Research Article

A Bayesian Generative Model for Surface Template Estimation

Jun Ma,[1] Michael I. Miller,[2] and Laurent Younes[3]

[1] *Center for Imaging Science, Department of Biomedical Engineering, The Johns Hopkins University,*
  320 Clark Hall, Baltimore, MD 21218, USA
[2] *Center for Imaging Science, Department of Biomedical Engineering, The Johns Hopkins University,*
  301 Clark Hall, Baltimore, MD 21218, USA
[3] *Center for Imaging Science, Department of Applied Math and Statistics, The Johns Hopkins University,*
  3245 Clark Hall, Baltimore, MD 21218, USA

Correspondence should be addressed to Jun Ma, junma@cis.jhu.edu

3D surfaces are important geometric models for many objects of interest in image analysis and Computational Anatomy. In this paper, we describe a Bayesian inference scheme for estimating a template surface from a set of observed surface data. In order to achieve this, we use the geodesic shooting approach to construct a statistical model for the generation and the observations of random surfaces. We develop a mode approximation EM algorithm to infer the maximum a posteriori estimation of initial momentum μ, which determines the template surface. Experimental results of caudate, thalamus, and hippocampus data are presented.

1. Introduction

3D surfaces are important geometric models for many objects of interest in image analysis and Computational Anatomy. For example, they are often used to represent the shape of 3D objects, the surface of human faces, and the boundaries of brain structures or of other human organs. Most data analysis methods in this domain are template-centered, and a proper estimation of a template plays an important role to obtain high quality results. This paper is devoted to the description of statistically supported template estimation method which is adapted to surface data sets.

Our approach will be to build a generative statistical shape model in which the template is a parameter. We will then estimate it using maximum likelihood. This model relies on the very natural setting for which an observed surface is a noisy version of a random deformation of the template. This is the most generic and most basic approach of the deformable template paradigm, even if we add a small refinement by including a prior distribution on the template, based on what we will call a *hypertemplate*. Even with this global scheme which is fairly simple, we will see

that implementing it in the context of surfaces will constitute a significant theoretical and numerical challenge.

At the exception of the recent work of [1], this approach significantly differs from what has been mostly proposed in the literature, in which most of the methods compute templates as averages over specific common parametrizations of the surfaces (using, for example, the sphere as a parameter space [2]). Parametric representations, however, are limited by the fact that, because they are defined a priori and independently for each object, they cannot be assumed to suitably align important features in a given data set of surfaces (i.e., give similar coordinates to similar features in the surfaces). This usually results in oversmoothed template surfaces (which is the equivalent of getting blurry template images in the case of image averaging). In [1], a similar diffeomorphic transformation model is used, but, as we will see, our Bayesian construction will provide a well-specified template whereas [1] needs to rely to topologically unconstrained approximations to end up with a manageable template.

In addition to the references above, there have been several publications addressing the issue of shape averaging

over a dataset, although most of them work with 3D volume data or landmark points set. In several cases, the average is based on metric properties of the space of shapes [3–7], and the template is computed as an intrinsic average, minimizing the sum of square distances to each element of the set (Fréchet mean). Such methods have been implemented in the context of diffeomorphic deformation models (which are also models of interest in the present paper) for landmark matching [8], for 3D average digital atlas construction [9], and to quantify variability in heart geometry [10]. Other definitions of the average, adapted to situations in which the data is corrupted by noise, have been proposed [11–13], based on variational approaches (but not relying on a generative statistical model).

Our approach to build a generative statistical shape model is reminiscent of the construction developed in [14] for linear models of deformations, and in [15] for large diffeomorphic deformations in 3D volume image averaging. Adapting these ideas to surfaces will however require new algorithms and numerical developments.

In order to present our model, we need to first provide some background materials and notation, describing in particular the geodesic shooting equations that we will use to generate deformed surfaces. We will then introduce a random statistical model describing the generation and the observations of random surfaces. We then develop a Mode Approximation EM algorithm for surface averaging, to estimate the template from observations. In the optimization part, we derive and implement a new variational scheme, which is also applicable to surface matching, providing an alternative approach to the one originally proposed in [16, 17]. Finally, we present and discuss experimental results on caudate, thalamus, and hippocampus data.

2. EPDiff for Surface Evolution

We will base our random shape model on the so-called EPDiff equation, which describes the evolution of deformable structures (like images, surfaces, or landmarks) under the action of groups of diffeomorphisms. It is a geodesic equation for a Riemannian metric on diffeomorphisms, and describes a momentum conservation law in the associated Hamiltonian system. The reader interested by the theory behind this equation can refer to [18–20], but most of this background will not be needed for the present paper, in which we will only use the specific form of the equations for surface evolution. The term EPDiff comes from its determination as an Euler-Poincaré equation in the group of diffeomorphisms, as introduced in [21]. One of its main interests here is that it provides a numerically stable, easily described, Hamiltonian evolution over diffeomorphisms, which will constitute an ideal support for our shape models.

The EPDiff equations describe the combined time evolution of a diffeomorphism, denoted $\phi(t, \cdot)$ and of what can be interpreted as a momentum, denoted $p(t, \cdot)$. The initial conditions are always $\phi(0, x) = x$ for ϕ, and some initial value, p_0, for p. This *initial momentum* will be a key component of the statistical model that will be built later on.

Let us start with the simplest form of the equation, which assumes that p_0 is a vector-valued function over $\mathbb{R}^d$, that is, $p_0 : \mathbb{R}^d \to \mathbb{R}^d$. It involves a smoothing kernel, K, defined on $\mathbb{R}^3 \times \mathbb{R}^3$, a typical choice being

$$K(x, y) = \text{cst} \exp\left(-\frac{\|x - y\|^2}{2\tau^2}\right). \tag{1}$$

Letting $\nabla_1 K$ denote the gradient of K with respect to its first variable, the corresponding EPDiff equation is

$$\frac{d\phi}{dt}(t, x) = \int_{\mathbb{R}^3} K(\phi(t, x), \phi(t, y)) p(t, y) dy,$$

$$\frac{dp}{dt}(t, x) = -\int_{\mathbb{R}^3} \nabla_1 K(\phi(t, x), \phi(t, y)) (p(t, x) \cdot p(t, y)) dy. \tag{2}$$

Here, the notation $a \cdot b$ refers to the usual dot product between vectors in $\mathbb{R}^3$. If K is smooth enough, this system has solutions for arbitrary large t, and the mapping $x \mapsto \phi(t, x)$ is a diffeomorphism at all times.

The interesting fact about these equations is that they can have singular variants that are described in a similar way and have the same existence properties. The simplest way to relate the variants to the previous equation is to replace the Lebesgue's measure in the integrals by another, possibly singular, measure. For example, taking a surface S_0 in $\mathbb{R}^3$, we can use the volume form on S_0 as a reference measure and obtain the equations (in which p_0 and $p(t, \cdot)$ need only to be defined on S_0)

$$\frac{d\phi}{dt}(t, x) = \int_{S_0} K(\phi(t, x), \phi(t, y)) p(t, y) d\omega_{S_0}(y),$$

$$\frac{dp}{dt}(t, x) = -\int_{S_0} \nabla_1 K(\phi(t, x), \phi(t, y))$$

$$\times (p(t, y) \cdot p(t, y)) d\omega_{S_0}(y), \tag{3}$$

where $d\omega_{S_0}$ is the volume form on S_0. Note that the first equation is defined for $x \in \mathbb{R}^3$, but it suffices to use it for $x \in S_0$ to obtain an equation for the evolving surface

$$S_t = \phi(t, S_0) = \{y = \phi(t, x), x \in S_0\}. \tag{4}$$

We can write a discrete form of the equations by replacing dy by a sum of Dirac measures (at points $x_1, \ldots, x_L$ in $\mathbb{R}^3$), which gives, letting $a_l(t) = p(t, x_l)$,

$$\frac{d\phi}{dt}(t, x) = \sum_{l=1}^{L} K(\phi(t, x), \phi(t, x_l)) a_l,$$

$$\frac{da_k}{dt} = -\sum_{l=1}^{L} (a_l \cdot a_k) \nabla_1 K(\phi(t, x_k), \phi(t, x_l)). \tag{5}$$

Similarly to (3), the first equation is valid for all $x \in \mathbb{R}^3$, but it suffices to solve it for $x = x_l$, $l = 1, \ldots, L$ to obtain the evolution of the point set

$$x_l(t) = \phi(t, x_l). \tag{6}$$

Also, (5) can be seen as a discretization of (3) in which $x_1, \ldots, x_L$ are the vertices of a triangulation of S_0, and $a_l(t) = p(t, x_l)\delta\sigma_{S_0}(x_l)$, where $\delta\sigma_{S_0}(x_l)$ is the area of a surface element around x_l.

The evolution of point sets is the most important from a practical point of view, since it is an ODE that can be easily implemented. Assuming a radial kernel $K(x, y) = \gamma(\|x - y\|^2)$ like in (1), and denoting $\gamma_{kl} = \gamma(\|x_k - x_l\|^2)$, and $\gamma'_{kl} = \gamma'(\|x_k - x_l\|^2)$, (5) can be rewritten as

$$\frac{dx_k}{dt} = \sum_{l=1}^{L} \gamma_{kl} a_l, \qquad \frac{da_k}{dt} = -2\sum_{l=1}^{L} \gamma'_{kl}(a_l \cdot a_k)(x_k - x_l). \tag{7}$$

Once the initial position of the vertices, $x(0) = (x_1(0), \ldots, x_L(0))$, and the initial momentum, $a(0) = (a_1(0), \ldots, a_L(0))$, are provided, the evolution of the point set is uniquely determined.

