 0 \end{bmatrix}, \quad \boldsymbol{\phi}^\wedge = \begin{bmatrix} \phi_x \\ \phi_y \\ \phi_z \end{bmatrix}^\wedge = \begin{bmatrix} 0 & -\phi_z & \phi_y \\ \phi_z & 0 & -\phi_x \\ -\phi_y & \phi_x & 0 \end{bmatrix}. \tag{3}$$

Finally, the elements of the vector space isomorphic to the tangent Lie algebra are equal to $\boldsymbol{\xi} = \left[\boldsymbol{\rho}\ \boldsymbol{\phi}\right]^T$.

The exponential mapping connecting the elements of the tangent vector space isomorphic to $\mathfrak{se}(3)$ with the *SE(3)* Lie group is defined as

$$\mathrm{Exp}\ :\ \mathbb{R}^6 \to \mathit{SE(3)}\ ;\ \boldsymbol{\xi} \mapsto \mathbf{T} = \mathrm{Exp}(\boldsymbol{\xi}) = \exp(\boldsymbol{\xi}^\wedge) = \begin{bmatrix} \exp(\boldsymbol{\phi}^\wedge) & \mathbf{V}(\boldsymbol{\phi})\boldsymbol{\rho} \\ \mathbf{0}^T & 1 \end{bmatrix}, \quad (4)$$

with $\mathbf{V}(\boldsymbol{\phi})$ being the left *SO(3)* group Jacobian. Inversely, the logarithmic mapping transforms the group element into the tangent vector and is defined as

$$\mathrm{Log}\ :\ \mathit{SE(3)} \to \mathbb{R}^6\ ;\ \mathbf{T} \mapsto \boldsymbol{\xi} = \mathrm{Log}(\mathbf{T}) = \log(\mathbf{T})^\vee = \begin{bmatrix} \mathbf{V}^{-1}(\boldsymbol{\phi})\mathbf{t} \\ \mathrm{Log}(\mathbf{R}) \end{bmatrix}, \quad (5)$$

with $\boldsymbol{\phi} = \mathrm{Log}(\mathbf{R})$.

Defining the Lie bracket as $[\mathbf{A}, \mathbf{B}] = \mathbf{AB} - \mathbf{BA}$, allows us to write the Lie brackets of $\mathfrak{se}(3)$ and $\mathfrak{so}(3)$

$$\left[\boldsymbol{\xi}_1^\wedge, \boldsymbol{\xi}_2^\wedge\right] = \boldsymbol{\xi}_1^\wedge \boldsymbol{\xi}_2^\wedge - \boldsymbol{\xi}_2^\wedge \boldsymbol{\xi}_1^\wedge = \left(\boldsymbol{\xi}_1^\curlywedge \boldsymbol{\xi}_2\right)^\wedge, \quad (6)$$

$$\left[\boldsymbol{\phi}_1^\wedge, \boldsymbol{\phi}_2^\wedge\right] = \boldsymbol{\phi}_1^\wedge \boldsymbol{\phi}_2^\wedge - \boldsymbol{\phi}_2^\wedge \boldsymbol{\phi}_1^\wedge = \left(\boldsymbol{\phi}_1^\wedge \boldsymbol{\phi}_2\right)^\wedge. \quad (7)$$

Where the '$\curlywedge$' operator is defined as

$$\boldsymbol{\xi}^\curlywedge = \begin{bmatrix} \boldsymbol{\rho} \\ \boldsymbol{\phi} \end{bmatrix}^\curlywedge = \begin{bmatrix} \boldsymbol{\phi}^\wedge & \boldsymbol{\rho}^\wedge \\ \mathbf{0}_{3\times 3} & \boldsymbol{\phi}^\wedge \end{bmatrix}. \quad (8)$$

The solution for $\mathbf{C}$ to $\exp(\mathbf{A})\exp(\mathbf{B}) = \exp(\mathbf{C})$, for non-commutative $\mathbf{A}$ and $\mathbf{B}$, is given by the *Baker-Campbell-Hausdorff* (BCH) formula,

$$\mathbf{C} = \log(\exp(\mathbf{A})\exp(\mathbf{B})) = \mathbf{A} + \mathbf{B} + \frac{1}{2}[\mathbf{A}, \mathbf{B}] + \text{ h.o.t.} \quad (9)$$

Due to linearity of the '$\vee$' operator, the BCH formula can be expressed in terms of members of vector spaces isomorphic to the associated Lie algebra, $\mathfrak{se}(3)$,

$$\log(\exp(\boldsymbol{\xi}_1^\wedge)\exp(\boldsymbol{\xi}_2^\wedge))^\vee = \boldsymbol{\xi}_1 + \boldsymbol{\xi}_2 + \frac{1}{2}\boldsymbol{\xi}_1^\curlywedge \boldsymbol{\xi}_2 + \text{h.o.t.} \quad (10)$$

Another important concept is the adjoint. The adjoint is a linear transform that maps the Lie algebra elements at $\mathbf{X}$ to Lie algebra elements at the identity, thus connecting the local and global elements. The adjoint of a Lie group at an element $\mathbf{X}$ is defined as

$$\boldsymbol{\tau}^\wedge \mapsto \mathrm{Ad}_{\mathbf{X}}\left(\boldsymbol{\tau}^\wedge\right) \triangleq \mathbf{X}\boldsymbol{\tau}^\wedge\mathbf{X}^{-1}. \quad (11)$$

Linearity of the adjoint action allows us to define the equivalent matrix operator that maps local tangent vectors to their global counterparts, ${}^{\mathbf{X}}\boldsymbol{\tau} \mapsto {}^{\mathcal{E}}\boldsymbol{\tau} = \mathbf{Ad}_{\mathbf{X}}{}^{\mathbf{X}}\boldsymbol{\tau}$.

