This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2015.2505690, IEEE Transactions on Industrial Electronics

A Stochastic Global Optimization Algorithm for the Two-Frame Sensor Calibration Problem

Junhyoung Ha, *Member, IEEE,* Donghoon Kang, *Member, IEEE,* and Frank C. Park, *Fellow, IEEE*

***Abstract*—In the two-frame sensor calibration problem, the objective is to find rigid-body homogeneous transformation matrices X, Y that best fit a set of equalities of the form $A_iX = YB_i$, $i = 1, \ldots, N$, where the $\{(A_i, B_i)\}$ are pairs of homogeneous transformations obtained from sensor measurements. The measurements are often subject to varying levels of noise, and the resulting optimization can have numerous local minima that exhibit high sensitivity in the choice of optimization parameters. As a first contribution, we present a fast and numerically robust local optimization algorithm for the two-frame sensor calibration objective function. Using coordinate-invariant differential geometric methods that take into account the matrix Lie group structure of the rigid-body transformations, our local descent method makes use of analytic gradients and Hessians, and a strictly descending fast step-size estimate to achieve significant performance improvements. As a second contribution, we present a two-phase stochastic geometric optimization algorithm for finding a stochastic global minimizer based on our earlier local optimizer. Numerical studies demonstrate the considerably enhanced robustness and efficiency of our algorithm over existing unit quaternion-based methods.**

***Index Terms*—Robot sensor calibration, hand-eye calibration, robot-world calibration, geometric optimization, stochastic global optimization.**

I. Introduction

In this paper we address the following version of the **two-frame sensor calibration** problem: given pairs of homogeneous rigid-body transformation matrices $\{(A_i, B_i)\}$, $i = 1, \ldots, N$, where each A_i and B_i is a 4×4 homogeneous rigid-body transformation matrix belonging to the Special Euclidean group $SE(3)$ of rigid-body motions, find $X, Y \in SE(3)$ that is a best fit to the N matrix equalities $A_iX = YB_i$, $i = 1, \ldots, N$. In its most common form, this problem is framed as a least-squares optimization problem over $X, Y \in SE(3)$, in which the objective function is given by

$$\min_{X,Y \in SE(3)} \sum_i \|A_iX - YB_i\|^2, \tag{1}$$

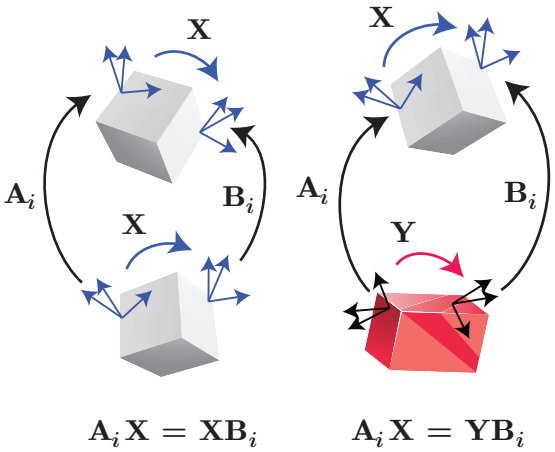
Manuscript received February 20, 2015; revised September 2, 2015; accepted October 15, 2015.

This research was supported in part by the Defense Acquisition Program Administration BMRR Center, SNU-IAMD, BK21+, MI Technology Innovation Program 10048320 and KIST institutional project.

J. Ha is with the Department of Cardiovascular Surgery, Boston Childrens Hospital, Harvard Medical School, Boston, Massachusetts, USA (e-mail: hjhdog1@gmail.com).

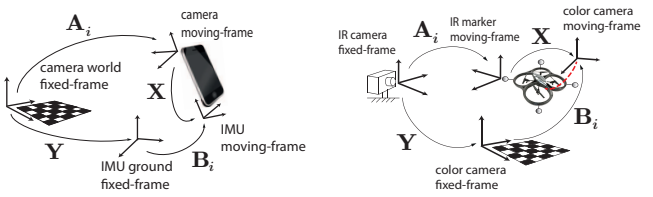
D. Kang is with the Imaging Media Center, Korea Institute of Science and Technology (KIST), Seoul, Korea (e-mail: kimbab.moowoo@gmail.com).

F.C. Park is with the Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul, Korea (e-mail: fcp@snu.ac.kr).



(a) One-frame problem (b) Two-frame problem

Fig. 1. Sensor calibration problems involving (a) one frame and (b) two frames.



(a) Mobile device with a built-in camera and an IMU. (b) UAV equipped with reflective IR markers and a built-in camera.

Fig. 2. Examples of the two-frame calibration problem

where $\|\cdot\|$ denotes the Frobenius matrix norm, i.e., $\|A\| = \sqrt{\text{tr}(AA^T)}$ for a matrix A.

Figure 1 illustrates the general context in which this problem arises. Given two rigid bodies, each with two reference frames attached at distinct locations, the relative displacements between each frame pair are respectively denoted X and Y. Here X and Y are unknown; to determine them, various sensors are used to measure the displacements $A_i, B_i \in SE(3)$ of the frames at various configurations of the two rigid bodies. In the absence of measurement noise, the loop closure equation $A_iX = YB_i$ must be satisfied for all measurements $i = 1, \ldots, N$. If the two rigid bodies (and the locations of the attached reference frames) are identical, then $X = Y$, and the loop closure equation simplifies to $A_iX = XB_i$; this is the one-frame sensor calibration problem.

The most well-known example of two-frame sensor calibration is the simultaneous hand-eye/robot-world calibration problem [1]. Hand-eye calibration alone is a one-frame calibration problem, as is the problem of robot-world calibration.

There is extensive literature on solution methods for these two classical problems, which we describe in detail below. What is worth emphasizing is that with the recent proliferation of devices that integrate multiple sensors of different types, the two-frame calibration problem is being encountered in a variety of different contexts beyond the traditional industrial settings. For example, relative rotations between cameras and inertial measurement units (IMUs) on a mobile device along with their corresponding reference world frames must satisfy equations of the form $A_iX = YB_i \in SO(3)$ as illustrated in Figure 2(a), where $SO(3)$ denotes the rotation group. Another application is aerial vehicle tracking (Figure 2(b)), in which an aerial vehicle equipped with an onboard camera identifies fixed markers in the environment while simultaneously being tracked by a ground-based camera.

The objective function (1) is in general nonconvex, and possesses local minima whose number and properties are strongly influenced by the level of noise in the measurements $\{(A_i, B_i)\}$. Usually this has not been a major concern for existing applications in hand-eye and robot-world calibration, where the measurement noise is relatively small and good initial guesses are available. Some of the more recent applications, however, rely on low-cost and noisy sensors that operate in highly unstructured and dynamic environments [2],[3]. In such cases it can be considerably more beneficial to search further for a "best" local minimizer rather than settling for the first local minimizer encountered, which is dependent on the choice of initial guess.

As stated earlier, there exists extensive literature on solution methods for the hand-eye and robot-world calibration problem. In [1] and [4], unit quaternion representations for rotations are used to develop linear least squares and iterative nonlinear methods for finding a local minimizer; in [5] dual quaternion representations for homogeneous transformations are used to find the rotation and translation components simultaneously, rather than in a decoupled way as in the previous two methods. All of these methods derive in part from earlier work on the hand-eye calibration problem, where various linear least-squares and nonlinear iterative methods for finding local minimizers have also been developed in [5]-[10]. More recently, in [11] a method for finding the globally optimal solution to the one-frame hand-eye calibration problem has been developed, in which the main result is a branch-and-bound search algorithm that ensures convergence to the global optimizer in the limit. Another class of methods includes extrema-seeking algorithms from the uncertain dynamical systems optimal control literature, in which Newton-type iterative algorithms are developed for determining extrema when the objective function is not available in analytic form [12]. These algorithms are formulated in a vector space setting, primarily for optimal feedback controller design applications.

In this paper we present a stochastic geometric optimization algorithm for finding a globally optimal solution to the two-frame calibration problem. The specific contributions are as follows:

- We first develop a fast and numerically robust local optimization algorithm that exploits the matrix Lie group structure of $SE(3)$. Our optimization algorithm evolves directly on the search space $SE(3)\times SE(3)$, and does not require local coordinates (e.g., Euler angles for rotations); solution constraints are exactly satisfied at each iteration, and the performance of the algorithm is invariant with respect to the choice of inertial reference frame.
  Whereas computational efficiency was not an issue in previous works on hand-eye and robot-world calibration, since calibration was performed infrequently and also offline, in the current work the local optimization algorithm is repeatedly called as part of a stochastic global optimization algorithm, making computational efficiency of paramount importance. Our performance improvements are achieved in part by the availability of exact **analytic** gradients and Hessians of the objective function. As is well-known, the speed and accuracy of nonlinear optimization algorithms are greatly enhanced by the availability of analytic gradients (and for Newton's method, Hessians)—finite difference approximations of gradients (which is what general optimization algorithms evaluate when analytic gradients are not provided) can lead to numerical error accumulations that result in instability and poor convergence.
  We derive two specific local search methods, steepest descent and Newton's method. For iterative descent algorithms, as a matter of practice it is usually far more beneficial to devote greater computational resources to finding a better search direction rather than in determining the optimal stepsize. Toward this end, we also derive an analytic formula for rapidly computing a strictly descending **stepsize estimate**. Also as a by-product of our analysis, we characterize in a rigorous way the existence and uniqueness of solutions to the ideal case (i.e., no noise in the measurements).
- We develop a two-phase stochastic global optimization algorithm, consisting of a global and local search phase, that extends the method of [13], [14] to the matrix Lie group $SE(3) \times SE(3)$ in a geometric way (i.e., invariant with respect to choice of local coordinates, and also to left and right translations). The local search phase relies upon the local algorithm developed earlier. In the stochastic global search phase, we reduce the original unbounded search space $SE(3) \times SE(3)$ to the **compact space** $SO(3) \times SO(3)$, and generate random samples on this space in a coordinate-invariant way. An optimal Bayesian stopping criterion is used to terminate the algorithm, with probabilistic confidence levels obtained for the resulting solution to be a global optimizer. Through both numerical and actual hardware experiments, we demonstrate the advantages of our global optimization algorithm over existing local quaternion-based approaches.