3. Generative Model for Surface Observation

3.1. Random Triangulated Surfaces. If a triangulated template surface T with vertices $x^{(T)}$ is given, and we solve, until time $t = 1$, (5) initialized with $x(0) = x^{(T)}$ and a random initial momentum $a(0) = \alpha$, the displaced vertices provide a random perturbation of the initial surface that will be denoted by T_α. This is stated in the following definition.

Definition 1. Let T be a triangulated surface with vertices $x^{(T)} = (x_1^{(T)}, \ldots, x_L^{(T)})$. Let $\alpha \in (\mathbb{R}^3)^L$ be a collection of L vectors in $\mathbb{R}^3$. Let $(x(t), a(t))$ be the solution of (5) with initial condition $x(0) = x^{(T)}$ and $a(0) = \alpha$. One defines T_α to be the triangulated surface with vertices $x^{(T_\alpha)} = x(1)$ and the same topology as T.

By letting α be random, we build T_α as a random deformation of T. This will form the "ideal", unobserved, surface, of which only a noisy version is observable (the noise process will be described in the next section).

Following [15], we will use a Bayesian formulation in which T is itself represented as a random deformation $T_{0,\mu} := (T_0)_\mu$, where T_0 is a fixed surface that we will call the hypertemplate, and μ is a prior initial momentum shooting from T_0 to T (same notation as in Definition 1). One of the main interests of using a hypertemplate is to fix the topology of T so that it belongs, by construction, to the same class of objects as T_0.

So, if N surfaces are observed, we need to model the probability distribution of the prior momentum, μ (starting at T_0), which specifies $T = T_{0,\mu}$ and of N deformation momenta $\alpha^{(1)}, \ldots, \alpha^{(N)}$ which specify the surfaces $T_{\alpha^{(1)}}, \ldots, T_{\alpha^{(N)}}$. We now provide a statistical model for the joint probability distribution of $\mu, \alpha^{(1)}, \ldots, \alpha^{(N)}$.

We first introduce some notation. Letting K be the kernel introduced in the previous section to define the geodesic shooting equations, we let Γ_T be the $3L$ by $3L$ matrix formed with the 3 by 3 blocks $K(x_k^{(T)}, x_l^{(T)})Id_{\mathbb{R}^3}$. We define, for a triangulated surface T with L vertices $x^{(T)}$, and $\alpha \in \mathbb{R}^{3L}$,

$$\|\alpha\|_T^2 = \alpha^* \Gamma_T \alpha = \sum_{k,l=1}^{L} K\left(x_k^{(T)}, x_l^{(T)}\right)(\alpha_k \cdot \alpha_l). \tag{8}$$

We define the joint distribution of $\mu, \alpha^{(1)}, \ldots, \alpha^{(N)}$ on $\mathbb{R}^{3L} \times (\mathbb{R}^{3L})^N$ by

$$p\left(\mu, \alpha^{(1)}, \ldots, \alpha^{(N)}\right) = \frac{1}{Z} \exp\left(-\frac{1}{2}\lambda\|\mu\|_{T_0}^2 - \frac{1}{2}\sum_{n=1}^{N}\left\|\alpha^{(n)}\right\|_{T_{0,\mu}}^2\right), \tag{9}$$

where λ is a fixed parameter regulating the weight on the hypertemplate.

There is a technical difficulty here, which is that one must make sure that this probability can be normalized (Z exists), which requires that the exponential is integrable. That this is true is not straightforward, and we have not been able to find a proof that works with any choice of the kernel K. One way to deal with this is to introduce a constant A_μ (which can be chosen arbitrarily large so that it does not interfere with the algorithms that will follow), and add to the model the constraint that $\|\mu\|_{T_0}$ is smaller than A_μ. Under such an assumption, one obtains (after integrating out the α's)

$$Z = (2\pi)^{3NL/2} \int_{\|\mu\|_{T_0} \leq A_\mu} \exp\left(-\frac{1}{2}\|\mu\|_{T_0}^2 - \frac{N}{2}\log\det\Gamma_{T_{0,\mu}}\right)d\mu. \tag{10}$$

This is finite, since, for any given μ, the transformation $x^{(T_0)} \to x^{(T_{0,\mu})}$ is the restriction of a diffeomorphism to the vertices of T_0 (as seen from (5)). This implies that $\Gamma_{T_{0,\mu}}$ is nonsingular, and its determinant is bounded away from 0 when μ is restricted to a compact space.

In fact, the choice $A_\mu = \infty$ can be proved to be acceptable for a large class of kernels. Those are kernels for which the smallest eigenvalue of Γ_T decreases at a speed which is at most polynomial in the minimal distance, h_T, between the vertices. A list of kernels satisfying this property can be found in [22]. For such kernels, we find that (L being fixed) $\log\det\Gamma_T = O(\log h_T)$. Just sketching the argument here, one can prove, using elementary properties of dynamical systems, that $h_T = O(\exp(-C\|\mu\|_{T_{0,\mu}}))$ for some constant C, so that the log determinant in (9) is linear in $\|\mu\|_{T_{0,\mu}}$ and Z is well defined, even with $A_{T_{0,\mu}} = \infty$. For very smooth kernels, including the Gaussian, bounds on the smallest eigenvalue of Γ_T are much worse (with a decay which is exponential in $(-1/h_T^2)$), and the previous argument does not work. Since the bounds in [22] hold uniformly with respect to the number of points, a polynomial bound may still be valid for a fixed L, although we were unable to discover it.

Notice that, conditionally to the template, the momenta α are independent and follow a Gaussian distribution with inverse covariance matrix given by Γ_T. An example of simulated random deformations obtained using such a model is provided in Figure 1.
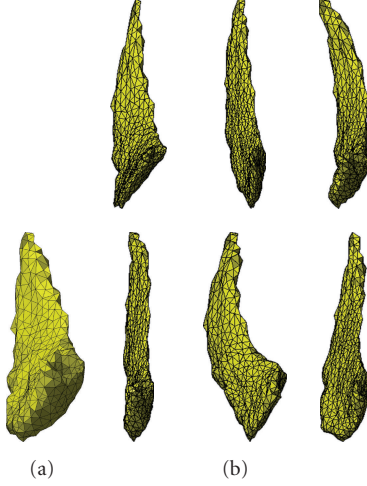
(a)   (b)

FIGURE 1: Random deformation of a template caudate surface (left). The six surfaces following the template are independent realization of the model described in (9).

3.2. Observation and Noise.

The second part of our generative model is to describe the observation process, which takes an ideal surface T_α generated according to the model above, and returns the noisy observable.

Modeling such a noise process is a tricky issue. Obvious choices (like adding noise to the vertices of T_α) do not work because one cannot assume that the observed surfaces are discretized consistently with the template. In this paper, we will work around this issue by assimilating the observation of a surface that of a singular Gaussian process.

For this, we consider that surfaces in $\mathbb{R}^3$ are not observable directly, but only via their action on test functions, that we will call *sensors*. We define a sensor to be a smooth vector field w over $\mathbb{R}^3$ (typically with small support). Given an oriented surface S, define

$$(S, w) = \int_S w(s) \cdot N_S(s) d\omega_S(s), \qquad (11)$$

where N_S is the normal to S. The real number, (S, w) is the measurement of S through the sensor w.

Now, modeling noisy surfaces will result in assuming that, given any w, the measurement (S, w) is a random variable. We will assume that it is Gaussian, and more generally, that, given m sensors, $w_1, \ldots, w_m$, the random vector $((S, w_j), j = 1, \ldots, m)$ is Gaussian.

S, via its action on sensors, is therefore modeled as a Gaussian random field. Given an ideal surface T_α, we will assume that its mean is given by

$$E((S, w)) = (T_\alpha, w) \qquad (12)$$

and the process is thereafter uniquely characterized by its

covariance operator

$$G(w, \widetilde{w}) = \mathrm{cov}((S, w), (S, \widetilde{w})). \qquad (13)$$

We will assume that this covariance is associated to a symmetric operator L_{obs} so that

$$G(w, \widetilde{w}) = \sigma^2 \int_{\mathbb{R}^d} (L_{\mathrm{obs}} w(x) \cdot \widetilde{w}(\widetilde{x})) dx \, d\widetilde{x}. \qquad (14)$$

(The apparently redundant parameter σ^2, which could have been included in L_{obs}, appears here because it can be easily estimated by the algorithm, with the operator L_{obs} remaining constant.)