It can be shown, [2, 27], that the adjoint matrix of a rigid transformation $\mathbf{X} = \exp({\boldsymbol{\xi}_{\mathbf{X}}}^{\wedge}) \in$ *SE(3)* evaluates to

$$\mathbf{Ad}_{\mathbf{X}} = \mathcal{T}_{\mathbf{X}} = \sum_{n=0}^{\infty} \frac{1}{n!} \left({\boldsymbol{\xi}_{\mathbf{X}}}^{\curlywedge}\right)^{n} = \begin{bmatrix} \mathbf{R}_{\mathbf{X}} & {\mathbf{t}_{\mathbf{X}}}^{\wedge}\mathbf{R}_{\mathbf{X}} \\ \mathbf{0}_{3\times 3} & \mathbf{R}_{\mathbf{X}} \end{bmatrix}, \tag{12}$$

while the adjoint matrix of a rotation matrix $\mathbf{R} \in$ *SO(3)* is simply $\mathbf{R}$.

Finally, given that the special Euclidean group is closed under matrix multiplication we define left (global) and right (local) multiplicative increment as

$$\delta\boldsymbol{\xi} \oplus \mathbf{X} = \exp((\delta\boldsymbol{\xi})^{\wedge}) \exp(\boldsymbol{\xi}^{\wedge}) \approx \exp((\boldsymbol{\xi} + \mathcal{J}_l^{-1}(\boldsymbol{\xi})\delta\boldsymbol{\xi})^{\wedge}), \tag{13}$$

$$\mathbf{X} \oplus \delta\boldsymbol{\xi} = \exp(\boldsymbol{\xi}^{\wedge}) \exp((\delta\boldsymbol{\xi})^{\wedge}) \approx \exp((\boldsymbol{\xi} + \mathcal{J}_r^{-1}(\boldsymbol{\xi})\delta\boldsymbol{\xi})^{\wedge}), \tag{14}$$

where $\mathcal{J}_l$ and $\mathcal{J}_r$ denote the left and right Jacobian of the *SE(3)* group with the inverses being equal to

$$\mathcal{J}_l^{-1}(\boldsymbol{\xi}) = \sum_{n=0}^{\infty} \frac{B_n}{n!} \left(\boldsymbol{\xi}^{\curlywedge}\right)^{n}, \quad \mathcal{J}_r^{-1}(\boldsymbol{\phi}) = \mathcal{J}_l^{-1}(-\boldsymbol{\phi}). \tag{15}$$

B_n denote the Bernoulli numbers and the analytic closed-form expressions for $\mathcal{J}_l$, $\mathcal{J}_r$ and their inverses can be found in [3].

3 Proposed Method

In this section, we formulate the batch calibration problem on the *SE(3)* Lie group and derive the exact and approximate Gauss-Newton solutions. The hand-eye calibration problem is modeled by the equation

$$\mathbf{AX} = \mathbf{XB}. \tag{16}$$

The unknown fixed rigid transformation between two sensors is represented by $\mathbf{X}$ and motion segments of a pair of sensors by $\mathbf{A}$ and $\mathbf{B}$.

To determine the optimal $\mathbf{X}$ from a set of measured motion segments $\mathbf{A}_i$, $\mathbf{B}_i$, $i = 1, \ldots, N$, it is necessary to minimize the objective function defined as the sum of *SE(3)* distances

$$\mathbf{X}^* = \underset{\mathbf{X} \in SE(3)}{\arg\min} \frac{1}{2} \sum_{i=1}^{N} \|\mathbf{f}_i\|_2^2, \tag{17}$$

with $\|\cdot\|_2^2$ denoting the Euclidean norm of a vector. Expressing the hand-eye problem via (16) leads us to define $\mathbf{f}_i$ as follows

$$\mathbf{f}_i \;:\; SE(3) \rightarrow \mathbb{R}^6 \;;\; \mathbf{f}(\mathbf{X}) = \log({\mathbf{A}_i}^{-1}\mathbf{X}\mathbf{B}_i\mathbf{X}^{-1})^{\vee}. \tag{18}$$

To find the optimal $\mathbf{X}^*$ and determine the Gauss-Newton step we expand the distance function around the current calibration estimate, $\mathbf{X}$,

$$\mathbf{f}_i(\delta\boldsymbol{\xi}_{\mathbf{X}} \oplus \mathbf{X}) \approx \mathbf{e}_i(\mathbf{X}) + \mathbf{G}_i(\mathbf{X})\delta\boldsymbol{\xi}_{\mathbf{X}}, \tag{19}$$

where $\mathbf{G}_i(\mathbf{X})$ denotes the Jacobian matrix comprised of Lie derivatives of the distance vector function $\mathbf{f}_i$ at the current estimate $\mathbf{X}$, and $\mathbf{e}_i(\mathbf{X})$ denotes the error. That allows us to define the Gauss-Newton update as

$$\delta\boldsymbol{\xi}_{\mathbf{X}} = -\left(\sum_{i=1}^{N} \mathbf{G}_i^T \cdot \boldsymbol{\Sigma}_i^{-1} \cdot \mathbf{G}_i\right)^{-1} \sum_{i=1}^{N} \mathbf{G}_i^T \cdot \boldsymbol{\Sigma}_i^{-1} \cdot \mathbf{e}_i, \tag{20}$$

with $\boldsymbol{\Sigma}_i$ being the covariance matrix of each measurement. The Gauss-Newton update is applied multiplicatively to the left in accordance to the relation (13). For brevity the subscript i is omitted from the remainder of the derivation.

