Supplementary Material: Analysis of Rotational Deformations from Directional Data

Jörn Schulz[1], Sungkyu Jung[2], Stephan Huckemann[3],
Michael Pierrynowski[4], J. S. Marron[5] and Stephen M. Pizer[5]

[1]University of Tromsø, [2]University of Pittsburgh, [3]University of Göttingen,

[4]McMaster University, Hamilton [5]University of North Carolina at Chapel Hill

March 26, 2014

1 Additional data analysis

In this section, some additional data analyses are discussed, in particular using datasets from point distribution models.

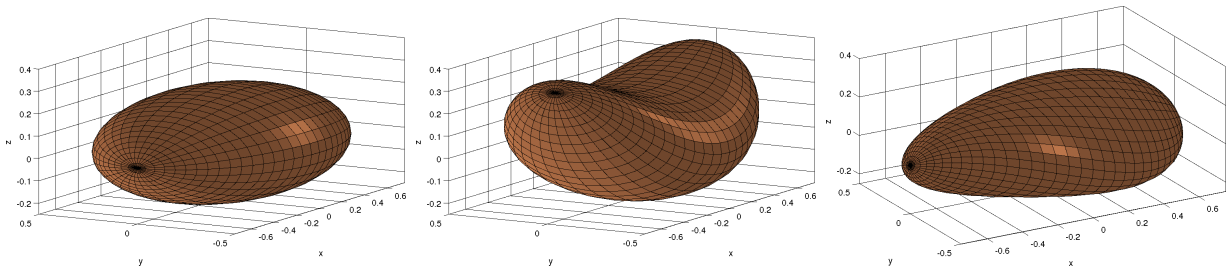
1.1 Rotationally deformed ellipsoids

We first discuss the procedures of obtaining the raw ellipsoid data. To avoid notational confusion, we denote a random vector by X_{ij} and their observed values by χ_{ij}, $i = 1, \ldots, n, j = 1, \ldots, K$. A point in $\mathbb{R}^3$ is described by $(x, y, z) \in \mathbb{R}^3$ in a fixed Cartesian coordinate system.

The surface of an ellipsoid can be parameterized by

$$\mathbf{s}(u,v) = \begin{pmatrix} x(u,v) \\ y(u,v) \\ z(u,v) \end{pmatrix} = \begin{pmatrix} r_a \sin(v) \\ r_b \sin(u)\cos(v) \\ r_c \cos(u)\cos(v) \end{pmatrix}, \qquad u \in [-\pi, \pi), v \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \tag{1.1}$$

where $r_a \geq r_b \geq r_c > 0$ are the length of the axes. We assume a default ellipsoid of $r_a = 0.75$, $r_b = 0.5$ and $r_c = 0.25$. For a parameter space $\Omega = [-\pi, \pi) \times [-\frac{\pi}{2}, \frac{\pi}{2}] \subset \mathbb{R}^2$, $\mathbf{s} : \Omega \to \mathbb{R}^3$ is



(a) Original (b) Bent (c) Twisted

Figure 1.1: Tube views of meshed surfaces of (a) an original ellipsoid with $r_a = 0.75$, $r_b = 0.5$ and $r_c = 0.25$, (b) bent ellipsoid with $\alpha = 0.6$, (c) twisted ellipsoid with $\theta = 0.6$.

a surface map in $\mathbb{R}^3$ with $(u, v) \xrightarrow{\mathbf{s}} (x(u, v), y(u, v), z(u, v))$. Without loss of generality, we assume that the first to third principal axes are x, y, z-axis in $\mathbb{R}^3$, respectively. The centroid of the ellipsoid is at origin $(0, 0, 0)'$.

Rotational bending around the y-axis ($c_b = (0, 1, 0)'$) is given by

$$B(\mathbf{s}) = R(c_b, f_b(x))\mathbf{s}(u, v), \tag{1.2}$$

and twisting around the x-axis ($c_t = (1, 0, 0)'$) is given by

$$T(\mathbf{s}) = R(c_t, f_t(x))\mathbf{s}(u, v), \tag{1.3}$$

where

$$\begin{aligned} R(c_b, f_b(x)) &= \begin{pmatrix} \cos(f_b(x)) & 0 & -\sin(f_b(x)) \\ 0 & 1 & 0 \\ \sin(f_b(x)) & 0 & \cos(f_b(x)) \end{pmatrix}, \\ R(c_t, f_t(x)) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(f_t(x)) & -\sin(f_t(x)) \\ 0 & \sin(f_t(x)) & \cos(f_t(x)) \end{pmatrix}. \end{aligned}$$

Here, $f_b(x) = \alpha x$ and $f_t(x) = \theta x$ for some $\alpha, \theta \in [-\pi/2, \pi/2]$ representing the overall size of the deformation. The amount of bending and twisting depends on the location on the x-axis. In addition to the rotational bending, we also consider quadratic bending around the y-axis which is defined by

$$B_q(\mathbf{s}) = (x, y, z + \alpha x^2)', x, y, z, \in \mathbb{R} \tag{1.4}$$

using the ellipsoid parametrization (1.1). In the following, the term *bending* is used for

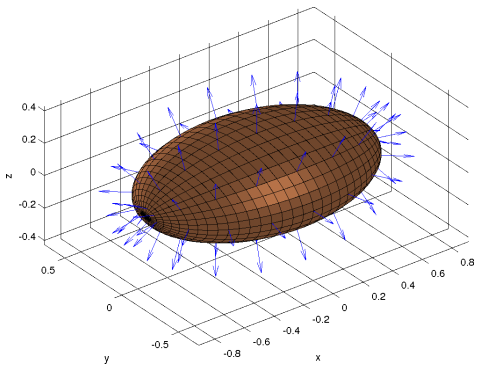


Figure 1.2: Visualization of a standard ellipsoid with attached boundary normals.

rotational bending, and *quadratic bending* will be specified explicitly.

Figure 1.1 shows an example of an original ellipsoid, bent ellipsoid with $\alpha = 0.6$ and twisted ellipsoid with $\theta = 0.6$.

1.2 Point distribution and boundary normals

We now discuss a parameterization of ellipsoids by a point distribution model. In particular, a mesh grid and attached boundary normals of the surface $\mathbf{s}(u, v)$ will be used. See Fig. 1.2.

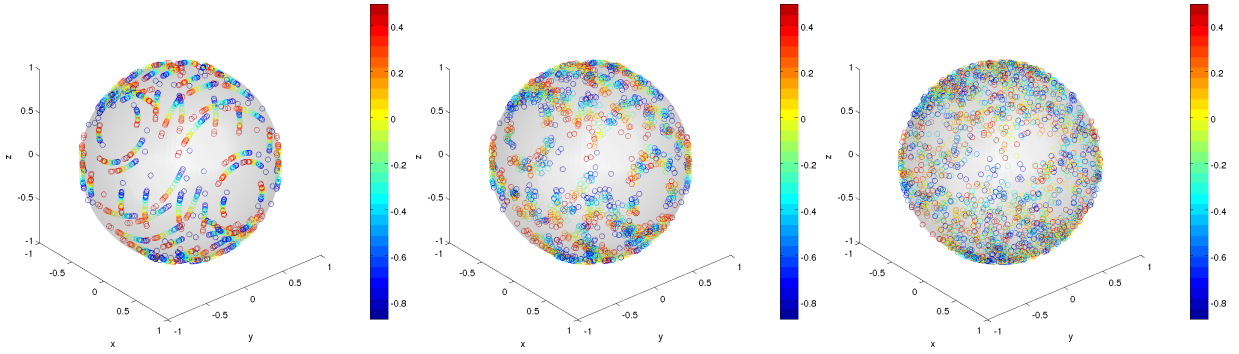
The size of surface mesh we used is 37×33. We chose $K = 9 \times 8 = 72$ vertices to attach normal direction vectors $\chi_{k_1 k_2} \in S^2, k_1 = 1, \ldots, 9, k_2 = 1, \ldots, 8$. For each k_1, the vertices where $\chi_{k_1 k_2}$ are attached have common x-coordinates

$$\{-0.738, -0.649, -0.482, -0.256, 0, 0.256, 0.482, 0.649, 0.738\},$$

the values of which influence the amount of deformation. Note that there are some directions that will not vary when the object is deformed. For example, the normals χ_{5k_2} $(1 \leq k_2 \leq 8)$, which are attached to vertices with zero x-coordinate, will stay still when the object is twisted or bent.

In the following we report four sets of experiments on the boundary normal ellipsoid data. As opposed to the s-rep data analysis, we are working with the raw data directly. The noise in the data is from the von Mises–Fisher distribution (Mardia and Jupp, 2000) with concentration parameter $\kappa > 0$. For each experiment, two levels of noise are used: $\kappa = 100, 1000$. The four sets of models are

- *Twisting* by (1.3), with $c_t = (1, 0, 0)'$, $\theta \sim N(0, \sigma_\theta^2)$, $\sigma_\theta = 0.3 \approx 17.2°$. See Fig. 1.3.
- *Bending* by (1.2) with $c_b = (0, 1, 0)'$, $\alpha \sim N(0, \sigma_\alpha^2)$, $\sigma_\alpha = 0.4 \approx 22.9°$. See Fig. 1.4.