To finalize our model, it remains to describe how S is discretized, that is, to make explicit a finite family of sensors through which S is measured. Let $(z_j, j \in J)$ form a regular grid of points in Ω. Let γ_s be a radial basis function (a Gaussian, e.g.,) and define, for $j \in J$ and $d = 1, 2, 3$

$$w_{j,d} = \gamma_s(x - z_j) e_d \qquad (15)$$

where e_d is the dth vector of the standard basis of $\mathbb{R}^3$ (this therefore specifies $3|J|$ sensors). The resulting observed variables are

$$y_{j,d} = (S, w_{j,d}) = \int_S \gamma_s(s - z_j) N_S^{(d)}(s) d\omega_S(s), \qquad (16)$$

where $N_S^{(d)} = N_S \cdot e_d$ is the dth coordinate of N_S. These variables are, by assumption, jointly Gaussian, with means $m_{j,d} = (T_\alpha, w_{j,d})$ and covariance matrix

$$\begin{aligned} g_{i,d,i',d'} &= G(w_{i,d}, w_{i',d'}) \\ &= \sigma^2 \delta_{d,d'} \int_{\mathbb{R}^d} L_{\mathrm{obs}} \gamma_s(x - z_i) \gamma_s(\widetilde{x} - z_{i'}) dx \, d\widetilde{x}. \end{aligned} \qquad (17)$$

Assuming that L_{obs} is translation invariant, the resulting expression is a function of $z_i - z_{i'}$ that we will denote

$$g_{i,d,i',d'} = \sigma^2 \delta_{dd'} \gamma_{\mathrm{obs}}(z_i - z_{i'}). \qquad (18)$$

Let $R_{\mathrm{obs}} = (r_{ij}^{(\mathrm{obs})})$ be the inverse matrix of the one with coefficients $(\gamma_{\mathrm{obs}}(z_j - z_j'), j, j' \in J)$. The log likelihood of the process will include error terms taking the form

$$\mathcal{E}_{\mathrm{obs}} = \frac{1}{\sigma^2} \sum_{j,j' \in J} \sum_{d=1}^{3} r_{j,j'}^{(\mathrm{obs})} (y_{j,d} - (T_\alpha, w_{j,d}))(y_{j',d} - (T_\alpha, w_{j',d})). \qquad (19)$$

Replacing $y_{j,d}$ by its expression in (16), we have

$$
\begin{aligned}
\sigma^2 \mathcal{E}_{\text{obs}} = \sum_{j,j' \in J} \sum_{d=1}^{3} r_{j,j'}^{(\text{obs})} \int_S \gamma_s\big(x - z_j\big) N_S^{(d)}(x) d\omega_S(x) \\
\times \int_S \gamma_s\big(x - z_{j'}\big) N_S^{(d)}(x) d\omega_S(x) \\
- 2 \sum_{j,j' \in J} r_{j,j'}^{(\text{obs})} \sum_{d=1}^{3} \int_S \gamma_s\big(x - z_j\big) N_S^{(d)}(x) d\omega_S(x) \\
\times \int_{T_\alpha} \gamma_s\big(x - z_{j'}\big) N_{T_\alpha}^{(d)}(x) d\omega_{T_\alpha}(x) \\
\times \sum_{j,j' \in J} r_{jj'}^{(\text{obs})} \sum_{d=1}^{3} \int_{T_\alpha} \gamma_s\big(x - z_j\big) N_{T_\alpha}^{(d)}(x) d\omega_{T_\alpha}(x) \\
\times \int_{T_\alpha} \gamma_s\big(x - z_{j'}\big) N_{T_\alpha}^{(d)}(x) d\omega_{T_\alpha}(x).
\end{aligned}
\tag{20}
$$

Let us analyze the first term. We have

$$
\begin{aligned}
\sum_{j,j' \in J} r_{j,j'}^{(\text{obs})} \sum_{d=1}^{3} \int_S \gamma_s\big(x - z_j\big) N_S^{(d)}(x) d\omega_S(x) \\
\times \int_S \gamma_s\big(x - z_{j'}\big) N_S^{(d)}(x) d\omega_S(x) \\
= \sum_{j,j' \in J} r_{j,j'}^{(\text{obs})} \sum_{d=1}^{3} \int_{S \times S} \gamma_s\big(x - z_j\big) N_S^{(d)}(x) \gamma_s \\
\times \big(y - z_{j'}\big) N_S^{(d)}(y) d\omega_S(x) d\omega_S(y) \\
= \int_{S \times S} \sum_{j,j' \in J} r_{j,j'}^{(\text{obs})} \gamma_s\big(x - z_j\big) \gamma_s\big(y - z_{j'}\big) \\
\times (N_S(x) \cdot N_S(y)) d\omega_S(x) d\omega_S(y) \\
= \int_{S \times S} K_{\text{obs}}(x, y) (N_S(x) \cdot N_S(y)) d\omega_S(x) d\omega_S(y),
\end{aligned}
\tag{21}
$$

with the notation

$$
K_{\text{obs}}(x, y) = \sum_{j,j' \in J} r_{j,j'}^{(\text{obs})} \gamma_s\big(x - z_j\big) \gamma_s\big(y - z_{j'}\big).
\tag{22}
$$

Treating the other terms similarly, we can rewrite the error term in the form

$$
\begin{aligned}
\mathcal{E}_{\text{obs}} = \frac{1}{\sigma^2} \int_{S \times S} K_{\text{obs}}(x, y) (N_S(x) \cdot N_S(y)) d\omega_S(x) d\omega_S(y) \\
- \frac{2}{\sigma^2} \int_{S \times T_\alpha} K_{\text{obs}}(x, y) \\
\times (N_S(x) \cdot N_{T_\alpha}(y)) d\omega_S(x) d\omega_{T_\alpha}(y) \\
+ \frac{1}{\sigma^2} \int_{T_\alpha \times T_\alpha} K_{\text{obs}}(x, y) \\
\times (N_{T_\alpha}(x) \cdot N_{T_\alpha}(y)) d\omega_{T_\alpha}(x) d\omega_{T_\alpha}(y)
\end{aligned}
\tag{23}
$$

and we will abbreviate this (introducing a notation for the right-hand side) as $\mathcal{E}_{\text{obs}} = (1/\sigma^2) \|S - T_\alpha\|_{\text{obs}}^2$. Thus, we can write

$$
p(y \mid T_\alpha, \sigma) = \text{cst} \exp\left(-\frac{1}{2\sigma^2} \|S - T_\alpha\|_{\text{obs}}^2\right).
\tag{24}
$$

We have the following important proposition.

Proposition 1. *Assume that γ_s is taken equal to the Green function of L_{obs}. Then, when the grid J becomes finer and K_{obs} is given by (22), one has*

$$
K_{\text{obs}}(x, y) \longrightarrow \gamma_{\text{obs}}(x - y).
\tag{25}
$$

See the appendix for a proof of this proposition (which requires some background on the theory of Hilbert spaces with a reproducing kernel) and possible extensions. For practical purposes, we will use γ_{obs} instead of K_{obs} in $\| \cdot \|_{\text{obs}}$, therefore assuming that the sensors are chosen according to the proposition. It is interesting to notice that the resulting norm in this case is precisely the norm that has been introduced in [16] to compare surfaces, when they are considered as elements of a reproducing kernel Hilbert space of *currents*. We refer to [16] for details on the mathematical construction.

In the rest of the paper, we develop a parametric procedure to estimate the template T from the observation of i.i.d. surfaces $S^{(1)}, S^{(2)}, \ldots, S^{(N)}$ generated as described above. This includes in particular N hidden deformation momenta $\alpha^{(1)}, \alpha^{(2)}, \ldots, \alpha^{(N)}$, such that the complete distribution of observed and unobserved variables is

$$
\begin{aligned}
& p\big(\mu, \alpha^{(1)}, S^{(1)}, \ldots, \alpha^{(N)}, S^{(N)}\big) \\
& = \text{cst} \exp\left(-\frac{\lambda}{2} \|\mu\|_{T_0}^2 - \frac{1}{2} \sum_{n=1}^{N} \left(\left\|\alpha^{(n)}\right\|_T^2 + \frac{1}{\sigma^2} \left\|S^{(n)} - T_{\alpha^{(n)}}\right\|_{\text{obs}}^2\right)\right),
\end{aligned}
\tag{26}
$$

where we have written for short $T = T_{0,\mu}$.

We now discuss the estimation of the parameter μ, and of the associated template $T_{0,\mu}$, which is the main purpose of this paper. This will be implemented with a mode approximation of the EM algorithm, as described in the next section.

4. Algorithms for Surface Template Estimation

4.1. Mode Approximation EM. Let $\Theta = (\alpha^{(1)}, \alpha^{(2)}, \ldots, \alpha^{(N)})$ be the hidden part of the process, representing the collection of initial momenta, and let $\mathbf{S} = (S^{(1)}, S^{(2)}, \ldots, S^{(N)})$ be the collection of observed surfaces. The complete distribution for the process, including the prior is given by (26).

The EM algorithm is an iterative method that updates a current estimation of μ using the following two E- and M-steps.

E-step: determine the conditional expectation $\widetilde{\mu} \mapsto E_\mu\{\log \pi(\widetilde{\mu}, \Theta, \mathbf{S}) \mid \mathbf{S}\}$.

M-step: maximize this expression with respect to $\widetilde{\mu}$ and replace the current estimation of μ by the obtained maximizer.

