# Rotation Averaging with Application to Camera-Rig Calibration

Yuchao Dai[1,2], Jochen Trumpf[2], Hongdong Li[3,2], Nick Barnes[3,2], and Richard Hartley[2,3]

[1] School of Electronics and Information,Northwestern Polytechnical University
Shaanxi Key Laboratory of Information Acquisition and Processing, China
[2] Research School of Information Sciences and Engineering
The Australian National University
[3] Canberra Research Lab, NICTA **

**Abstract.** We present a method for calibrating the rotation between two cameras in a camera rig in the case of non-overlapping fields of view and in a globally consistent manner. First, rotation averaging strategies are discussed and an $L_1$-optimal rotation averaging algorithm is presented which is more robust than the $L_2$-optimal mean and the direct least squares mean. Second, we alternate between rotation averaging across several views and conjugate rotation averaging to achieve a global solution. Various experiments both on synthetic data and a real camera rig are conducted to evaluate the performance of the proposed algorithm. Experimental results suggest that the proposed algorithm realizes global consistency and a high precision estimate.

## 1 Introduction

Multiple-camera systems have recently received much attention from the computer vision community. Two typical scenarios of applying multi-camera systems are (1) multi-camera networks for surveillance and (2) multi-camera rigs for motion recovery and geometry reconstruction. This paper is exclusively concerned with the latter case of multiple individual cameras rigidly mounted on a rig. Example applications of multi-camera rigs include camera tracking, 3D city modeling or creation of image panoramas and structure from motion [1–3].

Multi-camera systems use a set of cameras which are placed rigidly on a moving object like a vehicle with possibly non-overlapping or only slightly overlapping fields of view. In this case, images captured by different cameras do not share any or only a few common points. The system moves rigidly and correspondences between subsequent frames taken by the individual cameras are captured before and after the motion.

** NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program. The first author would like to thank the Chinese Scholarship Council and Prof. Mingyi He for his immeasurable support and encouragement.

This non-overlapping arrangement poses difficulties in calibrating the multi-camera rig. Recent work done by Pollefeys et al. suggests a simple approach using a flat planar mirror [4]. Since it requires the use of a mirror, it is less convenient to use.

Esquivel et al. [5] proposed an approach for rig parameter estimation from non-overlapping views using sequences of time-synchronous poses of each camera. The presented approach works in three stages: internal camera calibration, pose estimation and rig calibration. They solve the problem using the relative motion measurements directly. However, according to our analysis, multiple relative motions are not consistent in general; using an averaging of motion strategy we can estimate the relative motion with high precision and in a globally consistent manner.

Our main contributions are: $L_1$-optimal rotation averaging strategy; $L_2$-optimal quaternion mean; global minimum with respect to the quaternion distance metric for the conjugate rotation problem and iterative rotation averaging for rotation calibration of multi-camera rig with non-overlapping views.

## 2 Existing works on rotation averaging

Given several estimates of relative orientation of coordinate frames, a posteriori enforcement of global consistency has been shown to be an effective method of achieving improved rotation estimates. Govindu seems to be the first who introduced the idea of motion averaging for structure-from-motion computation in computer vision. He published a series of papers addressing this problem [6–8]. In [6] a simple linear least squares method is proposed where rotations in SO(3) are parameterized by quaternions and a closed-form linear least squares solution is derived. Although Govindu made a claim of optimality, the linear solution is not in fact optimal because the linear solution can not require each quaternion in the solution to have unit norm. It also ignores the difficulty that both a quaternion and its negative represent the same rotation, which can sometimes cause the method to fail.

The paper [7] further developed the above linear method by following a non-linear optimization on manifold approach. Because the set of all rotations carries the structure of a Lie group, it makes more sense to define the distance between two rotations as the geodesic distance on that Lie group. Based on this, the averaged "mean rotation" should be defined with respect to the geodesic distance. It will be made clear later that, while our new methods to be presented share the same spirit in this regard, Govindu's Lie-averaging algorithm uses a first order approximation only, whereas our approach makes no such approximation. Similar Lie-averaging techniques have been applied to the distributed calibration of a camera network [9], and to generalized mean-shifts on Lie groups [10]. A generic mathematical exposition of this topic can be found in [11].

Another paper by Govindu [8] basically tackles robustness problems where a RANSAC-type approach is adopted. In the present paper we demonstrate that the $L_1$-distance can be directly used for this purpose as the $L_1$-distance is well known to be robust. What we really have achieved here is that we give an $L_1$-based averaging algorithm and prove its global convergence.

Martinec and Pajdla [12] discussed rotation averaging using the "chordal" metric, defined by $d_{\text{chord}}(\mathtt{R}_1, \mathtt{R}_2) = \|\mathtt{R}_1 - \mathtt{R}_2\|_{\text{Fro}}$. Averaging using the chordal metric suffers from similar problems to quaternion averaging. An analysis of averaging on SO(3) under the chordal metric has recently appeared [13].

When covariance uncertainty information is available for each local measurement, Agrawal shows how to incorporate such information in the Lie-group averaging computation [14]. Alternatively, one could apply the belief propagation framework to take the covariance information into account [15].

