



UNIVERSITY OF LEEDS

This is a repository copy of *Cameras and Inertial/Magnetic Sensor Units Alignment Calibration*.

White Rose Research Online URL for this paper:
<http://eprints.whiterose.ac.uk/98804/>

Version: Accepted Version

Article:

Zhang, Z-Q (2016) *Cameras and Inertial/Magnetic Sensor Units Alignment Calibration*. IEEE Transactions on Instrumentation and Measurement, 65 (6). pp. 1495-1502. ISSN 0018-9456

<https://doi.org/10.1109/TIM.2016.2518418>

Reuse

Unless indicated otherwise, fulltext items are protected by copyright with all rights reserved. The copyright exception in section 29 of the Copyright, Designs and Patents Act 1988 allows the making of a single copy solely for the purpose of non-commercial research or private study within the limits of fair dealing. The publisher or other rights-holder may allow further reproduction and re-use of this version - refer to the White Rose Research Online record for this item. Where records identify the publisher as the copyright holder, users can verify any specific terms of use on the publisher's website.

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.



eprints@whiterose.ac.uk
<https://eprints.whiterose.ac.uk/>

Cameras and Inertial/Magnetic Sensor Units Alignment Calibration

Zhi-Qiang Zhang

Abstract—Due to the external acceleration interference/magnetic disturbance, the inertial/magnetic measurements are usually fused with visual data for drift-free orientation estimation, which plays an important role for a wide variety of applications, ranging from virtual reality, robot, computer vision, to bio-motion analysis and navigation. However, in order to perform data fusion, alignment calibration must be performed in advance to determine the difference between the sensor coordinate system and camera coordinate system. Since orientation estimation performance of the inertial/magnetic sensor unit is immune to the selection of the inertial/magnetic sensor frame original point, we therefore ignore the translational difference by assuming the sensor and camera coordinate systems sharing the same original point and focus on the rotational alignment difference only in this paper. By exploiting the intrinsic restrictions among the coordinate transformations, the rotational alignment calibration problem is formulated by a simplified hand-eye equation $AX = XB$ (A, X and B are all rotation matrices). A two-step iterative algorithm is then proposed to solve such simplified hand-eye calibration task. Detailed laboratory validation has been performed and the good experimental results have illustrated the effectiveness of the proposed alignment calibration method.

Index Terms—Inertial/magnetic, Cameras, Estimation, Orientation/Attitude, Optimization

I. INTRODUCTION

Inertial/magnetic sensor units have been widely used to determine orientation estimation, which plays an important role for a wide variety of applications, ranging from virtual reality, robot, computer vision, to bio-motion analysis and biomedical applications [1] [2] [3]. However, the inertial/magnetic sensor units inherently suffer from integration drift, and they are also usually susceptible to external acceleration interference/magnetic disturbance; therefore, inertial/magnetic sensor units are combined with cameras for drift-free orientation estimation, particularly for the vision-aided inertial navigation applications [4] [5] [6].

Thus far, extensive research has been performed on how to accurately determine attitude information by fusing inertial/magnetic sensor measurements and visual data. For example, Du et al. [7] incorporated Kalman filters (KFs) and adaptive multispace transformation (AMT) to track movements of the human hand and control the robot manipulator. Their method employed one inertial measurement unit and a 3-D camera (Kinect) to determine the orientation and translation of the human hand. Nam et al. [8] presented a method to estimate golf club trajectory (position and velocity) and club face orientation using an inertial sensor unit and a stereo camera both on

the golf club. Li et al. [9] and Tian et al. [10] also presented similar work for drift free orientation estimation. However, the achievable accuracy of orientation estimation is highly dependent on the quality of the sensor measurements given in the camera coordinate system in practice. Therefore, alignment calibration must be performed in advance to determine the difference between the sensor coordinate system and camera coordinate system.

In general, the differences between any two coordinate frames can be described by two parameters: a translation vector and a rotation matrix. The determination of the translation vector and rotation matrix is usually modeled as a hand-eye calibration problem $AX = XB$ (A, X and B are all homogeneous matrices). The aim is to determine the transformation matrix X , given at least two pairs of A and B . Thus far, hand-eye calibration has been studied extensively. For example, Chou et al. [11] used quaternion to transform the hand-eye calibration equation into two simple and structured linear systems with rank-deficient coefficient matrices. Closed-form solutions were derived using the generalized inverse method with singular value decomposition analysis. Daniilidis et al. [12] introduced of the idea of dual-quaternion parameterization, which facilitated a new simultaneous solution for the hand-eye rotation and translation using the singular value decomposition. Zhao et al. [13] presented a new hand-eye calibration algorithm based on screw motion constraints, which established a linear homogeneous system using quaternion. The computation of the null space with singular value decomposition was also implemented to yield an accurate solution of hand-eye transformation. Lébraly et al. [14] and Hu et al. [15] also presented similar work in their papers. However, all these methods were based on homogenous matrices or quaternion, and a closed-form solution to hand-eye calibration equation was provided, but they were always accompanied with sophisticated derivations. In the past years, researchers tend to move from the closed-form solution to iterative method due to its high efficiency and simplicity. The basic idea of iterative method is to minimize the difference between the left and right parts of the hand-eye equation or its variations. Thus far, a number of solutions have been proposed. For instance, Ruland et al. [16] proposed to integrate the hand-eye calibration problem into a branch-and-bound parameter space search. The presented method constituted the first guaranteed globally optimal estimator for simultaneous optimization of both components with respect to a cost function based on re-projection errors. Ackerman et al. [17] presented a unified algorithm which used gradient descent optimization on the Euclidean Group. They also applied filtering to update the

Z. Q. Zhang is with the Department of Computing, Imperial College London, SW7 2AZ, United Kingdom. (email: z.zhang@imperial.ac.uk)

calibration parameters on-line based on new incoming data. Heller et al. [18] presented several formulations of hand-eye calibration that led to multivariate polynomial optimization problems. Convex linear matrix inequality (LMI) relaxations was used to effectively solve these problems and to obtain globally optimal solutions. Wu et al. [19] presented the theory and implementation of neural networks for hand-eye calibration and inverse kinematics of a six degrees of freedom robot arm equipped with a stereo vision system. Similarly, Hubert et al. [20] and Prasse et al. [21] also derived their cost functions and solutions for the optimization problem. The aforementioned methods can all be used to solve the hand-eye calibration problem, albeit being complex to implement in practice.