The conditional expectation can be expanded as

$$E_\mu\{\log \pi(\widetilde{\mu}, \Theta, \mathbf{S}) \mid \mathbf{S}\}$$

$$= -\frac{\lambda}{2}||\widetilde{\mu}||^2_{T_0} - \frac{1}{2}\sum_{n=1}^{N} E_\mu\left(\left|\left|\alpha^{(n)}\right|\right|^2_{T} + \frac{1}{\sigma^2}\left|\left|S^{(n)} - T_{\alpha^{(n)}}\right|\right|^2_{\text{obs}} \mid S^{(n)}\right). \tag{27}$$

Considering the highly nonlinear relation between $\alpha^{(n)}$ and the deformed surface $T_{\alpha^{(n)}}$, an explicit computation in (27) is impossible. We will therefore rely on the classic mode approximation in the EM which replaces the conditional distribution by a Dirac measure taken at the conditional mode, yielding

$$E_\mu\{\log \pi(\widetilde{\mu}, \Theta, \mathbf{S}) \mid \mathbf{S}\}$$

$$\approx \frac{\lambda}{2}||\widetilde{\mu}||^2_{T_0} + \frac{1}{2}\sum_{n=1}^{N}\left|\left|\widehat{\alpha}^{(n)}\right|\right|^2_{T} + \frac{1}{\sigma^2}\left|\left|S^{(n)} - T_{\widehat{\alpha}^{(n)}}\right|\right|^2_{\text{obs}}\bigg\}, \tag{28}$$

where

$$\widehat{\alpha}^{(n)} = \arg\min_{\alpha}\left\{||\alpha||^2_{T} + \frac{1}{\sigma^2}\left|\left|S^{(n)} - T_{\alpha}\right|\right|^2_{\text{obs}}\right\}. \tag{29}$$

This results in a maximum a posteriori procedure that maximizes alternatively in μ and in the $\alpha^{(n)}$'s. Like in [15], we will refer to it as a mode approximation to the EM (MAEM) rather than a MAP algorithm, in order to strengthen the fact that it is an approximation of the maximum a posteriori procedure relying on the likelihood of the observed data only. As illustrated in [14], the MAEM can be biased (leading to inexact estimation of the template, even with a very large number of samples), especially when the noise is important, but it is obviously more feasible than the exact EM. Notice that the MAEM method is a special form of EM algorithm, and as such optimizes a lower bound of the log likelihood of the observed data.

We summarize the two steps of the ith iteration of the MAEM in our case. Suppose μ and the $\alpha^{(n)}$'s are the current variables to be updated. Then the next iteration is as follows.

MAE step: with μ (and therefore T) fixed, find, for $n = 1, \ldots, N$, $\widehat{\alpha}^{(n)}$ to minimize

$$||\alpha||^2_{T} + \frac{1}{\sigma^2}\left|\left|S^{(n)} - T_{\alpha}\right|\right|^2_{\text{obs}} \tag{30}$$

and replace $\alpha^{(n)}$ by $\widehat{\alpha}^{(n)}$.

M step: with $\alpha^{(n)}$ fixed, update μ with the minimizer of

$$\lambda||\widetilde{\mu}||^2_{T_0} + \sum_{n=1}^{N}\left(\left|\left|\alpha^{(n)}\right|\right|^2_{T_0,\widetilde{\mu}} + \frac{1}{\sigma^2}\left|\left|S^{(n)} - \left(T_0\widetilde{\mu}\right)_{\alpha^{(n)}}\right|\right|^2_{\text{obs}}\right). \tag{31}$$

We now discuss how each of these steps can be implemented.

4.2. MAE Step. Our goal in this section is to optimize (30). We work with fixed n and drop it from the notation to simplify the expressions. The objective function is

$$E(\alpha) = ||\alpha||^2_{T} + \frac{1}{\sigma^2}||S - T_{\alpha}||^2_{\text{obs}}. \tag{32}$$

This problem is equivalent to the surface matching algorithm considered in [16], with a slightly different formulation since [16] optimize an energy with respect to a time-dependent momentum instead of just the initial momentum (i.e., they solved simultaneously the geodesic estimation and the matching problems). These two formulations are equivalent when using continuous time (they produce the same minima), but they yield different results when discretized. In our setting, formulating the problem as in (32) is natural, and focuses on the modeled random variable, α.

We need to compute the variation of E with respect to α. This computation will be useful for the M-step also. We first discuss the discretization of the error term, which follows [16]. Let S be a triangulated surfaces, with vertices $x^{(S)} = (x_1^{(S)}, \ldots, x_L^{(S)})$ and faces $F^{(S)} = (f_1^{(S)}, \ldots, f_M^{(S)})$. Each face is represented by an ordered triple of vertices: $f = (x_1^f, x_2^f, x_3^f)$, and we define the face centers and area-weighted normals by

$$c_f = \frac{1}{3}\left(x_1^f + x_2^f + x_3^f\right),$$

$$N_f = \frac{1}{2}\left(x_2^f - x_1^f\right) \times \left(x_3^f - x_1^f\right). \tag{33}$$

Then a discrete approximation of $||S - S'||^2_{\text{obs}}$ is

$$U\left(x^{S'}\right) = \sum_{f,f' \in F^{(S)}} \gamma_{\text{obs}}\left(c_f - c_{f'}\right)\left(N_f \cdot N_{f'}\right)$$

$$+ \sum_{f,f' \in F^{(S')}} \gamma_{\text{obs}}\left(c_f - c_{f'}\right)\left(N_f \cdot N_{f'}\right) \tag{34}$$

$$- 2\sum_{f \in F^{(S)}, f' \in F^{(S')}} \gamma_{\text{obs}}\left(c_f - c_{f'}\right)\left(N_f \cdot N_{f'}\right),$$

where S is considered as fixed and U is therefore considered as a function of the vertices, $x^{(S')}$, of S'.

With this notation, we can write

$$E(\alpha) = ||\alpha||^2_{T} + \frac{1}{\sigma^2}U\left(x^{(T_\alpha)}\right). \tag{35}$$

We want to compute the gradient of E and for this, apply the chain rule to the transformations $\alpha \to T_\alpha \to U(x^{(T_\alpha)})$, yielding

$$\partial_\alpha U = \left(\frac{dx^{(T_\alpha)}}{d\alpha}\right)^* \partial_x U, \qquad (36)$$

where A^* is the transpose matrix of A.

The gradient of U with respect to $x^{(S')}$ has been computed in [16], and is given as follows. We denote by $F_l^{(S)}$ the set of faces (triangles) that contain a vertex x_l in a triangulated surface S. For $f \in F_l^{(S)}$, we let $e_l(f)$ denote the edge opposed to x_l in f, positively oriented in f. With $S' = T_\alpha$, we have

$$\frac{\partial U}{\partial x_l^{(S')}}\left(x^{(S')}\right)$$

$$= \sum_{f \in F_l^{(S')}} \sum_{g \in F^{(S)} \cup F^{(S')}} a_g \left(\frac{2}{3} \nabla \gamma_{\text{obs}}\left(c_f - c_g\right)\left(N_f \cdot N_g\right)\right.$$

$$\left. + \gamma_{\text{obs}}\left(c_f - c_g\right)e_l(f) \times N_g\right), \qquad (37)$$

where $a_g = 1$ if $g \in F^{(S')}$ and $a(g) = -1$ if $g \in F^{(S)}$.

Now let us derive the variation of $x^{(T_\alpha)}$ with respect to the initial momentum, α. We know that $x^{(T_\alpha)} = x(1)$, where x and a evolve according to the system (7)

$$\frac{dx_k}{dt} = \sum_{l=1}^{L} \gamma_{kl} a_l, \qquad \frac{da_k}{dt} = -2\sum_{l=1}^{L} \gamma'_{kl}(a_l \cdot a_k)(x_k - x_l) \qquad (38)$$

with $x(0) = x^{(T)}$ and $a(0) = \alpha$ (and $\gamma_{kl}, \gamma'_{kl}$ are short for $\gamma(\|x_k - x_l\|^2)$ and $\gamma'(\|x_k - x_l\|^2)$). Now an infinitesimal variation $\alpha \to \alpha + \delta\alpha$ in the initial condition induces infinitesimal variations $a + \delta a$ and $x + \delta x$ over time, and the pair $(\delta x, \delta a)$ obeys the following differential system, that can be obtained from a formal differentiation of (7):

$$\frac{d\delta x_k(t)}{dt} = \sum_{l=1}^{L} \gamma_{kl} \delta a_l + 2\sum_{l=1}^{L} \gamma'_{kl} a_l (x_k - x_l) \cdot (\delta x_k - \delta x_l), \qquad (39)$$

$$\frac{d\delta a_k(t)}{dt} = -2\sum_{l=1}^{L} \gamma'_{kl}(a_l \cdot \delta a_k + \delta a_l \cdot a_k)(x_k - x_l)$$

$$- 2\sum_{l=1}^{L} \gamma'_{kl}(a_l \cdot a_k)(\delta x_k - \delta x_l)$$

$$- 4\sum_{l=1}^{L} \gamma''_{kl}(a_l \cdot a_k)(x_k - x_l)((x_k - x_l) \cdot (\delta x_k - \delta x_l)) \qquad (40)$$

with $\gamma''_{kl} = \gamma''(\|x_k - x_l\|^2)$.

