

# Constructing Diffeomorphic Representations for the Groupwise Analysis of Non-Rigid Registrations of Medical Images

Stephen Marsland and Carole Twining

**Abstract**—Groupwise non-rigid registrations of medical images define dense correspondences across a set of images, defined by a continuous deformation field that relates each target image in the group to some reference image. These registrations can be automatic, or based on the interpolation of a set of user-defined landmarks, but in both cases, quantifying the normal and abnormal structural variation across the group of imaged structures implies analysis of the set of deformation fields. We contend that the choice of representation of the deformation fields is an integral part of this analysis. This paper presents methods for constructing a general class of multi-dimensional diffeomorphic representations of deformations. We demonstrate, for the particular case of the polyharmonic clamped-plate splines, that these representations are suitable for the description of deformations of medical images in both two and three dimensions, using a set of (2D) annotated MRI brain slices and a set of (3D) segmented hippocampi with optimised correspondences.

The class of diffeomorphic representations also defines a non-Euclidean metric on the space of patterns, and, for the case of compactly-supported deformations, on the corresponding diffeomorphism group. In an experimental study, we show that this non-Euclidean metric is superior to the usual ad hoc Euclidean metrics in that it enables more accurate classification of legal and illegal variations.

**Index Terms**—Diffeomorphisms, non-rigid registration, interpolating splines, geodesic interpolating splines

## I. INTRODUCTION

**N**ON-RIGID registration algorithms [1], [2], [3], [4], [5] automatically generate dense (i.e., pixel-to-pixel or voxel-to-voxel) correspondences between pairs and groups of images with the aim of aligning corresponding ‘structures’. The deformation fields implicit in this correspondence contain information about the variability of structures across the set. In order to analyse this variability quantitatively we need to be able to analyse the set of deformation fields. Such analysis must be based (either implicitly or explicitly) on a particular mathematical representation of the deformation field. This paper describes a suitable representation for the analysis of deformation fields, and demonstrates its use on medical images in 2D and 3D. The focus of the work is on a common low-dimensional representation of warps from a group of images, either based on a set of pairwise registrations (e.g., using one

of the registration algorithms referenced above), or from an explicitly groupwise algorithm [6].

There has been very little work on modelling dense 2D and 3D deformation fields. What work there has been has either used the densely-sampled deformation vectors directly (e.g., [7], [8]), or has employed a smooth, continuous representation of them (e.g., [9]). Neither of these methods guarantees that the deformation field is diffeomorphic (although B splines can be guaranteed diffeomorphic given certain non-trivial constraints on the control-point displacements [10]).

We contend that the appropriate representation should be continuous and diffeomorphic, as only a diffeomorphic representation allows a smooth and unambiguous one-to-one correspondence between all points in any pair of images. The importance of a one-to-one mapping is clear when the question of modelling and analysis is considered. Most modelling strategies are based on the idea of a training set of examples that all contain different instantiations of the same set of objects/structures – there are examples of modelling strategies where objects can appear or become hidden, but for the purposes of this paper, we will take the simplest assumption that all the relevant structures will appear in all the examples in our training set. In spatial terms, this translates to the assumption that there is some one-to-one spatial correspondence across the set of images. Note here that we are only considering the analysis of the spatial variability of generalised structures present in images, as opposed to the explicit modelling of the images themselves. Modelling images would mean considering pixel-value deformation data as well as pixel (position) deformation data, which leads to the fundamental problem that spatial and pixel-value deformations are *a priori* incommensurate.

The smoothness constraint may be physically violated in the case of intrasubject correspondence, where, for example, organs may slide against one another. However, from the point of view of pixelated images with finite resolution, such a sampled motion can always be represented as a continuous diffeomorphic deformation.

There will be specific instances from medical image analysis where the assumption of one-to-one correspondence fails, for example, where an additional structure such as a tumour appears. How this is represented post-registration, whether as an extreme spatial deformation, an extreme pixel-value deformation, or as some combination of the two, depends on the exact registration algorithm chosen. If the former, this would be indicated by the warp parameters assuming atypical

Joint first authors.

SM and CT are at Imaging Science and Biomedical Engineering, University of Manchester, Manchester M13 9PT, U.K., SM is at Institute of Information Sciences, Massey University, Private Bag 11222, Palmerston North, New Zealand. S.R.Marsland@massey.ac.nz, carole.twining@man.ac.uk

This research was supported by the MIAS IRC project, EPSRC grant number GR/N14248/01.

values when compared with those seen across the training set.

When we are considering the correspondence between discrete and bounded objects such as brains, it is also desirable that the warps themselves should be discrete and bounded. This leads us to suggest that a suitable representation is that of the group of continuous diffeomorphisms with some appropriate set of boundary conditions.

Such a representation can be constructed using an approach based on Geodesic Interpolating Splines (GIS) [11], [12]. In previous work [11], [13] it has been shown that this approach also allows the construction of a metric on the space of knotpoint positions of fixed cardinality. This metric is actually induced by a metric on the full group of Sobolev diffeomorphisms [13]; it should be noted that the polyharmonic clamped-plate splines described in the current paper have compact support, and that this has important theoretical implications when it comes to this construction. Further discussion of this point is beyond the scope of the current paper, and interested readers should consult the specialist mathematical literature – e.g., [14], [15].

There are obvious connections between interpolating the motion of a set of landmarks (either exactly or inexactly) for the purposes of registration, and of representing a densely-defined deformation in terms of the motion of some small set of knotpoints. In a sense, the representation problem we wish to consider is the inverse problem of landmark-based registration. We discuss the similarities and significant differences between these related topics in section III.

Work on the analysis of shape variability (a simpler problem than that of image deformation) has used a range of representations; examples include polygonal [16] or spline [17] representations based on a small set of corresponding points (landmarks), Fourier representations [18] or spherical harmonics [19], medial based representations [20], or combinations of these [21]. The importance of the choice of representation has been demonstrated by the fact that explicitly optimizing the representation can lead to improved model performance [22].

In this paper we demonstrate the construction of these diffeomorphic representations using a variety of spline bases. We show that these representations generate warps that are suitable for the task in hand, giving biologically ‘plausible’ warps in both two and three dimensions, whilst being of a relatively low dimensionality. We further study the significance of the metric (geodesic) distances between warps, and show that using them provides a measure of atypical variation that has greater discriminatory power than naïve measures based on the ad hoc use of a Euclidean metric on the space of warp parameters.