The exact formulation of the error function equals the distance function at the current estimate, $\mathbf{e}_{\mathbf{Exact}}(\mathbf{X}) = \mathbf{f}(\mathbf{X}) = \log(\mathbf{A}^{-1}\mathbf{X}\mathbf{B}\mathbf{X}^{-1})^{\vee}$, while the Jacobian matrix can be found through its definition,

$$\mathbf{G}_{\mathbf{Exact}} = (\mathcal{L}_{\boldsymbol{\varepsilon}}\mathbf{f}(\mathbf{X})) \triangleq \lim_{\delta\boldsymbol{\xi}_{\mathbf{X}}\to\mathbf{0}} \frac{\mathbf{f}(\delta\boldsymbol{\xi}_{\mathbf{X}} \oplus \mathbf{X}) - \mathbf{f}(\mathbf{X})}{\delta\boldsymbol{\xi}_{\mathbf{X}}} : \mathit{SE(3)} \to \mathbb{R}^{6\times 6}. \tag{21}$$

The numerator in (21) is equal to

$$\begin{aligned}
\mathbf{f}(\delta\boldsymbol{\xi}_{\mathbf{X}} \oplus \mathbf{X}) - \mathbf{f}(\mathbf{X}) &= \log(\mathbf{A}^{-1}\delta\mathbf{X}\mathbf{X}\mathbf{B}\mathbf{X}^{-1}\delta\mathbf{X}^{-1})^{\vee} - \log(\mathbf{A}^{-1}\mathbf{X}\mathbf{B}\mathbf{X}^{-1})^{\vee} \\
&= \left(\log(\mathbf{A}^{-1}\delta\mathbf{X}\mathbf{X}\mathbf{B}\mathbf{X}^{-1}\delta\mathbf{X}^{-1}) - \log(\mathbf{A}^{-1}\mathbf{X}\mathbf{B}\mathbf{X}^{-1})\right)^{\vee} \\
&= \log\left(\mathbf{A}^{-1}\delta\mathbf{X}\mathbf{X}\mathbf{B}\mathbf{X}^{-1}\delta\mathbf{X}^{-1}\mathbf{X}\mathbf{B}^{-1}\mathbf{X}^{-1}\mathbf{A}\right)^{\vee}.
\end{aligned} \tag{22}$$

First we define the tangent vector of $\delta\mathbf{X}$ as $\delta\boldsymbol{\xi}_{\mathbf{X}} = \log(\delta\mathbf{X})^{\vee}$. Then by applying the identity $\mathbf{X}\exp(\boldsymbol{\xi}^{\wedge})\mathbf{X}^{-1} = \exp(\mathbf{X}\boldsymbol{\xi}^{\wedge}\mathbf{X}^{-1})$, and the definition of the adjoint (11) to (22), we reach

$$\mathbf{f}(\delta\boldsymbol{\xi}_{\mathbf{X}} \oplus \mathbf{X}) - \mathbf{f}(\mathbf{X}) = \log\left(\mathbf{A}^{-1}\exp(\delta\boldsymbol{\xi}_{\mathbf{X}}{}^{\wedge})\exp((-\mathcal{T}_{\mathbf{X}}\mathcal{T}_{\mathbf{B}}\mathcal{T}_{X}{}^{-1}\delta\boldsymbol{\xi}_{\mathbf{X}})^{\wedge})\mathbf{A}\right)^{\vee}. \tag{23}$$

We simplify the expression further by using the definition the left increment (13) and the inverse of the left *SE(3)* group Jacobian (15),

$$\begin{aligned}
\mathbf{f}(\delta\boldsymbol{\xi}_{\mathbf{X}} \oplus \mathbf{X}) - \mathbf{f}(\mathbf{X}) &= \log\left(\mathbf{A}^{-1}\exp((\delta\boldsymbol{\xi}_{\mathbf{X}} - \mathcal{T}_{\mathbf{X}}\mathcal{T}_{\mathbf{B}}\mathcal{T}_{X}{}^{-1}\delta\boldsymbol{\xi}_{\mathbf{X}})^{\wedge})\mathbf{A}\right)^{\vee} \\
&= \log\left(\exp((\mathcal{T}_{\mathbf{A}}{}^{-1}(\mathbf{I}_6 - \mathcal{T}_{\mathbf{X}}\mathcal{T}_{\mathbf{B}}\mathcal{T}_{X}{}^{-1})\delta\boldsymbol{\xi}_{\mathbf{X}})^{\wedge})\right)^{\vee}.
\end{aligned} \tag{24}$$

From (24) we deduce the exact Jacobian of the distance function

$$\mathbf{G}_{\mathbf{Exact}}(\mathbf{X}) = \mathcal{T}_{\mathbf{A}}{}^{-1}(\mathbf{I}_6 - \mathcal{T}_{\mathbf{X}}\mathcal{T}_{\mathbf{B}}\mathcal{T}_{X}{}^{-1}). \tag{25}$$

3D rotations are generally non-commutative, transitively making rigid transformations generally non-commutative. Due to the nature of the hand-eye problem, in zero noise environments $\mathbf{A}^{-1}$ and $\mathbf{X}\mathbf{B}\mathbf{X}^{-1}$ are identical and, thus, commutative. If the motion segment measurements are only slightly noisy, we view $\mathbf{A}^{-1}$ and $\mathbf{X}\mathbf{B}\mathbf{X}^{-1}$ as nearly commutative, making $\|[\mathbf{A}^{-1}, \mathbf{X}\mathbf{B}\mathbf{X}^{-1}]^{\vee}\|_2^2$ small. In order to reduce computational complexity in calculating the optimization step, we develop an approximate solution justified by the previous argument.