One can rewrite it in the matrix form:

$$\frac{d}{dt}\begin{pmatrix} \delta x \\ \delta \alpha \end{pmatrix} = J(t)\begin{pmatrix} \delta x \\ \delta \alpha \end{pmatrix}, \qquad (41)$$

where $J(t) = \begin{pmatrix} J_{xx} & J_{xa} \\ J_{ax} & J_{aa} \end{pmatrix}$ with

$$J_{xx}(k,l) = \left(2\sum_{q=1}^{L} \gamma'_{kq} a_q \left(x_l - x_q\right)^*\right)\delta_{kl} - 2\gamma'_{kl} a_l(x_k - x_l)^*$$

$$J_{xa}(k,l) = \gamma_{kl} Id_{\mathbb{R}^3}.$$

$$J_{ax}(k,l) = -\left(2\sum_{q=1}^{L} \left(a_q \cdot a_l\right)\right.$$

$$\times \left(\gamma'_{kq} Id_{\mathbb{R}^3} + 2\gamma''_{kq}\left(x_l - x_q\right)\left(x_l - x_q\right)^*\right)\delta_{kl}\right)$$

$$\left. + 2(a_l \cdot a_k)\left(\gamma'_{kl} Id_{\mathbb{R}^3} + 2\gamma''_{kl}(x_k - x_l)(x_k - x_l)^*\right)\right).$$

$$J_{aa}(k,l) = -\left(2\left(\sum_{q=1}^{L} \gamma'_{kq}\left(x_l - x_q\right)a_q^*\right)\right)\delta_{kl} - 2\gamma'_{kl}(x_k - x_l)a_k^*. \qquad (42)$$

Solving this system with initial condition $\delta x(0) = 0$ and $\delta a(0) = \delta\alpha$ provides what we have denoted

$$\left(\frac{dx^{(T_\alpha)}}{d\alpha}\right)\delta\alpha. \qquad (43)$$

One does not need to compute all the coefficients of the matrix $dx^{(T_\alpha)}/d\alpha$ using this equation in order to apply the transpose in (36). This is fortunate because this would constitute a computationally demanding effort given that this matrix is $3L$ by $3L$ with L large. The right hand side of (36) can be in fact computed directly using a single dynamical system, given by

$$\frac{d}{dt}\begin{pmatrix} \eta_x \\ \eta_\alpha \end{pmatrix} = -J(t)^*\begin{pmatrix} \eta_x \\ \eta_\alpha \end{pmatrix}, \qquad (44)$$

where $J(t)$ is defined in (39). If (44) is solved from time $t = 1$ to time $t = 0$ with $\eta_x(1) = \partial_x U$ and $\eta_\alpha(1) = 0$, then

$$\partial_\alpha U = \eta_\alpha(0). \qquad (45)$$

This is a simple consequence of the theory of linear differential systems (a proof is provided in the Appendix for completeness). Note that the matrix $J(t)$ depends on the solution of (7) computed with initial conditions $x(0) = x^{(T)}$ and $a(0) = \alpha$. To emphasize this dependency, we will denote it $J(t) = J^{(T,\alpha)}(t)$ in the following.

Given this, we see that a variation $\alpha \to \alpha + \delta\alpha$ induces a first-order variation δE of the energy given by

$$\delta E = 2\langle \delta\alpha, \alpha\rangle_T + \frac{1}{\sigma^2}\delta\alpha \cdot \eta_\alpha(0), \qquad (46)$$

where the T-dot product is $\langle \delta\alpha \, , \, \alpha \rangle_T = \delta\alpha \cdot (\Gamma_T \alpha)$ and Γ_T is the matrix formed with 3 by 3 blocs $\gamma_{kl} Id_{\mathbb{R}^3}$.

We choose to operate the gradient descent with respect to this dot product and therefore choose a variation proportional to $\delta\alpha = -(2\alpha + \Gamma_T^{-1}\eta_\alpha(0))$. So, the algorithm to compute an optimal α is the following.

Algorithm 1 (MAE Step for Surface Template Estimation).

(1) Compute the variation $\partial_{x^{(T_\alpha)}} U$ using (37).

(2) Solve backward in time (44) initialized with $\eta_x(1) = \partial_{x^{(T_\alpha)}} U$ and $\eta_\alpha = 0$.

(3) Replace α by $\alpha - \epsilon(2\alpha + (1/\sigma^2)\Gamma_T^{-1}\eta_\alpha(0))$ (using a line-search to optimize ϵ).

This algorithm has to be applied N times (for all α_k, $k = 1, \ldots, N$) in the MAE step.

Remark 1. The matrix Γ_T being typically very badly conditioned, we numerically compute $\Gamma_T^{-1}\eta_\alpha$ after adding a small positive number to the diagonal of Γ_T. The inversion itself is computed using conjugate gradient.

4.3. M Step. There are many similarities between the M-step and the E-step variational problems, so that we will be able to only sketch the detail of the computation here. We need to minimize

$$\widetilde{E}(\mu) = \|\mu\|_{T_0}^2 + \frac{1}{\lambda}\sum_{n=1}^{N}\widetilde{U}^{(n)}\left(x^{(T_{0,\mu})}\right) \qquad (47)$$

with

$$\widetilde{U}^{(n)}\left(x^{(T)}\right) = \left\|\alpha^{(n)}\right\|_T^2 + \frac{1}{\sigma^2}\|S - T_{\alpha^{(n)}}\|_{\text{obs}}^2. \qquad (48)$$

Let us consider the variation of each term in the sum (fixing n, that we temporarily drop from the notation). Since

$$\|\alpha\|_T^2 = \sum_{k,l=1}^{L}\gamma\left(\left\|x_k^{(T)} - x_l^{(T)}\right\|^2\right)(\alpha_k \cdot \alpha_l), \qquad (49)$$

we can write

$$\frac{\partial\|\alpha\|_T^2}{\partial x_k^{(T)}} = 2\sum_{l=1}^{L}\gamma'\left(\left\|x_k^{(T)} - x_l^{(T)}\right\|^2\right)(\alpha_k \cdot \alpha_l)\left(x_k^{(T)} - x_l^{(T)}\right). \qquad (50)$$

The function U being defined as before, we see that the derivative of the second term is given by applying the chain rule again, this time in the form

$$\left(\frac{dx^{(T_\alpha)}}{dx^{(T)}}\right)^* \partial_{x^{(T_\alpha)}} U. \qquad (51)$$

Like in the previous section, the transpose of the differential applied to the gradient of U can be computed by

solving a dynamical system backward in time. In fact, it is the same system as with the variation in α, namely,

$$\frac{d}{dt}\begin{pmatrix}\eta_x \\ \eta_\alpha\end{pmatrix} = -J^{(T,\alpha)}(t)^*\begin{pmatrix}\eta_x \\ \eta_\alpha\end{pmatrix} \qquad (52)$$

still initialized with $\eta_x(1) = \partial_{x^{(T_\alpha)}} U$ and $\eta_\alpha(1) = 0$, but the relevant result now is $\eta_x(0)$. The gradient of $\widetilde{U}$ is then

$$\partial_{x^{(T)}} U = \partial_{x^{(T)}}\|\alpha\|_T^2 + \frac{1}{\sigma^2}\eta_x(0). \qquad (53)$$

Once this is computed, the next step is to compute (reintroducing n in the notation)

$$\left(\frac{dx^{(T_{0\mu})}}{d\mu}\right)^*\left(\sum_{n=1}^{N}\partial_{x^{(T)}} U^{(n)}\right). \qquad (54)$$

This follows a similar procedure, using (44), with T_0 instead of T and μ instead of α. This requires solving

$$\frac{d}{dt}\begin{pmatrix}\eta_x \\ \eta_\mu\end{pmatrix} = -J^{(T_0,\mu)}(t)^*\begin{pmatrix}\eta_x \\ \eta_\mu\end{pmatrix}, \qquad (55)$$

initialized with $\eta_x(1) = \partial_{x^{(T)}}\widetilde{U}$ and $\eta_\mu(1) = 0$. The variation of $\widetilde{E}$ associated to an infinitesimal variation of μ is then

$$\begin{aligned}\delta E &= \langle \delta\mu, \, 2\mu \rangle_{T_0} + \frac{1}{\lambda}\left(\delta\mu \cdot \eta_\mu(0)\right) \\ &= \left\langle \delta\mu, \, 2\mu + \frac{1}{\lambda}\Gamma_{T_0}^{-1}\eta_\mu(0)\right\rangle.\end{aligned} \qquad (56)$$

We summarize the M step in the following algorithm.

Algorithm 2 (M-Step Algorithm for Surface Template Estimation).

(1) For $n = 1, \ldots, N$:

(1.1) Compute $\partial_{x^{(T_{\alpha^{(n)}})}} U$ using (37).

(1.2) Solve system (44) backward in time with initial condition $\eta_x^{(n)}(1) = \partial_{x^{(T_{\alpha^{(n)}})}} U$ and $\eta_\alpha^{(n)}(1) = 0$.

(1.3) Compute

$$\partial_{x^{(T)}}\widetilde{U}^{(n)} = \partial_{x^{(T)}}\left\|\alpha^{(n)}\right\|_T^2 + \frac{1}{\sigma^2}\eta_x^{(n)}(0) \qquad (57)$$

using (50).

(2) Solve system (55) backward in time with

$$\eta_x(1) = \sum_{n=1}^{N}\partial_{x^{(T)}}\widetilde{U}^{(n)} \qquad (58)$$

and $\eta_\mu(1) = 0$.

(3) Replace μ by $\mu - \epsilon(\mu + \Gamma_{T_0}^{-1}\eta_\mu(0)/\lambda)$, ϵ being optimized with a line search.