In the above discussions, the problem in question is to find the averaged rotation $\bar{\mathtt{R}}$ from a set of rotations $\{\mathtt{R}_1, \ldots, \mathtt{R}_n\}$ measured in the same coordinate frame. In this paper, we consider two more challenging rotation averaging problems: rotation averaging over several views and conjugate rotation averaging. In the case of *conjugate rotations*, the distance is defined as $d(\mathtt{R}_i\mathtt{S}, \mathtt{S}\mathtt{L}_i)$ where rotation pairs $\mathtt{R}_i, \mathtt{L}_i$ are given, and the rotation $\mathtt{S}$ is to be found. One traditional way to solve the conjugate-rotation problem is by solving a Sylvester equation treating each of the rotations as a generic $3 \times 3$ matrix (e.g. used in robot hand-eye calibration) [16].

Most of the papers on rotation averaging in the vision literature have omitted any discussion of optimality or global convergence. In addition, it seems they all overlooked the ambiguity of the sign problem associated with the quaternion representation, which invalidates previously known algorithms in some configurations. We have obtained rigorous conditions for convergence for most of our algorithms, though space does not allow us to include all proofs here.

## 3 Problem Formulation

We consider a camera rig consisting of two cameras, denoted left and right, fixed rigidly with respect to each other and individually calibrated. The camera rig undergoes rigid motion and captures several image pairs. We denote the coordinate frames of the cameras at time $i$ by $\mathtt{M}_i^L$ and $\mathtt{M}_i^R$, respectively.

$$\mathtt{M}_i^L = \begin{bmatrix} \mathtt{L}_i & \mathbf{t}_i^L \\ \mathbf{0}^\top & 1 \end{bmatrix} \text{ and } \mathtt{M}_i^R = \begin{bmatrix} \mathtt{R}_i & \mathbf{t}_i^R \\ \mathbf{0}^\top & 1 \end{bmatrix} .$$

The first three rows of these matrices represent the projection matrices of the corresponding cameras, where image points are represented in coordinates normalized by the calibration matrix.

We denote the relative motion of $\mathtt{M}_0^R$ with respect to $\mathtt{M}_0^L$ by a transformation $\mathtt{M}^{LR}$, such that $\mathtt{M}^{LR} = \mathtt{M}_0^R\,(\mathtt{M}_0^L)^{-1}$. Since this relative motion remains fixed throughout the motion, we observe that $\mathtt{M}^{LR} = \mathtt{M}_i^R\,(\mathtt{M}_i^L)^{-1}$ for all $i$.

Next, the relative motion of $\mathtt{M}_j^L$ with respect to $\mathtt{M}_i^L$ is denoted by $\mathtt{M}_{ij}^L = \mathtt{M}_j^L\,(\mathtt{M}_i^L)^{-1}$. Similarly, $\mathtt{M}_{ij}^R = \mathtt{M}_j^R\,(\mathtt{M}_i^R)^{-1}$. Using the relation $\mathtt{M}_i^R = \mathtt{M}^{LR}\,\mathtt{M}_i^L$, we find

$$\mathtt{M}_{ij}^R = \mathtt{M}^{LR}\,\mathtt{M}_{ij}^L\,(\mathtt{M}^{LR})^{-1} \tag{1}$$

for all $i, j$. Now, we denote

$$\mathtt{M}^L_{ij} = \begin{bmatrix} \mathtt{L}_{ij} & \mathbf{t}^L_{ij} \\ \mathbf{0}^\top & 1 \end{bmatrix} \text{ and } \mathtt{M}^R_{ij} = \begin{bmatrix} \mathtt{R}_{ij} & \mathbf{t}^R_{ij} \\ \mathbf{0}^\top & 1 \end{bmatrix} .$$

Observe that the relative rotations $\mathtt{R}_{ij}, \mathtt{L}_{ij}$ and relative translations $\mathbf{t}^R_{ij}, \mathbf{t}^L_{ij}$ may be computed via the essential matrix for the $(i, j)$ image pairs.

Writing the transformation $\mathtt{M}^{LR}$ as $\begin{bmatrix} \mathtt{S} & \mathbf{s} \\ \mathbf{0}^\top & 1 \end{bmatrix}$, we deduce from (1) the equations

$$\mathtt{R}_{ij} = \mathtt{S}\mathtt{L}_{ij}\mathtt{S}^{-1} \tag{2}$$
$$\mathbf{t}^R_{ij} = \mathtt{S}\mathbf{t}^L_{ij} + (\mathtt{I} - \mathtt{L}_{ij})\mathbf{s} \tag{3}$$

**Calibration strategy.** Our prescribed task is to find the relative motion between the right and left cameras, namely the transformation $\mathtt{M}^{LR}$. Our method uses the following general framework.

1. Compute the relative rotations and translations $(\mathtt{R}_{ij}, \mathbf{t}^R_{ij})$ $(\mathtt{L}_{ij}, \mathbf{t}^L_{ij})$ for many pairs $(i, j)$ using the essential matrix.
2. Compute the relative rotation $\mathtt{S}$ from (2).
3. Solve linearly for $\mathbf{s}$ using (3).