## II. THE GEODESIC INTERPOLATING SPLINE

### A. Interpolating Splines

Consider a vector-valued spline function  $\vec{f}(\vec{x})$ ,  $\vec{x} \in \mathbb{R}^n$ ,  $\vec{f}(\vec{x}) \in \mathbb{R}^n$ , that interpolates between data values at a set of knotpoints  $\{\vec{x}_i : i = 1 \text{ to } N\}$ , where  $\vec{f}(\vec{x}_i) = \vec{f}_i$ . We will restrict ourselves to the class of splines that can be expressed as the minimiser of a functional Lagrangian of the

form [23]:

$$\begin{aligned} E[\vec{f}] &= \int_{\mathbb{R}^n} d\vec{x} \left\| L\vec{f}(\vec{x}) \right\|^2 + \sum_{i=1}^N \vec{\lambda}_i \cdot \left( \vec{f}(\vec{x}_i) - \vec{f}_i \right) \\ &= \int_{\mathbb{R}^n} d\vec{x} \vec{f}(\vec{x}) \cdot (L^\dagger L) \vec{f}(\vec{x}) \\ &\quad + \sum_{i=1}^N \vec{\lambda}_i \cdot \int_{\mathbb{R}^n} d\vec{x} \left( \vec{f}(\vec{x}) - \vec{f}_i \right) \delta(\vec{x} - \vec{x}_i), \end{aligned} \quad (1)$$

where  $L$  is some scalar differential operator, with Lagrange dual  $L^\dagger$ . The first term in the Lagrangian is the smoothing term, or the energy of the displacement field; the second term with the (vector-valued) Lagrange multipliers  $\{\vec{\lambda}_i\}$  ensure that the spline fits the data at the knotpoints<sup>1</sup>. A particular spline basis is then defined by specifying the choice of operator  $L$  and the boundary conditions on  $\vec{f}(\vec{x})$ .

The general form of the functional minimiser is obtained by solving the Euler-Lagrange equations of  $E$  using the technique of Green’s functions, as follows. Taking partial and functional derivatives, we obtain:

$$\begin{aligned} \frac{\partial E}{\partial \vec{\lambda}_i} &= \vec{f}(\vec{x}_i) - \vec{f}_i, \\ \therefore \frac{\partial E}{\partial \vec{\lambda}_i} &= 0 \Rightarrow \vec{f}(\vec{x}_i) = \vec{f}_i, \end{aligned} \quad (2)$$

$$\begin{aligned} \frac{\delta E}{\delta \vec{f}(\vec{x})} &= 2(L^\dagger L) \vec{f}(\vec{x}) + \sum_{i=1}^N \vec{\lambda}_i \delta(\vec{x} - \vec{x}_i) \\ \therefore \frac{\delta E}{\delta \vec{f}(\vec{x})} &= 0 \Rightarrow (L^\dagger L) \vec{f}(\vec{x}) = -\frac{1}{2} \sum_{i=1}^N \vec{\lambda}_i \delta(\vec{x} - \vec{x}_i). \end{aligned} \quad (3)$$

We hence see that the general solution to (3) can be written in the form:

$$\vec{f}(\vec{x}) = \vec{g}(\vec{x}) + \sum_{i=1}^N \vec{\alpha}_i G(\vec{x}, \vec{x}_i), \quad (4)$$

where:

the generalised ‘affine’ function<sup>2</sup>  $g$  is a solution of:

$$L\vec{g}(\vec{x}) = 0, \quad (5)$$

the Green’s function  $G$  is a solution of:

$$(L^\dagger L) G(\vec{x}, \vec{y}) \propto \delta(\vec{x} - \vec{y}), \quad (6)$$

and the set of  $\vec{\alpha}_i \in \mathbb{R}^n$  are the spline coefficients. Table I gives a selection of commonly used Green’s functions, for varying choice of operator  $L$  and varying boundary conditions on  $\vec{f}(\vec{x})$ .

The choice of a particular Green’s function depends on the boundary conditions and smoothness appropriate to the problem considered. For example, the clamped-plate spline (CPS) Green’s function is useful for discrete objects such

<sup>1</sup>The equivalent *smoothing* spline would be a minimum of this Lagrangian for a *particular* set of values of  $\{\vec{\lambda}_i\}$ .

<sup>2</sup>We use the term ‘affine’ in inverted commas because although in some cases this function does indeed correspond to the usual definition of an affine function (that is, a general linear function), this is not always the case. For example, if the operator  $L$  was a fourth-order differential operator, the generalised ‘affine’ function would actually be cubic.

TABLE I

A SELECTION OF GREEN'S FUNCTIONS.  $\mathbb{D}^n$  IS USED TO DENOTE THE UNIT BALL IN  $\mathbb{R}^n$ , AND  $\partial\mathbb{D}^n$  IS ITS BOUNDARY.

Name	Dim	$L^\dagger L$	Boundary conditions on $f(\vec{x})$	$G(\vec{x}, \vec{y})$
thin-plate (TPS) [24]	even	$(\nabla^2)^2$	asymptotically linear	$\ \vec{x} - \vec{y}\ ^{4-n} \log \ \vec{x} - \vec{y}\ $
thin-plate (TPS) [24]	odd	$(\nabla^2)^2$	asymptotically linear	$\ \vec{x} - \vec{y}\ ^{4-n}$
biharmonic clamped plate [25], [26] (CPS)	2	$(\nabla^2)^2$	$f = f' = 0$ on and outside $\partial\mathbb{D}^2$	$\ \vec{x} - \vec{y}\ ^2 (A^2 - 1 - \log A^2),$ $A(\vec{x}, \vec{y}) = \frac{\sqrt{\vec{x}^2 \vec{y}^2 - 2\vec{x} \cdot \vec{y} + 1}}{\ \vec{x} - \vec{y}\ }$
triharmonic clamped plate [25] (CPS)	3	$(\nabla^2)^3$	$f = f' = 0$ on and outside $\partial\mathbb{D}^3$	$\ \vec{x} - \vec{y}\  (A + \frac{1}{A} - 2),$ $A(\vec{x}, \vec{y}) = \frac{\sqrt{\vec{x}^2 \vec{y}^2 - 2\vec{x} \cdot \vec{y} + 1}}{\ \vec{x} - \vec{y}\ }$
Gaussian	$n$	$\exp\left(-\frac{\nabla^2}{4\beta}\right)$	asymptotically linear	$\exp(-\beta\ \vec{x} - \vec{y}\ ^2)$

as brains, whereas an image of knee cartilage would require asymptotically linear boundary conditions.