(a) S_1 (b) S_2 (c) S_3 (d) S_4 (e) S_5

(f) S_6 (g) S_7 (h) S_8 (i) S_9

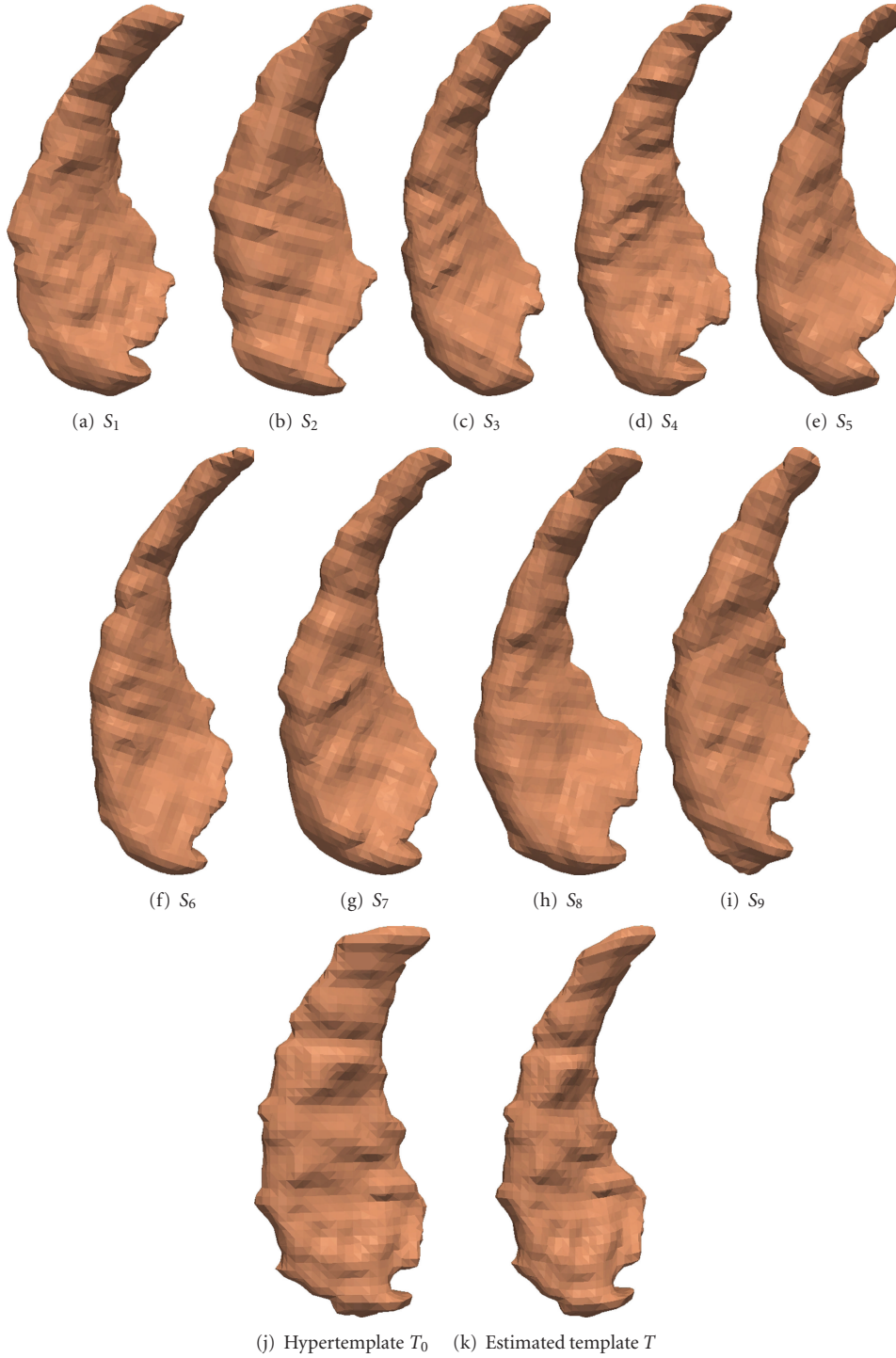(j) Hypertemplate T_0 (k) Estimated template T

Figure 2: Estimating the surface template from 9 caudate data. (a)–(i) observed surfaces. (j) is the hypertemplate. (k) is the result.

4.4. Surface Template Estimation Algorithm. We finally summarize the surface template estimation algorithm:

Algorithm 3 (Surface Template Estimation). Having the hypertemplate T_0 and observed surfaces $S^{(1)}, \ldots, S^{(N)}$, the goal is to estimate the template T. Let $T, \mu, \alpha^{(1)}, \ldots, \alpha^{(N)}$ denote the current estimation with initial guess $T = T_0$,

$\mu = 0$, $\alpha^{(n)} = 0$. Then, in the next iteration, update with the following steps:

(i) With T fixed, apply Algorithm 1 to update each $\alpha^{(n)}$, $n = 1, \ldots, N$.

(ii) With $\alpha^{(n)}$'s fixed, apply Algorithm 2 to obtain a new value for μ.

(a) S_1 (b) S_2 (c) S_3 (d) S_4 (e) S_5

(f) S_6 (g) S_7 (h) S_8 (i) S_9
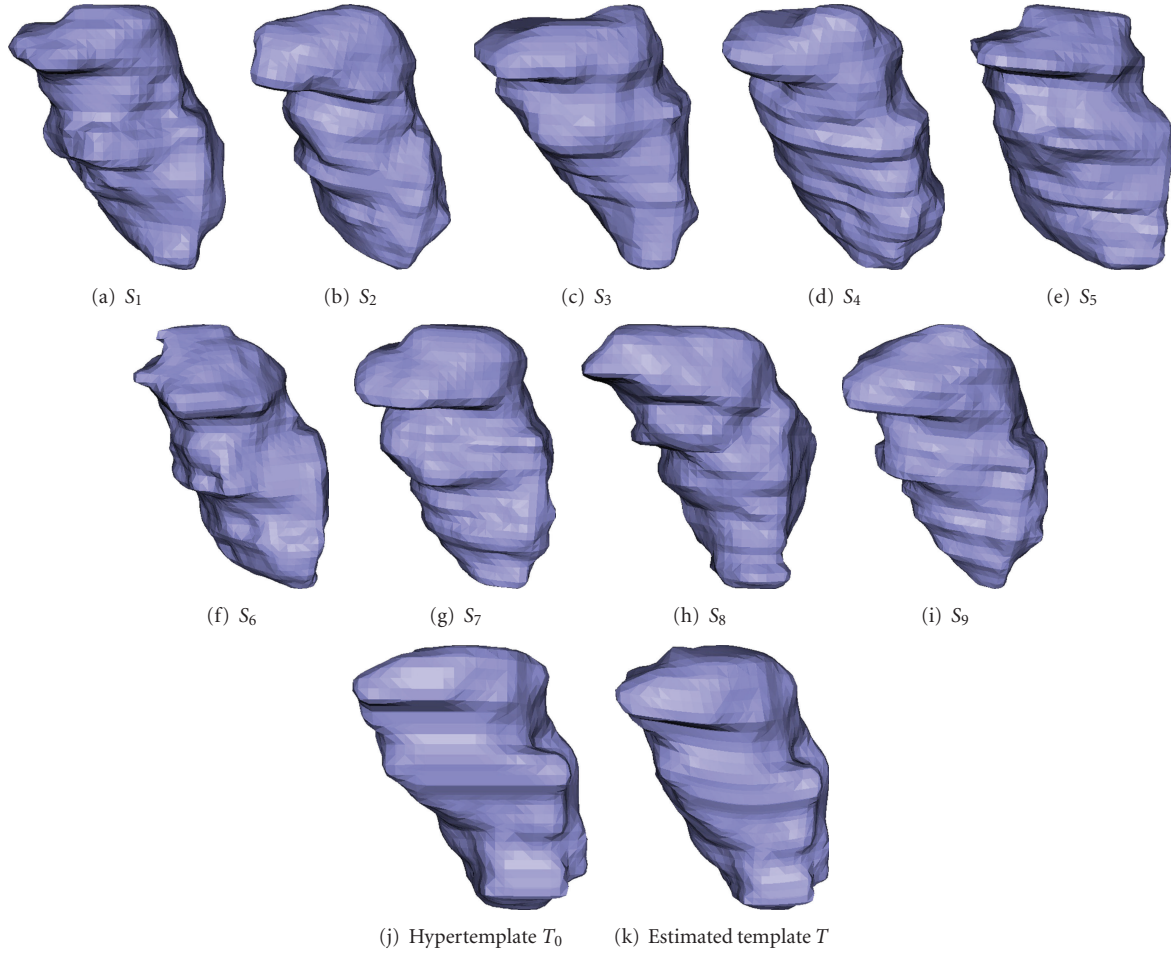
(j) Hypertemplate T_0 (k) Estimated template T

FIGURE 3: Estimating the surface template from 9 thalamus data. (a)–(i) observed surfaces. (j) is the hypertemplate. (k) is the result.

(iii) Solve (7) initialized with the hypertemplate T_0 and the newly obtained μ to update the estimated template, T.