In practice, the orientation estimation performance is immune to selection of the inertial/magnetic sensor frame original point [22] [23], thus we can ignore the translational difference by assuming the sensor and camera coordinate systems sharing the same original point; therefore, we will only focus on the rotational alignment difference in this paper. By exploiting the intrinsic restrictions among the coordinate transformations, the rotational alignment estimation problem is formulated by a simplified hand-eye equation $AX = XB$ (A, X and B are therefore simplified as rotation matrices). A two-step iterative algorithm is then proposed to solve such simplified hand-eye calibration task. Detailed laboratory validation has been performed and the good experimental results have illustrated the effectiveness of the proposed alignment calibration method.

The rest of the paper is organized as follows. The proposed rotational alignment calibration procedures, including the simplified hand-eye equation derivation and two-step iteration method are given in section II. Experimental results and conclusions are provided in sections III and IV, respectively.

II. OUR METHOD

A. Simplified hand-eye equation derivation

At any time t as shown in the Fig. 1, if we denote the orientations of inertial/magnetic sensor node and the camera in the global reference coordinate system as I_t and C_t , respectively, we can have:

$$C_t = R_0 I_t \quad (1)$$

where R_0 is the rotational alignment difference between the sensor coordinate system and the camera coordinate system. However, due to the difficulty to define the same reference coordinate system for the inertial/magnetic sensor node and the camera in practice, it may be not easy to the exact C_t and I_t . Therefore, we can consider another time slot k as

$$C_k = R_0 I_k \quad (2)$$

where I_k and C_k are the orientations of inertial/magnetic sensor node and the camera in the global reference coordinate system at time k . Similarly, C_k and I_k are difficult to acquire too. However, instead of calculating the absolute orientations in the camera frame, such as C_t and C_k , it is straightforward to derive the relative orientation difference

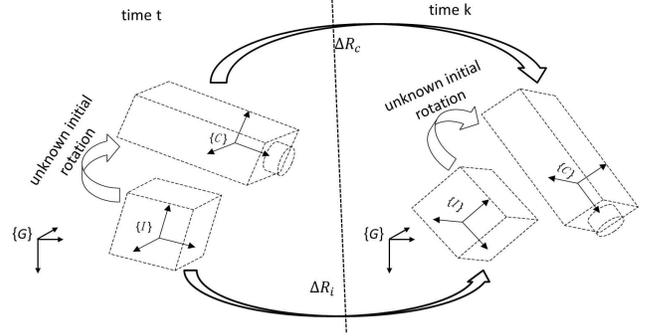


Figure 1. The illustration of the alignment difference between the inertial/magnetic sensor unit frame and the camera frame.

between them [24] [25]. There are also plenty of methods to fuse the inertial/magnetic sensor measurements to estimate the sensor orientation difference between time t and k [26] [27]. As shown in the Fig. 1, denote ΔR_c as the orientation difference in the camera coordinate system between time t and k , C_k can thus be taken as the combination of two rotations ΔR_c and C_t as

$$C_k = \Delta R_c C_t. \quad (3)$$

Similarly, denote ΔR_i as the orientation difference in sensor coordinate system, we can then have

$$I_k = \Delta R_i I_t. \quad (4)$$

Substitute Eq. (1) to Eq.(3) and Eq. (4) to Eq.(2), we can have

$$\begin{cases} C_k = \Delta R_c C_t = \Delta R_c R_0 I_t \\ C_k = R_0 I_k = R_0 \Delta R_i I_t \end{cases} \quad (5)$$

which means

$$\Delta R_c R_0 I_t = R_0 \Delta R_i I_t. \quad (6)$$

Since the I_t is a full rank rotational matrix, thus we can have the following simplified hand-eye equation:

$$\Delta R_c R_0 = R_0 \Delta R_i. \quad (7)$$

B. Two step iteration method

In order to estimate the R_0 , we can put the camera and sensor node together at different orientations. Given J orientation differences $\Delta R_{c,1}, \Delta R_{c,2} \cdots \Delta R_{c,J}$ in the camera frame, and their corresponding differences in the sensor frame $\Delta R_{i,1}, \Delta R_{i,2} \cdots \Delta R_{i,J}$, the estimate of R_0 can be written as a quadratic convex optimization problem:

$$\hat{R}_0 = \underset{R_0}{\operatorname{argmin}} \left\{ \sum_{j=1}^J \left\| \Delta R_{c,j} R_0 - R_0 \Delta R_{i,j} \right\|^2 \right\} \quad (8)$$

subject to

$$R_0 \cdot R_0^T = U \quad (9)$$

and

$$\det(R_0) = 1 \quad (10)$$

where $\|\cdot\|$ is the Frobenius norm, U is the identify matrix of order 3, and $\det(\cdot)$ is the determinant of a 3×3 matrix. There

are plenty of algorithms, such as active set algorithm [28], interior point algorithm [29], sequential quadratic programming (SQP) algorithm [30] and so on, have been proposed so far to solve the above constrained minimization problem, but these methods tend to calculate the Jacobian matrix and Hessian matrix, which are computationally expensive. In this paper, we propose a simple two step iteration method to solve the above constrained optimization problem.

Denote a $3J \times 3$ matrix \mathcal{H}_l as

$$\mathcal{H}_l = \begin{bmatrix} \Delta R_{c,1} \\ \Delta R_{c,2} \\ \vdots \\ \Delta R_{c,J} \end{bmatrix} \quad (11)$$

and a $3 \times 3J$ matrix \mathcal{H}_r as:

$$\mathcal{H}_r = [\Delta R_{i,1}, \Delta R_{i,2}, \dots, \Delta R_{i,J}] \quad (12)$$

thus R_0 should satisfy:

$$\begin{aligned} V2H(\mathcal{H}_l R_0) &= R_0 \mathcal{H}_r \\ \mathcal{H}_l R_0 &= H2V(R_0 \mathcal{H}_r) \end{aligned} \quad (13)$$

where $V2H(\cdot)$ is to convert a $3J \times 3$ matrix to a $3 \times 3J$ matrix while $H2V(\cdot)$ is the inverse operation of $V2H(\cdot)$, converting a $3 \times 3J$ matrix to a $3J \times 3$ matrix. Take the \mathcal{H}_l for example, $V2H(\mathcal{H}_l) = [\Delta R_{c,1}, \Delta R_{c,2}, \dots, \Delta R_{c,J}]$ and $H2V(V2H(\mathcal{H}_l)) = \mathcal{H}_l$.