(a) without noise (b) $\kappa = 1000$ (c) $\kappa = 100$

Figure 1.3: (Twisting) Boundary normals using different concentration parameter κ. The rotation around the x-axis is clearly visible in (a). Different colors represent different amounts of twisting parameter θ.

- *Quadratic bending* by (1.4) with above c_b and σ_α. See Fig. 1.5.
- *Hierarchical deformation* by bending (primary) and twisting (secondary) with the same set of parameters above. See Fig. 1.7.

From each model a random sample of size $n = 30$ or 100 is obtained, from which the estimate $\hat{c}$ of the axis and $\hat{\sigma}$ are obtained. This is repeated for 1000 times.

Twisting

Figure 1.3 shows 30 samples of 72 different normal directions from sets of twisted ellipsoids with different noise levels. Different colors represent different amounts of twisting parameter θ. Therefore, the number of colors correspond to the sample size.

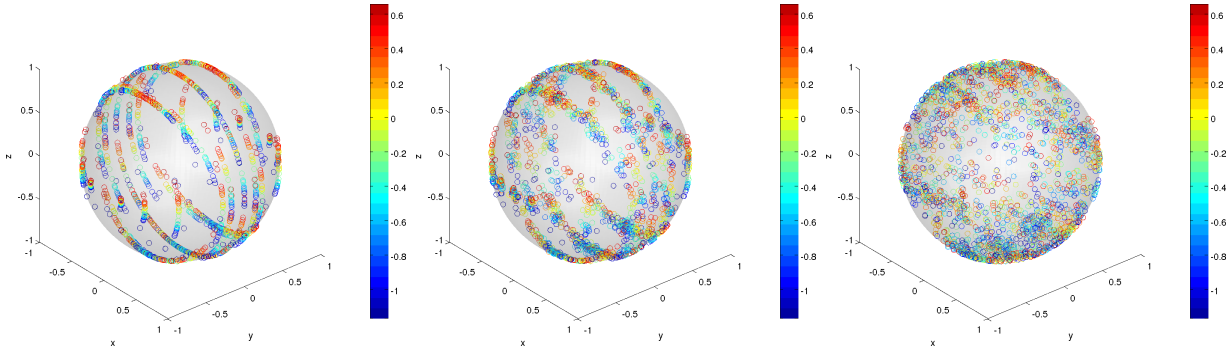
Table 1.1 shows the performance of our estimator based on 1000 simulations. The performance is measured by the mean and standard deviation of the absolute error $d_g(\hat{c}, c)$ and those of the estimated twisting parameter $\hat{\sigma}_\theta$. The accuracy is increased for larger n or κ. In general, we observe quite accurate estimates even for a larger noise level ($\kappa = 100$).

Bending

Figure 1.4 shows 30 samples of 72 different normal directions after rotational bending using different noise levels. The case of quadratic bending is illustrated in Fig. 1.5. Different colors represent different amounts of bending parameter α. The different effects of rotational and quadratic bending to the boundary normals can be compared by examination of Fig. 1.4a and Fig. 1.5a. Rotationally bent directions are at concentric small circles (Fig. 1.4a) while

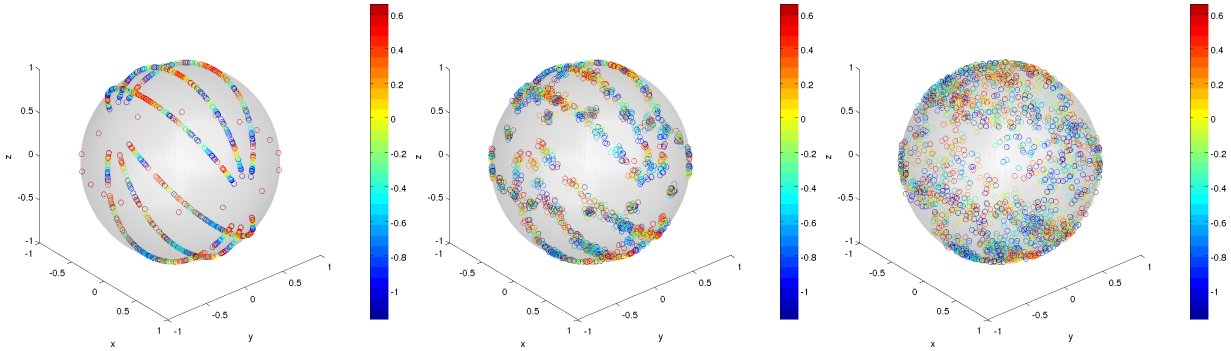
Table 1.1: Twisting: Mean absolute error for $\hat{c}$, and the estimates $\hat{\sigma}_\theta$

Twisting (unit: degrees)			
κ	n	$d_g(\hat{c}, c)$	$\hat{\sigma}_\theta(\sigma_\theta = 17.189)$
100	30	3.174 (2.294)	17.209 (2.152)
100	100	1.563 (0.890)	17.324 (1.250)
1000	30	0.561 (0.317)	17.045 (2.133)
1000	100	0.289 (0.164)	17.173 (1.235)



(a) without noise (b) $\kappa = 1000$ (c) $\kappa = 100$

Figure 1.4: (Rotational bending) Boundary normals using different concentration parameter κ. Different colors represent different amounts of bending parameter α.



(a) without noise (b) $\kappa = 1000$ (c) $\kappa = 100$

Figure 1.5: (Quadratic bending) Boundary normals using different concentration parameter κ. Different colors represent different amounts of bending parameter α.

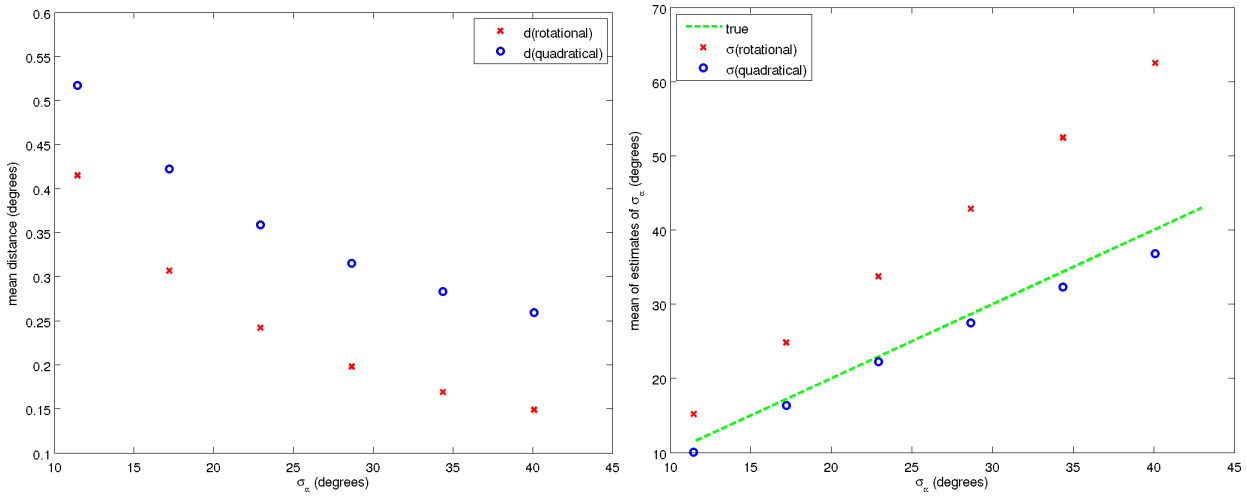
quadratically bent directions are at circles with different centers (Fig. 1.5a). Despite the major violation of our assumption of concentric circles, the proposed method surprisingly works well for quadratic bending models, as Table 1.2 summarizes.

Table 1.2 shows the performance of our estimator for the rotational and quadratic bending models. The absolute errors between the true axis c and the estimated rotation axis $\hat{c}$

Table 1.2: Rotational bending: Mean absolute error for $\hat{c}$, and the estimates $\hat{\sigma}_\alpha$

(unit: degrees)		Rotational bending		Quadratic bending	
κ	n	$d_g(\hat{c}, c)$	$\hat{\sigma}_\alpha(\sigma_\alpha = 22.918)$	$d_g(\hat{c}, c)$	$\hat{\sigma}_\alpha(\sigma_\alpha = 22.918)$
100	30	0.898 (0.492)	34.133 (4.429)	1.494 (0.871)	23.277 (3.389)
100	100	0.467 (0.261)	34.179 (2.592)	0.789 (0.470)	22.880 (2.454)
1000	30	0.242 (0.127)	33.739 (4.448)	0.359 (0.213)	22.203 (3.184)
1000	100	0.127 (0.069)	33.973 (2.581)	0.193 (0.112)	22.276 (1.891)

are virtually small for both models. The performance of the estimator is enhanced for larger κ and n. Moreover, as expected, the empirical errors are smaller for rotational bending than quadratic bending. Note that the estimates $\hat{\sigma}_\alpha$ of rotational bending model are biased upwards, which we discuss in section 1.2.1. The parameter σ_α affects the quality of estimators. In particular, larger σ_α leads to a greater bias of $\hat{\sigma}_\alpha$, meanwhile it yields a better estimate of $\hat{c}$ (Fig. 1.6).