### B. The Generalised 'Affine' Function

Taking the general form of  $\vec{f}(\vec{x})$  given above (4), substituting into (1), and applying the set of constraints  $\{\vec{f}(\vec{x}_i) = \vec{f}_i : i = 1, \dots, N\}$  (2), we obtain:

$$E[\{\vec{\alpha}_k\}] = \sum_{i,j} G(\vec{x}_i, \vec{x}_j) \vec{\alpha}_i \cdot \vec{\alpha}_j = G_{ij} \vec{\alpha}_i \cdot \vec{\alpha}_j, \quad (7)$$

where we define:  $G_{ij} \doteq G(\vec{x}_i, \vec{x}_j)$ .

The spline coefficients  $\{\vec{\alpha}_i\}$  and the generalised 'affine' function (5) are then obtained by optimising the above expression with the set of constraints (2), which we can now write as:

$$\vec{g}(\vec{x}_i) = \vec{f}_i - G_{ij} \vec{\alpha}_j \quad \forall i = 1, \dots, N. \quad (8)$$

For operators where the boundary conditions are that the deformation is asymptotically linear, the generalised 'affine' function is the general linear function. For this case, using the notation of Camion and Younes [11], let us define:

$$\begin{aligned} \{\vec{x}_i, i = 1, \dots, N\} &\in \mathbb{R}^n \\ \text{with Cartesian components: } \vec{x}_i &= \{x_i^\mu : \mu = 1, \dots, n\}. \\ Q &\doteq \begin{pmatrix} x_1^1 & x_1^2 & \dots & x_1^n & 1 \\ & & & \vdots & \\ x_N^1 & x_N^2 & \dots & x_N^n & 1 \end{pmatrix}. \end{aligned} \quad (9)$$

Hence the matrix  $Q$  is of size  $N \times (n+1)$ , with elements  $Q_{iA}$ , where  $i = 1, \dots, N$ ,  $A = 1, \dots, (n+1)$ .

By definition, the action of a general linear transformation  $g$  on a single point  $\vec{x} = \{x^\mu : \mu = 1, \dots, n\} \in \mathbb{R}^n$  can be written as:

$$\vec{x} \xrightarrow{g} \vec{g}(\vec{x}) \in \mathbb{R}^n, \quad g^\mu(\vec{x}) = \gamma_{n+1}^\mu + \sum_{\nu=1}^n \gamma_\nu^\mu x^\nu, \quad (10)$$

where  $\{\{\gamma_1^\mu, \dots, \gamma_{n+1}^\mu\} : \mu = 1, \dots, n\}$  are the set of  $n(n+1)$  independent parameters that define the

transformation. Using the index notation defined above, we can also write the set of parameters in the form  $\{\gamma_A^\mu : \mu = 1, \dots, n, A = 1, \dots, (n+1)\}$ . So, we can now see by combining (9) and (10) that the action of a general linear transformation on the entire set of knotpoints  $g : \{\vec{x}_i\} \mapsto \{\vec{g}(\vec{x}_i)\}$  can be written in the compact form:

$$g^\mu(\vec{x}_i) = Q_{iA} \gamma_A^\mu, \quad (11)$$

where again we use the summation convention on repeated indices.

Taking components, the constraint equation (8) can now be written as:

$$\begin{aligned} g^\mu(\vec{x}_i) &= f_i^\mu - G_{ij} \alpha_j^\mu = Q_{iA} \gamma_A^\mu, \\ \Rightarrow \alpha_j^\mu &= G_{jk}^{-1} f_k^\mu - G_{jk}^{-1} Q_{kA} \gamma_A^\mu. \end{aligned} \quad (12)$$

We could now substitute from (12) into (7) to obtain an explicit expression for the energy as a function of the parameters of the general linear function  $\{\gamma_A^\mu\}$ ; setting the first functional derivative of this expression to zero would then give the desired solution. However, it is simpler to use the chain rule for derivatives as follows:

$$\begin{aligned} \text{From (7):} \quad E[\{\vec{\alpha}_k\}] &= G_{ij} \vec{\alpha}_i \cdot \vec{\alpha}_j = \alpha_i^\mu G_{ij} \alpha_j^\mu \\ \therefore \frac{\partial E}{\partial \alpha_j^\nu} &= 2G_{jk} \alpha_k^\nu. \end{aligned} \quad (13)$$

$$\begin{aligned} \text{From (12):} \quad \alpha_j^\nu &= G_{jk}^{-1} f_k^\nu - G_{jk}^{-1} Q_{kA} \gamma_A^\nu \\ \therefore \frac{\partial \alpha_j^\nu}{\partial \gamma_A^\mu} &= -G_{jm}^{-1} Q_{mA} \\ \frac{\partial E}{\partial \gamma_A^\mu} &\equiv \frac{\partial E}{\partial \alpha_j^\nu} \frac{\partial \alpha_j^\nu}{\partial \gamma_A^\mu} \end{aligned} \quad (14)$$

$$\begin{aligned} \therefore \frac{\partial E}{\partial \gamma_A^\mu} &= 0 \Rightarrow G_{jk} \alpha_k^\nu G_{jm}^{-1} Q_{mA} = 0 \\ &\Rightarrow \alpha_m^\nu Q_{mA} = 0. \end{aligned} \quad (15)$$

<sup>3</sup>We adopt the summation convention that we sum over repeated indices.

Substituting for  $\alpha_m^\nu$  from (12) then gives the final results for

the affine and spline parameters in closed form:

$$\begin{aligned}\gamma_A^\mu &= (Q^T G^{-1} Q)_{AB}^{-1} Q_{Bi}^T G_{ij}^{-1} f_j^\mu, \quad (16) \\ \alpha_i^\mu &= G_{ij}^{-1} (f_j^\mu - Q_{jA} \gamma_A^\mu).\end{aligned}$$

For the case of the clamped-plate spline, the boundary conditions are that the deformation is zero on and outside the unit ball, which means that the generalised ‘affine’ function  $\vec{g}(\vec{x})$  in (4) is identically zero. The relation between the spline parameters  $\{\vec{\alpha}_i\}$  and the knotpoint displacements  $\{f_i^\mu\}$  for the CPS is hence given by setting the affine parameters  $\{\gamma_A^\mu\}$  to zero in (12). Unless stated otherwise, when using the CPS, we apply a Procrustes affine alignment (of the form given in (5)) to the sets of knotpoints *before* the boundary conditions are imposed.