In order to apply the two step iteration methods, we take the left side and right side of R_0 in Eq.(13) separately, and use R_0^l and R_0^r to represent them accordingly. Given an initial value for R_0^l as $R_{0,0}^l$, the R_0^l and R_0^r can be estimated as:

1. set index $n = 1$;
2. calculate $R_{0,n}^r$ as:

$$R_{0,n}^r = V2H(\mathcal{H}_l R_{0,n-1}^l) \cdot \mathcal{H}_r^+ \quad (14)$$

where $(\cdot)^+$ is the pseudo-inverse operator.

3. calculate $R_{0,n}^l$ as

$$R_{0,n}^l = \mathcal{H}_l^+ \cdot H2V(R_{0,n}^r \mathcal{H}_r). \quad (15)$$

4. set $n = n + 1$ and repeat steps 2–4 until $R_{0,n}^l$ and $R_{0,n}^r$ converge.
5. Recover the rotation matrix from $R_{0,n}^l$ using singularity value decomposition (SVD) related techniques. The SVD of the matrix $R_{0,n}^l$ can be calculated as:

$$R_{0,n}^l = U \Sigma \Lambda^T \quad (16)$$

where the columns of U contain the eigenvectors of $R_{0,n}^l (R_{0,n}^l)^T$, the columns of Λ contain the eigenvectors of $(R_{0,n}^l)^T R_{0,n}^l$, and the diagonal of Σ indicates the singular values of $R_{0,n}^l$. Thus we can have

$$\hat{R}_0^l = U \Lambda^T. \quad (17)$$

Similarly, we can also derive the rotational matrix \hat{R}_0^r from $R_{0,n}^r$. The final estimation for R_0 can thus be written as

$$\hat{R}_0 = \hat{R}_0^l \quad (18)$$

or

$$\hat{R}_0 = \hat{R}_0^r. \quad (19)$$

Theorem 1: The $R_{0,n}^l$ and $R_{0,n}^r$ can always converge to obtain the ground truth for R_0 via the two step iteration methods.

Proof: The purpose of the Eq. (8) is to minimize

$$\|V2H(\mathcal{H}_l R_0) - R_0 \mathcal{H}_r\| \quad (20)$$

or

$$\|\mathcal{H}_l R_0 - H2V(R_0 \mathcal{H}_r)\| \quad (21)$$

which means that $R_{0,n}^l$ and $R_{0,n}^r$ can converge to obtain the ground truth for R_0 only if:

$$\begin{aligned} &\|V2H(\mathcal{H}_l R_{0,n}^l) - R_{0,n+1}^r \mathcal{H}_r\| \\ &\leq \|V2H(\mathcal{H}_l R_{0,n}^l) - R_{0,n}^r \mathcal{H}_r\| \end{aligned} \quad (22)$$

and

$$\begin{aligned} &\|\mathcal{H}_l R_{0,n}^l - H2V(R_{0,n}^r \mathcal{H}_r)\| \\ &\leq \|\mathcal{H}_l R_{0,n-1}^l - H2V(R_{0,n}^r \mathcal{H}_r)\|. \end{aligned} \quad (23)$$

For Eq. (22), we can have

$$\begin{aligned} &\|V2H(\mathcal{H}_l R_{0,n}^l) - R_{0,n+1}^r \mathcal{H}_r\| \\ &= \|V2H(\mathcal{H}_l R_{0,n}^l) - V2H(\mathcal{H}_l R_{0,n}^l) \cdot \mathcal{H}_r^+ \mathcal{H}_r\| \\ &= \|V2H(\mathcal{H}_l R_{0,n}^l) (U - \mathcal{H}_r^+ \mathcal{H}_r)\| \end{aligned} \quad (24)$$

For any matrices Υ and A , $\|U - \Upsilon^+ \Upsilon\| < \|U - A^+ \Upsilon\|$ is always satisfied unless $\Upsilon = A$ [31], so

$$\begin{aligned} &\|V2H(\mathcal{H}_l R_{0,n}^l) - R_{0,n+1}^r \mathcal{H}_r\| \\ &\leq \|V2H(\mathcal{H}_l R_{0,n}^l) (U - V2H(\mathcal{H}_l R_{0,n}^l)^+ \cdot \\ &\quad V2H(\mathcal{H}_l R_{0,n-1}^l) \mathcal{H}_r^+ \mathcal{H}_r)\| \\ &= \|V2H(\mathcal{H}_l R_{0,n}^l) - V2H(\mathcal{H}_l R_{0,n-1}^l) \mathcal{H}_r^+ \mathcal{H}_r\| \\ &= \|V2H(\mathcal{H}_l R_{0,n}^l) - R_{0,n}^r \mathcal{H}_r\|. \end{aligned} \quad (25)$$

For Eq. (23), we can also have

$$\begin{aligned} &\|\mathcal{H}_l R_{0,n}^l - H2V(R_{0,n}^r \mathcal{H}_r)\| \\ &= \|\mathcal{H}_l \mathcal{H}_l^+ H2V(R_{0,n}^r \mathcal{H}_r) - H2V(R_{0,n}^r \mathcal{H}_r)\| \\ &= \|(\mathcal{H}_l \mathcal{H}_l^+ - U) H2V(R_{0,n}^r \mathcal{H}_r)\| \end{aligned} \quad (26)$$