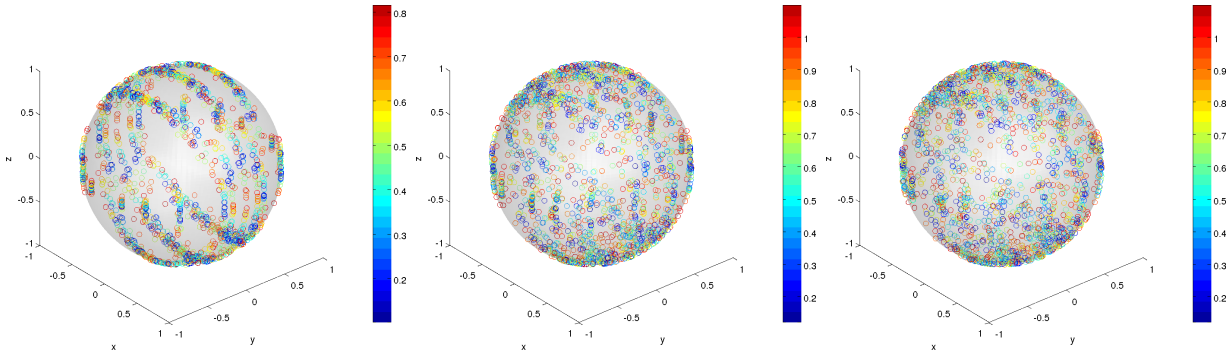


(a) (b)

Figure 1.6: (a) Mean absolute error $d_g(\hat{c}, c)$ after quadratic and rotational bending for different σ_α values. (b) Mean of estimated bending parameter $\hat{\sigma}_\alpha$ after quadratic and rotational bending for different σ_α values. The green dashed line marks the first diagonal.

Hierarchical motion

Figure 1.7 shows 30 samples of 64 different normal directions using different concentration parameters κ, twistings σ_θ and a fixed bending parameter $\sigma_\alpha = 0.4$. We have excluded 8 normal directions attached at $x = 0$. Different colors represent different amounts of absolute rotation, which are $|\alpha_j| + |\theta_j|$. In Fig. 1.7a we see rotations of normals along small circles



(a) $\epsilon_j = 0, \sigma_\theta = 0.1$ (b) $\epsilon_j = 0, \sigma_\theta = 0.3$ (c) $\kappa = 1000, \sigma_\theta = 0.3$

Figure 1.7: (Hierarchical deformation) Boundary normals using $\sigma_\alpha = 0.4$, different concentration parameter κ and twisting σ_θ.

around the y-axis. The clear motion pattern disappeared after an increase of σ_θ and κ in Figure 1.7b and Figure 1.7c.

Table 1.3 shows the performance of our estimator based on 1000 simulations under hierarchical rotational bending and twisting. As expected, the rotation axis estimates are less accurate than for single motions. Moreover, the estimate of the secondary rotation axis is less accurate than the estimate of the primary axis, particularly for $\kappa = 100$. The bias in the estimates of the rotation angle will be further discussed shortly.

Two initial value choices (from Principal Arc Analysis and random directions) are applied. In contrast to the s-rep analysis in the main article, the results in Table 1.3 are very similar for both choices.

In the four sets of analyses, we have shown accurate estimation results of the proposed method for the boundary normal data. In particular, the estimation from the quadratic bending model is surprisingly accurate despite the misspecified model. We believe that the proposed method will lead to robust estimates in other deformation models, which are similar to a rotational deformation.

1.2.1 Estimation of σ_α

A precise estimation of σ_α under the bending model is an interesting open problem. We have observed that the amount of swing is nonlinear, and conjecture that the change of the surface curvature in the object is responsible for the additional swing of the directions. Figure 1.8 exemplifies such a non-linear relationship.

In the case of rotational bending, we may assume that our estimate $\hat{\sigma}_\alpha$ and the parameter

Table 1.3: (Rotational bending and twisting) Mean absolute error for $\hat{c}_1$, $\hat{c}_2$, and the estimates $\hat{\sigma}_\alpha$, $\hat{\sigma}_\theta$.

(unit: degrees)		1st rotation axis		2nd rotation axis	
κ	n	$d_g(\hat{c}_1, c_1)$	$\hat{\sigma}_\alpha(\sigma_\alpha = 22.918)$	$d_g(\hat{c}_2, c_2)$	$\hat{\sigma}_\theta(\sigma_\theta = 17.189)$
initialization by 1st and 2nd principal component					
100	30	3.526 (2.775)	18.125 (2.370)	20.047 (16.717)	9.232 (1.598)
100	100	1.902 (1.444)	18.268 (1.337)	11.081 (13.477)	9.239 (1.143)
1000	30	2.683 (2.272)	17.785 (2.377)	8.570 (9.031)	9.119 (1.323)
1000	100	1.637 (1.126)	18.101 (1.349)	3.901 (2.459)	9.367 (0.691)
initialization by 1st principal component and a random direction					
100	30	3.496 (2.762)	18.125 (2.367)	19.133 (15.445)	9.295 (1.498)
100	100	1.866 (1.390)	18.260 (1.342)	8.944 (6.753)	9.390 (0.798)
1000	30	2.678 (2.272)	17.785 (2.377)	8.479 (8.702)	9.125 (1.299)
1000	100	1.635 (1.127)	18.102 (1.349)	3.892 (2.451)	9.367 (0.691)

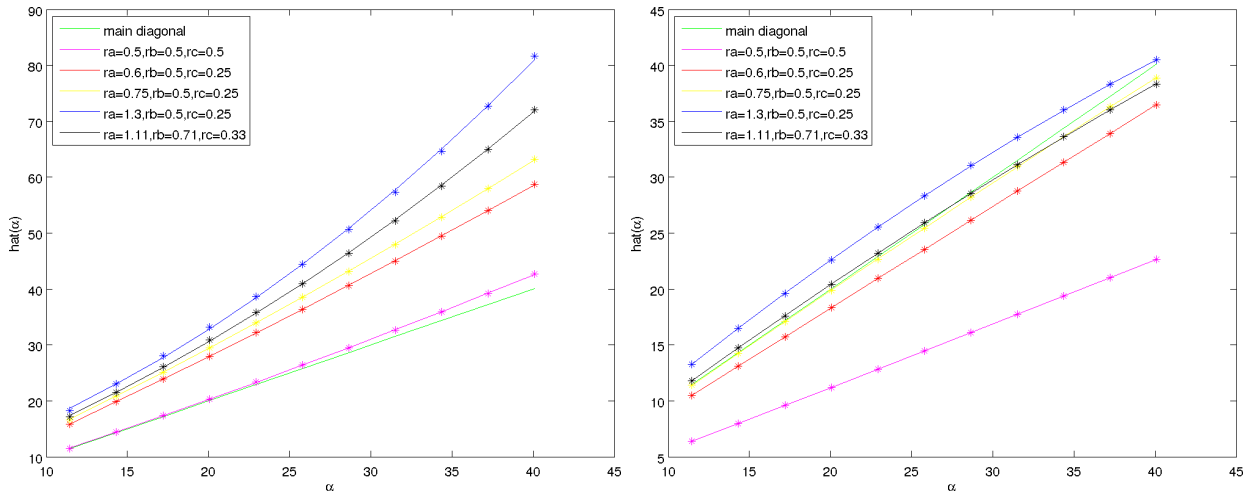
σ_α are related through a quadratic function as Fig.1.8a suggests. If such a quadratic function $f(\sigma_\alpha) = p_0 + p_1\sigma_\alpha + p_2\sigma_\alpha^2 = \hat{\sigma}_\alpha$ is known or can be estimated efficiently by a least square quadratic polynomial, one can correct the estimate of σ_α for fixed r_a, r_b and r_c by

$$\hat{\sigma}_\alpha^{\text{new}} = -\frac{1}{2p_2} \pm \sqrt{\frac{1}{4p_2^2}(p_1^2 - 4p_0p_2 + 4p_2\hat{\sigma}_\alpha)},$$

with $f''(\hat{\sigma}_\alpha^{\text{new}}) \geq 0$. A similar modification can be made for quadratic bending (Fig.1.8b).

In general, we believe that modeling based on the curvatures will improve our current estimator. Such a modeling is beyond the scope of this paper, and we list a few important notions of curvature that have potential in modeling.