Let us now consider using the splines defined in table I to represent the densely-sampled (i.e., at every pixel/voxel position in the image) deformation field of an image, where a point at original position  $\vec{x}$  is warped to a position  $\vec{x} + \vec{f}(\vec{x})$ . There will be some set of knotpoints (with associated initial and final positions) and knotpoint displacements for which the interpolated displacement field matches our given displacement field at all sample points, to some required level of accuracy (this point is discussed in detail in Section II-E.3). We will call such a set of knotpoints and knotpoint displacements a representation of the deformation field  $\vec{f}(\vec{x})$ . However, such an interpolated deformation field is only guaranteed to be diffeomorphic in the limit of sufficiently small displacements.

### C. Geodesic Interpolating Splines

The usual approach to constructing a large-deformation diffeomorphism is to consider such a deformation as an infinite sequence of infinitesimal deformations [11], [27], [12], [28]; that is, we have an infinite sequence of the spline-part generated by the Green’s function  $G$ , and an infinite sequence of infinitesimal affine transformations.

Consider a sequence of interpolating spline transformations being applied between the flowtimes  $t = 0$  and  $t = 1$ . Under this sequence of infinitesimal deformations, consider a particle  $P$  that follows the flow; it starts at a general point  $\vec{x}(0) \in \mathbb{R}^n$  and follows a path  $\vec{x}(t)$ . Between the times  $t$  and  $t + dt$ , the particle that has reached  $\vec{x}(t)$  then moves to  $\vec{x}(t+dt) \approx \vec{x}(t) + dt \frac{d\vec{x}(t)}{dt}$ . We define the time-dependant Eulerian velocity field  $\vec{v}(t, \vec{y})$  thus:

$$\vec{v}(t, \vec{x}(t)) \doteq \frac{d\vec{x}(t)}{dt}, \quad (17)$$

so that the particle  $P$  follows a path defined by this time-varying velocity field. We can now see that the analogue of the deformation field  $\vec{f}(\vec{x})$  for the segment  $t$  to  $t + dt$  is the quantity  $dt \cdot \vec{v}(t, \vec{x})$ . The analogue of the total deformation field

energy for the time segment  $t$  to  $t + dt$  is then given by:

$$\begin{aligned}\Delta E &= \int_{\mathbb{R}^n} d\vec{x} \vec{v}(t, \vec{x}) (L^\dagger L) \vec{v}(t, \vec{x}), \\ \text{and } E[\vec{v}(t, \vec{x})] &\doteq \int_0^1 dt \int_{\mathbb{R}^n} d\vec{x} \vec{v}(t, \vec{x}) (L^\dagger L) \vec{v}(t, \vec{x}), \quad (18)\end{aligned}$$

We now consider a set of knotpoints, with paths  $\{\vec{x}_i(t) : i = 1, \dots, N\}$  whose motion we wish to interpolate. In the Lagrangian for the interpolating splines (1), the constraint on the deformation field is that it reproduces the displacement of the knotpoints – the obvious analogue of this is that the velocity field is always such that the knotpoints move according to it. That is:

$$\frac{d\vec{x}_i(t)}{dt} \doteq \vec{v}_i(t) = \vec{v}(t, \vec{x}_i(t)) \quad \forall t \in [0, 1] \text{ and } \forall i = 1, \dots, N. \quad (19)$$

For a given set of knotpoint flowpaths  $\{\vec{x}_i(t)\}$ , the Lagrangian is:

$$\begin{aligned}E[\{\vec{x}_i(t)\}] &= \int_0^1 dt \int_{\mathbb{R}^n} d\vec{x} \vec{v}(t, \vec{x}) \cdot (L^\dagger L) \vec{v}(t, \vec{x}) \\ &+ \sum_{i=1}^N \int_0^1 dt \int_{\mathbb{R}^n} d\vec{x} \vec{\lambda}_i(t) \cdot \left( \vec{v}(t, \vec{x}) - \frac{d\vec{x}_i(t)}{dt} \right) \delta(\vec{x} - \vec{x}_i(t)), \quad (20)\end{aligned}$$

where the set of functions  $\{\vec{\lambda}_i(t)\}$  are the Lagrange multipliers. Taking derivatives:

$$\begin{aligned}\frac{\delta E}{\delta \vec{\lambda}_i(t)} &= \vec{v}(t, \vec{x}_i(t)) - \frac{d\vec{x}_i(t)}{dt} \\ \therefore \frac{\delta E}{\delta \vec{\lambda}_i(t)} &= 0 \Rightarrow \vec{v}(t, \vec{x}_i(t)) = \frac{d\vec{x}_i(t)}{dt}. \quad (21) \\ \frac{\delta E}{\delta \vec{v}(t, \vec{x})} &= 2 (L^\dagger L) \vec{v}(t, \vec{x}) + \sum_{i=1}^N \vec{\lambda}_i(t) \delta(\vec{x} - \vec{x}_i(t)) \\ \therefore \frac{\delta E}{\delta \vec{v}(t, \vec{x})} &= 0 \\ \Rightarrow (L^\dagger L) \vec{v}(t, \vec{x}) &= -\frac{1}{2} \sum_{i=1}^N \vec{\lambda}_i(t) \delta(\vec{x} - \vec{x}_i(t)). \quad (22)\end{aligned}$$

So, in an analogous manner to (4), (5), and (6), the general solution can be written in the form:

$$\vec{v}(t, \vec{x}) = \vec{g}(t, \vec{x}) + \sum_{i=1}^N \alpha_i(t) G(\vec{x}, \vec{x}_i(t)), \quad (23)$$

where the Green’s function  $G(\cdot, \cdot)$  is as defined previously (6). The spline parameters  $\{\alpha_i(t)\}$  and the generalised ‘affine’ function  $\vec{g}(t, \vec{x})$  now depend continuously on the flowtime  $t$ ,

<sup>5</sup>Velocity fields for which this integral exists, and for which the integral is finite, belong to the relevant Sobolev space of functions; the associated diffeomorphism group is the group of Sobolev diffeomorphisms, which is different to, say, the group of  $C^k$  or  $C^\infty$  diffeomorphisms. See [14] for further details.

<sup>4</sup>Note that the expression for  $\gamma_a^\mu$  differs from [11], where there is a typographical error and the second factor of  $G^{-1}$  has been omitted.

giving us the infinite sequence of infinitesimal spline transformations and generalised ‘affine’ transformations.