Both these equations may be solved linearly. The rotation equation may be written as $\mathtt{S}\mathtt{L}_{ij} = \mathtt{R}_{ij}\mathtt{S}$, which is linear in the entries of $\mathtt{S}$. In solving for the translation $\mathbf{s}$, we note that the relative translations $\mathbf{t}^L_{ij}$ and $\mathbf{t}^R_{ij}$ are known only up to scale factors $\lambda_{ij}$ and $\mu_{ij}$. Then (3) may be written more exactly as $\lambda_{ij}\mathbf{t}^R_{ij} = \mu_{ij}\mathtt{S}\mathbf{t}^L_{ij} + (\mathtt{I} - \mathtt{L}_{ij})\mathbf{s}$, where everything is known except for $\mathbf{s}$ and the scales $\lambda_{ij}$ and $\mu_{ij}$. Three image pairs are required to solve these equations and find $\mathbf{s}$.

The strategy outlined here is workable, but relies on accurate measurements of the rotations $\mathtt{L}_{ij}$ and $\mathtt{R}_{ij}$. In the following sections of this paper, we will explain our strategies for rotation averaging that will lead to significantly improved results in practice. Although we have implemented the complete calibration algorithm, including estimation of the translation $\mathbf{s}$, for the rest of this paper, we will consider only rotation estimation.

## 4 Averaging Rotations

The relative rotation estimates $\mathtt{R}_{ij}$ and $\mathtt{L}_{ij}$ obtained from individual estimates using the essential matrix will not be consistent. In particular, ideally, there should exist rotations $\mathtt{L}_i$, $\mathtt{R}_i$ and $\mathtt{S}$ such that $\mathtt{L}_{ij} = \mathtt{L}_j\mathtt{L}_i^{-1}$ and $\mathtt{R}_{ij} = \mathtt{R}_j\mathtt{R}_i^{-1} = \mathtt{S}\mathtt{L}_{ij}\mathtt{S}^{-1}$. If these two conditions are satisfied, then the relative rotation estimates $\mathtt{R}_{ij}$ and $\mathtt{L}_{ij}$ are consistent. In general they will not be, so we need to adjust them by a process of rotation averaging.

A distance measure $d : \mathrm{SO}(3) \times \mathrm{SO}(3) \rightarrow \mathbb{R}$ is called *bi-invariant* if $d(\mathtt{S}\mathtt{R}_1, \mathtt{S}\mathtt{R}_2) = d(\mathtt{R}_1, \mathtt{R}_2) = d(\mathtt{R}_1\mathtt{S}, \mathtt{R}_2\mathtt{S})$ for all $\mathtt{S}$ and $\mathtt{R}_i$. Given an exponent $p \geq 1$ and a set of

$n \geq 1$ rotations $\{\mathtt{R}_1, \ldots, \mathtt{R}_n\} \subset \mathrm{SO}(3)$ we define the $L^p$-mean rotation with respect to $d$ as

$$d^{\mathrm{p}}\text{-mean}(\{\mathtt{R}_1, \ldots, \mathtt{R}_n\}) = \underset{\mathtt{R} \in \mathrm{SO}(3)}{\operatorname{argmin}} \sum_{i=1}^{n} d^p(\mathtt{R}_i, \mathtt{R}). \tag{4}$$

### 4.1 The geodesic $L_2$-mean

The *geodesic* distance function $d_{\mathrm{geod}}(\mathtt{R}, \mathtt{S})$ is defined as the rotation angle $\angle(\mathtt{R}\mathtt{S}^\top)$. It is related to the angle-axis representation of a rotation in which a rotation is represented by the vector $\theta\mathbf{v}$, where $\mathbf{v}$ is a unit 3-vector representing the axis, and $\theta$ is the angle of rotation about that axis. We denote by $\log(\mathtt{R})$ the angle-axis representation of $\mathtt{R}$. Then $d(\mathtt{R}, \mathtt{S}) = \| \log(\mathtt{R}\mathtt{S}^\top) \|$. The inverse of this mapping is the exponential $\mathtt{R} = \exp(\theta\mathbf{v})$.

The associated $L_2$-mean is usually called the Karcher mean [17] or the geometric mean [11]. A necessary condition [11, (3.12)] for $\mathtt{R}$ to be a $d^2_{\mathrm{geod}}$-mean of $\{\mathtt{R}_1, \ldots, \mathtt{R}_n\}$ is given by $\sum_{i=1}^{n} \log(\mathtt{R}^\top \mathtt{R}_i) = 0$.