Most common curvature measures are the principal curvatures (κ_1, κ_2) with $\kappa_1 \geq \kappa_2$, Gaussian curvature, and mean curvature. These measures are calculated from the first and second fundamental form (see Gray (1998) and Kühnel (2006)). Koenderink (1990) indicated that Gaussian curvature and mean curvature are not representatives of local shapes because Gaussian curvature is identical for all local approximations for which the ratio of the principal curvatures (κ_1, κ_2) is equal. Therefore, he introduced the two alternative quantities: *shape*



(a) Rotational bending (b) Quadratic bending

Figure 1.8: Relationship between $\hat{\alpha}$ and α for various (r_a, r_b, r_c).

index S and *curvedness* C,

$$S = \frac{2}{\pi} \tan^{-1}\left(\frac{\kappa_2 + \kappa_1}{\kappa_2 - \kappa_1}\right), \qquad (\kappa_1 \geq \kappa_2) \tag{1.5}$$

$$C = \frac{2}{\pi} \ln\left(\sqrt{\frac{1}{2}(\kappa_1^2 + \kappa_2^2)}\right). \tag{1.6}$$

1.3 Quadratic bending of s-reps

The proposed method is also applied to the fitted s-reps of quadratically bent ellipsoids. After fitting s-reps to the raw images discussed in Section 5.1 of the main article, we obtained the estimated axis of rotation of $\hat{c}_b = (-0.026, 0.999, 0.002)$ with a distance of 1.517° to the true y-axis $c_b = (0, 1, 0)$ and a standard error of $\hat{\text{se}}_B(\hat{c}_b) = 0.86°$. Similar to mesh ellipsoid surfaces, the distance of the rotation axis estimate to the true axis is lower in case of rotational bending compared to quadratic bending.

1.4 Standard error estimation using bootstrap

In general, the bootstrap algorithm for estimating standard errors can be formulated as follows:

1. Select B independent bootstrap samples $\tilde{\mathbf{x}}^{(1)}, \ldots, \tilde{\mathbf{x}}^{(B)}$ drawn from $\mathbf{x} = (x_1, \ldots, x_n)$ with $x_i = (x_{ij})_{j=1,\ldots,K}$.

2. Evaluate the bootstrap replications by the statistic $s(\mathbf{x}^{(b)})$ for all $b \in \{1, \ldots, B\}$.

3. Estimate the standard error by the standard deviation of the B replications

$$\hat{\text{se}}_B = \left(\frac{1}{B-1} \sum_{b=1}^{B} [s(\mathbf{x}^{(b)}) - s(\mathbf{x}^{(\cdot)})]^2 \right)^{\frac{1}{2}}$$

where $s(\mathbf{x}^{(\cdot)}) = \frac{1}{B} \sum_{b=1}^{B} s(\mathbf{x}^{(b)})$.

In our case, the bootstrap replications result in estimates $\hat{c}^{(1)}, \ldots, \hat{c}^{(B)}$ with $\hat{c}^{(b)} \in S^2$ where the number of bootstrap replications B is chosen by $B = 200$. The mean $\hat{c}_g^{(\cdot)}$ is given by the Fréchet mean

$$\hat{c}_g^{(\cdot)} = \underset{x \in S^2}{\operatorname{argmin}} \sum_{b=1}^{B} d_g^2(x, \hat{c}^{(b)}).$$

Thereby, the standard error estimate of the rotation axis is given by

$$\hat{\text{se}}_B(\hat{c}) = \left(\frac{1}{B-1} \sum_{b=1}^{B} d_g^2(\hat{c}^{(b)}, \hat{c}_g^{(\cdot)}) \right)^{\frac{1}{2}}.$$

The standard error of the standard deviation $\hat{\text{se}}_B(\hat{\sigma})$ can be estimated by plug-in $s(\mathbf{x}^{(b)}) = \hat{\sigma}^{(b)}$ in the above equation.

2 Simulation results for the hierarchical deformation model

Table 2.1 summarizes the numerical performance of estimation of the hierarchical deformation model (11) as discussed in Section 4 of the main article. We have used the data-driven method to choose the initial values as described in Section 3 of the main article. The results are comparable to those from the simpler models in Section 4 and are fairly successful.

3 Discussion of model bias

A possibly important issue that is left unanswered is whether the parameters c_0 and $r = d_g(c_0, \mu)$ of the model

$$X = R(c_0, \theta)\mu \oplus \epsilon \tag{3.1}$$

Table 2.1: Numerical performance over 1000 replications, for hierarchical deformations.

Hierarchical rotations (unit: degrees)			$d_g(\hat{c}_1, c_1)$	$d_g(\hat{c}_2, c_2)$	$\hat{\sigma}_\theta(\sigma_\theta = 22.5)$	$\hat{\sigma}_\psi(\sigma_\psi = 15)$
Body 1	$\kappa = 100$	$n = 30$	4.527 (3.591)	4.929 (3.007)	22.693 (3.031)	15.890 (2.096)
		$n = 100$	2.201 (1.206)	2.944 (1.550)	22.698 (1.589)	15.844 (1.126)
	$\kappa = 1000$	$n = 30$	2.084 (1.364)	1.275 (0.701)	22.385 (2.984)	14.940 (1.941)
		$n = 100$	1.123 (0.741)	0.652 (0.338)	22.492 (1.542)	15.030 (1.110)
Body 2	$\kappa = 100$	$n = 30$	2.617 (1.762)	3.066 (3.099)	22.440 (2.959)	15.094 (2.011)
		$n = 100$	1.366 (0.746)	1.682 (0.870)	22.542 (1.562)	15.219 (1.073)
	$\kappa = 1000$	$n = 30$	1.099 (1.171)	0.921 (2.349)	22.339 (2.983)	14.872 (1.945)
		$n = 100$	0.568 (0.354)	0.438 (0.236)	22.470 (1.543)	14.981 (1.099)

are the minimizer $(\tilde{c}, \tilde{r})$ of the least squares problem

$$\min_{c,r} \mathbb{E}\rho^2\{\delta(c,r), X\} = \min_{c,r} \mathbb{E}\{d_g(c,X) - r\}^2. \tag{3.2}$$

Rivest (1999) has shown that when c_0 is known, the minimizer $\tilde{r} = \text{argmin}_r\, \mathbb{E}\{d_g(c_0, X) - r\}^2$ is biased towards $\pi/2$, i.e. $\tilde{r} > r = d_g(c_0, \mu)$ if $r < \pi/2$. In fact, given any c for the axis of rotation, $\tilde{r}_c = \mathbb{E}\{d_g(c,X)\}$ minimizes $\mathbb{E}\{d_g(c,X) - r\}^2$. Now to focus on c, $\min_{c,r} \mathbb{E}\{d_g(c,X) - r\}^2 = \min_c \mathbb{E}\{d_g(c,X) - \tilde{r}_c\}^2 = \min_c \text{Var}\{d_g(c,X)\}$. Therefore c_0 of (3.1) is the minimizer of (3.2) if for all $c \in S^2$,

$$\text{Var}\{d_g(c_0, X)\} \leq \text{Var}\{d_g(c,X)\}. \tag{3.3}$$

This inequality may be satisfied when $r\sigma_\theta$ is large compared to the standard deviation of the error ϵ. If σ_θ or r is 0, there is no variation due to the rotation of $R(c_0, \theta)$, which makes the model unidentifiable. Heuristically, small σ_θ makes the estimation difficult. Likewise, the variation due to rotation is small if the rotation radius $r = d_g(c, \mu)$ is small. The standard deviation of the length $r\theta$ of the arc on $\delta(c,r)$ is $r\sigma_\theta$. Hence, the hypothesis:

(H) If (3.3) is not satisfied, the least-squares estimator may be biased.

While we have not succeeded in finding analytic forms of conditions that satisfy (3.3), a simulation study has been carried out to support our hypothesis (H). For $c_0 = e_3 = (0,0,1)'$, $\mu_r = \mu(r) = \cos(r)c_0 + \sin(r)e_1$, and $\theta \overset{\text{iid}}{\sim} N(0, \sigma_\theta^2)$, X is distributed as the von Mises–Fisher distribution with center $R(c,\theta)\mu_r$ and the concentration parameter $\kappa = 100$. We then evaluated the minimizer $\tilde{c}$ of $\text{Var}\{d_g(c,X)\}$ based on 5×10^5 Monte-Carlo random observations of X, for different combinations of $r \in \{\pi/16, \pi/8, \pi/4, \pi/3, \pi/2\}$ and $\sigma_\theta \in \{0.1, 0.2, 0.5, 1\}$ in radian. The result of the experiment is summarized as the distance

Table 3.1: Distance to true axis (measured in degrees)

	σ_θ				
r	0.01	0.1	0.2	0.5	1
$\pi/16 = 11.25$	11.25	11.19	10.99	9.10	2.29
$\pi/8 = 22.50$	22.50	22.36	21.89	2.55	0.20
$\pi/4 = 45.00$	44.98	44.81	42.73	0.22	0.00
$\pi/3 = 60.00$	59.97	59.79	3.72	0.16	0.00
$\pi/2 = 90.00$	90.00	90.00	0.49	0.02	0.02

between $\tilde{c}$ and c_0 in Table 3.1.

It can be checked from Table 3.1 that the distance between $\tilde{c}$ and c_0 is smaller for larger values of σ_θ and r. Moreover, for sufficiently large σ_θ and r, $d_g(\tilde{c}, c_0) = 0$, which leads to c_0 from the model (3.1) satisfying (3.3). On the other hand, when σ_θ and r are small with respect to the variance of the error, the rotation (3.1) becomes unidentifiable. This is further illustrated at Figure 3.1, which shows the scatter of 100 random observations from model (3.1), with $(r, \sigma_\theta) = (\pi/16, 0.2)$ on the left and $(\pi/4, 0.5)$ on the right panel. The left panel suggests that when (r, σ_θ) are small, the rotation about the axis c_0 is not distinguished and the optimal $\tilde{c}$ is near μ and $d_g(\tilde{c}, c_0)$ is large (cf. Table 3.1). The right panel illustrates that the rotation is visually identified for large values of (r, σ_θ), and for such a case, the parameter c_0 is the solution of the least squared problem (3.2).