The paper is organized as follows. After formulating the problem and characterizing the existence and uniqueness of solutions in Section II, we present the local least-squares iterative algorithm in Section III. A two-phase stochastic global optimization algorithm is described in Section IV. Experimental results involving numerical case studies are presented in Section V.

II. Problem Formulation and Existence of Solutions

A. Mathematical Preliminaries

We first recall some facts and formulae involving the rotation group $SO(3)$ and the special Euclidean group $SE(3)$. Elements of $SO(3)$ are given by the 3×3 real matrices R satisfying $R^T R = I$ and $\det R = 1$. The group $SE(3)$ is defined to be

$$SE(3) = \left\{ \left[\begin{array}{cc} R & p \\ 0 & 1 \end{array} \right] \middle| \; R \in SO(3), \; p \in \mathbb{R}^3 \right\}.$$

$SO(3)$ is a matrix Lie group, and its associated Lie algebra, denoted $so(3)$, is given by the set of 3×3 real skew-symmetric matrices with the matrix commutator as Lie bracket. We adopt the following notation: given $r \in \mathbb{R}^3$, its 3×3 skew-symmetric representation is denoted $[r]$, where

$$[r] = \left[\begin{array}{ccc} 0 & -r_3 & r_2 \\ r_3 & 0 & -r_1 \\ -r_2 & r_1 & 0 \end{array} \right].$$

The matrix exponential map $\exp : so(3) \to SO(3)$ provides a useful local coordinate parametrization for $SO(3)$. If r is expressed as $r = \omega\theta$, where $\theta = \|r\|$ and $\omega = r/\|r\|$ is a unit vector, then the exponential formula is given by

$$e^{[\omega]\theta} = I + \sin\theta\,[\omega] + (1 - \cos\theta)\,[\omega]^2.$$

The inverse of the exponential, or logarithm, is also defined in closed-form as follows: for any $R \in SO(3)$ such that $Tr(R) \neq -1$,

$$\log R = \frac{\theta}{2\sin\theta}(R - R^T),$$

where θ satisfies $1 + 2\cos\theta = Tr(R)$. Restricting θ to the range $[0, \pi]$ defines a unique logarithm for all $R \in SO(3)$ that satisfy $Tr(R) \neq -1$. In the event that $Tr(R) = -1$, the logarithm $[\omega] = \log R$ has two antipodal solutions $\pm\omega$ which are determined from the relation $R = I + (2/\pi^2)[\omega]^2$. A straightforward calculation also establishes that in the general case, $\|\log R\| = \theta$, where $\|\cdot\|$ denotes the Frobenius norm.

B. Existence and Uniqueness of Solutions to $AX = YB$

The equation $AX = YB$ on $SE(3)$ can be expressed as the pair of equations

$$\begin{aligned} R_A R_X &= R_Y R_B \\ R_A p_X + p_A &= R_Y p_B + p_Y, \end{aligned}$$

where $R_A, R_B \in SO(3)$ and $p_A, p_B \in \mathbb{R}^3$ are assumed to be known, and $R_X, R_Y \in SO(3)$ and $p_X, p_Y \in \mathbb{R}^3$ are unknown. In [1], it is shown that any solution to the above equations must satisfy a corresponding set of linear equations, but the existence and uniqueness of solutions is not precisely characterized. In this section, we make mathematically precise statements about the existence and uniqueness of solutions (X, Y) to $AX = YB$; detailed proofs of the ensuing propositions are provided in Appendix A.

1) Existence and Uniqueness of Solutions to $R_A R_X = R_Y R_B$ on $SO(3)$: Given $R_A, R_B \in SO(3)$, we now consider the existence and uniqueness of solutions to the equation $R_A R_X = R_Y R_B$, where $R_X, R_Y \in SO(3)$ are unknown.

Proposition 1. *Given two rotation matrix pairs $\{(R_{A_1}, R_{B_1}), (R_{A_2}, R_{B_2})\}$, consider the pair of equations*

$$R_{A_1} R_X = R_Y R_{B_1}, \quad R_{A_2} R_X = R_Y R_{B_2}, \tag{2}$$

where $R_X, R_Y \in SO(3)$ are unknown. Define $\alpha, \beta \in \mathbb{R}^3$ as follows:

$$\begin{aligned} [\alpha] &= \log(R_{A_1} R_{A_2}^T) \\ [\beta] &= \log(R_{B_2}^T R_{B_1}). \end{aligned}$$

If $\|\alpha\| = \|\beta\|$, then (2) has a one-parameter family of solutions $(R_X, R_Y)_t$ given by

$$R_X = R_{A_2}^T e^{[\alpha]t} \Theta_p = R_{A_2}^T \Theta_p e^{[\beta]t} \tag{3}$$

$$R_Y = e^{[\alpha]t} \Theta_p R_{B_2}^T = \Theta_p e^{[\beta]t} R_{B_2}^T, \tag{4}$$

where $t \in [0, 2\pi]$, and $\Theta_p \in SO(3)$ is any particular solution to $\Theta\beta = \alpha$.

Proposition 2. *Given the three rotation pairs (R_{A_i}, R_{B_i}), $i = 1, 2, 3$, define*

$$\begin{aligned} [\alpha_{jk}] &= \log(R_{A_j} R_{A_k}^T) \\ [\beta_{jk}] &= \log(R_{B_k}^T R_{B_j}), \end{aligned}$$

and the 3×3 matrices

$$\begin{aligned} \Phi &= \left[\begin{array}{ccc} \alpha_{21} & \alpha_{31} & \alpha_{21} \times \alpha_{31} \end{array} \right] \\ \Psi &= \left[\begin{array}{ccc} \beta_{21} & \beta_{31} & \beta_{21} \times \beta_{31} \end{array} \right]. \end{aligned}$$

There exists a unique solution pair (R_X, R_Y) to the set of equations

$$R_{A_i} R_X = R_Y R_{B_i} \;, \; i = 1, 2, 3,$$

given by

$$R_X = R_{A_1}^T \Phi \Psi^{-1}, \quad R_Y = \Phi \Psi^{-1} R_{B_1}, \tag{5}$$

if and only if both matrices Φ and Ψ are nonsingular and $\Phi^T \Phi = \Psi^T \Psi$.

2) Existence and Uniqueness of Solutions (X, Y) to $AX = YB$ on $SE(3)$:

Proposition 3. *Given two rigid-body transformation matrix pairs $\{(A_1, B_1), (A_2, B_2)\}$, consider the pair of equations*

$$A_1 X = Y B_1, \quad A_2 X = Y B_2, \tag{6}$$

where $X, Y \in SE(3)$ are unknown. Define $\alpha, \beta \in \mathbb{R}^3$ and $\mathbb{A} \in \mathbb{R}^{6\times 6}$ as follows:

$$\begin{aligned} [\alpha] &= \log(R_{A_1} R_{A_2}^T) \\ [\beta] &= \log(R_{B_2}^T R_{B_1}). \\ \mathbb{A} &= \left[\begin{array}{cc} R_{A_1} & -I \\ R_{A_2} & -I \end{array} \right] \end{aligned}$$

If $\|\alpha\| = \|\beta\|$, then (6) has a $(7 - rank(\mathbb{A}))$-parameter family of solutions (X, Y).

Proposition 4. *Given the three rigid-body motion pairs (A_i, B_i), $i = 1, 2, 3$, and corresponding rotation pairs (R_{A_i}, R_{B_i}) and translation pairs (p_{A_i}, p_{B_i}), define the same Φ and Ψ as in Proposition 2 and*

$$\mathbb{A} = \begin{bmatrix} R_{A_1} & -I \\ R_{A_2} & -I \\ R_{A_3} & -I \end{bmatrix} \in \mathbb{R}^{9\times 6}$$

$$\eta = \begin{bmatrix} \Phi\Psi^{-1}R_{B_1}p_{B_1} - p_{A_1} \\ \Phi\Psi^{-1}R_{B_2}p_{B_2} - p_{A_2} \\ \Phi\Psi^{-1}R_{B_3}p_{B_3} - p_{A_3} \end{bmatrix} \in \mathbb{R}^9.$$

Then there exists a unique solution of (X, Y) if and only if both Φ and Ψ are nonsingular with $\Phi^T\Phi = \Psi^T\Psi$, $rank(\mathbb{A}) = 6$, and η is linearly dependent on the column vectors of $\mathbb{A}$.

Propositions 1 and 2 describe the existence and uniqueness of solutions when noise-free measurements on the rotation group $SO(3)$ are available. Propositions 3 and 4 are more general versions of the existence and uniqueness conditions for the solution on the Euclidean group $SE(3)$ under the assumption of noise-free measurements.

III. Local Least Squares Minimization

A. Least Squares Objective Function

In practice there will not exist an exact solution to $A_iX = YB_i$ since the measurements A_i and B_i are corrupted by sensor noise. In this section we consider the minimization of the following least squares criterion:

$$\frac{1}{2}\sum_{i=1}^{N} \|A_iX - YB_i\|^2$$

where several choices of $\|\cdot\|$ are available. Here, we define $\|\cdot\|^2$ as $\|P - Q\|^2 = \|R_P - R_Q\|_F^2 + \zeta\|p_P - p_Q\|^2$ where $R_P, R_Q \in SO(3)$ and $p_P, p_Q \in \mathbb{R}^3$ are the rotations and the translations of $P, Q \in SE(3)$ respectively. Here, $\|\cdot\|_F$ denotes the Frobenius norm and $\zeta \in \mathbb{R}_+$ is a weighting factor for the translation error. The least squares criterion becomes as follows:

$$\frac{1}{2}\sum_{i=1}^{N} \|A_iX - YB_i\|^2 = \frac{1}{2}\sum_{i=1}^{N}(6 - 2Tr(R_X^T R_{A_i}^T R_Y R_{B_i}) + \zeta\|R_{A_i}p_X + p_{A_i} - R_Y p_{B_i} - p_Y\|^2). \quad (7)$$

The above Equation (7) can be established using the general matrix trace identity $Tr(ABC) = Tr(CAB) = Tr(BCA)$ for matrices A, B, C. In general this problem is not a convex nor even a quasi-convex problem, typically possessing multiple local minima. Applying stochastic global optimization techniques to this problem directly is problematic because of the infinite volume of the search space $SE(3) \times SE(3)$ resulting from the unbounded space of pure translations in $SE(3)$.