5. Result and Discussion

We applied the algorithm to surface data of human brain's caudate, thalamus, and hippocampus. All data are courtesy of Center for Imaging Science at Johns Hopkins University. Each surface has around 5–10 thousand triangle cells. We randomly chose one as the hypertemplate and the others as observed surfaces. In these experiments, we set $\lambda = 1.0$ and $\sigma^2 = 1.0$.

Figures 2 and 3 are the template estimation result for caudate and thalamus, respectively.

We also applied the algorithm to 101 hippocampus surface data in the BIRN Project (Biomedical Informatics Research Network). In Figure 4, Panels (a)–(h) are 8 examples of the 101 observations. Panel (i) is the hypertemplate and Panel (j) is the estimated template.

The result is visually satisfying in the sense that the estimated template is found to agree with a qualitative representation of the population. For example, in the caudate experiment, the estimated template has an obviously narrower upper tip than the hypertemplate. This captures the population characteristic since most observed data have narrower upper tips. Notice that the obtained template does not look smoother than the rest of the population, as would typically yield template estimation methods that average over fixed parametrizations.

Figure 5 shows how the energy in (31) changes with the iteration in the caudate experiment. This confirms the effectiveness and convergence of the algorithm. One can see the energy drops quickly in the first twenty iterations, then gradually slows down. After the 35th iteration, the energy changes little and the estimated template remains stable.

In our model, the hypertemplate can be provided by an atlas obtained from other studies, although we here simply choose one of the surfaces in the population. Actually, as Figure 6 shows, different choices for the hypertemplate yield very similar results.

6. Conclusion

In this paper, we have presented a Bayesian approach for surface template estimation. We have built, for this purpose, a generative statistical model for surfaces: the construction first applies a random initial momentum to the template

(a) S_1 (b) S_2 (c) S_3 (d) S_4

(e) S_5 (f) S_6 (g) S_7 (h) S_8

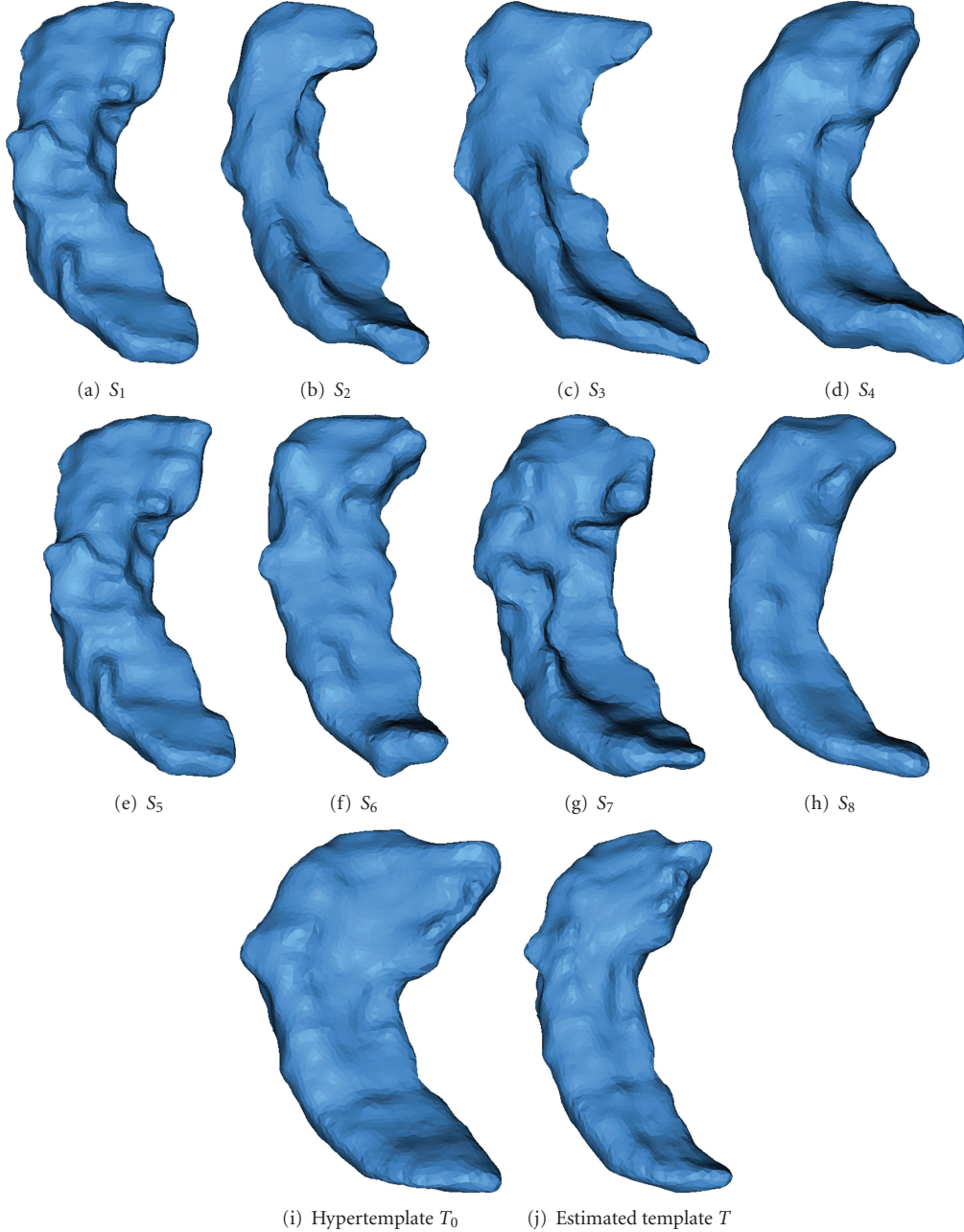(i) Hypertemplate T_0 (j) Estimated template T

FIGURE 4: Estimating the surface template from 101 data. (a)–(h) are 8 examples out of 101 observed surfaces. (i) is the hypertemplate. (j) is the result.

surface, then assumes an observation process using test functions and noise. The template is assumed to be generated as a random deformation of a *hypertemplate*, completing the Bayesian model. We used a mode approximation EM scheme to estimate the surface template, and introduced for this purpose a novel surface matching algorithm optimizing with respect to the initial momentum. The procedure has been tested with caudate, thalamus, and hippocampus surface data, showing its effectiveness and convergence, and also experimentally proved to be robust to variations in the choice of the hypertemplate.

Appendices

A. Proof of Proposition 1

Let us assume that $L_{obs}\gamma_s = \delta_0$, that is, γ_s is the Green Kernel of the operator L_{obs}. This assumption implies, in particular, that $\gamma_s = \gamma_{obs}$. Given a smooth function f, denote

$$f_J(x) = \sum_{j,j' \in J} r_{j,j'}^{(obs)} \gamma_{obs}\left(x - z_j\right) f\left(z_{j'}\right). \qquad (A.1)$$
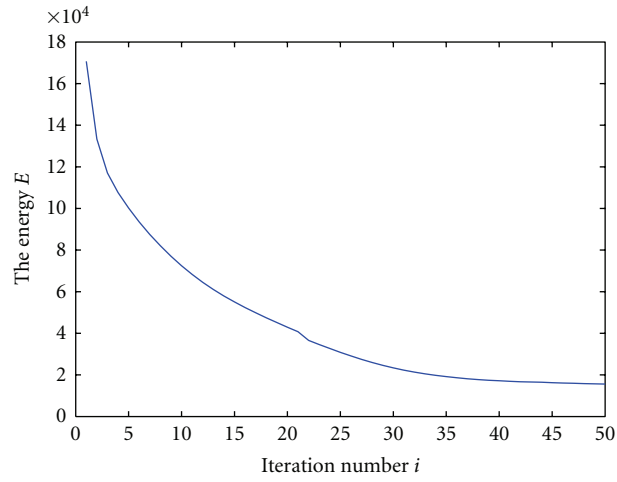
FIGURE 5: Energy change with the iteration.

(a) Hypertemplate 1

(b) Result 1

(c) Hypertemplate 2

(d) Result 2

(e) Hypertemplate 3

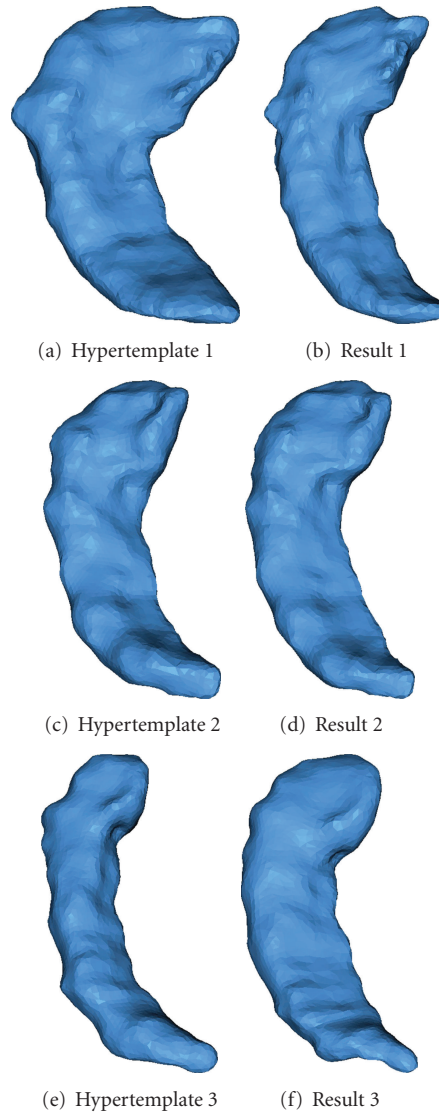(f) Result 3

FIGURE 6: For the same observed population, we choose different surfaces as hypertemplate. The results only have minor differences.