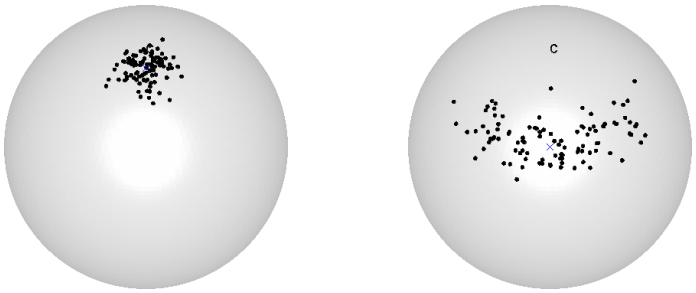


Figure 3.1: The scatter of 100 random observations from the model (3.1), with $(r, \sigma_\theta) = (\pi/16, 0.2)$ on the left and $(\pi/4, 0.5)$ on the right panel. The north pole, depicted as c in the figure, is the true axis of rotation c_0. The blue x visualizes μ.

4 Numerical studies with misspecified parameters

In this section, we study the impact of parameter misspecification of the estimator, particularly of the function f_j that model the relationships between the rotation angles θ_j. We study parameter misspecification over different rotational deformation situations as described in

Section 4 in the main article.

Recall, Section 4 in the main article reports studies for two different objects. The first object (Body 1) consists of $K = 4$ directions, while the second object (Body 2) contains $K = 8$ directions. The von Mises–Fisher distribution is used for the distribution of errors. Three rotation models (indexed by equation number in the main article) are considered for each object:

- Model (2)–*Rigid rotation*: $c = (1, 0, 0)'$, $\theta_j = f_j(\theta) = \theta$, where $\theta \sim N(0, \sigma_\theta^2)$ and $\sigma_\theta = \pi/12 \approx 15°$.
- Model (10)–*Twisting*: $c = (0, 1, 0)'$, $\theta_j = f_j(\theta) = 1_{j\in I_1}\theta - 1_{j\in I_2}\theta$, where $\theta \sim N(0, \sigma_\theta)^2$, $\sigma_\theta = \pi/8 \approx 22.5°$. Here, I_1 and I_2 are the partitions of $\{1, \ldots, K\}$ according to the right and left sides whereas $I_1 = \{1, 2\}$ and $I_2 = \{3, 4\}$ for Body 1 and $I_1 = \{1, \ldots, 4\}$ and $I_2 = \{5, \ldots, 8\}$ for Body 2.
- Model (11)–*Hierarchical deformations*: $c_1 = (1, 0, 0)'$, $c_2 = (1/\sqrt{2}, -1/\sqrt{2}, 0)'$, $\theta_j = \theta$ and $\psi_j = f_j(\psi) = 1_{j\in I_1}\psi - 1_{j\in I_2}\psi$, where $\theta \sim N(0, \sigma_\theta)^2$, $\sigma_\theta \approx 22.5°$ and $\psi \sim N(0, \sigma_\psi)^2$, $\sigma_\psi \approx 15°$. The I_1 and I_2 are the same partition used in the twisting model above.

The hierarchical model represents deformations by a rigid rotation and oblique twist. For each model, we generate $n = 30,100$ rotationally deformed objects with different error concentration parameters $\kappa = 100, 1000$. These are replicated 1000 times, and the estimation quality is measured by $d_g(\hat{c}, c)$ and $\hat{\sigma}_\theta = \sum_{i=1}^n \hat{\theta}_i^2/n$.

The estimation results using correct parameters are reported for the single deformation models in Table 1 in the main article and for the hierarchical deformation model in Table 2.1 above.

4.1 Parameter f_j

The modeling of the function f_j can be challenging as discussed in Section 1.2.1 or for the s-rep model in case of bent, and bent and twisted ellipsoids in chapter 5 in the main article. Therefore, it is crucial to study the impact of misspecification of f_j.

Table 4.1 reports mean and standard deviations of the measures for the single deformation models. The true underlying rigid rotation deformation is indicated by $f_j(\theta) = \theta, j = 1, \ldots, 4$ for Body 1 and is misspecified by $f_j(\theta) = \theta, j = 1, 2$ and $f_j(\theta) = -\theta, j = 3, 4$. Body 2 is misspecified by $f_j(\theta) = (j/10)\theta$ instead of $f_j(\theta) = \theta$ for $j = 1, \ldots, 8$. In both cases the

Table 4.1: Numerical performance over 1000 replications, for single deformation models.

Rotation (unit: degrees)		Body 1 $d_g(\hat{c}, c)$	Body 1 $\hat{\sigma}_\theta$ ($\sigma_\theta = 15$)	Body 2 $d_g(\hat{c}, c)$	Body 2 $\hat{\sigma}_\theta$ ($\sigma_\theta = 15$)
$\kappa = 100$	$n = 30$	4.133 (2.26)	3.314 (0.44)	2.905 (1.60)	6.771 (0.85)
	$n = 100$	2.235 (1.18)	3.308 (0.24)	1.560 (0.83)	6.816 (0.51)
$\kappa = 1000$	$n = 30$	1.166 (0.64)	1.037 (0.13)	0.841 (0.46)	6.698 (0.88)
	$n = 100$	0.656 (0.34)	1.039 (0.07)	0.448 (0.22)	6.744 (0.46)
Twisting (unit: degrees)		Body 1 $d_g(\hat{c}, c)$	Body 1 $\hat{\sigma}_\theta$ ($\sigma_\theta = 22.5$)	Body 2 $d_g(\hat{c}, c)$	Body 2 $\hat{\sigma}_\theta$ ($\sigma_\theta = 22.5$)
$\kappa = 100$	$n = 30$	2.761 (1.51)	3.669 (0.48)	4.062 (3.67)	17.150 (2.12)
	$n = 100$	1.482 (0.78)	3.658 (0.25)	1.889 (1.35)	17.055 (1.27)
$\kappa = 1000$	$n = 30$	0.803 (0.43)	1.139 (0.15)	1.017 (0.74)	16.760 (2.19)
	$n = 100$	0.446 (0.23)	1.147 (0.08)	0.536 (0.36)	16.857 (1.17)

misspecification does not effect the estimated rotation axis $\hat{c}$ but leads to a poor prediction of $\hat{\sigma}_\theta$. The true underlying twisting motion of model (10) is given by $f_j(\theta) = \theta, j = 1, 2,\ f_j(\theta) = -\theta, j = 3, 4$ for Body 1 and is misspecified by $f_j(\theta) = \theta, j = 1, 4,\ f_j(\theta) = -\theta, j = 2, 3$. Body 2 is misspecified by $f_j(\theta) = \theta,\ f_l(\theta) = -0.5\theta$ whereas the true parameter is given by $f_j(\theta) = \theta,\ f_l(\theta) = -\theta$ for $j = 1, \ldots, 4$ and $k = 5, \ldots, 8$. The comparison of Table 4.1 to Table 1 in the main article shows that a misspecification of the parameter f_j does not effect the rotation axis estimation of a single deformation by fitting concentric small circles whereas the predicted rotation angle is biased by misspecification of f_j. The specification of f_j models the relationships between the rotation angles θ_j and is therefore crucial for their prediction.

On the other hand, the partition I_1 and I_2 is not explicitly used in the estimation procedure of the rotation axis. The partition I_1 and I_2 is implied by the function f_j. A partition I_1 and I_2 of indices $\{1, \ldots, K\}$ represents sets of direction vectors that rotate together, i.e., the partition models the deformation type. In the simulated examples, the partitions are chosen to model the bending and twisting deformation accordingly. The specification of f_j gives the possibility to incorporate additional prior knowledge about the statistical model of the rotation angle θ_j, e.g., linear or quadratic relation by the distance of the direction to the rotation axis. Nevertheless, the modeling of the function f_j can be challenging as discussed before. There are real data applications where the definition of a partition is naturally motivated, e.g., by the physical structure of the body. An example is to estimate the rotational deformation between the upper and lower leg as studied in Section 6 in the main article. This example can be extended to all joints inside the human body and to all objects which are connected by a joint. On the other hand, there is a group of data sets where such a partition is not obvious, e.g., in the s-rep model of the hippocampus. A first approach could

Table 4.2: True and misspecified parameter f_j for Body 1 in the hierarchical deformation model (11).

	$f_{11}(\theta)$	$f_{12}(\theta)$	$f_{13}(\theta)$	$f_{14}(\theta)$	$f_{21}(\psi)$	$f_{22}(\psi)$	$f_{23}(\psi)$	$f_{24}(\psi)$
True	θ	θ	θ	θ	ψ	ψ	$-\psi$	$-\psi$
Table 4.4	θ	θ	$-\theta$	$-\theta$	ψ	ψ	ψ	ψ
Table 4.5	θ	$-\theta$	$-\theta$	θ	$-\psi$	ψ	ψ	$-\psi$

Table 4.3: True and misspecified parameter f_j for Body 2 in the hierarchical deformation model (11).