We now show that the function can be easily reduced to a quadratic function on $SO(3) \times SO(3)$, which from the compactness of $SO(3)$ has a search space of bounded volume. To see why, first note that

$$\min_{X,Y\in SE(3)} \frac{1}{2}\sum_{i=1}^{N} \|A_iX - YB_i\|^2 = \min_{R_X,R_Y\in SO(3)} \left(\min_{p_X,p_Y\in\mathbb{R}^3} \frac{1}{2}\sum_{i=1}^{N} \|A_iX - YB_i\|^2 \right).$$

Since $\frac{1}{2}\sum_{i=1}^{N} \|A_iX - YB_i\|^2$ is a least squares criterion, it is a convex quadratic function with respect to p_X and p_Y in which closed-form solutions to the sub-minimization $\min_{p_X,p_Y\in\mathbb{R}^3} \frac{1}{2}\sum_{i=1}^{N} \|A_iX - YB_i\|^2$ are available. Note that the closed-form solutions $p_X^*(R_X, R_Y)$ and $p_Y^*(R_X, R_Y)$ are functions of R_X and R_Y. By substituting these into (7), we define

$$\begin{aligned} J(R_X, R_Y) &= \min_{p_X,p_Y\in\mathbb{R}^3} \frac{1}{2}\sum_{i=1}^{N} \|A_iX - YB_i\|^2 \\ &= \frac{1}{2}\sum_{i=1}^{N} \big(6 - 2Tr(R_X^T R_{A_i}^T R_Y R_{B_i}) \\ &\quad + \zeta\|R_{A_i}p_X^* + p_{A_i} - R_Y p_{B_i} - p_Y^*\|^2\big). \end{aligned}$$

The entire problem now reduces to

$$\min_{R_X,R_Y\in SO(3)} J(R_X, R_Y), \quad (8)$$

which is a minimization on $SO(3) \times SO(3)$. We finally derive the following expression for $\hat{J}(R_X, R_Y)$:

$$\begin{aligned} J(R_X, R_Y) = \frac{1}{2}\sum_{i=1}^{18} \lambda_i \left(Tr(P_iR_X) + Tr(Q_iR_Y)\right)^2 \\ + Tr(P_0R_X) + Tr(Q_0R_Y) + c, \end{aligned} \quad (9)$$

where $P_i, Q_i \in \mathbb{R}^{3\times 3}$ and $\lambda_i, c \in \mathbb{R}$ are obtained by the eigenvalue analysis of the original function (see Appendix B). Note that the minimization (8) is not a cyclic minimization (namely, it does not optimize over the variables in a cyclical fashion, by optimizing over one variable while keeping the remaining fixed, and iterating this procedure over all variables). We remark that the time complexity of the function evaluation of $\hat{J}(R_X, R_Y)$ reduces from $O(N)$ to $O(1)$. This leads to gradient and Hessian evaluation algorithms with complexity $O(1)$ as well, greatly enhancing our stochastic global optimization algorithm by reducing the function evaluation times for a large number of sample points.

B. Local Geometric Minimization

The objective function (9) is minimized through generalizations of the steepest descent algorithm and Newton's method to the search space $SO(3) \times SO(3)$. It is instructive to first review the standard vector space versions of these algorithms. Given a twice-differentiable objective function $J(x)$, $x \in \mathbb{R}^n$, the steepest descent algorithm applies the following iteration until a suitable convergence criterion is met:

$$x_{k+1} = x_k + m_k d_k,$$

where the search direction $d_k \in \mathbb{R}^n$ is taken to be the gradient of $J(x)$ at x_k, i.e.,

$$d = -\nabla J(x_k),$$

and the stepsize m_k is a positive scalar, typically chosen so as to minimize J along the search direction:

$$m_k = \arg\min_{m \in \mathbb{R}} J(x + m_k d_k). \tag{10}$$

In Newton's method, the search direction is taken to be

$$d = -[\nabla^2 J(x_k)]^{-1} \nabla J(x_k),$$

where $\nabla^2 J(x_k) \in \mathbb{R}^{n \times n}$ denotes the Hessian of J at x_k (recall that the gradient and Hessian correspond to the first- and second-order terms in the Taylor series expansion of $J(x)$:

$$J(x+h) = J(x) + \nabla J(x) h + \frac{1}{2} h^T \nabla^2 J(x) h + \dots \ .$$

The following sections present a geometric generalization of these standard vector space optimization methods. In a geometric space like $SO(3) \times SO(3)$, the straight lines in the stepsize computation (10) are now replaced by minimal geodesics (which are difficult to compute in the case of general Riemannian manifolds, but fairly straightforward in the case of compact matrix Lie groups like the rotation group). Not only exact analytic gradients and Hessians that enhance the performance and convergence of local optimization algorithms are derived in Section III-B1, but also "strictly descending stepsize estimate" that enables one to use a traditional stochastic optimization algorithm in our geometric search space is presented in Section III-B2. The matrix Lie group structure of $SO(3)$ is then exploited to generate a geodesic curve along the direction and update the state on the curve with strictly descending stepsize estimate.

1) Gradient and Hessian Formulae: Because the objective function (9) is defined on $SO(3) \times SO(3)$, which as is well-known is not a vector space, appropriate notions of the gradient and Hessian are needed. We first expand (R_X, R_Y) about (R_{X_k}, R_{Y_k}) via the matrix exponential as follows:

$$\begin{aligned} R_X &= R_{X_k}(I + [\omega_{R_X}] + \frac{1}{2}[\omega_{R_X}]^2 + \dots) \\ R_Y &= R_{Y_k}(I + [\omega_{R_Y}] + \frac{1}{2}[\omega_{R_Y}]^2 + \dots). \end{aligned}$$

$\hat{J}(R_X, R_Y)$ can now expanded up to second order and the gradient and Hessian can be obtained by differentiating it with respect to ω_{R_X} and ω_{R_Y}; the constant and first-order terms then correspond to the gradient and Hessian, respectively. The specific derivations and formulae are given in Appendix C.

2) Strictly Descending Stepsize: We first expand (R_X, R_Y) about (R_{X_k}, R_{Y_k}) via the exponential mapping as follows:

$$\begin{aligned} R_X &= R_{X_k} e^{[\omega_{R_X}]t} \\ &= R_{X_k}(I + \tfrac{\sin\|\omega_{R_X}\|t}{\|\omega_{R_X}\|}[\omega_{R_X}] + \tfrac{1-\cos\|\omega_{R_X}\|t}{\|\omega_{R_X}\|^2}[\omega_{R_X}]^2) \\ R_Y &= R_{Y_k} e^{[\omega_{R_Y}]t} \\ &= R_{Y_k}(I + \tfrac{\sin\|\omega_{R_Y}\|t}{\|\omega_{R_Y}\|}[\omega_{R_Y}] + \tfrac{1-\cos\|\omega_{R_Y}\|t}{\|\omega_{R_Y}\|^2}[\omega_{R_Y}]^2) \end{aligned}$$

where $t \in \mathbb{R}$ is the stepsize variable and (ω_X, ω_Y) is the given search direction. The line search procedure is then given by

$$t^* = \arg\min_{t \in \mathbb{R}} \phi(t), \tag{11}$$

where $\phi(t) = J(R_{X_k} e^{[\omega_{R_X}]t}, R_{Y_k} e^{[\omega_{R_Y}]t}) = c_0 + c_1 \sin t_X + c_2 \cos t_X + c_3 \sin t_Y + c_4 \cos t_Y + c_5 \sin t_X \sin t_Y + c_6 \sin t_X \cos t_Y + c_7 \cos t_X \sin t_Y + c_8 \cos t_X \cos t_Y$ with $\hat{\omega}_{R_X} = \omega_{R_X}/\|\omega_{R_X}\|$, $\hat{\omega}_{R_Y} = \omega_{R_Y}/\|\omega_{R_Y}\|$, $t_X = \|\omega_{R_X}\|t$ and $t_Y = \|\omega_{R_Y}\|t$. The coefficients c_i can be obtained by straightforward calculation. When replacing $\tan t_X$ and $\tan t_Y$ by x and y, respectively so that $\cos t_X = \frac{1-x^2}{x^2+1}$, $\cos t_Y = \frac{1-y^2}{y^2+1}$, $\sin t_X = \frac{2x}{x^2+1}$, $\sin t_Y = \frac{2y}{y^2+1}$, the above line search reduces to a multivariate polynomial root-finding problem for which exact solutions are available [15].

The use of the exact stepsize formula reduces the number of iterations required for convergence. However, there are several associated drawbacks from both the theoretical and computational perspective. First, the function value along the geodesic between (R_{X_k}, R_{Y_k}) and $(R_{X_{k+1}}, R_{Y_{k+1}})$ may not be strictly descending, and the updated point may end up in a different region of attraction. These features can violate the assumptions of the optimal Bayesian stopping rules which are essential in typical stochastic global optimization algorithms [13]. Bayesian stopping rules are derived under the assumption that the entire path generated from the local search is contained in one region of attraction, and the function value over each path segment generated in each update of the local search is monotonically decreasing. Second, the computational costs involved in the calculation of the exact stepsize can actually increase the total computation time. It is widely accepted in optimization that as a matter of practice, it is far better to spend time and resources in computing the search direction and to calculate the stepsize in a rapid manner so that they ensure the objective function decreases.

For these reasons, we present a stepsize estimate that is computationally more efficient and always guarantees a strictly descending function value. The estimate is based on the fact that for a twice-differentiable function $f(t)$, if $f'(0) > 0$ and $|f''(t)| \leq c$ for all t, then $f'(t) \geq 0$ in the interval $-\frac{f'(0)}{c} \leq t \leq \frac{f'(0)}{c}$. Conversely, if $f'(0) < 0$ and $|f''(t)| \leq c$ for all t, then $f'(t) \leq 0$ in the interval $\frac{f'(0)}{c} \leq t \leq -\frac{f'(0)}{c}$. From the above equation, the stepsize estimate $t^* = -\frac{f'(0)}{c}$ ensures that $f(t^*) < f(0)$.