The mean is unique provided the given rotations $\mathtt{R}_1, \ldots, \mathtt{R}_n$ do not lie too far apart [17, Theorem 3.7], more precisely if $\{\mathtt{R}_1, \ldots, \mathtt{R}_n\}$ lie in an open ball $B(\mathtt{R}, \pi/2)$ of geodesic radius $\pi/2$ about some rotation $\mathtt{R}$. For this case Manton [18] has provided the following convergent algorithm where the inner loop of the algorithm is computing the average in the tangent space and then projecting back.

```
Set $\mathtt{R} := \mathtt{R}_1$. Choose a tolerance $\varepsilon > 0$.
loop
    Compute $\mathbf{r} := \frac{1}{n} \sum_{i=1}^{n} \log\left(\mathtt{R}^\top \mathtt{R}_i\right)$.
    if $\|\mathbf{r}\| < \varepsilon$ then
        return $\mathtt{R}$
    end if
    Update $\mathtt{R} := \mathtt{R} \exp(\mathbf{r})$.
end loop
```

**Algorithm 1**: computing the Karcher mean on SO(3)

### 4.2 The geodesic $L_1$-mean

Another interesting mean with respect to the geodesic distance $d_{\mathrm{geod}}$ is the associated $L_1$-mean

$$d_{\mathrm{geod}}\text{-mean}(\{\mathtt{R}_1, \ldots, \mathtt{R}_n\}) = \underset{\mathtt{R} \in \mathrm{SO}(3)}{\operatorname{argmin}} \sum_{i=1}^{n} d_{\mathrm{geod}}(\mathtt{R}_i, \mathtt{R}) , \tag{5}$$

which we might assume to be more robust to errors.

We propose a Riemannian gradient descent algorithm with geodesic line search to compute the $L_1$-mean. As long as Algorithm 2 avoids arbitrarily small but fixed $\delta$-neighborhoods of the $\mathtt{R}_i$s, convergence to the set of critical points of

```
1: Set $\mathtt{R} := d^2_{\text{geod}}$-mean$(\{\mathtt{R}_1, \ldots, \mathtt{R}_n\})$. Choose a tolerance $\varepsilon > 0$.
2: loop
3:     Compute $\mathbf{r} := \sum_{i=1}^n \log(\mathtt{R}^\top \mathtt{R}_i)/\|\log(\mathtt{R}^\top \mathtt{R}_i)\|$.
4:     Compute $s^* := \operatorname{argmin}_{s \geq 0} f(\mathtt{R}\exp(s\mathbf{r}))$.
5:     if $\|s\mathbf{r}\| < \varepsilon$ then
6:         return $\mathtt{R}$
7:     end if
8:     Update $\mathtt{R} := \mathtt{R}\exp(s\mathbf{r})$.
9: end loop
```

**Algorithm 2**: computing the geodesic $L_1$-mean on SO(3)

$f$ follows from [19, Corollary 4.3.2] applied to a modification of $f$ obtained by smoothing $f$ within those $\delta$-neighborhoods [20].

Note that possibly the easiest way to implement the line search in Step 4 is a Fibonacci search on a large enough interval. We have suggested to initialize the algorithm with the Karcher mean, but other initializations would of course be possible.

### 4.3 Quaternion averaging

A rotation $\mathtt{R}$ may be represented by a quaternion $\mathbf{r}$, which is a unit 4-vector, defined as follows. If $\mathbf{v}$ is the axis of the rotation and $\theta$ is the angle of the rotation about that axis, then $\mathbf{r}$ is defined as $\mathbf{r} = (\cos(\theta/2), \mathbf{v}\sin(\theta/2))$. We may think to define a distance $d_{\text{quat}}(\mathtt{S}, \mathtt{R})$ between two rotations to be $d_{\text{quat}}(\mathtt{R}, \mathtt{S}) = \|\mathbf{r} - \mathbf{s}\|$. Unfortunately, this simple equation will not do, since both $\mathbf{r}$ and $-\mathbf{r}$ represent the same rotation, and it is not clear which one to choose. However, this is resolved by defining

$$d_{\text{quat}}(\mathtt{R}, \mathtt{S}) = \min(\|\mathbf{r} - \mathbf{s}\|, \|\mathbf{r} + \mathbf{s}\|) \ .$$

Since quaternions satisfy the condition $\|\mathbf{r} \cdot \mathbf{t}\| = \|\mathbf{r}\| \, \|\mathbf{t}\|$, where $\mathbf{r} \cdot \mathbf{t}$ represents the quaternion product, it is easily verified that the quaternion distance is bi-invariant.

The relationship of this to the geodesic distance is as follows. Let $d_{\text{geod}}(\mathtt{R}, \mathtt{S}) = d_{\text{geod}}(\mathtt{I}, \mathtt{R}^\top \mathtt{S}) = \theta$, which is equal to the angle of the rotation $\mathtt{RS}^\top$. Then simple trigonometry provides the relationship $d_{\text{quat}}(\mathtt{I}, \mathtt{R}^\top \mathtt{S}) = 2\sin(\theta/4)$. For small rotations, we see that $d_{\text{quat}}(\mathtt{R}, \mathtt{S}) \approx d_{\text{geod}}(\mathtt{R}, \mathtt{S})/2$.

The following theorem shows how the $L_2$ quaternion mean of a set of rotations $\mathtt{R}_i$ may be computed, it is defined as $\operatorname{argmin}_{\mathtt{R}} \sum_{i=1}^n d^2_{\text{quat}}(\mathtt{R}, \mathtt{R}_i)$ [20].