Hierarchical rotations		j							
		1	2	3	4	5	6	7	8
True	$f_{1j}(\theta)$	θ	θ	θ	θ	θ	θ	θ	θ
	$f_{2j}(\psi)$	ψ	ψ	ψ	ψ	$-\psi$	$-\psi$	$-\psi$	$-\psi$
Table 4.4	$f_{1j}(\theta)$	0.5θ	0.5θ	0.5θ	0.5θ	0.5θ	0.5θ	0.5θ	0.5θ
	$f_{2j}(\psi)$	ψ	ψ	ψ	ψ	$-\psi$	$-\psi$	$-\psi$	$-\psi$
Table 4.5	$f_{1j}\theta)$	0.1θ	0.2θ	0.3θ	0.4θ	θ	θ	θ	θ
	$f_{2j}(\psi)$	ψ	ψ	0.7ψ	0.7ψ	-0.2ψ	-0.2ψ	-0.3ψ	-0.3ψ

be to define the partition on the basis of an observation of a medical expert. An automatic detection of partitions and clusters is an interesting future research direction.

In contrast to the single deformation model, a misspecification of f_j has an impact in the hierarchical deformation model by the iterative back-and-forward deformations of the random direction vectors. Therefore, a misspecification of the parameter might guide the iterative estimation procedure to fall in a local minimum, and leads to a poor estimation of the rotation axes and angles. Table 4.2 and Table 4.3 report the true and misspecified f_j for the simulation studies whereas the estimation results are summarized in Table 4.4 and Table 4.5. At first we have changed the order of the original parameters f_1, f_2 for Body 1 in Table 4.4. The first estimated rotation axis $\hat{c}_1$ is around $(1/\sqrt{2}, -1/\sqrt{2}, 0)'$ and the second estimated rotation axis $\hat{c}_2$ is around $(1, 0, 0)'$, i.e., the estimator has interchanged the true underlying deformations which results in a distance of approximately 45 degree of $\hat{c}_1$ to c_1 and $\hat{c}_2$ to c_2. Nevertheless, the order of generalized rotational deformations are not interchangeable in general, and a misspecification might bias the results. In a second set, we have misspecified f_{1j} globally by a factor of 0.5 and kept the original f_{2j} for Body 2. The means and standard deviations in Table 4.4 show only small changes compared to Table 2.1 and are very accurate. A global scale change does not change the relationships between the rotation angles. Both cases demonstrate the performance of the hierarchical estimation procedure in case of a moderate misspecification by reasonable estimates.

In addition, two cases with drastically misspecified parameters are reported. In a third

Table 4.4: Numerical performance over 1000 replications, for hierarchical deformations using moderate misspecified parameters.

Hierarchical rotations (unit: degrees)			$d_g(\hat{c}_1, c_1)$	$d_g(\hat{c}_2, c_2)$	$\hat{\sigma}_\theta(\sigma_\theta = 22.5)$	$\hat{\sigma}_\psi(\sigma_\psi = 15)$
Body 1	$\kappa = 100$	$n = 30$	45.223 (3.61)	44.792 (3.69)	15.013 (2.00)	22.725 (3.01)
		$n = 100$	45.188 (1.73)	44.423 (1.79)	14.992 (1.06)	22.731 (1.58)
	$\kappa = 1000$	$n = 30$	43.648 (1.29)	44.257 (2.67)	14.454 (1.89)	22.442 (2.99)
		$n = 100$	43.705 (0.72)	43.981 (1.42)	14.482 (1.06)	22.510 (1.54)
Body 2	$\kappa = 100$	$n = 30$	2.617 (1.76)	3.066 (3.09)	22.440 (2.96)	15.094 (2.01)
		$n = 100$	1.100 (1.17)	0.921 (2.35)	22.339 (2.98)	14.872 (1.94)
	$\kappa = 1000$	$n = 30$	1.366 (0.74)	1.683 (0.87)	22.542 (1.56)	15.219 (1.07)
		$n = 100$	0.569 (0.35)	0.438 (0.23)	22.470 (1.54)	14.981 (1.09)

Table 4.5: Numerical performance over 1000 replications, for hierarchical deformations using drastically misspecified parameters..

Hierarchical rotations (unit: degrees)			$d_g(\hat{c}_1, c_1)$	$d_g(\hat{c}_2, c_2)$	$\hat{\sigma}_\theta(\sigma_\theta = 22.5)$	$\hat{\sigma}_\psi(\sigma_\psi = 15)$
Body 1	$\kappa = 100$	$n = 30$	11.576 (4.66)	33.737 (4.94)	5.423 (2.58)	2.619 (3.45)
		$n = 100$	11.272 (2.38)	33.372 (2.48)	4.749 (1.02)	0.966 (1.44)
	$\kappa = 1000$	$n = 30$	11.228 (3.91)	33.586 (4.06)	3.597 (1.98)	1.186 (2.34)
		$n = 100$	11.183 (2.19)	33.445 (2.25)	3.295 (0.32)	0.413 (0.26)
Body 2	$\kappa = 100$	$n = 30$	12.204 (5.01)	33.456 (5.11)	15.917 (1.88)	214.219 (20.16)
		$n = 100$	11.337 (2.46)	33.959 (2.47)	16.180 (1.15)	214.553 (10.75)
	$\kappa = 1000$	$n = 30$	11.838 (4.40)	33.787 (4.46)	15.727 (2.01)	219.414 (20.99)
		$n = 100$	11.289 (2.30)	33.978 (2.34)	16.316 (1.20)	219.587 (10.45)

scenario, f_1 and f_2 are misspecified so as to generate different deformations for Body 1 in Table 4.5. Both the means and the standard deviations show poor estimation results. In a last set we modified f_1 and f_2 by keeping the underlying deformation direction of each direction vector for Body 2 but changing the amount of deformation locally. Also in this scenario the hierarchical estimator shows poor estimation results in Table 4.5.

4.2 Primary and secondary rotation

The hierarchical model is a first attempt to model and estimate more then one rotational deformation. The order of two rotations is not interchangeable and is considered as a part of the statistical model which attempts to describe the nature as well as possible. Therefore, we assume a primary rotation $R(c_1, \theta_j)$ and a secondary rotation $R(c_2, \theta_j)$ in our hierarchical model. The order of rotation can be misspecified in two different ways in the proposed approach. At first, we might interchange f_1 and f_2 as discussed in Section 4.1 above for Body 1 in Table 4.4. In addition, the primary and secondary rotation has to be specified for

Table 4.6: Numerical performance over 1000 replications, for hierarchical deformations with misspecified order of primary and secondary rotation axis.

Hierarchical rotations (unit: degrees)			$d_g(\hat{c}_1, c_1)$	$d_g(\hat{c}_2, c_2)$	$\hat{\sigma}_\theta(\sigma_\theta = 22.5)$	$\hat{\sigma}_\psi(\sigma_\psi = 15)$
Body 1	$\kappa = 100$	$n = 30$	4.624 (2.66)	4.876 (2.85)	22.720 (3.00)	15.896 (2.10)
		$n = 100$	2.258 (1.30)	2.949 (1.54)	22.701 (1.58)	15.847 (1.12)
	$\kappa = 1000$	$n = 30$	2.044 (1.27)	1.279 (0.70)	22.382 (2.98)	14.947 (1.94)
		$n = 100$	1.124 (0.72)	0.656 (0.33)	22.492 (1.54)	15.033 (1.11)
Body 2	$\kappa = 100$	$n = 30$	2.590 (1.47)	2.992 (1.75)	22.439 (2.96)	15.094 (2.01)
		$n = 100$	1.323 (0.72)	1.688 (0.87)	22.541 (1.56)	15.220 (1.07)
	$\kappa = 1000$	$n = 30$	1.063 (0.67)	0.849 (0.47)	22.336 (2.98)	14.873 (1.94)
		$n = 100$	0.567 (0.35)	0.438 (0.23)	22.470 (1.54)	14.982 (1.09)

the initialization.

Table 4.6 summarizes the results in estimation of the hierarchical deformation model (11) with interchanged primary and secondary rotation for the initialization. The results are similar to Table 2.1. The estimator converges to the same results in this example. Nevertheless, a misspecification might increase the risk that the estimation procedure converges in a different local minimum.

4.3 Discussion

The introduction of f_j in our model has advantages and disadvantages. We study generalized rotational deformations on the basis of directional data, and the rotation of a direction from one to another place on the sphere is not uniquely defined in the hierarchical case. Therefore, different functions f_j can describe different rotational deformations. Prior knowledge is necessary in order to restrict the rotational directions to avoid the convergence of the optimizer into a different local minimum and to avoid an overfit of c_1 and c_2 as mentioned Section 4 in the main article.

The development of a method to predict the function f_j from a given data set is an interesting open research question. Furthermore, an automatic classification of spoke directions into a partition I_1 and I_2 and a final classification of the deformation type into rigid rotation, bending or twisting are of future interest.