Similar to equation (25), we can have

$$\begin{aligned} &\|\mathcal{H}_l R_{0,n}^l - H2V(R_{0,n}^r \mathcal{H}_r)\| \\ &\leq \|(\mathcal{H}_l \mathcal{H}_l^+ H2V(R_{0,n-1}^r \mathcal{H}_r) \cdot \\ &\quad H2V(R_{0,n}^r \mathcal{H}_r)^+ - U) H2V(R_{0,n}^r \mathcal{H}_r)\| \\ &= \|\mathcal{H}_l \mathcal{H}_l^+ H2V(R_{0,n-1}^r \mathcal{H}_r) - H2V(R_{0,n}^r \mathcal{H}_r)\| \\ &= \|\mathcal{H}_l R_{0,n-1}^l - H2V(R_{0,n}^r \mathcal{H}_r)\| \end{aligned} \quad (27)$$

■

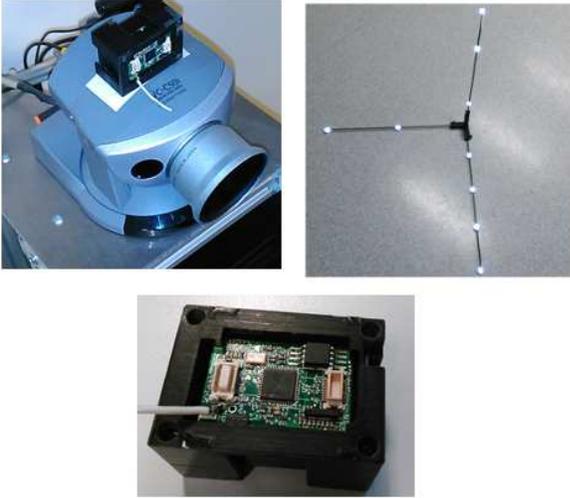


Figure 2. The BSN node was mounted onto the top camera of the robot. To simplify the orientation derivation from the captured images, the camera was facing the calibration wands all the time from different orientations.

III. EXPERIMENTAL AND SIMULATION RESULTS

In order to evaluate the performance of the proposed alignment calibration algorithm, detailed simulation and laboratory experiments were carried out. The simulation study was based on the Monte Carlo simulation, which was carried out in a workstation with 3.40 GHz Intel Core i7 processor and 16G RAM. For the experimental results presented in this paper, we used the Body Sensor Network (BSN) platform [32] developed by our lab, which consists of three stackable daughter boards: the sensor board, the main processor board, and the battery board. They are connected via a stackable connector design. Each BSN node used is equipped with an Analog Devices ADXL330 [33] for 3D acceleration measurement, an InvenSense ITG-3200 digital gyroscope [34] for 3D angular velocity measurement, and a Honeywell HMC5843 [35] for 3D magnetic field measurement. In order to calculate the rational difference between an inertial/magnetic sensor unit and a camera, the BSN sensor node was placed on top of a camera as shown in the Fig. 2. The BSN sensor node was properly calibrated to provide accurate orientation estimation using the method presented in [27] [36] [37]. Similarly, to simplify the orientation derivation from the captured images, calibration wands (the middle right one in the Fig. 2), which consist of 9 marker points positioned in 3-D space at known coordinates was used in our experiment. The method presented in the [38] was thus applied to extract the camera orientation.

A. Simulation study

Since it is quite challenging to acquire the ground-truth of the rotational alignment difference between inertial/magnetic sensor units and cameras, we resort to simulation study with known parameters. In this simulation, the estimation of the rotational alignment difference R_0 was studied when the camera was rotated into randomly selected 20 different orientations, given by the relative motions $\Delta R_{c,1}, \Delta R_{c,2} \cdots \Delta R_{c,20}$. The

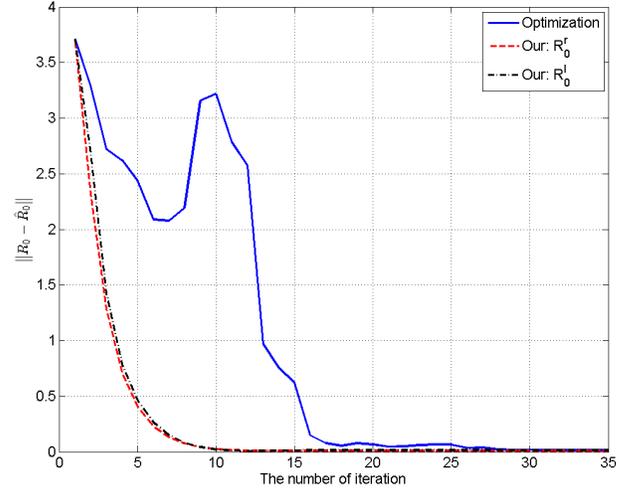


Figure 3. Estimation results for matrix R_0 , showing that after 10 iterations, the Frobenius norm $\|R_0 - \hat{R}_0\|$ converges to 0, i.e., $R_0 = \hat{R}_0$.

R_0 was randomly set to

$$R_0 = \begin{bmatrix} 0.9099 & 0.0180 & -0.4144 \\ 0.3423 & 0.5315 & 0.7748 \\ 0.2342 & -0.8468 & 0.4775 \end{bmatrix},$$

thus the relative motions in the sensor coordinate frame were calculated as:

$$\Delta R_{i,j} = R_0^T \Delta R_{c,j} R_0, \quad j = 1, 2, 3 \cdots 20.$$

To simulate the orientation estimation error in the $\Delta R_{i,j}$, a random selected 3×1 vector $v_{i,j}$ with less than 0.02 magnitude (to make sure the rotation angle is less than 1°) was applied to generate a small rotational error matrix for each j as:

$$\delta R_{i,j} = [v_{i,j} \times] + U$$

where $[\cdot \times]$ is the skew-symmetric matrix operator. The SVD technique given in Eqs. (16)(17) was also applied to $\delta R_{i,j}$ to make it a perfect rotational matrix. Thus the $\Delta R_{i,j}$ used in our simulation is

$$\Delta R_{i,j} = \delta R_{i,j} \Delta R_{i,j}.$$

Similarly, a small rotation error was also added to $\Delta R_{c,j}$ using the same method.