**Theorem 1.** *Let $\mathtt{R}_i; i = 1, \ldots, n$ be rotations, and suppose that there exists a rotation $\mathtt{S}$ such that $d_{\text{geod}}(\mathtt{R}_i, \mathtt{S})$ is less than $\pi/2$. Let $\mathbf{r}_i$ be the quaternion representation of $\mathtt{R}_i$ chosen with sign such that $\|\mathbf{r}_i - \mathbf{s}\|$ is the smaller of the two choices. Then the $L_2$ quaternion mean of the rotations $\mathtt{R}_i$ is represented by the quaternion $\bar{\mathbf{r}}/\|\bar{\mathbf{r}}\|$, where $\bar{\mathbf{r}} = \sum_{i=1}^n \mathbf{r}_i$.*

### 4.4 The conjugate averaging problem

We now consider the problem of conjugate averaging. This problem is motivated by the second step of the calibration algorithm outlined in Section 3. The general form of this problem is as follows. Let $(\mathtt{R}_i, \mathtt{L}_i); i = 1, \ldots, n$ be pairs of rotations. (In Section 3 these rotations have two subscripts, namely $\mathtt{R}_{ij}, \mathtt{L}_{ij}$). The conjugate averaging problem is to find the rotation $\mathtt{S}$ that minimizes

$$\sum_{i=1}^{n} d^p(\mathtt{R}_i\mathtt{S}, \mathtt{S}\mathtt{L}_i) \ . \tag{6}$$

This problem has not been explicitly addressed in the context of multi-camera rigs, as far as we know, though it has been studied as the "hand-eye coordination problem" in robotics [16]. We give here an optimal solution for the $L_2$ quaternion distance metric under certain conditions.

We make the observation that if $\mathtt{R}_i$ and $\mathtt{L}_i$ are exactly conjugate, then they have the same rotation angle. In general, we assume that they do not differ by too much. One condition we need to give a closed form solution to this problem is that the rotations $\mathtt{R}_i$ and $\mathtt{L}_i$ should not be too large. In fact, we assume that the angle $\theta$ associated with $\mathtt{R}_i$ or $\mathtt{L}_i$ is less than some angle $\theta_{\max} < \pi$. For the application we are interested in, where $\mathtt{R}_i$ and $\mathtt{L}_i$ are relative rotations between two positions of a camera, the rotation angle of $\mathtt{R}_i$ can not be very large. If for instance the rotation $\mathtt{R}$ between two positions of a camera approaches $\pi$, then at least for normal cameras, there will be no points visible in both images, and hence no way to estimate the rotation $\mathtt{R}$. Normally, the rotation $\mathtt{R}_{ij}$ between two positions of the camera will not exceed the field of view of the camera, otherwise there will not be any matched points for the two cameras (except possibly for points lying between the two camera positions).

We now state the conditions under which we can guarantee an optimal solution to the conjugate averaging problem.

1. The rotations $\mathtt{L}_i$ and $\mathtt{R}_i$ satisfy the conditions $\angle(\mathtt{L}_i) < \theta_{\max}$ and $\angle(\mathtt{R}_i) < \theta_{\max}$.
2. In the optimal solution to problem (6), the errors $d_{\text{geod}}(\mathtt{R}_i\mathtt{S}, \mathtt{S}\mathtt{L}_i) < \alpha_{\max}$.
3. $\theta_{\max} + \alpha_{\max}/2 < \pi$.

Thus, we are assuming that the errors plus angles are not too large. In particular, since $\alpha_{\max} \leq \pi$, we see that the last two conditions always hold if $\theta_{\max} < \pi/2$.

**Linear solution.** We now outline a linear algorithm for estimating the matrix $\mathtt{S}$, under the $L_2$ quaternion distance. Let $\mathbf{r}_i$ and $\mathbf{l}_i$ be quaternion representatives of the rotations $\mathtt{R}_i$ and $\mathtt{L}_i$, chosen such that $\mathbf{r}_i = (\cos(\theta_i/2), \sin(\theta_i/2)\mathbf{v})$ with $\theta_i < \pi$. This means that the first component $\cos(\theta_i/2)$ of the quaternion is positive. This fixes the choice between $\mathbf{r}_i$ and $-\mathbf{r}_i$. We define $\mathbf{l}_i$ similarly.

Now, consider the equation $\mathtt{R}_i\mathtt{S} = \mathtt{S}\mathtt{L}_i$, and write it in terms of quaternions as $\mathbf{r}_i{\cdot}\mathbf{s}{-}\mathbf{s}{\cdot}\mathbf{l}_i = \mathbf{0}$. As before, $\cdot$ represents quaternion multiplication. Since quaternion multiplication is bilinear in terms of the entries of the two quaternions involved, this gives a homogeneous linear equation in terms of the entries of $\mathbf{s}$. Stacking all these equations into one and finding the solution such that $\|\mathbf{s}\| = 1$, we may solve for $\mathbf{s}$. This gives a simple linear way to solve this problem. Under the conditions stated above, we can prove that this algorithm finds the global minimum with respect to the quaternion distance metric [20].