Based on the derivations given in Appendix D, our stepsize estimate is

$$t^* = -\frac{\phi'(0)}{c} \tag{12}$$

where

$$\begin{aligned} &\phi'(0) = \\ &\textstyle\sum_{i=1}^{18} \lambda_i Tr(P_i R_{X_k} + Q_i R_{Y_k}) \\ &Tr(P_i R_{X_k}[\omega_{R_X}] + Q_i R_{Y_k}[\omega_{R_Y}]) \\ &+ Tr(P_0 R_{X_k}[\omega_{R_X}] + Q_0 R_{Y_k}[\omega_{R_Y}]) \end{aligned} \tag{13}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2015.2505690, IEEE Transactions on Industrial Electronics

and

$$c = |\lambda|_{max}\left(\|[\omega_{R_X}]\|^2 + \|[\omega_{R_Y}]\|^2\right) + \sqrt{6}|\lambda|_{max}\sqrt{\|[\omega_{R_X}]^2\|^2 + \|[\omega_{R_Y}]^2\|^2} + \sqrt{3}\left(\|P_0 R_{X_k}[\omega_{R_X}]^2\| + \|Q_0 Y_k[\omega_{R_Y}]^2\|\right). \quad (14)$$

(R_X, R_Y) is then updated by

$$R_{X_{k+1}} = R_{X_k} e^{[\omega_{R_X}]t^*} \quad (15)$$

$$R_{Y_{k+1}} = R_{Y_k} e^{[\omega_{R_Y}]t^*}. \quad (16)$$

C. Determining Initial Guess (R_{X_0}, R_{Y_0})

From the rotational parts $\{(R_{A_i}, R_{B_i})\}_{i=1,\ldots,N}$ of pose data pairs $\{(A_i, B_i)\}_{i=1,\ldots,N}$, one can choose two independent equations with any $k \in [1, N]$, e.g., $R_{A_k} R_X = R_Y R_{B_k}$ and $R_{A_i} R_X = R_Y R_{B_i}$ $(i \neq k)$. From the above two equations, R_Y can be eliminated by $R_{A_k}^T R_{A_i} R_X = R_X R_{B_k}^T R_{B_i}$. Let us denote $[\alpha_{1i}] \triangleq \log(R_{A_k}^T R_{A_i})$ and $[\beta_{1i}] \triangleq \log(R_{B_1}^T R_{B_i})$, so that the above reduces to $R_X \beta_{1i} = \alpha_{1i}$. In the literature [6], a closed-form solution for the minimum of $\sum \|R_{A_i} R_X - R_X R_{B_i}\|^2$ is given, in which

$$R_X = (U^T U)^{-1/2} U^T, \quad (17)$$

where $U = \sum_i \beta_{1i} \alpha_{1i}^T$. In a similar fashion, one can eliminate R_X rather than R_Y to obtain $R_{A_k} R_{A_i}^T R_Y = R_Y R_{B_k} R_{B_i}^T$. Let us define $\tilde{\beta}_{1i}$ and $\tilde{\alpha}_{1i}$ as $[\tilde{\alpha}_{1i}] \triangleq \log(R_{A_k} R_{A_i}^T)$ and $[\tilde{\beta}_{1i}] \triangleq \log(R_{B_k} R_{B_i}^T)$. Applying the same closed-form solution, R_Y can be obtained by

$$R_Y = (V^T V)^{-1/2} V^T, \quad (18)$$

where $V = \sum_i \tilde{\beta}_{1i} \tilde{\alpha}_{1i}^T$. In this way one can obtain a reasonable set of initial values (R_{X_0}, R_{Y_0}) for (R_X, R_Y).

D. Summary of Local Search Algorithm

The proposed local search algorithm is summarized in Table I. Like all descent algorithms, our algorithm consists of direction finding and line searching step. For the geometric steepest descent method the search direction is given by

$$\begin{bmatrix} \omega_{R_X} \\ \omega_{R_Y} \end{bmatrix} = -\nabla J, \quad (19)$$

while for Newton's method

$$\begin{bmatrix} \omega_{R_X} \\ \omega_{R_Y} \end{bmatrix} = -[\nabla^2 J]^{-1} \nabla J, \quad (20)$$

where ∇J and $\nabla^2 J$ are as given in Equations (25) and (26) in Appendix C.

IV. Stochastic Global Optimization

In this section we propose a two-phase stochastic global optimization method for our nonlinear objective function (8). There are many variations of this method as described in [13], [14] but the main features are the same:

1) Generate uniform samples on the search space S.
2) Determine whether or not to apply local search to each sample.
3) Apply local search to samples selected in 2.
4) Add newly discovered local minimizer x^* to a set of local minimizers $\mathbb{X}^*$ (which is initially empty) and assign samples to their minimizers.
5) Check the optimal Bayesian stopping criterion.
6) Stop or go back to 1.

TABLE I
Local Search Algorithm

	Algorithm
1	**Initialization**
	Set (R_{X_0}, R_{Y_0}) using Equations (17),(18).
	Set $k = 0$.
2	**Set search direction**
	Find ω_{R_X} and ω_{R_Y} :
	Equation (19) is used for steepest descent, while
	Equation (20) is used for Newton's method.
3	**Update**
	Compute stepsize t^* using Equations (12)-(14).
	Find $(R_{X_{k+1}}, R_{Y_{k+1}})$ using Equations (15),(16).
4	**Check Local Convergence**
	If local convergence criterion is satisfied
	break and return $(R_{X_{k+1}}, R_{Y_{k+1}})$
	Else
	$k \leftarrow k + 1$
	go to **Step 2**

A. Uniform Random Sampling on $SO(3)$

There are several ways to generate a random sequence of rotations uniformly. The main difficulties are in the choice of a convenient parametrization for $SO(3)$, and sampling from the parameter space in a way such that the resulting samples on $SO(3)$ are uniform [16]. The traditional way of performing uniform random sampling on $SO(3)$ is the subgroup algorithm presented in [17]. The group of orthogonal matrices $O(n)$ contains $O(n-1)$ as its subgroup. By randomly sampling an element of $O(n-1)$ and coset representatives for $O(n-1)$ in $O(n)$, it generates a uniform random sample of $O(n)$.

B. Resampling for Local Search

After generating the random samples, there is a resampling step before applying the local search algorithm in Table I to each sample. There exist many variations of the stochastic global optimization with their own resampling strategies. Most of the variations focus on reducing the number of attempts at local search. For example, clustering methods evaluate function values of samples and the distances between samples to avoid unnecessary multiple attempts at local search in the same region of attraction. They assign some samples to their minimizers without applying local search. However, though they can save computation time, the quality of the solution is generally not dependent on the selection of the variation. For this reason we do not introduce any specific variation here.

C. Optimal Bayesian Stopping Rules

Two possible stopping rules are as follows:
(i) Using the posterior expectation of the number of local minima:

$$\frac{w(N-1)}{N-w-2} < w + \epsilon. \quad (21)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2015.2505690, IEEE Transactions on Industrial Electronics

(ii) Using the posterior expected relative size of the non-observed regions of attraction:

$$\frac{w(w+1)}{N(N-1)} < \delta. \tag{22}$$

Here, w is the number of local minima discovered and N is the number of samples assigned to the minimizers. ϵ and δ are the stopping criteria. Note that N is not necessarily the same as the total number of attempts at local search. Some samples can be assigned to a nearby minimizer by checking the location and function value according to the resampling strategy.

To make reliable use of (21) and (22), one should ensure that the search space S is finite and the local search procedure is strictly descending and completely contained in S. We have already resolved the finite search space issue by reducing the search space to $SO(3) \times SO(3)$. The strictly descending step-size is also addressed in the previous section with associated geometric local search algorithm.

D. Summary of Stochastic Global Optimization Algorithm

A stochastic global optimization algorithm using the local search algorithm is presented in Table II.

TABLE II
STOCHASTIC GLOBAL OPTIMIZATION ALGORITHM

Stochastic Global Optimization Algorithm	
1	**Initialization**
	Set X^* to empty.
	Set $N = 0$.
2	**Uniform Random Sampling on** $SO(3)$
	Generate n random samples of $(R_X, R_Y) \in SO(3) \times SO(3)$
3	**Local Search**
	Apply local search described in Table I to the samples generated in Step **2**. (*Note: fewer than n local search calls are required if the reduction schemes presented in [13], [14] are applied.*)
	Assign k samples to corresponding local minimizer and insert newly discovered minimizer (R_X^*, R_Y^*) into X^*. (k *and* n *depend on choice of reduction scheme.*)
4	**Check Convergence**
	$N \leftarrow N + k$
	Check stopping rules (21) and (22).
	If convergence criterion is satisfied
	break and return best minimizer (R_X^*, R_Y^*) in $\mathbb{X}^*$
	Else
	go to **Step 2**

V. EXPERIMENTS

In this section, we compare the performances of our geometric algorithms against the state-of-the-art local unit quaternion-based nonlinear optimization [4] using synthetic and real data.

A. Synthetic Data

We generate synthetic data to simulate an aerial vehicle with infrared markers and onboard camera in Figure 2(b). Using these synthetic data, we present experimental results for three different optimization methods: (i) stochastic global geometric algorithm, (ii) local geometric algorithm, and (iii) local unit quaternion-based nonlinear optimization.

For the local geometric approach, the initial guess $(R_{X_0}, R_{Y_0}) \in SO(3) \times SO(3)$ can be obtained by using the

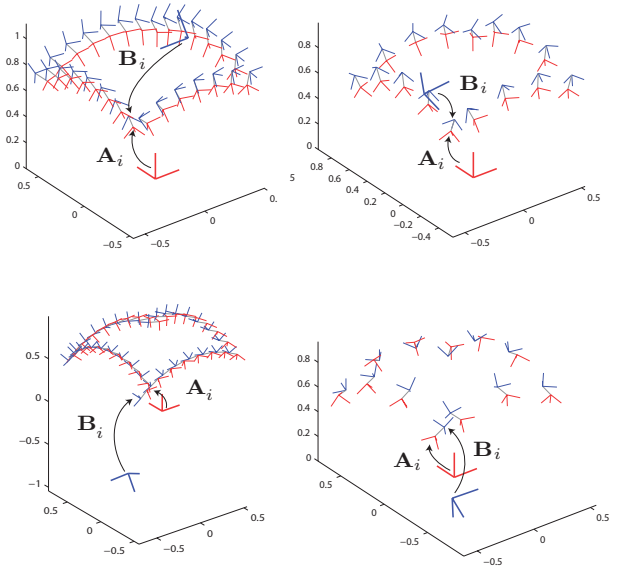


Fig. 3. Synthetically-generated continuous trajectories of pose data (A_i, B_i) to simulate a real aerial vehicle with infrared markers and onboard camera in Figure 2(b).