At each instant of flowtime  $t$ , the closed-form solution for the spline parameters  $\{\alpha_i(t)\}$  and the generalised ‘affine’ function  $\bar{g}(t, \vec{x})$  is given by applying the constraint given in (21). The results are the exact analogues of those given previously, except that (as above) at each time instant  $t$ , the components of the spline displacements  $\{f_i^\mu\}$  are now replaced by the components of the time-dependant velocities of the knotpoints  $\{v_i^\mu(t)\}$  (19). We define:

$$Q(t) \doteq \begin{pmatrix} x_1^1(t) & x_1^2(t) & \dots & x_1^n(t) & 1 \\ & & & \vdots & \\ & & & & \\ x_N^1(t) & x_N^2(t) & \dots & x_N^n(t) & 1 \end{pmatrix}, \quad (24)$$

$$G_{ij}(t) \doteq G(\vec{x}_i(t), \vec{x}_j(t)). \quad (25)$$

A general *time-dependant* linear transformation is given by:

$$\begin{aligned} g^\mu(t, \vec{x}) &= \gamma_{n+1}^\mu(t) + \sum_{\nu=1}^n \gamma_\nu^\mu(t) x^\nu \\ \Rightarrow g^\mu(t, \vec{x}_i(t)) &= Q_{iA}(t) \gamma_A^\mu(t), \end{aligned} \quad (26)$$

so that, in the linear case, the solution for the affine and spline parameters is:

$$\gamma_A^\mu(t) = (Q^T(t) G^{-1}(t) Q(t))_{AB}^{-1} Q_{Bi}^T(t) G_{ij}^{-1}(t) v_j^\mu(t), \quad (27)$$

$$\alpha_i^\mu(t) = G_{ij}^{-1}(t) (v_j^\mu(t) - Q_{jA}(t) \gamma_A^\mu(t)). \quad (28)$$

For the case of the clamped-plate GIS, the generalised ‘affine’ function is identically zero, so that:

$$\alpha_i^\mu(t) = G_{ij}^{-1}(t) v_j^\mu(t). \quad (29)$$

To summarize, for a given set of knotpoint paths  $\{\vec{x}_i(t)\}$ , we can solve for the spline parameters  $\{\alpha_i(t)\}$ , which then give a total associated energy (the generalization of equation (5)) of:

$$E[\vec{\alpha}_k(t)] = \int_0^1 dt G(\vec{x}_i(t), \vec{x}_j(t)) (\vec{\alpha}_i(t) \cdot \vec{\alpha}_j(t)). \quad (30)$$

However, this solution is only for a given fixed set of knotpoint paths – in practice, we wish to also optimise over the paths of the knotpoints with the initial and final knotpoint positions (i.e., the points  $\{\vec{x}_i(0)\}$  and  $\{\vec{x}_i(1)\}$ ) held fixed. This means that we no longer have a closed-form solution. We therefore have to numerically optimise the expression for the energy in (30), where the free variables are the knotpoint paths between their fixed endpoints. The implementation of this is described in section II-D.

Then, as in section II-B, we take the initial and final knotpoint positions to be a representation of the diffeomorphism generated by the interpolation. We denote such a geodesic interpolating spline (GIS) diffeomorphism by  $\omega(\{\vec{x}_i(0)\}, \{\vec{x}_i(1)\})$ .

As was shown by Camion and Younes [11], who considered using the GIS for inexact landmark matching, the value of the energy (30) when optimised w.r.t. the variation of knotpoint paths with fixed endpoints has the important property that it

can be considered as the square of a geodesic distance function  $d$  on the space of knotpoint configurations (that is, patterns) of a fixed cardinality  $N$ . That is:

$$d^2(e, \omega) = E_{opt}[\omega], \text{ where } \omega = \omega(\{\vec{x}_i(0)\}, \{\vec{x}_i(1)\}), \quad (31)$$

where  $e$  is the identity transformation on the space. This distance function has the important property that it is invariant under right-multiplication by an arbitrary element  $\Phi$  of the diffeomorphism group. That is:

$$d(e, \omega) = d(e \circ \Phi, \omega \circ \Phi) \quad (32)$$

where  $\circ$  is the group multiplication<sup>6</sup>, defined by:

$$\begin{aligned} \text{if } \Phi : \vec{x} \mapsto \Phi(\vec{x}), \Psi : \vec{x} \mapsto \Psi(\vec{x}), \\ \text{then } \Psi \circ \Phi : \vec{x} \mapsto \Psi(\Phi(\vec{x})). \end{aligned} \quad (33)$$

To see the importance of this invariance, consider the following. We take a set of  $M$  registered images, with a dense correspondence defined across the set. We select an image to be the spatial reference, which then defines the set of spatial warps  $\{\omega_A : A = 1, \dots, M-1\}$  between it and the rest of the images in the set. We can then compute the pairwise warp distance between image  $\mathcal{P}$  and image  $\mathcal{Q}$  thus:

$$d(\omega_{\mathcal{P}}, \omega_{\mathcal{Q}}). \quad (34)$$

Suppose we now choose a different image, image  $\mathcal{B}$  say, as the new spatial reference. This means that the warp from the new reference to image  $\mathcal{P}$  is given by  $\omega_{\mathcal{P}} \circ \omega_{\mathcal{B}}^{-1}$ . The pairwise warp distance between image  $\mathcal{Q}$  and image  $\mathcal{P}$  with respect to the new reference is

$$d(\omega_{\mathcal{P}} \circ \omega_{\mathcal{B}}^{-1}, \omega_{\mathcal{Q}} \circ \omega_{\mathcal{B}}^{-1}). \quad (35)$$

It is then clear that the pairwise warp distances between a set of registered images is only invariant to the choice of spatial reference image if the pairwise distance function is right-invariant.

The formulation of the GIS is right-invariant because it is written in terms of the Eulerian velocity field (17). Consider a path  $\Phi(t)$  in the diffeomorphism group, whose action on a point  $\vec{x}_0 \in \mathbb{R}^n$  is given by:

$$\Phi(t) : \vec{x}_0 \mapsto \Phi(t, \vec{x}_0) \doteq \vec{x}(t). \quad (36)$$

Note that if  $\vec{x}(0) \neq \vec{x}_0$ , this corresponds to a path in the diffeomorphism group that does not start at the identity  $e$ . The Eulerian velocity field associated with this flow is:

$$\vec{v}_\Phi(t, \vec{x}(t)) \doteq \frac{\partial \Phi(t, \vec{x}_0)}{\partial t} \equiv \frac{\partial \Phi(t, \Phi^{-1}(t, \vec{x}(t)))}{\partial t}. \quad (37)$$

If we consider the flow  $\Phi(t) \circ \Psi$ , where  $\Psi$  is some fixed element of the diffeomorphism group, it is trivial to show that the Eulerian velocity field defined above is invariant under this transformation – and hence we see that the energy of the flow, and hence the metric distance along the flow, is right-invariant.