Following a similar procedure to [14], we apply the increment to the left to the current estimate of the calibration parameters $\mathbf{X}$ in the distance function,

$$\mathbf{f}(\delta\boldsymbol{\xi}_{\mathbf{X}} \oplus \mathbf{X}) = \log\left(\exp(-\boldsymbol{\xi}_{\mathbf{A}}{}^{\wedge})\exp(\delta\boldsymbol{\xi}_{\mathbf{X}}{}^{\wedge})\mathbf{X}\exp(\boldsymbol{\xi}_{\mathbf{B}}{}^{\wedge})\mathbf{X}^{-1}\exp(-\delta\boldsymbol{\xi}_{\mathbf{X}}{}^{\wedge})\right)^{\vee}. \tag{26}$$

Again, using the identity $\mathbf{X}\exp(\boldsymbol{\xi}^{\wedge})\mathbf{X}^{-1} = \exp(\mathbf{X}\boldsymbol{\xi}^{\wedge}\mathbf{X}^{-1})$, and the definition of the adjoint (11) we reach

$$\mathbf{f}(\delta\boldsymbol{\xi}_{\mathbf{X}} \oplus \mathbf{X}) = \log\left(\exp(-\boldsymbol{\xi}_{A}{}^{\wedge})\exp(\delta\boldsymbol{\xi}_{\mathbf{X}}{}^{\wedge})\exp((\underbrace{\mathcal{T}_{\mathbf{X}}\boldsymbol{\xi}_{B}}_{\boldsymbol{\xi}_{XB}})^{\wedge})\exp(-\delta\boldsymbol{\xi}_{\mathbf{X}}{}^{\wedge})\right)^{\vee}. \tag{27}$$

Repeating the previous step gives the following formulation of the perturbed distance function

$$\mathbf{f}(\delta\boldsymbol{\xi}_{\mathbf{X}} \oplus \mathbf{X}) = \log\left(\exp(-\boldsymbol{\xi}_{\mathbf{A}}{}^{\wedge})\exp((\mathcal{T}_{\delta\mathbf{X}}\boldsymbol{\xi}_{\mathbf{XB}})^{\wedge})\right)^{\vee}. \tag{28}$$

After applying the BCH formula on $\mathfrak{se}(3)$, as in (10), to the first order of approximation, we are left with

$$\mathbf{f}(\delta\boldsymbol{\xi}_{\mathbf{X}} \oplus \mathbf{X}) \approx -\boldsymbol{\xi}_{\mathbf{A}} + \mathcal{T}_{\delta\mathbf{X}}\boldsymbol{\xi}_{\mathbf{XB}} - \frac{1}{2}\boldsymbol{\xi}_{\mathbf{A}}{}^{\curlywedge}\mathcal{T}_{\delta\mathbf{X}}\boldsymbol{\xi}_{\mathbf{XB}}. \tag{29}$$

For small $\|\delta\mathbf{X}\|$ the adjoint can be approximated with $\mathcal{T}_{\delta\mathbf{X}} \approx \mathbf{I}_6 + \delta\boldsymbol{\xi}_{\mathbf{X}}{}^{\curlywedge}$. Along with the equality $\mathbf{x}^{\curlywedge}\mathbf{y} = -\mathbf{y}^{\curlywedge}\mathbf{x}$, that leads us to

$$\mathbf{f}(\delta\boldsymbol{\xi}_{\mathbf{X}} \oplus \mathbf{X}) \approx \underbrace{-\boldsymbol{\xi}_{\mathbf{A}} + \boldsymbol{\xi}_{\mathbf{XB}} - \frac{1}{2}\boldsymbol{\xi}_{\mathbf{A}}{}^{\curlywedge}\boldsymbol{\xi}_{\mathbf{XB}}}_{\mathbf{e}_1} + \underbrace{\left(-\boldsymbol{\xi}_{\mathbf{XB}}{}^{\curlywedge} + \frac{1}{2}\boldsymbol{\xi}_{\mathbf{A}}{}^{\curlywedge}\boldsymbol{\xi}_{\mathbf{XB}}{}^{\curlywedge}\right)}_{\mathbf{G}_1}\delta\boldsymbol{\xi}_{\mathbf{X}}. \tag{30}$$

The 0th order error and Jacobian functions can be extracted from the 1st order formulation

$$\mathbf{e}_0(\mathbf{X}) = -\boldsymbol{\xi}_{\mathbf{A}} + \boldsymbol{\xi}_{\mathbf{XB}}, \quad \mathbf{G}_0(\mathbf{X}) = -\boldsymbol{\xi}_{\mathbf{XB}}{}^{\curlywedge}. \tag{31}$$

To determine the influence of the approximation order in the BCH formula on the calibration accuracy, we compare results reached by using the exact formulation, the 1st and the 0th order approximate formulations. Approximate Gauss-Newton solutions were derived by leveraging the near-commutative nature of $\mathbf{A}^{-1}$ and $\mathbf{XBX}^{-1}$ matrices making up the distance function on the $SE(3)$ group and applying the BCH formula. Additionally, the novel formulation of the perturbed distance function through the adjoint transformation on $SE(3)$ made the proposed approximate solutions computationally less costly than the approximate solution previously suggested in [14].