Equations (17) and (18). In these experiments, we set ζ in Equation (7) to 1.

We randomly set a pair of ground truth rotations $(R_{X,true}, R_{Y,true}) \in SO(3) \times SO(3)$ by using the uniform random sampling method described in the previous section IV-A. Two true translations $(p_{X,true}, p_{Y,true}) \in \mathbb{R}^3 \times \mathbb{R}^3$ are randomly set. Taking into account the relative scale between the aerial vehicle and the workspace, we normalize $(p_{X,true}, p_{Y,true})$ differently as $\|p_{X,true}\| = 0.1$ and $\|p_{Y,true}\| = 1$. With the above pair of rotations and translations, we construct true values of $(X_{true}, Y_{true}) \in SE(3) \times SE(3)$. We now generate a set of N random pose measurements, $\{A_i\}_{i=1,\dots N} \in SE(3)$, where the number of measurements $N \in [12, 40]$ is also randomly generated. To ensure realistic trajectories for the aerial vehicle within the limited workspace, the translation component of each A_i is confined to lie on a sphere with varying radius. The other set of pose measurements $\{B_i\}_{i=1,\dots N}$ are uniquely determined as $B_i = Y_{true}^T A_i X_{true}$.

Noisy data sets are then generated by multiplying the rotations (R_{A_i}, R_{B_i}) of the trajectories by random noisy rotation matrices $(e^{[v_A]}, e^{[v_B]})$ as follows: $R_{A_i} \leftarrow R_{A_i} e^{[v_A]}$ and $R_{B_i} \leftarrow R_{B_i} e^{[v_B]}$ where $v_A, v_B \in \mathbb{R}^3 \sim \mathcal{N}(0, (\gamma\pi)^2 I)$ are Gaussian noises and $\gamma \in \mathbb{R}$ is the noise level. The translation (p_{A_i}, p_{B_i}) are also corrupted by $p_{A_i,x} \leftarrow p_{A_i,x} + p_{A_i,x} w_A$ and $p_{B_i,x} \leftarrow p_{B_i,x} + p_{A_i,x} w_B$ where $w_A, w_B \in \mathbb{R}^3 \sim \mathcal{N}(0, \gamma^2 I)$. The y and z components of translation vectors p_{A_i} and p_{B_i} are corrupted in the same manner with the x component. For each noise level, 1,000 trials of independent experiments are performed to regress the random effect in a single experiment. We perform this experiment 100 times with different true values $(X_{true}, Y_{true}) \in SE(3) \times SE(3)$ and measurements $\{A_i, B_i\}_{i=1,\dots N} \in SE(3) \times SE(3)$ as shown in Figure 3. As a result, we ultimately perform 100 $\times$ 1,000 experiments with

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2015.2505690, IEEE Transactions on Industrial Electronics

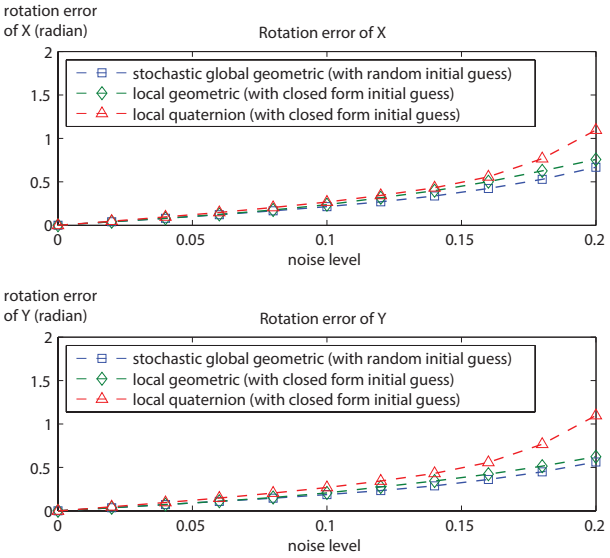


Fig. 4. Geodesic angle errors with increasing noise level.

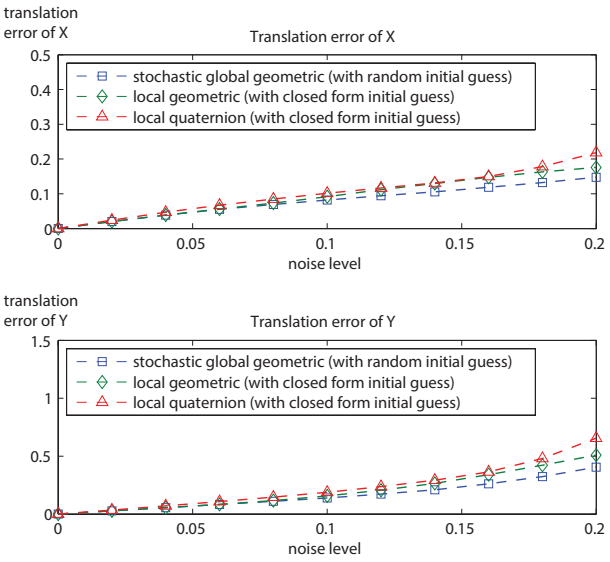


Fig. 5. Translation errors with increasing noise level.

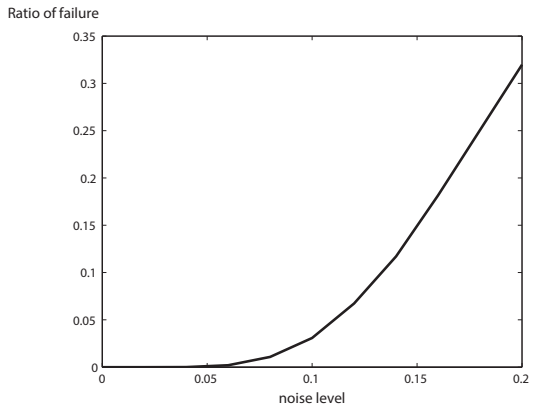


Fig. 6. Ratio of deviations from a stochastic global solution in the local geometric method.

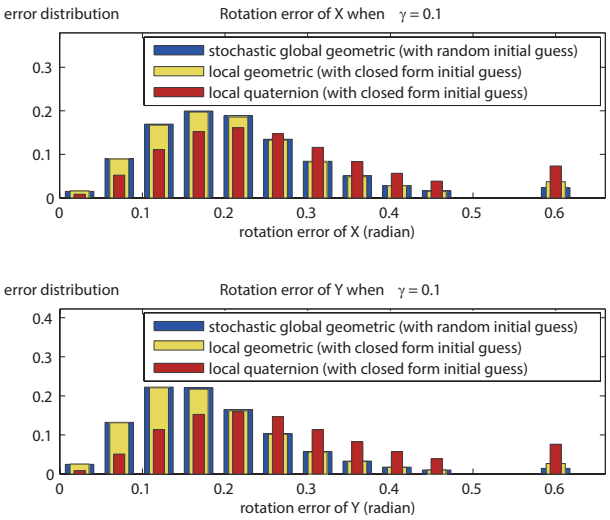


Fig. 7. Distribution of geodesic angle errors when $\gamma = 0.1$.

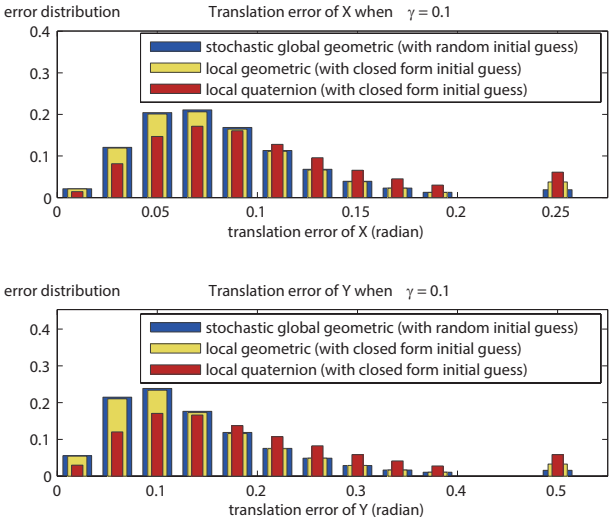


Fig. 8. Distribution of translation errors when $\gamma = 0.1$.

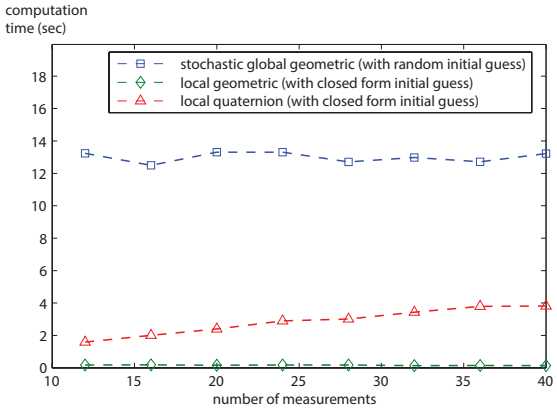


Fig. 9. Average computation times with respect to the number of measurements when $\gamma = 0.1$. 100 experimental trials are averaged for each number of measurements.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2015.2505690, IEEE Transactions on Industrial Electronics

100 sets of independent pose trajectories and 1,000 different noise distributions at each noise level.

We evaluate our stochastic global geometric algorithm and some local algorithms by comparing the accuracy of the estimates (X_{est}, Y_{est}) with the synthetically generated ground truth values (X_{true}, Y_{true}). Figures 4 and 5 show the average rotation errors (in terms of geodesic angle) and translation errors, respectively, for noise levels varying within the range $\gamma \in [0, 0.2]$. With increasing noise levels γ, the stochastic global geometric algorithm shows superior performance over both the local quaternion-based method and the local geometric method. As the noise level increases, the error of the local geometric algorithm deviates from that of the stochastic global geometric algorithm. This can be attributed to the local solution tending to converge to another local minimizer with increasing noise. The ratio of deviations from the stochastic global solution in the local geometric method is shown in Figure 6. Note that the ratio of deviations from the stochastic global solution increases with noise, since large noise levels increase the number of local minima that attract the solution to the problem at each iteration.