### 4.5 Iterative Rotation Averaging for Camera Rig Calibration

The cost function that we minimize is the residual error in the rotation measurements $\mathtt{R}_{ij}$ and $\mathtt{L}_{ij}$, defined by

$$\min_{\mathtt{S},\mathtt{L}_i} \sum_{(i,j)\in\mathcal{N}} d^p(\mathtt{L}_{ij}, \mathtt{L}_j\mathtt{L}_i^{-1}) + d^p(\mathtt{R}_{ij}, \mathtt{S}\mathtt{L}_j\mathtt{L}_i^{-1}\mathtt{S}^{-1}) \qquad (7)$$

There seems to be no direct method of minimizing this cost function under any of the metrics we consider. Therefore, our strategy is to minimize the cost function by using rotation averaging to update each $\mathtt{L}_i$ in turn, then conjugate rotation averaging to find $\mathtt{S}$. At each step of this algorithm, the total cost decreases, and hence converges to a limit. We do not at present claim a rigorous proof that the algorithm converges to even a local minimum, though that seems likely under most reasonable conditions. In particular, the sequence of estimates must contain a convergent subsequence, and the limit of this subsequence must be at least a local minimum with respect to each $\mathtt{L}_i$ and $\mathtt{S}$ individually.

Initial values for each $\mathtt{L}_i$ are easily found by propagating from a given rotation $\mathtt{L}_0$ assumed to be the identity, and then obtaining the initial $\mathtt{S}$ through conjugate averaging.

The complete rotation estimation procedure follows.

```
Choose a tolerance $\epsilon > 0$.
Estimate initial values of $\mathtt{L}_i$ through rotation propagation.
Estimate $\mathtt{S}$ from $\mathtt{R}_{ij}\mathtt{S} = \mathtt{S}\mathtt{L}_{ij}$ solving the quaternion least squares problem.
loop
    Update each $\mathtt{L}_j$ in turn by averaging all the rotations $\mathtt{L}_{ij}\mathtt{L}_i$ and $\mathtt{S}^{-1}\mathtt{R}_{ij}\mathtt{S}\mathtt{L}_i$.
    Recompute and update $\mathtt{S}$ from the equation $\mathtt{R}_{ij}\mathtt{S} = \mathtt{S}\mathtt{L}_j\mathtt{L}_i^{-1}\mathtt{S}$ using conjugate
    rotation averaging.
    if the RMS error has decreased by less then $\epsilon$ since the last iteration, then
        return $S$
    end if
end loop
```

**Algorithm 3**: Iterative Rotation Averaging

## 5 Experiments

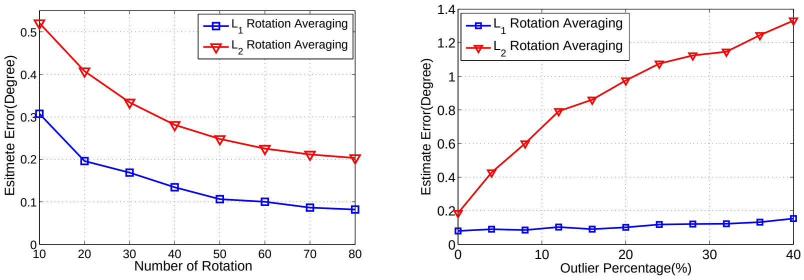
To evaluate the performance of the proposed algorithms, we conducted experiments on both synthetic data and real images. A comparison with other methods is presented to show the improved accuracy of the proposed method.

In the experiments reported below, we used $L_1$ and $L_2$ geodesic rotation averaging to compute each $\mathtt{L}_i$, but used quaternion averaging for the conjugate rotation averaging to compute $\mathtt{S}$. Theoretically this is not ideal, but we find it works well in practice and we will discuss all the possible combinations in future work.

### 5.1 Synthetic Rotation Averaging

In the first group of synthetic experiments, we evaluate the performance of $L_1$ rotation averaging and $L_2$ rotation averaging on a bunch of rotation measurements. First we generate a random rotation $\mathbf{r}$ and the corresponding rotation matrix $\mathtt{R}$. A normally distributed angle $\theta$ with mean 0 and standard deviation $\sigma$ is generated to simulate the effect of random rotation noise. The rotation axis is generated uniformly in the cube $[-1, 1]^3$ and then normalized to a unit vector $\mathbf{r}$. Then the rotation noise is expressed as $\theta\mathbf{r}$ and the corresponding rotation matrix is denoted $\mathtt{R}_{err}$. Finally the simulated rotation measurement is taken as $\mathtt{R}\mathtt{R}_{err}$.

All the results are obtained as the mean of 200 trials. The evaluation metric is the angle between the ground truth rotations and the estimates.