4 Experimental Results

Here we present results of the performed experiments on synthetic and real world datasets where we investigated the accuracy of calibration with varying orders of approximation, rigid motion representations and noise levels. In previous work, [14, 20, 16], regarding on-manifold optimization and the hand-eye problem, it was assumed that two sides, or at least the rotational parts, of (16) are commutative without it being explicitly stated. In noisy environments that is not strictly true; however, with the following experiments we will demonstrate that in low noise data the Gauss-Newton solution derived with the assumption of commutativity produces sufficiently accurate results.

We compare the following approaches. First, the optimization algorithms based on the proposed approximate methods derived from the 0th order approximation, Eq. (31), and the 1st order approximation, Eq. (30), of the objective function using the BCH formula, are referred to as **SE3_0** and **SE3_1**, respectively. A similar method proposed in [14] is referred to as **SE3_JL**. To determine the influence of rigid motion representation on the calibration outcome, we have re-derived a solution on the commonly used *SO(3)* $\times \mathbb{R}^3$ group [29, 20] following a similar procedure to the one used in Section 3

$$\mathbf{e}_{SO3R3}(\mathbf{X}) = \begin{bmatrix} (\mathbf{I}_3 - \mathbf{R_A})\mathbf{t_X} + \mathbf{R_X}\mathbf{t_B} - \mathbf{t_A} \\ -\boldsymbol{\phi}_\mathbf{A} + \mathbf{R_X}\boldsymbol{\phi}_\mathbf{B} \end{bmatrix}, \tag{32}$$

$$\mathbf{G}_{SO3R3}(\mathbf{X}) = \begin{bmatrix} \mathbf{I}_3 - \mathbf{R_A} & -(\mathbf{R_X}\mathbf{t_B})^\wedge \\ \mathbf{0} & -(\mathbf{R_X}\boldsymbol{\phi}_\mathbf{B})^\wedge \end{bmatrix}. \tag{33}$$

We denote the algorithm, based on the error function and Jacobian on $SO(3) \times \mathbb{R}(3)$, by **SO3R3**. Finally, the algorithm based on (25), the exact calculation of the error and Jacobian functions, is denoted by **Exact**.

All of the methods were implemented in *Python3* and ran on a machine with i7-6700HQ CPU at 2.6 GHz × 8 and 16 GB of 2133 MHz DDR4 RAM. Furthermore, all of the methods were initialized with $\mathbf{X}_0 = \mathbf{I}_4$, with the ground truth calibration being $\mathbf{t}_\mathbf{GT}^T = [0.0 \;\; -0.20 \; 0.0]$ m, $\boldsymbol{\phi}_\mathbf{GT}^T = [-1.21 \;\; -1.21 \;\; -1.21]$ rad.

The following error metrics are used to evaluate the results

$$E_\mathbf{R} = \| \log\left(\mathbf{R}_{\mathrm{GT}}\mathbf{R}_{\mathrm{est}}^T\right)^\vee \|_2^2, \tag{34}$$

$$E_\mathbf{t} = \|\mathbf{t}_{\mathrm{GT}} - \mathbf{t}_{\mathrm{est}}\|_2^2\,. \tag{35}$$

In (34) and (35), $\mathbf{R}_{\mathrm{GT}}$ and $\mathbf{t}_{\mathrm{GT}}$ denote components of the ground truth rigid transform between sensors, and $\mathbf{R}_{\mathrm{est}}$ and $\mathbf{t}_{\mathrm{est}}$ components of the estimated rigid transformation.

4.1 Simulation Results

In the first synthetic experiment, we have generated a reasonably informative, yet achievable, 3D synthetic trajectory shown in Fig. 1. It was inspired by the Bernoulli lemniscate plane curve and the 3D trajectory simulator *Bernoulli-Lemniscate* from the *SRRG* toolbox [10]. Our 3D trajectory is described by the following set of parametric equations, with $a = 1.5\,\mathrm{m}$ denoting the half-width in

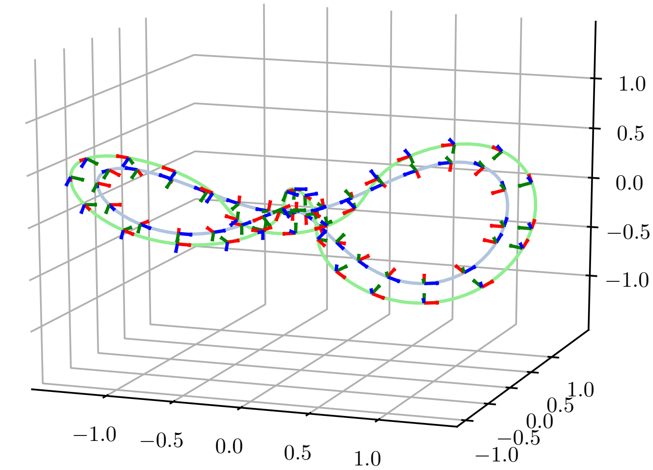

Fig. 1. The synthetic 3D Bernoulli lemniscate trajectory. Green and blue parametric curves represent trajectories of two sensors with associated local coordinate systems.