Fig. 3 shows the iterative results for R_0 estimation, while Fig. 4 presents the value of the cost function $\sum_{j=1}^J \|\Delta R_{c,j} R_0 - R_0 \Delta R_{i,j}\|$. In the figures, the R_0 estimations based on R_0^l and R_0^r are both given. It is obvious that either R_0^l or R_0^r can both generate accurate estimation for R_0 and minimise the cost function. Meanwhile, We also implemented the SQP algorithm to optimize the constrained problem in equation (8) for comparison purpose, and the results derived from the SQP algorithm are also shown in the Fig. 3 and Fig. 4. As we can see from the figures, it is very clear that our proposed iterative method is relatively faster to converge. After about 10 iterations, the estimation

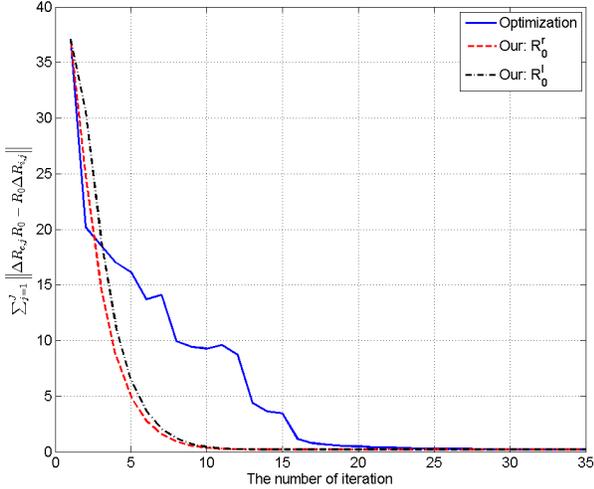


Figure 4. The value of the cost function $\sum_{j=1}^J \|\Delta R_{c,j} R_0 - R_0 \Delta R_{i,j}\|$, showing that after 10 iterations, the cost function value converges to 0, i.e., $R_0 = \hat{R}_0$.

for R_0 is already very close to their respective ground-truth values, and the value of the cost function is almost 0. Although the optimization method can also converge to the ground-truth of R_0 , convergence speed is much slower and it needs more than 30 iterations to achieve less than 1% error. Meanwhile, the convergence process of our method to find R_0 is much smoother. The estimation of R_0 will get closer to the ground truth, and the value of the cost function will get smaller after each iteration. In contrast, the estimation of R_0 using the SQP method may divert from the ground truth although the value of the cost function gets smaller after some certain iterations. We also noticed that the optimization method took about 2 seconds to complete all the iterations, while our method only took less than 0.05 second in our simulation. In fact, the SQP algorithm usually requires to calculate the value of cost function more than 10 times within an iteration, and it also involves sophisticated Hessian and Jacobian matrix operations, which are very computationally expensive. However, our proposed method only requires some basic matrix operations, such as multiplication and inverse, which therefore make our method much more efficient than the traditional optimization method.

In theory, the alignment different between the camera and inertial sensor is constant. However, in practice, particularly in our applications, the sensor node and camera are attached together using tapes. Every time we put the sensor node on the camera (as shown in the Fig 2), an alignment calibration must have to be done. Meanwhile, when the robot maneuvers on any uneven surface, there are always some small inter-movement between camera and sensor node since they are not rigidly connected. Therefore, online re-calibration must be performed during the experiments, which significantly requires the efficiency and simplicity of the calibration algorithm. The proposed method have shown its strength meet the requirements for such applications.

Table I
ITERATIVE RESULTS OVER 1000 SIMULATIONS (SHOWN AS MEAN \pm STD)

	$\ R_0 - \hat{R}_0\ $		
	Optimization	Our right	our left
Iteration 2	3.488 \pm 0.250	2.339 \pm 0.275	2.769 \pm 0.212
Iteration 5	3.100 \pm 0.716	0.483 \pm 0.231	0.536 \pm 0.221
Iteration 10	2.272 \pm 0.522	0.051 \pm 0.052	0.062 \pm 0.053
Iteration 15	0.451 \pm 0.233	0.012 \pm 0.012	0.016 \pm 0.011
Iteration 20	0.156 \pm 0.086	0.009 \pm 0.004	0.010 \pm 0.005
Iteration 30	0.053 \pm 0.038	0.009 \pm 0.004	0.010 \pm 0.005
Iteration 50	0.015 \pm 0.003	0.009 \pm 0.004	0.010 \pm 0.005

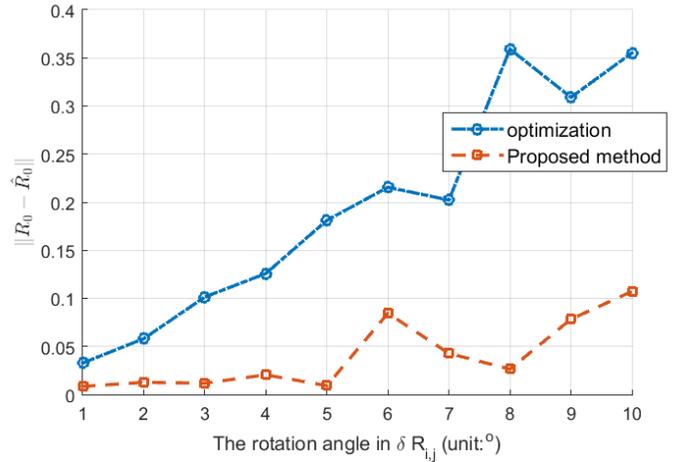


Figure 5. Estimation results for matrix R_0 : the variations of Frobenius norm $\|R_0 - \hat{R}_0\|$ when the angle related to $\delta R_{i,j}$ was increased from 1° to 10° while maintaining the noise for $\Delta R_{c,j}$ at 1° .

The simulation was repeated for another 1000 times, and statistical results for R_0 are given in Table I. It can be seen that the proposed two step iterative method (either based on R_0^l or R_0^r) converges after 15 iterations with negligible errors, while the traditional optimization based methods needs at least 30 iterations. In conclusion, the above analysis has shown that the proposed two step iteration method can estimate the rotational difference between inertial/magnetic sensor units and cameras accurately and efficiently.