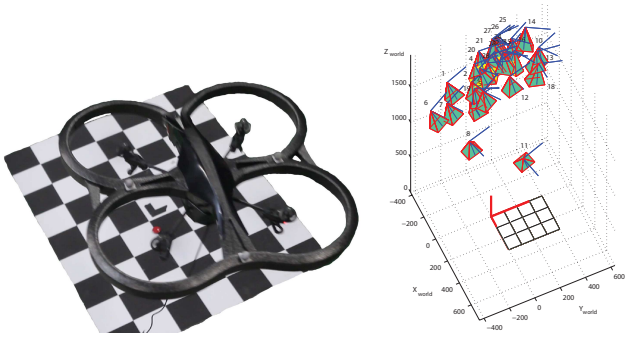
Figures 7 and 8 show error distributions in the rotations and translations over 100 $\times$ 1,000 experimental trials, with the noise level γ set to 0.1. The error distribution for the stochastic global geometric method tends to be much more sharply concentrated at small error values when compared to the results obtained for the local geometric and local quaternion-based methods. For problems in which large noise levels create numerous local minima, our stochastic global optimization method outperforms local optimization methods in terms of accuracy and robustness.

Figure 9 shows average computation times of the algorithms with respect to the number of measurement used in the calibration. 100 experimental trials are averaged for each number of measurements, with the noise level γ set to 0.1. The computation times of our global and local algorithms are invariant to the number of measurements since the time complexity of evaluating the objective function has been reduced to $O(1)$ as stated in Section III-A. The average computation times of our global and local algorithms are 0.16 sec. and 12.99 sec., respectively. On the other hand, the computation time of the quaternion-based method depends on the number of measurements used in the calibration. It increases from 1.59 sec. to 3.81 sec. as the number of measurements increases from 12 to 40.

B. Real Data

In this section, we present experimental results of our calibration algorithm using real data obtained from actual multiple networked infrared (IR) cameras (OptiTrack™) and a color camera on a commercial UAV (Parrot AR.Drone™ 2.0) as shown in Figure 10(a).

We attach an additional wired color camera (Logitech™ webcam C905) onto the AR.Drone to support time synchronization with the OptiTrack IR camera system via a host computer. Whenever the webcam captures planar checkerboard images on the ground, the pose data A_i of the reflective IR markers (illustrated in Figure 2(b)) are recorded. The webcam pose data B_i with respect to the coordinate frame of the checkerboard are easily computed by using a standard camera calibration toolbox. Figure 10(b) shows the trajectory of the webcam on a UAV for this experiment.



(a) Our experimental setup: UAV (b) Trajectory of a camera on the UAV.

Fig. 10. Experiments using a UAV equipped with IR markers and a camera

In general, when dealing with real data, the ground truth pair (X, Y) is unknown. Therefore, we employ a two-step verification procedure for our experiments, involving an estimation step and a validation step [7]. In the estimation step, we randomly select N_0 measurement pairs from a total of N_t measurement pairs (A_i, B_i), $(i = 1 \dots N_t)$. In each trial of random selection of measurement pairs, we compute the optimal estimate (X_{est}, Y_{est}) using only a randomly selected number N_0 of measurement pairs. For the validation step, we compute the estimation error by using the remaining $N_r = (N_t - N_0)$ data pairs excluding the previous N_0 randomly selected data pairs. We perform this two-step verification step N_e times.

The geodesic rotation error E_{geod} and the translation error E_t for each two-step verification trial are defined as

$$E_{geod} = \frac{1}{N_r} \sum_{j=1}^{N_r} \| \log(R_{A_j} R_X (R_Y R_{B_j})^{-1}) \|$$

$$E_t = \frac{1}{N_r} \sum_{j=1}^{N_r} \| R_{A_j} p_X + p_{A_j} - R_Y p_{B_j} - p_Y \|.$$

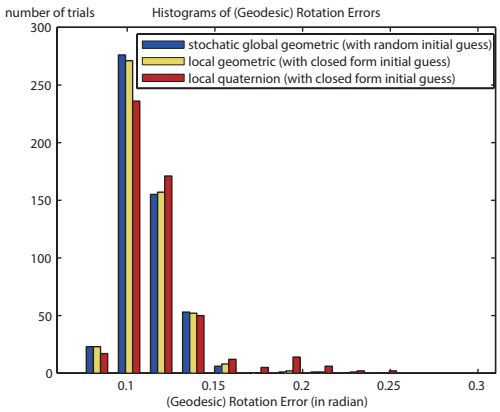


Fig. 11. Histogram of geodesic rotation error, E_{geod}, with 515 number of trials.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2015.2505690, IEEE Transactions on Industrial Electronics

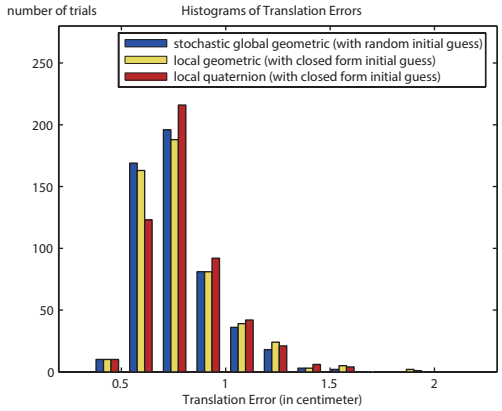


Fig. 12. Histogram of translation error, E_t, with 515 number of trials.

The histograms in Figures 11 and 12 show the rotation and translation estimation errors for several methods. For these experiments, $N_0 = 7$, $N_t = 28$ and $N_e = 515$. Note that our stochastic global geometric method with random initial guess outperforms the existing quaternion-based nonlinear optimization method with a closed-form initial guess as shown in Table III.

TABLE III
AVERAGE ESTIMATION ERRORS, $\frac{1}{N_e}\sum_{k=1}^{N_e} E_{geod}$ (IN RADIAN) AND $\frac{1}{N_e}\sum_{k=1}^{N_e} E_t$ (IN CENTIMETER); $N_0 = 7, N_t = 28, N_e = 515$

	stochastic global geometric	local geometric	local quaternion
$\frac{1}{N_e}\sum_{k=1}^{N_e} E_{geod}$	0.1108	0.1115	0.1167
$\frac{1}{N_e}\sum_{k=1}^{N_e} E_t$	0.7725	0.7902	0.8050

VI. CONCLUSION

This paper has presented a fast and numerically robust local optimization algorithm for the two-frame sensor calibration problem. Using coordinate-invariant differential geometric methods that take into account the matrix Lie group structure of the rigid-body transformations, the proposed local descent method makes use of analytic gradients and Hessians, and a strictly descending fast step-size estimate to achieve significant performance improvements. As a second contribution, a two-phase stochastic geometric optimization algorithm for finding a stochastic global minimizer is derived based on our earlier local optimizer. In both cases the calibration problem is formulated as a minimization of the objective function $\sum_i \|A_i X - Y B_i\|^2$ on the space $SE(3) \times SE(3)$. After deriving necessary and sufficient conditions for the existence and uniqueness of exact solutions (i.e., those cases where $J = 0$ is achievable), we perform both synthetic and real experiments that verify the advantages of our stochastic global optimization method over existing local quaternion-based methods.

APPENDIX A
EXISTENCE AND UNIQUESS OF SOLUTIONS TO $AX = YB$ ON $SE(3)$

A. Proof of Proposition 1: When two pairs of measurements $\{(R_{A_1}, R_{B_1}), (R_{A_2}, R_{B_2})\}$ on $SO(3) \times SO(3)$ are given

Proof: First, note that the general form of the solution to $R_A R_X = R_Y R_B$ is given by $R_X = R_A^T \Theta$, $R_Y = \Theta R_B^T$, where $\Theta = R_A R_X \in SO(3)$ is arbitrary. Now, if (X, Y) is a solution to (2), then there exists some $\Theta_1, \Theta_2 \in SO(3)$ such that $R_X = R_{A_1}^T \Theta_1 = R_{A_2}^T \Theta_2$ and $R_Y = \Theta_1 R_{B_1}^T = \Theta_2 R_{B_2}^T$. Eliminating Θ_1,

$$\Theta_2 R_{B_2}^T R_{B_1} \Theta_2^T = R_{A_1} R_{A_2}^T.$$

Taking the logarithm of both sides, $\Theta_2 [\beta] \Theta_2^T = [\alpha]$, or equivalently, $\Theta_2 \beta = \alpha$, where $[\alpha] = \log(R_{A_1} R_{A_2}^T)$ and $\beta = \log(R_{B_2}^T R_{B_1})$. As detailed in [6], a solution exists only if $\|\alpha\| = \|\beta\|$, and is given by the one-parameter family

$$\Theta_2 = e^{[\alpha]t} \Theta_p = \Theta_p e^{[\beta]t},$$

where $t \in [0, 2\pi]$ and Θ_p is any particular solution to $\Theta \beta = \alpha$. Substituting into $R_X = R_{A_2}^T \Theta_2$ and $R_Y = \Theta_2 R_{B_2}^T$ leads to the main result. ■

B. Proof of Proposition 2: When three pairs of measurements (R_{A_i}, R_{B_i}), $i = 1, 2, 3$ on $SO(3) \times SO(3)$ are given

Proof: We first prove the forward direction. Setting $\Theta_i = R_{A_i} R_X = R_Y R_{B_i}$, $i = 1, 2, 3$, we have four equations associated with $R_X = R_{A_i}^T \Theta_i$ and $R_Y = \Theta_i R_{B_i}^T$:

$$\begin{array}{lll} R_{A_1}^T \Theta_1 & = R_{A_2}^T \Theta_2 & = R_{A_3}^T \Theta_3 \\ \Theta_1 R_{B_1}^T & = \Theta_2 R_{B_2}^T & = \Theta_3 R_{B_3}^T. \end{array}$$

After some manipulation, $\Theta_1 R_{B_1}^T R_{B_j} \Theta_1^T = R_{A_j} R_{A_1}^T, \quad j = 2, 3$. Taking the logarithm of both sides,

$$\Theta_1 \beta_{j1} = \alpha_{j1}, \quad j = 2, 3. \tag{23}$$

Adding the following independent equation $\Theta_1(\beta_{21} \times \beta_{31}) = \alpha_{21} \times \alpha_{31}$ leads to $\Theta_1 \Psi = \Phi$. Under our assumptions about Φ and Ψ the solution Θ_1 is given by $\Theta_1 = \Phi \Psi^{-1}$. It is straightforwardly verified that $\det \Theta_1 = 1$ and $\Theta_1^T \Theta_1 = I$, and therefore a rotation matrix as required.