the x axis direction

$$x(t) = a\frac{\cos(t)}{1+\sin^2(t)}, \tag{36}$$

$$y(t) = \sin(t)x(t), \tag{37}$$

$$z(t) = \cos(t)y(t). \tag{38}$$

The trajectory of the second sensor was generated by applying the $\mathbf{X}_{\mathrm{GT}}$ transformation. After extracting equally sampled ($\mathit{\Delta} t = 0.02\,\mathrm{s}$) motion segments $\mathbf{A}_i,\ \mathbf{B}_i,\ i = 1,\dots,315$, from the synthetic trajectory, Gaussian noise was added in the local frame. Each instance of the Gaussian noise was extracted from a distribution $\mathcal{N}(0,\, \mathbf{\Sigma})$, $\mathbf{\Sigma} = \sigma^2\mathbf{I}_6$, with units of σ being in meters for the translational parameters $\boldsymbol{\rho}$ and radians for the rotational parameters $\boldsymbol{\phi}$. The standard deviation of noise σ in our experiments varied from 0.001 to 0.009 for both the translational and rotational part.

To analyze the performance of the benchmarked methods we have generated 100 trajectories per noise level, with noise drawn from the aforementioned distribution. The results are shown in Fig. 2. It is evident that, within the applied noise range, methods based on *SE(3)* and *SO(3)* $\times\ \mathbb{R}^3$ representation of rigid motions, produce almost identical results. Comparing the performance of the exact solution with the 0th and 1st order solutions, we can see that in this noise and motion range the 1st order approximate methods **SE3_1** and **SE3_JL**

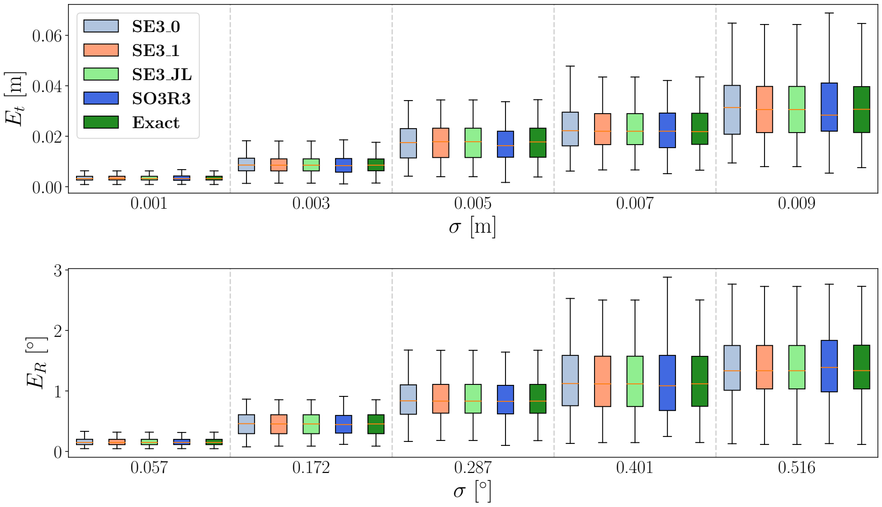

Fig. 2. Translational (top) and rotational (bottom) error of the benchmarked approaches with increasing noise for the Bernoulli lemniscate trajectory. The translational (top) and rotational (bottom) error are shown on the y axis in units of [m] and [°]. The standard deviation of noise σ is shown on the x axis.

are effectively equivalent to the exact solution. Equally, the 0th order approximate method yields similarly accurate solutions within the aforementioned noise range.

To eliminate the influence of trajectory informativeness on the calibration accuracy, in the second synthetic experiment we have analyzed the methods' performance on 100 randomly generated trajectories drawn form a Gaussian distribution. Randomly generated trajectories are generally the most informative possible trajectories but are difficult to achieve in real life situations. The distribution was adjusted to have a similar range of translational $(0.01-0.05\ \mathrm{m})$ and rotational displacements $(0-7\ ^{\circ})$ as a distribution of motion segments of the 3D Bernoulli lemniscate trajectory. Again, noise drawn from a Gaussian distribution $(\mathcal{N}(0,\, \mathbf{\Sigma}),\ \mathbf{\Sigma} = \sigma^2 \mathbf{I}_6)$ was added in the local frame. Randomly generated trajectories consist of $N = 315$ motion segments, lasting $\mathit{\Delta} t = 0.02\,\mathrm{s}$. The results are shown in Fig. 3. Testing the algorithms on randomly generated trajectories has confirmed that all of the on-manifold optimization schemes achieve similar accuracy in the studied noise range, with the calibration results being slightly improved by the more informative random trajectories.

In order to examine the runtime of the five benchmarked methods, we have averaged runtime over 100 randomly generated trajectories over the full noise range. The importance of our approximate *SE(3)* based solutions is evident when comparing the execution times of the benchmarked methods as shown in Fig. 4. The **Exact** and **SO3R3** methods have a longer time of execution due to multiple matrix multiplication operations needed to evaluate the Jacobian and error