In our second simulation, we considered how the noise strength would affect the performance of the proposed method. In our simulation, the angle related to $\delta R_{i,j}$ was increased from 1° to 10° while maintaining the noise for $\Delta R_{c,j}$ at 1° . Fig. 5 shows the variations of Frobenius norm $\|R_0 - \hat{R}_0\|$. It is obvious that there are some increments of $\|R_0 - \hat{R}_0\|$ when the noise level increases, but the increase speed of the proposed method is much slower than that of the traditional optimization method, which illustrates that the proposed method is more resilient to the noise.

B. Experimental Results

We then applied the proposed two step iteration method to estimate the alignment difference between the BSN node

Table II
THE RMS, MEAN, SD AND CORRELATION COEFFICIENTS OF THE ESTIMATED ATTITUDE COMPARED TO THE ONE EXTRACTED FROM CAMERA IMAGE FRAMES.

	Optimization Calibration		Our Calibration (R_0^l)		Our Calibration (R_0^r)		Sensor frame	
	RMS (Mean,SD)	Correlation Coefficient	RMS (Mean,SD)	Correlation Coefficient	RMS (Mean,SD)	Correlation Coefficient	RMS (Mean,SD)	Correlation Coefficient
Roll	0.6382 (-0.1346±0.6238)	0.9995	0.6876 (-0.1413±0.5961)	0.9996	0.6287 (-0.3736±0.6710)	0.9994	15.7528 (-0.9192±15.7268)	0.6668
Pitch	0.8312 (-0.2547±0.7913)	0.9997	0.7787 (-0.2723±0.7296)	0.9998	0.8130 (-0.2754±0.7649)	0.9997	68.3401 (-56.0996±39.0301)	-0.1062
Yaw	0.8327 (-0.1064±0.8259)	0.9990	0.9892 (-0.6121±0.7770)	0.9991	0.8856 (-0.4114±0.7843)	0.9991	63.1261 (-53.6427±33.2787)	0.1801

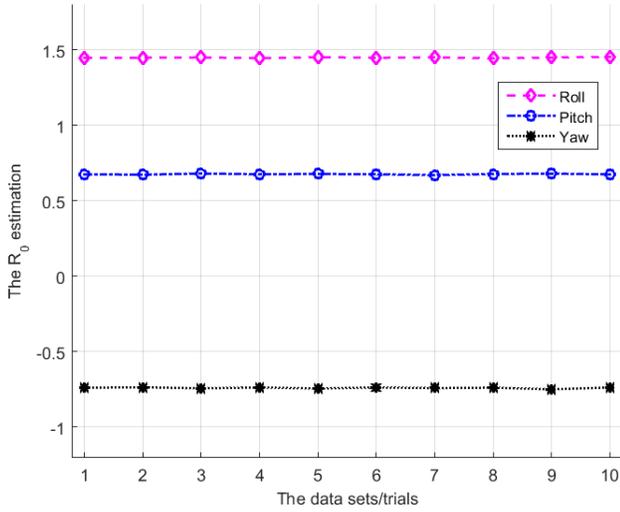
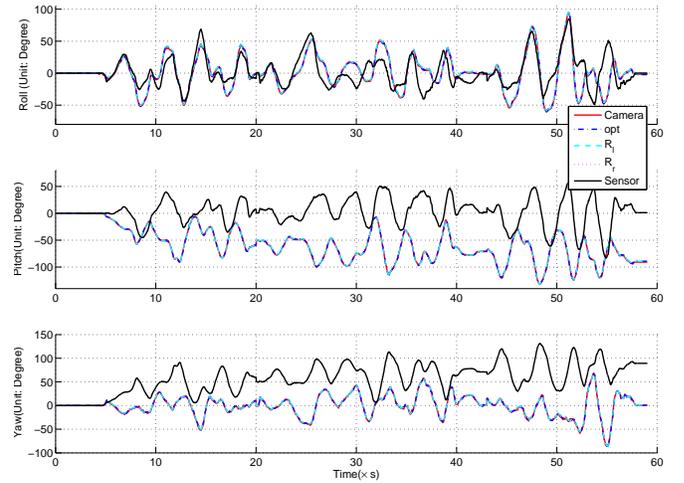


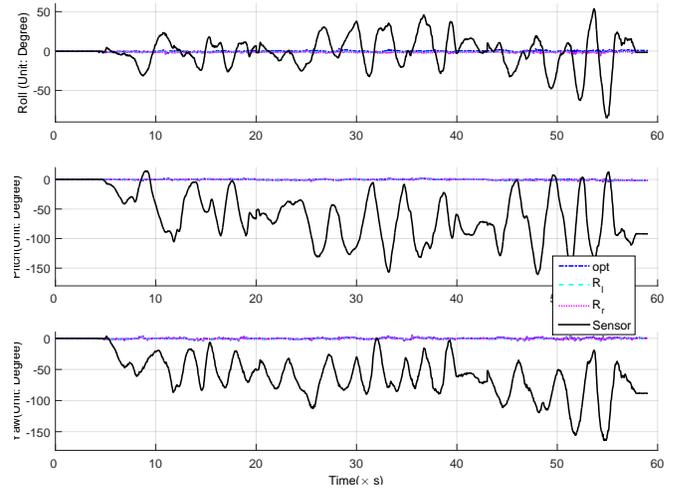
Figure 6. The BSN and robot's top camera alignment calibration results. During the experiments, the same two-step iteration method was applied on 10 independent data sets. Although there is no ground-truth for the alignment difference R_0 , the estimation results have shown good consistency, which illustrates the robustness of our proposed method.

and camera, as shown in the Fig. 2. The sensor node and the top camera on the robot were attached together. We then moved the camera and the sensor node together to different orientations to evaluate the reproducibility of the proposed method, since it is challenging to find the ground truth of alignment difference in practice. To make sure the camera orientation is derivable, the calibration wands are always within the camera's field of view. Ten data sets have been acquired, and in each data set, the camera and sensor node were randomly placed at 10-20 different orientations facing the calibration wands. At each orientation, the camera and sensor node were kept stationary for at least 5s. Instead of using all the measurements for each orientation, only the mean value of these measurements was used to increase the signal-to-noise ratio (SNR).