5 Computational complexity of the algorithm

The computational complexity of the proposed estimation procedure is now reported in two forms: the asymptotic time complexity and finite sample time measurements.

The asymptotic time complexity of the proposed algorithm depends on the number of samples n and the number of direction vectors K. In particular the optimization problem,

$$(\hat{c}, \hat{\mathbf{r}}) = \underset{c, r_1, \ldots, r_K}{\operatorname{argmin}} \sum_{i=1}^{n} \sum_{j=1}^{K} d_g^2\{\delta(c, r_j), X_{ij}\},$$

is the dominant part of the algorithm. Our algorithm to solve this nonlinear least squares problem is doubly iterative. The outer loop consists of applications of the exponential and inverse exponential maps whose time complexity is $O(nK)$. The inner loop iteratively updates $c^\dagger \in \mathbb{R}^3$ and $r_j^\dagger \in (0, \pi/2)$ by the Levenberg–Marquardt algorithm, where each iteration requires the asymptotic time complexity of $O(nK)$ mainly due to the computation of Jacobian matrix. Notice, that the computation time for inverting a 3×3 matrix is dominated by other terms and is ignored. Overall, if a finite number of iterations is assumed, then the asymptotic time complexity of the proposed algorithm is $O(nK)$.

We now turn to our attention to the complexity of the algorithm in real time, which we believe is more useful for practitioners. Computation times were measured by a personal computer with a Intel(R) Xeon(TM) 3.73GHz processor.

We have tested the estimations of three different rotational deformations

- *Model (2) - Rigid rotation*,
- *Model (10) - Twisting* and
- *Model (11) - Hierarchical deformations*

as described in Section 4 (the numbers correspond to the equation number in the main article), with two different types of objects to be deformed.

We first examined the computation times for a set of well-controlled objects. Using the Body 1 (as plotted in Fig. 2 of the main article) consisting of the original $K = 4$ directions, we duplicated the same direction vectors to increase the number of directions ($K = 8, 16, 32, 64, 128$) so that the effect on computation time of the different locations of direction vectors is minimized. For each sample size $n = 30, 60, 120$, we have repeated the

Table 5.1: Median time measurements in seconds over 100 replications.

n	Model	K					
		4	8	16	32	64	128
30	(2)	0.04	0.07	0.13	0.26	0.61	1.81
	(10)	0.11	0.16	0.26	0.40	0.79	2.00
	(11)	0.54	0.84	1.61	3.17	6.91	19.23
60	(2)	0.04	0.07	0.13	0.28	0.68	2.03
	(10)	0.10	0.18	0.29	0.42	0.85	2.18
	(11)	0.75	1.39	2.72	6.55	13.84	30.80
120	(2)	0.04	0.07	0.13	0.29	0.76	2.34
	(10)	0.12	0.18	0.27	0.44	0.92	2.50
	(11)	1.29	2.50	4.90	10.03	20.37	53.91

Table 5.2: Minimal time measurements in seconds over 100 replications..

n	Model	K					
		4	8	16	32	64	128
30	(2)	0.02	0.05	0.08	0.19	0.48	1.59
	(10)	0.03	0.07	0.12	0.25	0.61	1.75
	(11)	0.23	0.40	0.74	1.50	3.21	7.71
60	(2)	0.02	0.04	0.08	0.18	0.54	1.78
	(10)	0.03	0.05	0.14	0.27	0.68	1.89
	(11)	0.37	0.68	1.32	2.61	5.52	12.37
120	(2)	0.02	0.04	0.09	0.22	0.63	2.12
	(10)	0.03	0.09	0.15	0.33	0.79	2.38
	(11)	0.64	1.21	2.39	4.85	9.99	21.61

estimation $R = 100$ times to measure the computation time required to obtain the estimates $\hat{c}$. Note that we have used von Mises-Fisher distribution with $\kappa = 1000$ for the i.i.d. errors.

Tables 5.1 and 5.2 report the median computation time and the minimal computation time among the R repetitions, respectively. With large numbers of sample and directions, the computation requires less than a minute on average, while it takes less than a second for moderate sizes of sample and directions.

From an inspection of Table 5.1, there is a trend for the computation time increasing approximately linear with K and also with n for all models.

By comparing the minimal time (Table 5.2) and the median time (Table 5.1), we have noticed that the computation time varies by a large amount. See for example model (11) with $n = 60$, $K = 128$; The median time is over 30 seconds while the shortest time is only 12 seconds. This is due to the slow convergence of the iterative algorithm for a particular choice of observations.

Finally, Table 5.3 reports the quality of the repeated rotation axis estimations by the

Table 5.3: Quality of the repeated rotation axis estimations for the time measurements by $d_g(\hat{c}, c)$ in degree.

n	Model	rotation axis	K					
			4	8	16	32	64	128
30	(2)	1	1.18	0.92	0.63	0.40	0.33	0.22
	(10)	1	0.88	0.61	0.40	0.29	0.20	0.14
	(11)	1	2.18	2.10	1.99	1.89	1.86	1.81
	(11)	2	1.43	1.09	0.77	0.60	0.48	0.43
60	(2)	1	0.74	0.63	0.42	0.30	0.21	0.16
	(10)	1	0.59	0.42	0.29	0.18	0.14	0.10
	(11)	1	1.42	1.28	1.24	1.20	1.16	1.17
	(11)	2	0.88	0.64	0.49	0.35	0.28	0.21
120	(2)	1	0.60	0.39	0.30	0.20	0.15	0.10
	(10)	1	0.37	0.30	0.20	0.14	0.10	0.07
	(11)	1	1.09	0.97	0.99	0.90	0.89	0.89
	(11)	2	0.61	0.44	0.29	0.22	0.18	0.12

mean distance of $d_g(\hat{c}, c)$, which shows higher accuracy for larger K and n as discussed in Section 4 of the article.

We also have examined the computation times with another set of objects whose base direction vectors are determined randomly. As shown in the following, the additional randomness leads to more variation in the computation times. Nevertheless, the computation time exhibits again the approximate linear increase for K and n.

The second set of objects to be deformed has $K = 8, 16, 32, 64, 128$ directions vectors, each of which is obtained from uniform distribution on a hemisphere. With $n = 30, 60, 120$ samples, we also report the time measurements from $R = 100$ replications.

Table 5.4: Median time measurements in seconds over 100 replications.

n	Model	K				
		8	16	32	64	128
30	(2)	0.08	0.14	0.26	0.63	1.83
	(10)	0.18	0.29	0.49	0.94	2.10
	(11)	1.99	6.39	17.18	51.44	130.85
60	(2)	0.08	0.14	0.28	0.70	2.04
	(10)	0.22	0.32	0.51	0.96	2.27
	(11)	2.74	6.75	24.25	63.20	265.95
120	(2)	0.08	0.14	0.30	0.77	2.40
	(10)	0.22	0.32	0.48	0.98	2.54
	(11)	9.79	18.86	42.81	129.75	305.54

Table 5.4 and 5.5 report the median and the minimum computation time in seconds, respectively. As expected, the time increases with larger K and larger n. Due to the uncon-

Table 5.5: Minimal time measurements in seconds over 100 replications..

n	Model	K				
		8	16	32	64	128
30	(2)	0.05	0.08	0.20	0.58	1.78
	(10)	0.07	0.12	0.25	0.59	1.84
	(11)	0.42	0.84	2.30	3.45	8.03
60	(2)	0.05	0.10	0.23	0.59	1.80
	(10)	0.07	0.16	0.31	0.72	1.99
	(11)	0.72	1.36	2.75	5.99	18.78
120	(2)	0.04	0.10	0.24	0.63	2.23
	(10)	0.06	0.17	0.32	0.71	2.31
	(11)	1.25	2.43	4.91	15.24	33.16

Table 5.6: Quality of the repeated rotation axis estimations for the time measurements by $d_g(\hat{c}, c)$ in degree.

n	Model	rotation axis	K				
			8	16	32	64	128
30	(2)	1	0.92	0.64	0.45	0.33	0.21
	(10)	1	0.67	0.44	0.30	0.22	0.15
	(11)	1	1.20	0.59	0.40	0.23	0.16
	(11)	2	2.57	0.73	0.55	0.37	0.25
60	(2)	1	0.64	0.53	0.31	0.23	0.15
	(10)	1	0.43	0.26	0.22	0.14	0.10
	(11)	1	1.05	0.37	0.25	0.18	0.11
	(11)	2	2.67	0.60	0.35	0.23	0.17
60	(2)	1	0.42	0.33	0.23	0.14	0.11
	(10)	1	0.31	0.21	0.15	0.10	0.07
	(11)	1	0.86	0.27	0.17	0.11	0.07
	(11)	2	2.39	0.37	0.25	0.17	0.12

trolled model complexity (originated from the random base directions) the time difference between the median and the minimum time is larger than the previous controlled case. We conjecture that the computation times are heavily dependent on the convergence of the Levenberg–Marquardt algorithm.

In addition to the computation time, Table 5.6 reports the quality of the repeated rotation axis estimations by the mean distance of $d_g(\hat{c}, c)$, which again shows higher accuracy for larger K and n as discussed in Section 4 of the article.