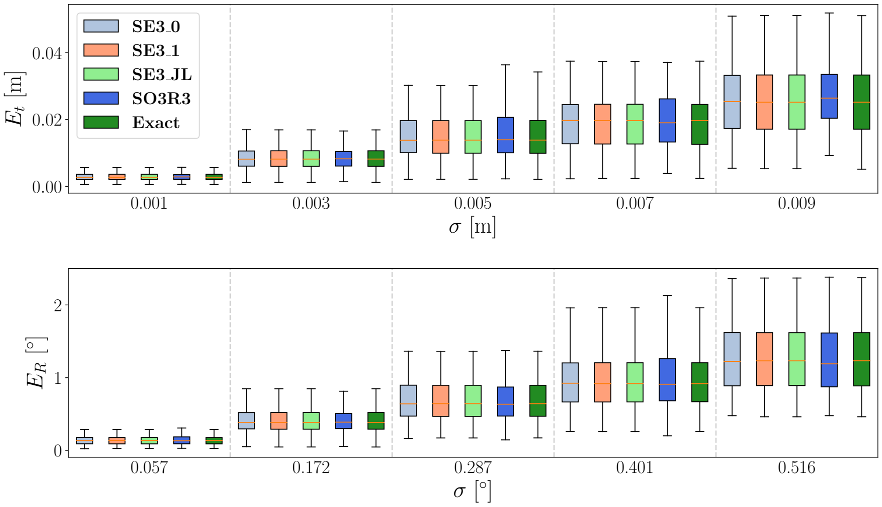

Fig. 3. Translational (top) and rotational (bottom) error of benchmarked approaches with increasing noise for randomly generated trajectories. The translational (top) and rotational (bottom) errors are shown on the y axis in units of [m] and [°]. The standard deviation of noise σ is shown on the x axis.

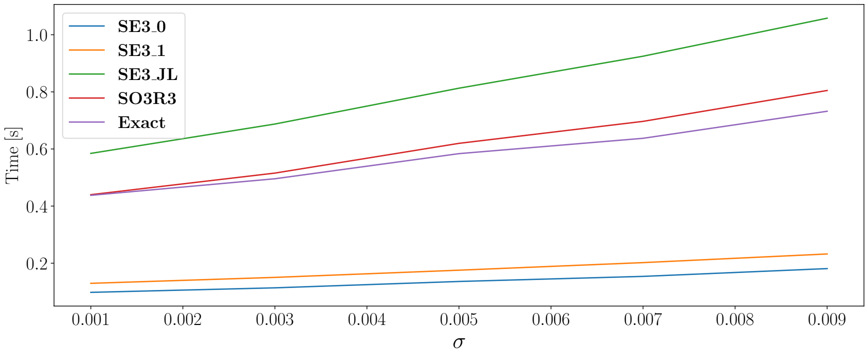

Fig. 4. Evaluation time of the benchmarked methods in seconds averaged over 100 noise instances.

matrices. Additionally, the **SE3_JL** method proposed in [14] has the longest runtime. That is caused by the complex structure of the analytical expression for $\mathcal{J}_l^{-1}$, used to calculate the approximate Jacobian matrix. On average, all of the these methods achieved convergence in an equal number of iterations. The comparatively low execution time of the proposed approximate methods, **SE3_0** and **SE3_1**, opens up the practical possibility of using them in more complex optimization techniques. Although the optimization using regression models that account for the measurement noise, e.g., [16], are generally more accurate, they

Table 1. Calibration results on the RGB-D dataset.

Method	$E_{\mathbf{t}}$ (m)	$E_{\mathbf{R}}$ (°)	Runtime (s)
SE3_0	0.01614	1.206	**0.1400**
SE3_1	0.01553	1.195	0.1735
SE3_JL	0.01553	1.191	0.8283
SO3R3	0.02088	**1.182**	0.6295
Exact	**0.01541**	1.195	0.5526

are time consuming, making the fast analytic Jacobians key in formulating an on-line solution.

4.2 Real World Data

For the real-world data experiment, we used the dataset presented in [4]. It includes 200 estimated motion segments of two rigidly mounted Xtion RGB-D cameras and the ground truth calibration between them, with motion estimation performed by the KinectFusion algorithm. The ground truth calibration was provided with the dataset and the covariance matrices of each measurement are assumed to be constant. From Table 1 we can conclude that all of the methods based on the *SE(3)* representation of rigid motion, **SE3_0**, **SE3_1**, **SE3_JL**, **SO3R3**, and **Exact**, performed similarly well at determining the rotational calibration parameters. However, the **SO3R3** method yielded somewhat less accurate translational calibration parameters than the *SE(3)* based methods. Again, the benefits of using proposed approximate methods, **SE3_0** and **SE3_1**, becomes evident when comparing the runtime. The number of iterations across the benchmarked methods was similar, with **SE3_0** and **SE3_1** having significantly smaller runtime. Again, the conclusions drawn from the synthetic experiments have been confirmed with real-world data. The accuracy of the *SE(3)* based methods is largely the same, while the $SO(3) \times \mathbb{R}^3$ based method has slightly higher rotational accuracy but lower translational accuracy. The **SE3_0** and **SE3_1** proposed methods vastly outperform other on-manifold optimization methods in regard to execution time.

5 Conclusion

In this paper, we have proposed an on-manifold solution to the batch hand-eye calibration problem. Framing the hand-eye problem on the *SE(3)* manifold guarantees the group closure of the solution and removes the need for renormalization or reorthogonalization during the optimization process. That in turn lowers the computational cost. Additionally, formulating the calibration problem directly on the *SE(3)* manifold allowed us to derive analytic exact and approximate Jacobians needed to calculate the iterative Gauss-Newton solution. The derived analytic expressions make for an efficient optimization scheme, all the while retaining the accuracy of conventional optimization approaches. We validated the proposed solutions and compared them to alternative approaches on synthetic and real-world data. The results explicitly demonstrate that in low noise situations the hand-eye problem can be considered near commutative, and that the novel approximate on-manifold iterative solution is the most efficient out of the benchmarked solutions, while retaining on par performance.

For future work, the superior efficiency of the proposed method opens up the practical possibility of extending the Gauss-Newton based solution to more complex regression models.