We also note that this metric provides us with a principled way of defining warps that interpolate between any two given

<sup>6</sup>Note that the exact definition of the group multiplication differs between authors, and depending on the exact definition, the same function can be denoted as either left or right invariant.

warps [26]; the optimised flow in the space of diffeomorphisms gives a geodesic on the space of warps, and the geodesic distance allows us to calculate a warp on this geodesic halfway between the two initial warps.

#### D. Implementation Issues

We now consider the numerical optimisation of (30) w.r.t. the knotpoint paths  $\{\bar{x}_i(t)\}$  with fixed endpoints. To do this, we first replace each continuum path variable  $\{\bar{x}_i(t), t \in [0, 1]\}$  by a corresponding piecewise-linear approximant  $\{\bar{q}_i(\tau) : \tau = 1, \dots, T + 1\}$ , where:

$$\bar{q}_i(\tau) \doteq \bar{x}_i(t) \text{ where } t = (\tau - 1)\Delta, \forall \tau = 1, \dots, T + 1 \quad (38)$$

and  $\Delta = \frac{1}{T}$ . Since the piecewise-linear approximant path is only defined at a set of  $T + 1$  discrete time values, we also have to replace the continuous time-derivative operator  $\frac{d}{dt}$  by an appropriate finite-difference operator  $D$ . We define:

$$D\bar{q}_i \left( \tau + \frac{1}{2} \right) \doteq \frac{\bar{q}_i(\tau + 1) - \bar{q}_i(\tau)}{\Delta} \text{ for } \tau = 1, \dots, T, \quad (39)$$

If  $\bar{v}_i(t)$  is the velocity along the corresponding continuum path  $\bar{x}_i(t)$ , then we can see that:

$$D\bar{q}_i \left( \tau + \frac{1}{2} \right) \approx \bar{v}_i \left( \Delta \left( \tau - \frac{1}{2} \right) \right). \quad (40)$$

However, what we actually require is an approximant to the continuum velocity at times  $t = (\tau - 1)\Delta$ . We define:

for  $\tau = 2, \dots, T$ :

$$D\bar{q}_i(\tau) \doteq \frac{1}{2} \left( D\bar{q}_i \left( \tau - \frac{1}{2} \right) + D\bar{q}_i \left( \tau + \frac{1}{2} \right) \right) \quad (41)$$

$$\approx \bar{v}_i((\tau - 1)\Delta),$$

$$D\bar{q}_i(1) \doteq D\bar{q}_i \left( \frac{3}{2} \right), \quad D\bar{q}_i(T + 1) \doteq D\bar{q}_i \left( T + \frac{1}{2} \right). \quad (42)$$

We note that this second-order definition of the finite-difference operator  $D$  is explicitly time-symmetric. This is in contrast to the first-order approximation:

$$D^{(1)}\bar{q}_i(\tau) \doteq \frac{\bar{q}_i(\tau + 1) - \bar{q}_i(\tau)}{\Delta} \text{ for } \tau = 1, \dots, T, \quad (43)$$

$$\approx \bar{v}_i((\tau - 1)\Delta),$$

which was the form used by both Camion & Younes [11] and Joshi & Miller [12]. The consequences of this explicitly non-time-symmetric derivative are investigated in the next section.

We now have the sets of variables  $\{\bar{q}_i(\tau)\}$  and  $\{D\bar{q}_i(\tau)\}$ . Using these, we define the quantities  $Q_{iA}(\tau)$  and  $G_{ij}(\tau)$  by using (24) and (25), where we simply replace  $\bar{x}_i(t)$  by  $\bar{q}_i(\tau)$ . Similarly, the solutions for the spline parameters  $\{\alpha_i^\mu(\tau)\}$  (and the affine parameters  $\{\gamma_A^\mu(\tau)\}$  for the linear case) are given by taking either equation(s) (27) and (28) (linear case) or (29) (clamped-plate case), and replacing  $G^{-1}(t)$  by  $G^{-1}(\tau)$ ,  $Q(t)$  by  $Q(\tau)$ , and  $v_j^\mu(t)$  by  $Dq_j^\mu(\tau)$ . So, for example, in the clamped-plate case, the spline parameters are given by inverting the equation:

$$D\bar{q}_i(\tau) = \sum_{j=1}^N \bar{\alpha}_j(\tau) G(\bar{q}_j(\tau), \bar{q}_i(\tau)). \quad (44)$$

Once we have obtained the spline parameters  $\{\alpha_i^\mu(\tau)\}$ , we can then calculate the energy for this particular set of knotpoint paths:

$$E[\{\bar{q}_i\}] = \Delta \sum_{i,j=1}^N \sum_{\tau=1}^{T+1} G(\bar{q}_i(\tau), \bar{q}_j(\tau)) (\bar{\alpha}_i(\tau) \cdot \bar{\alpha}_j(\tau)), \quad (45)$$

which is the discrete-sum approximation to the time integral in (30). Finding the optimal paths of the knotpoints between their constrained endpoints is then reduced to optimising the energy (45) with respect to the free variables  $\{\bar{q}_i(\tau), \tau = 2, \dots, T\}$ . The difference between this implementation and other implementations and approaches to diffeomorphic spline construction are discussed in the next section. In the implementation used in this paper the standard MATLAB optimisation routines were used, in effect a line search, since no gradient was computed. A gradient descent algorithm for the optimisation of the paths is described in [29].

Once we have found a set of optimum knotpoint paths, we then have to calculate the movement of a sample point,  $\vec{y}$  say, under this diffeomorphism. Suppose that at timestep  $\tau$ , this point has moved to position  $\vec{p}(\tau)$ . By definition,  $\vec{p}(1) = \vec{y}$ . We compute:

$$D\vec{p}(\tau) = \vec{g}(\tau, \vec{p}(\tau)) + \sum_{i=1}^N \bar{\alpha}_i(\tau) G(\vec{p}(\tau), \bar{q}_i(\tau)), \quad (46)$$

which is the interpolant of the velocity field. For the clamped-plate case  $\vec{g}(\tau, \cdot) \equiv 0$ , whereas for the linear case:

$$g^\mu(\tau, \vec{p}(\tau)) = \gamma_{n+1}^\mu(\tau) + \sum_{\nu=1}^n \gamma_\nu^\mu(\tau) p^\nu(\tau), \quad (47)$$

where  $\{p^\nu(\tau)\}$  are the Cartesian components of  $\vec{p}(\tau)$ , and the  $\gamma_s$  are the affine parameters as given above. We then calculate the position of the point at the next timestep,  $\tau + 1$ :

$$\vec{p}(\tau + 1) \doteq \vec{p}(\tau) + \Delta D\vec{p}(\tau), \quad (48)$$

and hence the whole of the path for this sample point<sup>7</sup>.