We now prove the reverse direction. Suppose Φ is singular, in which case $\det \Phi = \|\alpha_{21} \times \alpha_{31}\|^2 = 0$, or equivalently, $\alpha_{21} = c\alpha_{31}$ for some constant $c \in \mathbb{R}$. For a particular solution Θ_1 to Equation (23), $e^{[\alpha_{31}]} \Theta_1$ is also a solution and it follows that the solution is not unique. Likewise, in a similar way it can be shown that Ψ must be nonsingular in order for the solution to be unique. Finally, Suppose $\Phi^T \Phi \neq \Psi^T \Psi$. For nonsingular Ψ, no solution exists since $\Theta_1 = \Phi \Psi^{-1}$ does not satisfy $\Theta_1^T \Theta_1 = I$. The case of singular Ψ has already been proven to have multiple solutions. ■

C. Proof of Proposition 3: When two pairs of measurements $\{(A_1, B_1), (A_2, B_2)\}$ on $SE(3) \times SE(3)$ are given

Proof: It is already shown that there exists a one-parameter family of solutions of rotations (R_X, R_Y) if $\|\alpha\| = \|\beta\|$ in Proposition 1. For any rotation pair solution (R_X, R_Y), (p_X, p_Y) satisfies the following equations:

$$\begin{bmatrix} R_{A_1} & -I \end{bmatrix} \begin{bmatrix} p_X \\ p_Y \end{bmatrix} = R_Y b_1 - a_1$$

$$\begin{bmatrix} R_{A_2} & -I \end{bmatrix} \begin{bmatrix} p_X \\ p_Y \end{bmatrix} = R_Y b_2 - a_2.$$

Since there are six unknowns with 6 linear equations, there exists a unique solution for (p_X, p_Y) if and only if $rank(\mathbb{A}) = 6$.

If $rank(\mathbb{A})$ is less than six, we have additionally $6-rank(\mathbb{A})$ parameters more for the solution of the translation (p_X, p_Y). There already exists a one-parameter family of solutions for the rotation (R_X, R_Y), so that (6) has a $(7 - rank(\mathbb{A}))$-parameter family of solutions (X, Y). ■

D. Proof of Proposition 4: When three pairs of measurements (A_i, B_i) $i = 1, 2, 3$ on $SE(3) \times SE(3)$ are given

Proof: Proposition 2 shows that there exists a unique solution (R_X, R_Y) if and only if both Φ and Ψ are nonsingular and $\Phi^T\Phi = \Psi^T\Psi$, with the solution R_Y given by $R_Y = \Phi\Psi^{-1}B_1$. For the unique rotation solution (R_X, R_Y), (p_X, p_Y) must satisfy the following equations:

$$\left[\begin{array}{cc} R_{A_i} & -I \end{array}\right] \left[\begin{array}{c} p_X \\ p_Y \end{array}\right] = \Phi\Psi^{-1}R_{B_i}p_{B_i} - p_{A_i},\ i = 1, 2, 3$$

or equivalently $\mathbb{A}\left[\begin{array}{c} p_X \\ p_Y \end{array}\right] = \eta$. This is a typical over-constrained linear equation with six unknowns. As is well-known, this linear equation has a unique solution if and only if $\mathbb{A}$ has full column rank and η is in the range space of $\mathbb{A}$. ■

APPENDIX B
DERIVATION OF REDUCED OBJECTIVE FUNCTION

Given $X = \left[\begin{array}{cc} R_X & p_X \\ 0 & 1 \end{array}\right]$ and $Y = \left[\begin{array}{cc} R_Y & p_Y \\ 0 & 1 \end{array}\right]$, define

$$\eta_1 = \left[\begin{array}{c} r_X \\ r_Y \end{array}\right], \eta_2 = \left[\begin{array}{c} p_X \\ p_Y \end{array}\right], \eta = \left[\begin{array}{c} \eta_1 \\ \eta_2 \end{array}\right]$$

where $r_X = vec(R_X)$, $r_Y = vec(R_Y)$. Then (7) can be expressed as a quadratic function of η:

$$L(\eta) = \frac{1}{2}\eta^T \left[\begin{array}{cc} H_{11} & H_{12} \\ H_{12}^T & H_{22} \end{array}\right]\eta + \left[\begin{array}{c} f_1 \\ f_2 \end{array}\right]^T \eta + r$$

where $H_{11} \in \mathbb{R}^{18\times 18}$, $H_{12} \in \mathbb{R}^{18\times 6}$,$H_{22} \in \mathbb{R}^{6\times 6}$, $f_1 \in \mathbb{R}^{18}$, $f_2 \in \mathbb{R}^6$ and $r \in \mathbb{R}$ are obtained from (A_i, B_i) by a straightforward calculation. Since $J(X, Y)$ is a least squares criterion, this is a convex quadratic minimization with respect to η_2 for a given η_1. The minimizer η_2^* and the minimum are obtained as follows:

$$\begin{aligned} \eta_2^* &= -H_{22}^{-1}(H_{12}^T\eta_1 + f_2) \\ \min_{\eta_2\in\mathbb{R}^6} L(\eta) &= \frac{1}{2}\eta_1^T\tilde{H}\eta_1 + \tilde{f}^T\eta_1 + c \end{aligned} \quad (24)$$

where $\tilde{H} = H_{11} - H_{12}H_{22}^{-1}H_{12}^T$, $\tilde{f} = f_1 - H_{12}H_{22}^{-1}f_2$, $c = -\frac{1}{2}f_2^TH_{22}^{-1}f_2 + r$. Here $\tilde{H}$ is a 18×18 symmetric matrix; eigenvalue decomposition of $\tilde{H}$ results in an orthogonal eigenvector matrix X and real-valued eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_{18}$:

$$\begin{aligned} \tilde{H} &= X\Lambda X^T \\ \Lambda &= diag(\lambda_1, \lambda_2, \ldots, \lambda_{18}) \end{aligned}$$

Now we define 3×3 matrices (P_i, Q_i), $i = 0, 1, 2, \ldots, 18$ which satisfy

$$\begin{aligned} X &= \left[\begin{array}{ccc} vec(P_1^T) & \ldots & vec(P_{18}^T) \\ vec(Q_1^T) & \ldots & vec(Q_{18}^T) \end{array}\right] \\ \tilde{f} &= \left[\begin{array}{c} vec(P_0^T) \\ vec(Q_0^T) \end{array}\right]. \end{aligned}$$

Since $Tr(AB) = vec(A^T)vec(B)$ for any matrices $A, B \in \mathbb{R}^{3\times 3}$, (24) is equivalent to

$$\begin{aligned} J(R_X, R_Y) &= \frac{1}{2}\sum_{i=1}^{18}\lambda_i\left(Tr(P_iR_X) + Tr(Q_iR_Y)\right)^2 \\ &\quad + Tr(P_0R_X) + Tr(Q_0R_Y) + c \end{aligned}$$

where $J(R_X, R_Y) = \min_{\eta_2\in\mathbb{R}^6} L(\eta)$.

APPENDIX C
DERIVATIONS OF GRADIENT AND HESSIAN

$\hat{J}(R_X, R_Y)$ can be expanded up to second order as follows:

$$\begin{aligned} J(R_X, R_Y) \approx J(R_{X_k}, R_{Y_k}) &- \frac{1}{2}\sum_{i=1}^{18}\lambda_i\left((Tr([\alpha_i][\omega_{R_X}])^2\right. \\ &+ Tr([\beta_i][\omega_{R_Y}])^2 + 2\gamma_iTr([\alpha_i]_k[\omega_{R_X}]) \\ &+ \gamma_iTr(M_i[\omega_{R_X}]^2) + 2\gamma_iTr([\beta_i][\omega_{R_Y}]) \\ &\left.+\gamma_iTr(N_i[\omega_{R_Y}]^2) + 2Tr([\alpha_i][\omega_{R_X}])Tr([\beta_i][\omega_{R_Y}])\right) \\ &- Tr([\alpha_0][\omega_{R_X}]) - Tr([\beta_0][\omega_{R_Y}]) \\ &- \frac{1}{2}Tr(M_0[\omega_{R_X}]^2) - \frac{1}{2}Tr(N_0[\omega_{R_Y}]^2) \end{aligned}$$

where $[\alpha_i] = \frac{1}{2}(R_{X_k}^TP_i^T - P_iR_{X_k})$, $[\beta_i] = \frac{1}{2}(R_{Y_k}^TQ_i^T - Q_iR_{Y_k})$, $M_i = \frac{1}{2}(R_{X_k}^TP_i^T + P_iR_{X_k})$, $N_i = \frac{1}{2}(R_{Y_k}^TQ_i^T + Q_iR_{Y_k})$, $\gamma_i = Tr(M_i + N_i)$.

Differentiating the above expansion with respect to ω_{R_X} and ω_{R_Y}, the constant and first-order terms then correspond to the gradient and Hessian, respectively. In this regard the following proposition, which follows from a straightforward calculation, is useful.

Proposition 5. *Given $A, B \in \mathbb{R}^{3\times 3}$ and $[w] \in so(3)$,*

$$\begin{aligned} \left[\frac{\partial}{\partial\omega}Tr(A[\omega])\right] &= A^T - A \\ \left[\frac{\partial}{\partial\omega}Tr(B[\omega]^2)\right] &= -(B + B^T)[\omega] - [\omega](B + B^T). \end{aligned}$$

Using the above proposition the gradient and Hessian are then given by

$$\nabla J = \left[\begin{array}{c} s_X \\ s_Y \end{array}\right] \quad (25)$$

$$\nabla^2 J = -\left[\begin{array}{cc} \mathcal{A} & \mathcal{B} \\ \mathcal{B}^T & \mathcal{C} \end{array}\right] \quad (26)$$

where $s_X = 2\sum_{i=1}^{18}\lambda_i\gamma_i\alpha_i + 2\alpha_0$, $s_Y = 2\sum_{i=1}^{18}\lambda_i\gamma_i\beta_i + 2\beta_0$, $\mathcal{A} = \sum_{i=1}^{18}\lambda_i(4\alpha_i\alpha_i^T + \gamma_i(M_i - Tr(M_i)I) + M_0 - Tr(M_0)I$, $\mathcal{B} = 4\sum_{i=1}^{18}\lambda_i\alpha_i\beta_i^T$ and $\mathcal{C} = \sum_{i=1}^{18}\lambda_i(4\beta_i\beta_i^T + \gamma_i(N_i - Tr(N_i)I)) + N_0 - Tr(N_0)I$.