Fig. 6 shows the estimation results of R_0 based on the 10 independent data sets. As we can see from the figure, the estimation results for R_0 are similar throughout all the trials performed, and the deviations are very small. The consistency



(a) Euler angle



(b) Euler angle difference

Figure 7. The orientation extracted from image frames, and the sensor based orientation estimation results given in the sensor frame and camera frame. The coordinate conversion was completed using the R_0 estimate from our two step iteration method and the traditional optimization method, respectively.

among all the 10 trials indicates the good repeatability of the proposed method. It is also worth mentioning that although there is no ground-truth for the alignment difference R_0

between the BSN sensor node and the top camera on the robot, the consistency of the data illustrates the robustness and reproducibility of our proposed method.

After applying the alignment calibration method to the BSN sensor nodes and the camera, we then projected the sensor based orientation estimation results back to the camera frame coordinate as $R_0 \Delta R_{i,j} R_0^T$, and compared the difference between the projection and $\Delta R_{c,j}$. The smaller the difference is, the more accurate the alignment calibration is. In our experiments, we rotated the sensor node and camera slowly to minimise the linear acceleration interference. Meanwhile, the rotation movement was within a small volume to make sure the magnetic field was constant. Therefore, the IMU sensor node can provide accurate orientation information in a short time using method presented in [27]. Meanwhile, the orientation in the camera coordinate can also be actually derived based on the method presented in [38]. The orientation derived in sensor coordinate system is then projected to the camera coordinate. Fig. 7(a) shows the sensor based orientation estimation results given in the sensor frame and camera frame. The red line is the orientation estimation derived from camera images, while the black one is orientation estimated from sensor measurement. The cyan, magenta and blue lines are the projection of the sensor orientation estimation to the camera coordinate using the hand eye calibration equation, where R_0 was given by the proposed two step iteration method and the traditional optimization method, respectively. Fig. 6(b) shows differences between the red line and the other four lines. It is evident that there are significant differences in the orientation estimation in the sensor coordinate system and camera coordinate system. This is mainly due to the rotational alignment difference between these two coordinate systems, which should be compensated before using the camera and sensor node together. It is obviously that the proposed two step iteration method can estimate the alignment difference between the BSN and the camera, and convert sensor orientation estimation to the camera frame accurately. We also noticed that although the converge speeds of optimization based methods are slower than our proposed iterative method, they can also provide accurate sensor frame to camera frame conversion. The quantitative comparison results between the orientation extracted from the images frames and three projections are shown in Table II. For comparison purpose, the quantitative result between the orientations in camera frame and sensor frame are also include in the Table. From the results derived, it is evident that the proposed method significantly reduces the effect of the sensor frame and camera frame alinement difference. There is also an excellent correlation between the orientation extracted from images and the one derive from sensor node after coordinate conversion.

The above analyses have shown that the proposed two step iteration method can transfer the sensor frame orientation results to camera coordinate preciously, which indicates that the calibration method can estimate rotational alignment difference between the inertial/magnetic sensor unit and the camera accurately.

IV. CONCLUSION AND FUTURE WORK

In conclusion, we focused on the estimation of rotational alignment difference between inertial/magnetic sensor unit and the camera in this paper. By exploiting the intrinsic restrictions among the coordinate transformations, the rotational alignment calibration problem was formulated as simplified hand-eye equation $AX = XB$. A two-step iterative algorithm was then derived to solve such hand-eye calibration task. Such method was then applied to align BSN sensor node with the top camera on a robot. The experimental results show that such rotational alignment difference can be estimated efficiently, and the sensor orientation estimation can be converted to the camera coordinate system accurately.

It is expected that the method will be used for a range of orientation estimation applications, including robotic navigation and human biomotion analysis. In the future, fusion of inertial/magnetic sensor units and camera images will be also investigated, particularly when there are long-term external interference for sensor unit and occlusion for camera.

REFERENCES

- [1] X. Meng, Z.-Q. Zhang, J.-K. Wu, and W.-C. Wong, "Hierarchical information fusion for global displacement estimation in microsensors motion capture," *Biomedical Engineering, IEEE Transactions on*, vol. 60, no. 7, pp. 2052–2063, 2013.
- [2] Z. Zhang, A. Panousopoulou, and G.-Z. Yang, "Wearable sensor integration and bio-motion capture: A practical perspective," in *Body Sensor Networks*. Springer, 2014, pp. 495–526.
- [3] B. Wang, Z. Deng, C. Liu, Y. Xia, and M. Fu, "Estimation of information sharing error by dynamic deformation between inertial navigation systems," *Industrial Electronics, IEEE Transactions on*, vol. 61, no. 4, pp. 2015–2023, April 2014.
- [4] W. Wang and G. Xie, "Online high-precision probabilistic localization of robotic fish using visual and inertial cues," *Industrial Electronics, IEEE Transactions on*, vol. 62, no. 2, pp. 1113–1124, Feb 2015.
- [5] A. Erdem and A. Ercan, "Fusing inertial sensor data in an extended kalman filter for 3d camera tracking," *Image Processing, IEEE Transactions on*, vol. 24, no. 2, pp. 538–548, Feb 2015.
- [6] G. Panahandeh and M. Jansson, "Vision-aided inertial navigation based on ground plane feature detection," *Mechatronics, IEEE/ASME Transactions on*, vol. 19, no. 4, pp. 1206–1215, Aug 2014.
- [7] G. Du, P. Zhang, and D. Li, "Human manipulator interface based on multisensory process via kalman filters," *Industrial Electronics, IEEE Transactions on*, vol. 61, no. 10, pp. 5411–5418, Oct 2014.
- [8] C. N. K. Nam, H. J. Kang, and Y. S. Suh, "Golf swing motion tracking using inertial sensors and a stereo camera," *Instrumentation and Measurement, IEEE Transactions on*, vol. 63, no. 4, pp. 943–952, April 2014.
- [9] M. Li and A. I. Mourikis, "3-d motion estimation and online temporal calibration for camera-imu systems," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 5709–5716.
- [10] Y. Tian, W. R. Hamel, and J. Tan, "Accurate human navigation using wearable monocular visual and inertial sensors," *Instrumentation and Measurement, IEEE Transactions on*, vol. 63, no. 1, pp. 203–213, 2014.
- [11] J. C. Chou and M. Kamel, "Finding the position and orientation of a sensor on a robot manipulator using quaternions," *The International Journal of Robotics Research*, vol. 10, no. 3, pp. 240–254, 1991.
- [12] K. Daniilidis, "Hand-eye calibration using dual quaternions," *The International Journal of Robotics Research*, vol. 18, no. 3, pp. 286–298, 1999.
- [13] Z. Zhao and Y. Liu, "A hand-eye calibration algorithm based on screw motions," *Robotica*, vol. 27, no. 02, pp. 217–223, 2009.
- [14] P. Lébraly, E. Royer, O. Ait-Aider, and M. Dhome, "Calibration of non-overlapping cameras - application to vision-based robotics," in *Proceedings of the British Machine Vision Conference*, 2010, pp. 10.1–10.12.