The final question that needs to be considered is how to choose the initialisation of the knotpoint paths  $\bar{q}_i(\tau)$ . The simplest choice is a constant-velocity straight line between the two endpoints, but for the polyharmonic clamped-plate spine we have found quicker optimisation results by initializing using the geodesic paths for a single knotpoint, that is, computing the solution to (30) for the case  $N = 1$ . See the Appendix for details.

#### E. Illustrative Examples

We here provide a few examples to demonstrate some of the important points of our formulation of the GIS, and show how these properties carry over to the numerical implementation given previously.

<sup>7</sup>It should be noted that this update is not time symmetric, and that we should be using the interpolant of the velocity field at timestep  $\tau + \frac{1}{2}$ . However, to do this we would first have to estimate the position at this time. Iterative solutions are possible, which give convergence to a time-symmetric, self-consistent result. However, we have found in practice that the first-order estimate given in the text is sufficient.

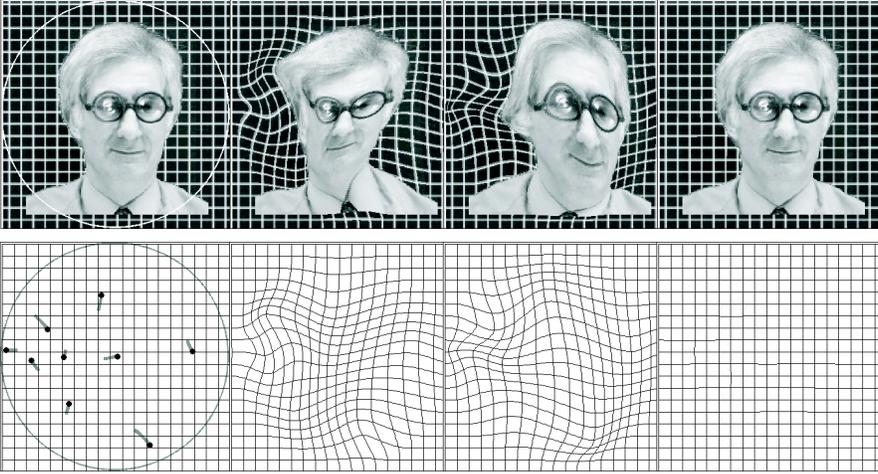


Fig. 1. *Top Row, from left:* The unwarped 190x190 pixel image with boundary circle, the forward warp applied to the image, the backward warp applied to the image, the concatenation of the forward and backward warps applied to the image, *Bottom Row:* The corresponding grids. For the purposes of illustration, these are only plotted every 10 pixels. The knotpoint initial positions (black dots) and optimum paths (gray lines) are shown on the initial grid.

1) *Inverse Consistency and Right Invariance:* We first note that the right-invariance of the continuum metric distance is *exact*, since the problem is formulated in terms of the Eulerian velocity field. In practice, the metric distance between continuously-defined warps  $\Phi$  and  $\Psi$  is *defined* to be  $d(e, \Phi \circ \Psi^{-1})$ , since this is what is calculated numerically (that is, all flows start at the identity). Hence, the numerically-calculated values of  $d(\Phi, \Psi)$  are right-invariant by definition, providing that the numerical implementation is also time-symmetric. Time symmetry is equivalent to the requirement that  $d(e, \Phi) = d(e, \Phi^{-1})$ . In terms of knotpoint endpoints  $\{\vec{x}_i(0)\}$  and  $\{\vec{x}_i(1)\}$ , this can be expressed as the requirement that the forward GIS warp  $\omega(\{\vec{x}_i(0)\}, \{\vec{x}_i(1)\})$  is the exact inverse of the corresponding backward GIS warp  $\omega(\{\vec{x}_i(1)\}, \{\vec{x}_i(0)\})$ .

To test this, we considered a  $190 \times 190$  pixel image, with unit boundary circle as shown in Fig. 1. We selected 9 knotpoint positions at random within the boundary circle and applied random displacements to these knotpoints to generate the set of endpoints  $\{\vec{x}_i(0)\}$  and  $\{\vec{x}_i(1)\}$ . In the figure, we show the backward and forward GIS warps between these sets of endpoints, calculated using  $T = 20$  discrete time-steps, using the biharmonic CPS GIS. If the time symmetry is retained numerically, then concatenating these two warps should give the identity. We compute the inverse consistency error (ICE) for each pixel in the original image by computing the straight-line distance between the original position, and the position after applying the concatenated warps. A plot of the ICE is shown in Fig. 2. The maximum ICE is 0.2675 pixels, with a mean value (taken over the interior of the circle only) of only 0.0548 pixels, and this is for an original warp with maximum displacement of 20.3 pixels, and a mean displacement of 6.5 pixels. We note that the apparent larger errors in the sub-sampled grid shown in Fig. 1 are due to the finite resolution of MATLAB figures.

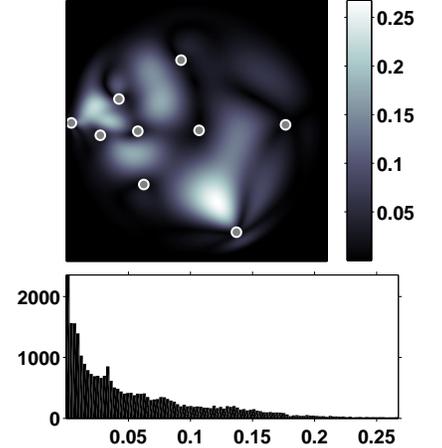


Fig. 2. The inverse consistency error (ICE) for Fig. 1, in units of pixel size. *Top:* a plot of the ICE, with the initial positions of the knotpoints (gray circles) shown for reference, *Bottom:* the histogram of the ICE for pixel centres inside the boundary circle only.

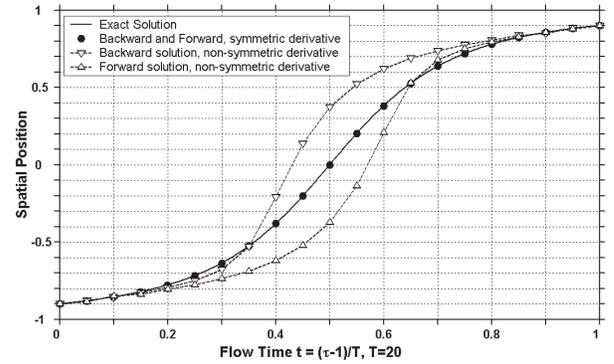


Fig. 3. Knotpoint position as a function of flowtime for the 1D biharmonic CPS GIS with one knotpoint. *Solid black line:* the exact solution (See Appendix). *Black circles:* the forward and backward solutions for the second-order difference operator. Note that these lie precisely on the exact solution given by the solid black line. *Upward pointing triangles:* Forward solution for the first-order operator. *Downward pointing triangles:* Backward solution for the first-order operator.