Acknowledgment

This research has been supported by the European Regional Development Fund under the grant KK.01.2.1.02.0119 (A-UNIT).

References

1. Andreff, N., Horaud, R., Espiau, B.: On-line hand-eye calibration. In: Second International Conference on 3-D Digital Imaging and Modeling (3DIM '99). (1999) 430–436
2. Barfoot, T.: State estimation for robotics. Cambridge University Press, Cambridge, United Kingdom New York, NY, USA (2017)
3. Barfoot, T.D., Furgale, P.T.: Associating uncertainty with three-dimensional poses for use in estimation problems. IEEE Transactions on Robotics **30**(3) (June 2014) 679–693
4. Brookshire, J., Teller, S.: Extrinsic calibration from per-sensor egomotion. In: Proceedings of Robotics: Science and Systems, Sydney, Australia (July 2012)
5. Carrera, G., Angeli, A., Davison, A.J.: SLAM-based automatic extrinsic calibration of a multi-camera rig. In: 2011 IEEE International Conference on Robotics and Automation (ICRA). (2011) 2652–2659
6. Censi, A., Franchi, A., Marchionni, L., Oriolo, G.: Simultaneous calibration of odometry and sensor parameters for mobile robots. IEEE Transactions on Robotics **29**(2) (2013) 475–492
7. Chen, H.: A screw motion approach to uniqueness analysis of head-eye geometry. In: Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. (1991) 145–151
8. Chou, J.C.K., Kamel, M.: Finding the position and orientation of a sensor on a robot manipulator using quaternions. The International Journal of Robotics Research **10**(3) (June 1991) 240–254
9. Daniilidis, K., Bayro-Corrochano, E.: The dual quaternion approach to hand-eye calibration. In: Proceedings of 13th International Conference on Pattern Recognition, IEEE (1996)
10. Della Corte, B., Andreasson, H., Stoyanov, T., Grisetti, G.: Unified motion-based calibration of mobile multi-sensor platforms with time delay estimation. IEEE Robotics and Automation Letters **4**(2) (2019) 902–909
11. Dornaika, F., Horaud, R.: Simultaneous robot-world and hand-eye calibration. IEEE Transactions on Robotics and Automation **14**(4) (1998) 617–622

12. Heller, J., Havlena, M., Pajdla, T.: Globally optimal hand-eye calibration using branch-and-bound. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**(5) (2016) 1027–1033
13. Horaud, R., Dornaika, F.: Hand-eye calibration. The International Journal of Robotics Research **14**(3) (June 1995) 195–210
14. Hu, X., Olesen, D., Per, K.: Calibration of extrinsic transformation using manifold optimization. IFAC-PapersOnLine **52**(8) (2019) 124–129
15. Huang, J.K., Wang, S., Ghaffari, M., Grizzle, J.W.: Lidartag: A real-time fiducial tag system for point clouds. IEEE Robotics and Automation Letters **6**(3) (2021) 4875–4882
16. Huang, K., Stachniss, C.: Extrinsic multi-sensor calibration for mobile robots using the Gauss-Helmert model. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). (2017) 1490–1496
17. Ishikawa, R., Oishi, T., Ikeuchi, K.: Lidar and camera calibration using motions estimated by sensor fusion odometry. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). (2018) 7342–7349
18. Levinson, J., Thrun, S.: Automatic online calibration of cameras and lasers. In: Robotics: Science and Systems (RSS). (2013)
19. Malti, A., Barreto, J.P.: Robust hand-eye calibration for computer aided medical endoscopy. In: 2010 IEEE International Conference on Robotics and Automation (ICRA). (2010) 5543–5549
20. Nguyen, H., Pham, Q.C.: On the covariance of $\boldsymbol{X}$ in $\boldsymbol{AX} = \boldsymbol{XB}$. IEEE Transactions on Robotics **34**(6) (2018) 1651–1658
21. Pachtrachai, K., Vasconcelos, F., Chadebecq, F., Allan, M., Hailes, S., Pawar, V., Stoyanov, D.: Adjoint transformation algorithm for hand–eye calibration with applications in robotic assisted surgery. Annals of Biomedical Engineering **46**(10) (July 2018) 1606–1620
22. Pandey, G., McBride, J.R., Savarese, S., Eustice, R.M.: Automatic extrinsic calibration of vision and lidar by maximizing mutual information. Journal of Field Robotics **32**(5) (2015) 696–722
23. Park, F.C., Martin, B.J.: Robot sensor calibration: solving AX=XB on the euclidean group. IEEE Transactions on Robotics and Automation **10**(5) (1994) 717–721
24. Peršić, J., Marković, I., Petrović, I.: Extrinsic 6DOF calibration of a radar–LiDAR–camera system enhanced by radar cross section estimates evaluation. Robotics and Autonomous Systems **114** (April 2019) 217–230
25. Richardson, A., Strom, J., Olson, E.: Aprilcal: Assisted and repeatable camera calibration. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. (2013) 1814–1821
26. Shiu, Y.C., Ahmad, S.: Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form AX=XB. IEEE Transactions on Robotics and Automation **5**(1) (1989) 16–29
27. Solà, J., Deray, J., Atchuthan, D.: A micro Lie theory for state estimation in robotics. arXiv e-prints (December 2018) arXiv:1812.01537
28. Strobl, K.H., Hirzinger, G.: Optimal hand-eye calibration. In: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). (2006) 4647–4653
29. Tsai, R.Y., Lenz, R.K.: A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. IEEE Transactions on Robotics and Automation **5**(3) (June 1989) 345–358