- [15] J.-S. Hu and Y.-J. Chang, "Automatic calibration of hand eye workspace and camera using hand-mounted line laser," *Mechatronics, IEEE/ASME Transactions on*, vol. 18, no. 6, pp. 1778–1786, Dec 2013.
- [16] T. Ruland, T. Pajdla, and L. Kruger, "Globally optimal hand-eye calibration," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, June 2012, pp. 1035–1042.
- [17] M. K. Ackerman, A. Cheng, E. Boctor, and G. Chirikjian, "Online ultrasound sensor calibration using gradient descent on the euclidean group," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4900–4905.
- [18] J. Heller, D. Henrion, and T. Pajdla, "Hand-eye and robot-world calibration by global polynomial optimization," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, May 2014, pp. 3157–3164.
- [19] H. Wu, W. Tizzano, T. T. Andersen, N. A. Andersen, and O. Ravn, "Hand-eye calibration and inverse kinematics of robot arm using neural network," in *Robot Intelligence Technology and Applications 2*. Springer, 2014, pp. 581–591.
- [20] U. Hubert, J. Stuckler, and S. Behnke, "Bayesian calibration of the hand-eye kinematics of an anthropomorphic robot," in *Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on*, Nov 2012, pp. 618–624.
- [21] C. Prasse, J. Stenzel, A. Böckenkamp, B. Rudak, K. Lorenz, F. Weichert, H. Müller, and M. ten Hompel, "New approaches for singularization in logistic applications using low cost 3d sensors," in *Sensing Technology: Current Status and Future Trends IV*. Springer, 2015, pp. 191–215.
- [22] A. Baak, T. Helten, M. Müller, G. Pons-Moll, B. Rosenhahn, and H.-P. Seidel, "Analyzing and evaluating markerless motion tracking using inertial sensors," in *Trends and Topics in Computer Vision*. Springer, 2012, pp. 139–152.
- [23] T. Helten, M. Muller, H.-P. Seidel, and C. Theobalt, "Real-time body tracking with one depth camera and inertial sensors," in *Computer Vision (ICCV), 2013 IEEE International Conference on*, Dec 2013, pp. 1105–1112.
- [24] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [25] S. Chen, "Kalman filter for robot vision: a survey," *Industrial Electronics, IEEE Transactions on*, vol. 59, no. 11, pp. 4409–4420, 2012.
- [26] J. Lee and E. Park, "Minimum-order kalman filter with vector selector for accurate estimation of human body orientation," *Robotics, IEEE Transactions on*, vol. 25, no. 5, pp. 1196–1201, 2009.
- [27] Z.-Q. Zhang, X.-L. Meng, and J.-K. Wu, "Quaternion-based kalman filter with vector selection for accurate orientation tracking," *Instrumentation and Measurement, IEEE Transactions on*, vol. 61, no. 10, pp. 2817–2824, 2012.
- [28] W. W. Hager and H. Zhang, "A new active set algorithm for box constrained optimization," *SIAM Journal on Optimization*, vol. 17, no. 2, pp. 526–557, 2006.
- [29] A. S. Nemirovski and M. J. Todd, "Interior-point methods for optimization," *Acta Numerica*, vol. 17, pp. 191–234, 2008.
- [30] J. Nocedal and W. S. J., *Numerical Optimization, Second Edition*. Springer Series in Operations Research, Springer Verlag, 2006.
- [31] C. B. Moler, "Least squares," in *Numerical Computing with Matlab*. Society for Industrial and Applied Mathematics, 2004, ch. 5, pp. 139–163.
- [32] Z.-Q. Zhang and G.-Z. Yang, "Calibration of miniature inertial and magnetic sensor units for robust attitude estimation," *Instrumentation and Measurement, IEEE Transactions on*, vol. 63, no. 3, pp. 711–718, 2014.
- [33] Analog Devices ADXL330. [Online]. Available: <http://www.analog.com/en/sensors/inertial-sensors/adxl330/products/product.html>
- [34] Invensense ITG-3200. [Online]. Available: <http://invensense.com/mems/gyro/itg3200.html>
- [35] Honeywell HMC5483. [Online]. Available: www.magneticsensors.com/datasheets/HMC5483.pdf
- [36] Z. Zhang and G. Yang, "Micromagnetometer calibration for accurate orientation estimation," *IEEE Trans. Biomed. Engineering*, vol. 62, no. 2, pp. 553–560, 2015.
- [37] Z.-Q. Zhang, "Two-step calibration methods for miniature inertial and magnetic sensor units," *Industrial Electronics, IEEE Transactions on*, vol. 62, no. 6, pp. 3714–3723, 2015.
- [38] E. Malis and M. Vargas, "Deeper understanding of the homography decomposition for vision-based control," Research Report RR-6303, 2007.



Zhi-Qiang Zhang received his B.E. degree in Computer Science and Technology from School of Electrical Information and Engineering, Tianjian University, China, in 2005, and the Ph.D. degree from the Sensor Network and Application Research Center, Graduate University, Chinese Academy of Sciences in 2010. He is currently a Research Associate in Department of Computing, Imperial College London. His research interests include Body Sensor Network, Information Fusion and Targets Tracking.