APPENDIX D
DERIVATION OF STRICTLY DESCENDING STEPSIZE ESTIMATE

In Section III it is stated that once the direction $(\omega_{R_X}, \omega_{R_Y})$ is determined, the strictly descending stepsize is given by

$$t^* = -\frac{\phi'(0)}{c}$$

where $\phi'(t) = J(R_{X_k}e^{[\omega_{R_X}]t}, R_{Y_k}e^{[\omega_{R_Y}]t})$ and c is an upper bound of $|\phi''(t)|$. $\phi'(t)$ and $\phi''(t)$ as follows:

$$\begin{aligned}\phi'(t) &= \sum_{i=1}^{18} \lambda_i Tr(P_i R_{X_k} e^{[\omega_{R_X}]t} + Q_i R_{Y_k} e^{[\omega_{R_Y}]t}) \\ & Tr(P_i R_{X_k}[\omega_{R_X}] e^{[\omega_{R_X}]t} + Q_i R_{Y_k}[\omega_{R_Y}] e^{[\omega_{R_Y}]t}) \\ & + Tr(P_0 R_{X_k}[\omega_{R_X}] e^{[\omega_{R_X}]t} + Q_0 R_{Y_k}[\omega_{R_Y}] e^{[\omega_{R_Y}]t})\end{aligned}$$

$$\begin{aligned}&\phi''(t) = \\ &\textstyle\sum_{i=1}^{18} \lambda_i \Big(\big(Tr(P_i R_{X_k}[\omega_{R_X}] e^{[\omega_{R_X}]t} + Q_i R_{Y_k}[\omega_{R_Y}] e^{[\omega_{R_Y}]t})\big)^2 \\ &+ \sum_{i=1}^{18} Tr(P_i R_{X_k} e^{[\omega_{R_X}]t} + Q_i R_{Y_k} e^{[\omega_{R_Y}]t}) \\ &Tr(P_i R_{X_k}[\omega_{R_X}]^2 e^{[\omega_{R_X}]t} + Q_i R_{Y_k}[\omega_{R_Y}]^2 e^{[\omega_{R_Y}]t}) \Big) \\ &+ Tr(P_0 R_{X_k}[\omega_{R_X}]^2 e^{[\omega_{R_X}]t} + Q_0 R_{Y_k}[\omega_{R_Y}]^2 e^{[\omega_{R_Y}]t}). \quad (27)\end{aligned}$$

We remark that the P_i, Q_i are from the eigenvectors of $\tilde{H}$ in (24) and it follows that, using the maximum eigenvalue of $\tilde{H}$ (in terms of absolute value), an upper bound for the first sum in (27) is $|\lambda|_{max}\left(\|X_k[\omega_{R_X}]e^{[\omega_{R_X}]t}\|^2 + \|Q_i R_{Y_k} e^{[\omega_{R_Y}]t}\|^2\right)$. An upper bound for the second sum in (24) can be found similarly. Since a linear transformation by a rotation matrix $R \in SO(3)$ is an isometry (i.e., $\|RU\| = \|UR\| = \|U\|$ for any $U \in \mathbb{R}^{3\times 3}$), it follows that the strictly descending stepsize is

$$t^* = -\frac{\phi'(0)}{c}$$

where

$$\begin{aligned}&\phi'(0) = \\ &\textstyle\sum_{i=1}^{18} \lambda_i Tr(P_i R_{X_k} + Q_i R_{Y_k}) Tr(P_i R_{X_k}[\omega_{R_X}] + Q_i R_{Y_k}[\omega_{R_Y}]) \\ &+ Tr(P_0 R_{X_k}[\omega_{R_X}] + Q_0 R_{Y_k}[\omega_{R_Y}])\end{aligned}$$

and

$$\begin{aligned}c = &|\lambda|_{max}\left(\|[\omega_{R_X}]\|^2 + \|[\omega_{R_Y}]\|^2\right) \\ &+\sqrt{6}|\lambda|_{max}\sqrt{\|[\omega_{R_X}]^2\|^2 + \|[\omega_{R_Y}]^2\|^2} \\ &+\sqrt{3}\left(\|P_0 R_{X_k}[\omega_{R_X}]^2\| + \|Q_0 Y_k[\omega_{R_Y}]^2\|\right).\end{aligned}$$

REFERENCES

[1] H. Zhuang, Z.S. Roth, and R. Sudhakar, "Simultaneous robot/world and tool/flange calibration by solving homogeneous transformation equations of the form $AX = YB$," *IEEE Trans. Robot. Autom.*, vol. 10, no. 4, pp. 549-554, 1994.

[2] S.-H.P. Won, F. Golnaraghi, and W.W. Melek, "A fastening tool tracking system using an IMU and a position sensor with Kalman filter and a fuzzy expert system," *IEEE Trans. Ind. Electron.*, vol. 56, no. 5, pp. 1782-1792, 2009.

[3] Y-.K. Kim, I.G. Jane, Y. Kim, K-.S. Kim, and S. Kim, "Structural optimization of a novel 6-DOF pose sensor system for enhancing noise robustness at a long distance," *IEEE Trans. Ind. Electron.*, vol. 61, no. 10, pp. 5622-5631, 2014.

[4] F. Dornaika and R. Horaud, "Simultaneous robot-world and hand-eye calibration," *IEEE Trans. Robot. Autom.*, vol. 14, no. 4, pp. 617-622, 1998.

[5] A. Li, L. Wang, and D. Wu, "Simultaneous robot-world and hand-eye calibration using dual-quaternions and Kronecker product," *Int. J. Physical Sciences*, vol. 5, no. 10, pp. 1530-1436, 2010.

[6] F.C. Park, and B. Martin, "Robot sensor calibration: solving $AX = XB$ on the Euclidean group," *IEEE Trans. Robot. Autom.*, vol. 10, no. 5, pp. 717-721, 1994.

[7] K. Daniliidis, "Hand eye calibration using dual quaternions," *Int. J. Robotics Research*, vol. 18, no. 3, pp. 286-298, 1999.

[8] Y. Motai, and A. Kosaka, "Hand-eye calibration applied to viewpoint selection for robotic vision," *IEEE Trans. Ind. Electron.*, vol. 55, no. 10, pp. 3731-3741, 2008.

[9] Z. Zhao, and Y. Weng, "A flexible method combining camera calibration and hand-eye calibration," *Robotica*, vol. 31, no. 5, pp. 747-756, 2013.

[10] M. Shah, "Solving the robot-world/hand-eye calibration problem using the Kronecker product," *ASME J. Mechanisms and Robotics*, vol. 5, no. 3, pp. 15-20, 2012.

[11] J. Heller, M. Havlena, and T. Pajdla, "Globally optimal hand-eye calibration using branch-and-bound," *IEEE Trans. Pattern Anal. Machine Intell.* (early access article; online published).

[12] A. Ghaffari, M. Krstic, and D. Nesic, "Multivariable Newton-based extremum seeking," *Automatica*, vol. 48, pp. 1759-1767, 2012.

[13] A.H.G. Rinooy Kan, and G.T. Timmer, "Stochastic global optimization methods part I: clustering methods," *Mathematical Programming*, vol. 39, no. 1, pp. 27-56, 1987.

[14] A.H.G. Rinooy Kan, and G.T. Timmer, "Stochastic global optimization methods part II: multi level methods," *Mathematical Programming*, vol. 39, no. 1, pp. 57-78, 1987.

[15] P. Dreesen, K. Batselier, and B.D. Moor, "Back to the roots: polynomial system solving, linear algebra, and systems theory," *Proc. 16th IFAC Symposium on System Identification*, 2012.

[16] A. Yershova, S. Lavalle, and C. Mitchell, "Generating uniform incremental grids on SO(3) using the Hopf fibration," *Proc. Workshop on Algorithmic Foundations of Robotics*, pp. 1-15, 2008.

[17] P. Diaconis, and M. Shahshahani, "The subgroup algorithm for generating uniform random variables," *Problems in Engineering and Information Sciences*, vol. 1, pp. 15-32, 1987.

Junhyoung Ha (M'15) received his B.S. and Ph.D. degrees in mechanical engineering from Seoul National University, Seoul, Korea, in 2008 and 2015, respectively.

During the Ph.D. study, he was a research assistant for the Robotics Laboratory in the School of Mechanical and Aerospace Engineering at Seoul National University. He is currently with Boston Childrens Hospital, Harvard Medical School, Boston. His research interests are in medical robotics, continuum robot, control and optimization.

Donghoon Kang (M'12) received his B.S. and M.S. degrees in mechanical engineering from POSTECH, Pohang, Korea, in 1997 and 1999, respectively. He is currently working toward the Ph.D. degree at Seoul National University, Seoul, Korea.

From 1999 to 2000, he was with Samsung SDI Inc. as an electronic engineer. Since 2000, he has been at Korea Institute of Science and Technology (KIST), where he is currently a research scientist. His research interests are in signal processing, motion estimation and 3D visual SLAM.

Frank C. Park (F'13) received his B.S. in electrical engineering from MIT in 1985, and Ph.D. in applied mathematics from Harvard University in 1991.

From 1991 to 1995 he was assistant professor of mechanical and aerospace engineering at the University of California, Irvine. Since 1995, he has been professor of mechanical and aerospace engineering at Seoul National University. His research interests are in robot mechanics, planning and control, vision and image processing, and related areas of applied mathematics.

He has been an IEEE Robotics and Automation Society Distinguished Lecturer, and received best paper awards for his work on visual tracking and parallel robot design. He has served on the editorial boards of the Springer Handbook of Robotics, Springer Advanced Tracts in Robotics (STAR), Robotica, and the ASME Journal of Mechanisms and Robotics. He has held adjunct faculty positions at the NYU Courant Institute and the Interactive Computing Department at Georgia Tech. He is a fellow of the IEEE, current editor-in-chief of the IEEE Transactions on Robotics, and developer of the EDX course Robot Mechanics and Control I,II.