(a) (b)

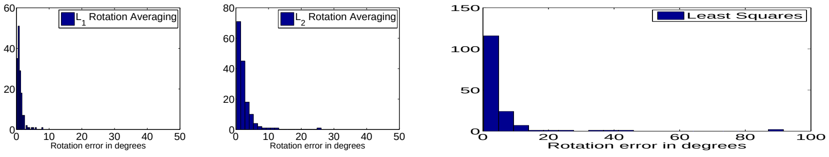
**Fig. 1.** Performance comparison of $L_1$-optimal rotation averaging and $L_2$-optimal rotation averaging. (a) Angle difference between the ground truth rotations and the averaging results for various numbers of rotations, where normally distributed rotation noise with standard deviation parameter $\sigma = 2$ degrees is added and no outliers are included. (b) Angle difference between the ground truth rotations and the averaging results on 100 rotations for various levels of outliers, where normally distributed rotation noise with standard deviation parameter $\sigma = 2$ degrees is added, and the outliers are simulated using normally distributed noise with standard deviation parameter $\sigma = 20$ degrees followed by selecting the samples with an angle error larger than 5 degrees.

From both figures in Figure 1 we conclude that the $L_1$-mean is more robust than the $L_2$-mean, especially in the presence of outliers.

### 5.2 Synthetic Camera Rig

To simulate a camera rig system, a rig with two cameras is generated with various numbers of frames. First the relatives rotation $\mathtt{S}$ of the camera rig is randomly generated. Second, the orientation $\mathtt{L}_i$ of the left camera is generated and the corresponding orientation $\mathtt{R}_i$ of the right camera is obtained. Third, whether a pair of frames has an epipolar geometry relationship is determined according to some probability distribution. If there exists epipolar geometry, the relative rotation measurement is obtained as $\mathtt{L}_{ij} = \mathtt{L}_j \mathtt{L}_i{}^{-1}$ and $\mathtt{R}_{ij} = \mathtt{R}_j \mathtt{R}_i{}^{-1}$. A random error rotation is applied to simulate noise in the rotation measurements.

To evaluate the performance of $L_1$-mean based rig rotation calibration, $L_2$-mean based rig rotation calibration and direct least squares rig rotation calibration, we conducted 200 separate experiments on synthetic camera rig data which contains 20 frames of motion. The possibility of existence of a relative measurement is 0.5 and 10% outliers are added where the rotation error is larger than 5 degrees. The histograms of the resulting errors are illustrated as Figure 2 and the histograms imply that our proposed $L_1$ rotation calibration estimates the rotation better than $L_2$ rotation calibration and direct least squares.



(a) (b) (c)

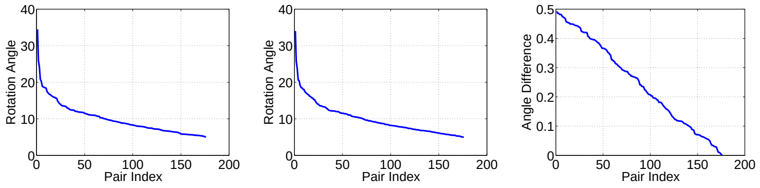
**Fig. 2.** (a) Histogram of rotation error using $L_1$ rotation averaging. It shows a mean of 1.12 degrees and standard deviation of 1.05 degrees. (b) Histogram of rotation error using $L_2$ rotation averaging. It shows a mean of 2.41 degrees and standard deviation of 2.75 degrees. (c) Histogram of rotation error using direct least squares. It shows a mean of 5.14 degrees and standard deviation of 11.65 degrees.

### 5.3 Experiments on Real Images

As a real example of a two-camera rig system, we have used a pair of wide-angle cameras to capture sequences of images. Images are captured at each camera illustrated in Figure 3. Feature points on the images are extracted using SIFT and tracked through image sequences. These tracked features are transformed to image vectors on the unit sphere given the individual intrinsic calibrations. Outliers in the tracked features are removed using RANSAC [21] to fit the essential matrix using the normalized 8 point algorithm. Pairwise relative pose is obtained through decomposition of the essential matrix, and two frames bundle adjustment is utilized to refine the estimate, thus obtaining the relative rotations $\mathtt{L}_{ij}, \mathtt{R}_{ij}$. Finally, $L_1$ and $L_2$ algorithms are applied to calibrate the camera rig, obtaining the relative rotation $\mathtt{S}$ and relative translation $\mathbf{s}$.

**Fig. 3.** Images captured by camera rig with non-overlapping views

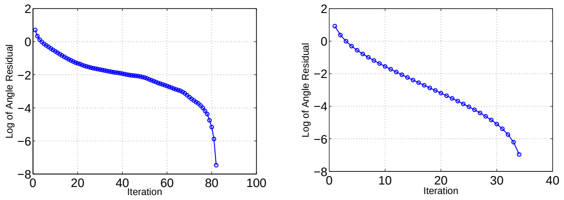
The image sequences captured by the left camera and the right camera contain 200 frames individually. As some pairs of image frames do not supply relative motion estimates, we ultimately obtained 11199 pairs of relative motion estimates. Since relative rotation estimates $\mathtt{L}_{ij}$ and $\mathtt{R}_{ij}$ should have equal angle rotations, we use this criterion along with a minimum rotation angle requirement to select the best image pairs for further processing. After rotation selection, we obtained 176 pairs of synchronized motions. The distributions of the rotation angles and angle differences for these pairs are shown in Figure 4.