6 Estimation procedure for knee motion during gait

In section 6 in the main article we have studied a real data example: the deformation of the upper and lower leg by two potentially non-orthogonal rotations at the knee joint during gait. These two rotations are flexion-extension about a right-to-left (medial-lateral) axis and the internal-external rotation of the lower-leg relative to the upper leg about an axis directed along the long axis of the lower leg. The data set consists of T time dependent observations $\mathbf{M}_1, \ldots, \mathbf{M}_T$ whereas each $\mathbf{M}_i$ is a collection of markers $\mathbf{M}_i = (M_{i1}, \ldots, M_{i\widetilde{K}})$ on the upper and lower leg with $M_{ij} \in \mathbb{R}^3, j = 1, \ldots, \widetilde{K}$. Let $\widetilde{I}_1$ and $\widetilde{I}_2$ be a partition of indices $\{1, \ldots, \widetilde{K}\}$ representing groups of $\widetilde{K}_1$ markers $\widetilde{I}_1$ on the upper leg and $\widetilde{K}_2$ markers $\widetilde{I}_2$ on the lower leg. Two markers $\mathbf{M}_{\iota_1}, \iota_1 \in \widetilde{I}_1$ and $\mathbf{M}_{\iota_2}, \iota_2 \in \widetilde{I}_2$ are chosen as basis points at the upper and lower leg. Set $I_1 = \widetilde{I}_1 \setminus \{\iota_1\}$, $I_2 = \widetilde{I}_2 \setminus \{\iota_2\}$, $K = \widetilde{K} - 2$, $K_1 = \widetilde{K}_1 - 1$ and $K_2 = \widetilde{K}_2 - 1$ then directions $\mathbf{X}_i = (X_{i1}, \ldots, X_{iK})$ are derived by

$$X_{ij} = \frac{M_{ij} - M_{i\iota_1}}{\|M_{ij} - M_{i\iota_1}\|}, j \in I_1, \quad X_{ij} = \frac{M_{ij} - M_{i\iota_2}}{\|M_{ij} - M_{i\iota_2}\|}, j \in I_2$$

for $i = 1, \ldots, T$, which are invariant to translation and size changes.

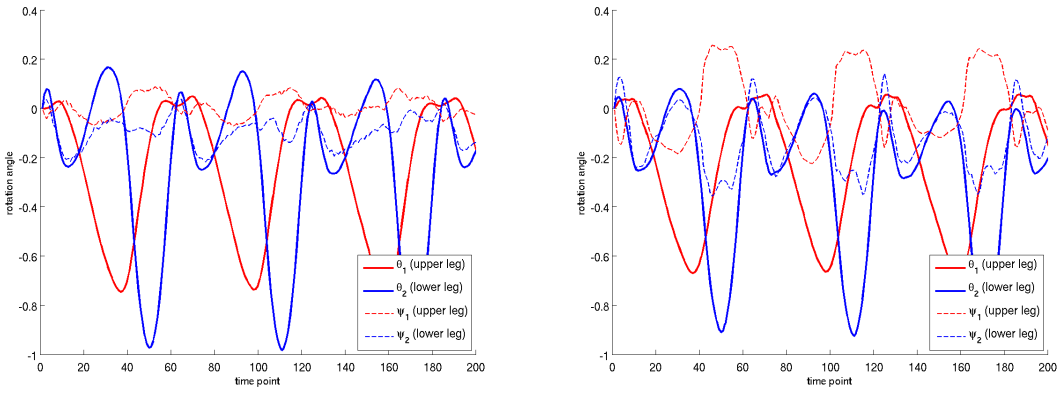
For the sake of convenience, we assume the observations $\mathbf{X}_1, \ldots, \mathbf{X}_T$ are independent and modify the single rotation model (10) in the main article to

$$X_{ij} = R(c, \theta_{ij}) X_{1j} \oplus \epsilon_{ij} \quad (j = 1, \ldots, K). \tag{6.1}$$

A more careful modeling of the time dependencies is left for future work, e.g., by an autoregressive model as suggested by (Rivest, 2001, Section 4.1).

The first (flexion-extension) rotation axis c_1 is estimated by the estimation procedure (7) in the main article and describes a bending deformation of the upper and lower leg around the knee. The lower leg rotates relative to the upper leg whereas the upper leg rotates relative to the pelvis. At first, we estimate the rotation angles θ_i^u of the upper leg in order to estimate the rotation angles θ_i^l of the lower leg relative to the upper leg. The least squares estimators $(\hat{c}_1, \hat{\mathbf{r}}_1)$ can be used to estimate the parameters of (6.1) with

$$\theta_{ij} = \operatorname{atan2}\{\langle P_{(\hat{c}_1, \hat{r}_{1j})} X_{ij}, \hat{c}_1 \times X_{1j} \rangle, \langle P_{(\hat{c}_1, \hat{r}_{1j})} X_{ij}, X_{1j} - \hat{c}_1 \cos(\hat{r}_j) \rangle\}. \tag{6.2}$$



(a) bone (b) skin

Figure 6.1: Predicted rotation angles $(\theta_1, \psi_1) = (\hat{\theta}^u, \hat{\psi}^u)$ of directions on the upper leg and angles $(\theta_2, \psi_2) = (\hat{\theta}^l, \hat{\psi}^l)$ of directions on the lower leg for the first 200 time points. (a) Estimated rotation angles of directions derived from bone markers. (b) Estimated rotation angles of directions derived from skin markers. (angle units in radian)

The predictor of $\theta_i^u, i = 1, \ldots, T$ is

$$\hat{\theta}_i^u = \frac{1}{K_1} \sum_{j=1}^{K_1} \theta_{ij}, \quad j \in I_1.$$

Next, the directions are de-rotated by $\widetilde{X}_{ij} = R(\hat{c}_1, -\hat{\theta}_i^u) X_{ij}$ for $j = 1 \ldots, K$ and $i = 1, \ldots, T$. The directions $\widetilde{X}_{ij}, j \in I_1$ are directions on the upper leg and do not rotate about $\hat{c}_1$ after the inverse deformation. The remaining rotation of the lower leg relative to the upper leg is then

$$\hat{\theta}_i^l = \frac{1}{K_2} \sum_{j=1}^{K_2} \tilde{\theta}_{ij}, \quad j \in I_2 \text{ with}$$

$$\tilde{\theta}_{ij} = \text{atan2}\{\langle P_{(\hat{c}_1, \hat{r}_{1j})} \widetilde{X}_{ij}, \hat{c}_1 \times \widetilde{X}_{1j} \rangle, \langle P_{(\hat{c}_1, \hat{r}_{1j})} \widetilde{X}_{ij}, \widetilde{X}_{1j} - \hat{c}_1 \cos(\hat{r}_j) \rangle\}.$$

Finally, we obtain a set of de-rotated directions $\mathbf{Z}$ by $Z_{ij} = \widetilde{X}_{ij}, \ j \in I_1$ and $Z_{ij} = R(\hat{c}_1, -\hat{\theta}_i^l) \widetilde{X}_{ij}, \ j \in I_2$ for $i = 1, \ldots, T$.

The estimation of the second (internal-external) rotation axis $(\hat{c}_2, \hat{\mathbf{r}}_2)$ and $\hat{\psi}^u, \hat{\psi}^l$ follows the same steps of the above paragraph using using $\mathbf{Z}_i$ instead of $\mathbf{X}_i$ for $i = 1, \ldots, T$.

In addition to the estimates $\hat{c}_1$ and $\hat{c}_2$, we estimate the standard errors $\hat{\text{se}}_B(\hat{c}_1), \hat{\text{se}}_B(\hat{c}_2)$. Moreover, we estimate the rotation axes $\hat{c}_{1j}$ and $\hat{c}_{2j}$ for each marker $j = 1 \ldots, K$. Therewith, we can quantify the estimations by the dispersion $\sigma_{d_g(\hat{c}_1, \hat{c}_{1j})}$ of the geodesic distance $d_g(\hat{c}_1, \hat{c}_{1j})$ and $\sigma_{d_g(\hat{c}_2, \hat{c}_{2j})}$ respectively.

Figure 6.1 reports the predicted rotation angles $(\hat{\theta}_i^u, \hat{\psi}_i^u)$ of the upper leg relative to the laboratory reference frame and $(\hat{\theta}_i^l, \hat{\psi}_i^l)$ of the lower leg relative to the upper leg for the first 200 time points. The larger observed rotation angles around the second rotation axis for the skin data is due to the well-known deformation of the skin surface.

References

Gray, A. (1998), *Modern Differential Geometry of Curves and Surfaces with Mathematica*, CRC Press, 2nd ed.

Koenderink, J. J. (1990), *Solid Shape*, The MIT Press.

Kühnel, W. (2006), *Differential Geometry*, vol. 16, Student Mathematical Library, 2nd ed.

Mardia, K. V. and Jupp, P. E. (2000), *Directional Statistics*, Chichester: Wiley.

Rivest, L.-P. (1999), "Some Linear Model Techniques for Analyzing Small-Circle Spherical Data," *Canadian Journal of Statistics*, 27, 623–638.

— (2001), "A Directional Model for the Statistical Analysis of Movement in Three Dimensions," *Biometrika*, 88, 779–791.