Define the vector space

$$V_J = \left\{ g \in V : g(x) = \sum_{j \in J} \gamma_{\text{obs}}\left(x - z_j\right)\rho_j, \ \rho_j \in \mathbb{R}, \ j \in J \right\}. \tag{A.2}$$

Introduce the RKHS V with scalar product given by

$$\langle w \, , \, w' \rangle_V = \int_{\mathbb{R}^d} (L_{\text{obs}}w)w' \, dx. \tag{A.3}$$

Then f_J can be identified to the orthogonal projection of f on V_j, because $r^{(\text{obs})}$ is the inverse of the Gram matrix (matrix of dot products) of $(\gamma_{\text{obs}}(\cdot - z_j), j \in J)$, which are the generators of V_J, and

$$f\left(z_j\right) = \left\langle f \, , \, \gamma_{\text{obs}}\left(\cdot, z_j\right) \right\rangle_V \tag{A.4}$$

by assumption.

Now, let J_m be a sequence of progressively finer grids with resolution ϵ_m tending to 0 when m tends to infinity. We prove that $\|f_{J_m} - f\|_V \to 0$, for which it suffices to prove that $\bigcup V_{J_m}$ is dense in V. This is equivalent to the fact that no nonzero g in V can be orthogonal to all the V_{J_m}'s. But g orthogonal to V_{J_m} is only possible when, for all $j \in J_m$,

$$g\left(z_j\right) = \left\langle g, \ \gamma_w\left(\cdot - z_j\right) \right\rangle_V = 0. \tag{A.5}$$

Since the z_j's form an arbitrarily fine grid in V and functions in V are continuous, this implies $g = 0$.

So, $f_J \to f$ in V, which implies, for example, pointwise convergence as soon as V is embedded in the set of continuous functions (that is, if γ_{obs} is continuous). This directly implies Proposition 1 in the case $\gamma_s = \gamma_{\text{obs}}$, by taking $f(x) = \gamma_{\text{obs}}(x - y)$ for a given y.

Extensions of this result is when γ_s is obtained by applying some linear operator, say A, to γ_{obs}. One then has

$$K_{\text{obs}}(\cdot, \tilde{x}) = A\left(\sum_{j,j' \in J} r_{j,j'}^{(w)} \gamma_{\text{obs}}\left(\cdot - z_j\right)\gamma_s\left(\tilde{x} - z_{j'}\right) \right) \tag{A.6}$$

and passing to the limit in the sum (for which one needs $\gamma_s \in V$ and A continuous on V), one gets $K_{\text{obs}}(x, \tilde{x}) \to A^2\gamma_{\text{obs}}$.

B. Transposing Linear Differential Equations

We here justify the procedure described in Section 4, and prove the following fact: consider the solution, $z(t) \in \mathbb{R}^p$, of the ordinary differential equation

$$\partial_t z = J(t)z, \tag{B.1}$$

where J is a time-dependent operator (a p by p matrix). Then, for any $u \in \mathbb{R}^p$, we have

$$u \cdot z(1) = \eta(0) \cdot z(0), \tag{B.2}$$

where η is the solution of the differential equation

$$\partial_s \eta = -J(s)^* \eta \tag{B.3}$$

with $\eta(1) = u$.

Let us prove this result. First remark that since z is the solution of a linear equation, it depends linearly on the initial condition, $z(0)$. More precisely, let $M(s,t)$ be the p by p matrix such that

$$\partial_t M(s,t) = J(t)M(s,t) \tag{B.4}$$

and $M(s,s) = Id_{\mathbb{R}^p}$. Then $z(t) = M(0,t)z(0)$, and, obviously, $\eta(0) = M(0,1)^*u$. Using the identity $M(t,s)M(s,t) = Id_{\mathbb{R}^p}$, we obtain

$$0 = M(t,s)(\partial_s M(s,t)) + (\partial_s M(t,s))M(s,t)$$
$$= (\partial_s M(s,t))M(t,s) + J(s) \tag{B.5}$$

so that $\partial_s M(s,t) = -M(s,t)J(s)$. Computing the transposed equation yields and taking $t = 1$

$$\partial_s M(s,1)^* = -J(s)^* M(s,1)^*. \tag{B.6}$$

Thus, if $\eta(s) = M(s,1)^*u$, we have $\eta(1) = u$ and $\partial_s \eta = -J(s)^* \eta$ as announced.

Acknowledgments

References

[1] S. Durrleman, X. Pennec, A. Trouvé, and N. Ayache, "A forward modelto build unbiased atlases from curves and surfaces," in *Proceedings of the International Workshop on the Mathematical Foundations of Computational Anatomy (MFCA '08)*, New York, NY, USA, September 2008.

[2] L. Shen, H. Farid, and M. A. McPeek, "Modeling three-dimensional morphological structures using spherical harmonics," *Evolution*, vol. 63, no. 4, pp. 1003–1016, 2008.

[3] A. Guimond, J. Meunier, and J.-P. Thirion, "Average brain models: a convergence study," *Computer Vision and Image Understanding*, vol. 77, no. 2, pp. 192–210, 2000.

[4] W. Mio, A. Srivastava, and S. Joshi, "On shape of plane elastic curves," *International Journal of Computer Vision*, vol. 73, no. 3, pp. 307–324, 2007.

[5] P. T. Fletcher, S. Joshi, C. Lu, and S. M. Pizer, "Principal geodesic analysis for the study of nonlinear statistics of shape," *IEEE Transactions on Medical Imaging*, vol. 23, no. 8, pp. 995–1005, 2004.

[6] H. Le and A. Kume, "The Fréchet mean shape and the shape of the means," *Advances in Applied Probability*, vol. 32, no. 1, pp. 101–113, 2000.

[7] X. Pennec, "Intrinsic statistics on riemannian manifolds: basic tools for geometric measurements," *Journal of Mathematical Imaging and Vision*, vol. 25, no. 1, pp. 127–154, 2006.

[8] M. Vaillant, M. I. Miller, L. Younes, and A. Trouvé, "Statistics on diffeomorphisms via tangent space representations," *NeuroImage*, vol. 23, no. 1, pp. S161–S169, 2004.

[9] M. F. Beg and A. Khan, "Computation of average atlas using lddmmand geodesic shooting," in *Proceedings of the IEEE International Symposium on Biomedical Imaging*, Arlington, Va, USA, April 2006.

[10] P. A. Helm, L. Younes, M. F. Beg et al., "Evidence of structural remodeling in the dyssynchronous failing heart," *Circulation Research*, vol. 98, no. 1, pp. 125–132, 2006.

[11] S. Joshi, B. Davis, M. Jomier, and G. Gerig, "Unbiased diffeomorphic atlas construction for computational anatomy," *NeuroImage*, vol. 23, supplement 1, pp. S151–S160, 2004.

[12] B. Avants and J. Gee, "Symmetric geodesic shape averaging and shape interpolation," in *Proceedings of the Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis, Workshops CVAMIA and MMBIA*, vol. 3117, pp. 99–110, 2004.

[13] B. Avants and J. C. Gee, "Geodesic estimation for large deformation anatomical shape averaging and interpolation," *NeuroImage*, vol. 23, supplement 1, pp. S139–S150, 2004.

[14] S. Allassonnière, Y. Amit, and A. Trouvé, "Towards a coherent statistical framework for dense deformable template estimation," *Journal of the Royal Statistical Society. Series B*, vol. 69, no. 1, pp. 3–29, 2007.

[15] J. Ma, M. I. Miller, A. Trouvé, and L. Younes, "Bayesian template estimation in computational anatomy," *NeuroImage*, vol. 42, no. 1, pp. 252–261, 2008.

[16] M. Vaillant and J. Glaunès, "Surface matching via currents," in *Proceedings of the 19th International Conference on Information Processing in Medical Imaging (IPMI '05)*, vol. 3565, Glenwood Springs, Colo, USA, July 2005.

[17] M. F. Beg, M. I. Miller, A. Trouvé, and L. Younes, "Computing large deformation metric mappings via geodesic flows of diffeomorphisms," *International Journal of Computer Vision*, vol. 61, no. 2, pp. 139–157, 2005.

[18] V. I. Arnol'd, V. I. Arnold, K. Vogtmann, and A. Weinstein, *Mathematical Methods of Classical Mechanics*, Springer, Berlin, Germany, 1989.

[19] J. E. Marsden and T. S. Ratiu, *Introduction to Mechanics and Symmetry*, Springer, Berlin, Germany, 1999.

[20] L. Younes, *Shapes and Diffeomorphisms*, Springer, Berlin, Germany, 2010.

[21] D. D. Holm, J. E. Marsden, and T. S. Ratiu, "The Euler-Poincaré equations and semidirect products with applications to continuum theories," *Advances in Mathematics*, vol. 137, no. 1, pp. 1–81, 1998.

[22] R. Schaback, "Error estimates and condition numbers for radial basis function interpolation," *Advances in Computational Mathematics*, vol. 3, no. 1, pp. 251–264, 1995.