(a) (b) (c)

**Fig. 4.** (a) Angle distribution of the left camera. (b) Angle distribution of the right camera. (c) Distribution of the difference between the angle of the left camera and the angle of the right camera

The convergence process is shown in Figure 5 with the $L_1$ rotation averaging and quaternion conjugate result corresponding to an angle of 143.1° and the $L_2$ rotation averaging result corresponding to an angle of 169.4°. Measured from the scene, the ground truth is about 140°.



(a) (b)

**Fig. 5.** Convergence process on real camera rig image sequences. (a) Log of Angle Residual of $L_1$ rotation averaging. (b) Log of Angle Residual of $L_2$ rotation averaging.

## 6 Conclusion and Future Work

Rotation averaging is an important component of our method of camera rig calibration. Individual rotation estimation is sensitive to outliers and geometrically critical configurations. It was shown that our new $L_1$ rotation averaging method gives markedly superior results to $L_2$ methods. Global bundle adjustment is recommended for final polishing. Previous computer vision literature has largely ignored issues such as convergence and optimality of rotation averaging algorithms. We have addressed this issue. Our complete analysis will be made available in an extended version of this paper.

## References

1. Kim, J.H., Li, H., Hartley, R.: Motion estimation for multi-camera systems using global optimization. In: Proc. Comput. Vis. Pattern Recognit. (2008) 1–8
2. Kim, J.H., Hartley, R., Frahm, J.M., Pollefeys, M.: Visual odometry for non-overlapping views using second-order cone programming. In: Proc. Asian Conf. on Computer Vision. (2007) 353–362
3. Li, H., Hartley, R., Kim, J.H.: A linear approach to motion estimation using generalized camera models. In: Proc. Comput. Vis. Pattern Recognit. (2008) 1–8
4. Kumar, R., Ilie, A., Frahm, J.M., Pollefeys, M.: Simple calibration of non-overlapping cameras with a mirror. In: Proc. Comput. Vis. Pattern Recognit. (2008) 1–7
5. Esquivel, S., Woelk, F., Koch, R.: Calibration of a multi-camera rig from non-overlapping views. In: DAGM07. (2007) 82–91
6. Govindu, V.M.: Combining two-view constraints for motion estimation. In: Proc. Comput. Vis. Pattern Recognit. (2001) 218–225
7. Govindu, V.M.: Lie-algebraic averaging for globally consistent motion estimation. In: Proc. Comput. Vis. Pattern Recognit. (2004) 684–691
8. Govindu, V.M.: Robustness in motion averaging. In: Proc. Asian Conf. on Computer Vision. (2006) 457–466
9. Tron, R., Vidal, R., Terzis, A.: Distributed pose averaging in camera networks via consensus on SE(3). In: Second ACM/IEEE International Conference on Distributed Smart Cameras. (2008) 1–10
10. Tuzel, O., Subbarao, R., Meer, P.: Simultaneous multiple 3d motion estimation via mode finding on Lie groups. In: Int. Conf. Computer Vision. (2005) 18–25
11. Moakher, M.: Means and averaging in the group of rotations. SIAM J. Matrix Anal. Appl. **24**(1) (2002) 1–16 (electronic)
12. Martinec, D., Pajdla, T.: Robust rotation and translation estimation in multiview reconstruction. In: Proc. Comput. Vis. Pattern Recognit. (2007) 1–8
13. Sarlette, A., Sepulchre, R.: Consensus optimization on manifolds. SIAM J. Control Optim. **48**(1) (2009) 56–76
14. Agrawal, M.: A Lie algebraic approach for consistent pose registration for general euclidean motion. In: Int. Conf. Intelligent Robots and Systems. (2006) 1891–1897
15. Devarajan, D., Radke, R.J.: Calibrating distributed camera networks using belief propagation. EURASIP J. Appl. Signal Process. **2007**(1) (2007)
16. Park, F., Martin, B.: Robot sensor calibration: solving AX=XB on the euclidean group. Robotics and Automation, IEEE Transactions on **10**(5) (1994) 717–721
17. Grove, K., Karcher, H., Ruh, E.A.: Jacobi fields and Finsler metrics on compact Lie groups with an application to differentiable pinching problems. Math. Ann. **211** (1974) 7–21
18. Manton, J.H.: A globally convergent numerical algorithm for computing the centre of mass on compact Lie groups. In: Proceedings of the Eighth Int. Conf. on Control, Automation, Robotics and Vision, Kunming, China (2004) 2211–2216
19. Absil, P.A., Mahony, R., Sepulchre, R.: Optimization algorithms on matrix manifolds. Princeton University Press, Princeton, NJ (2008)
20. Dai, Y., Trumpf, J., Li, H., Barnes, N., Hartley, R.: On rotation averaging in multi-camera systems. Technical report, Northwestern Polytechnical University and Australian National University (2009) To Appear.
21. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. Commun. Assoc. Comp. Mach. **24** (1981) 381–395