2) *Geodesic Paths:* From a numerical point of view, the issue of time-symmetry is also related to the optimal paths of the knotpoints. In (43) and (41) we gave the first and second order finite-difference operators that approximate to the continuum time-derivative. The first-order form is the one used by both Camion & Younes [11] and Joshi & Miller [12] in their implementations of inexact landmark matching. The second-order form, which was the form used in our implementation, is explicitly time-symmetric. To investigate the consequences of this, we will consider the simplest case of one knotpoint ( $N = 1$ ) in one dimension ( $n = 1$ ). The exact solution for both the knotpoint path and the optimal energy can be computed in closed form (see Appendix for details). Figure 3 shows the numerical results for one knotpoint moving from  $-0.9$  to  $0.9$  when using either the first-order (43) or second order (41)

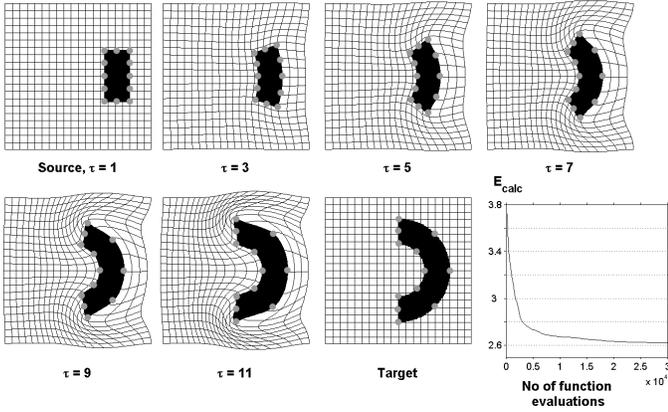


Fig. 4. An example artificial warp from a 200x200 pixel source (top left) to a target using  $T = 10$  timesteps. The boundary circle is the circumcircle of the image. The knotpoint positions are indicated by the gray circles, the grids are plotted at every 10 pixels, and all pictures are to the same scale. The plot at the bottom right shows the calculated energy during the optimisation against the cumulative number of function evaluations. 30,000 function evaluations took 950 seconds of CPU time.

approximants to the time derivative. Both the forward and backward solutions are computed. The same initial estimate of the knotpoint path (a constant velocity straight line) was used in both cases. It can be seen that the result using the second-order derivative is time symmetric (in that the forward and backward results match), and matches the exact solution. In contrast, the result obtained using just the first-order approximation to the knotpoint velocity is neither time-symmetric (that is, the forward and backward solutions do not match), nor does it match the exact result.

We can also compare the computed values of the energy with the exact result from (56) of  $E = 51.1579$ ; the first-order case gives  $E = 55.2862$ , which is 8.1% greater than the exact result, whereas the second-order case gives  $E = 50.4854$ , which is 1.3% lower. Note that we would not expect an exact agreement, since the numerical result is only a finite-sum approximation to the true energy integral, with  $T = 20$  timesteps.

To give a further example of the geodesic paths traced out in the diffeomorphism group, we considered the artificial matching problem shown in Fig. 4. The source and target are 200x200 pixel binary images, the source shape is a rectangle and the target is C-shaped object, with 10 corresponding points as shown in the figure. We used the biharmonic CPS, with the boundary circle taken to be the circumcircle of the image, so that the corners remain fixed.

As can be seen in the figure, the original rectangle evolves in a sensible manner, despite the large deformation required. We also show the calculated energy  $E_{\text{calc}}$  plotted as a function of the cumulative number of function evaluations during the optimisation. The 30,000 function evaluations took just under 16 minutes of CPU-time for an un-compiled MATLAB implementation running on a single 1.8Ghz Pentium 4 processor.

3) *Approximating an Arbitrary Warp* : So far, we have shown how to generate diffeomorphic warps given knotpoint endpoints. The actual problem we wish to solve is given a diffeomorphic warp, how do we represent that warp within

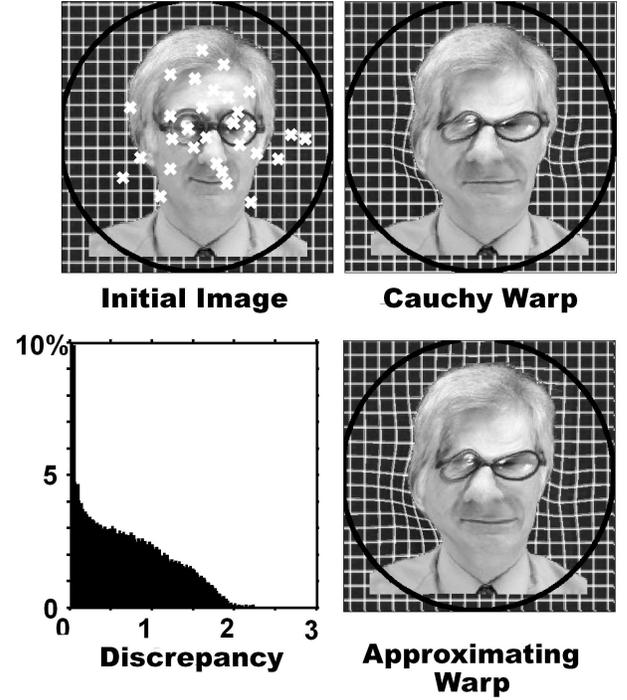


Fig. 5. *Top Row*: The unwarped image (the same image as was used in Fig. 1), with initial knotpoint positions (white crosses) and the unit circle (black line), and the Cauchy warped image. *Bottom Row*: The distribution of point-to-point Euclidean distance discrepancies between the Cauchy warp and its approximant for all pixel positions within the unit circle, in units of the unwarped pixel size, and the approximating warp generated by the set of knotpoints.

the GIS representation?

So, suppose we are given a warp that is defined on a dense set of points  $Y_0 = \{\vec{y}_0\}$  (typically the set of all pixel/voxel positions in an image), and the warped point positions are given by  $Y_1 = \{\vec{y}_1\}$ . To approximate the warp, we use the following iterative greedy algorithm:

- For the current set of knotpoint endpoints  $\{\vec{x}_i(0)\}, \{\vec{x}_i(1)\}, i = 1, \dots, N$ , compute the GIS warp  $\omega_{(N)} = \omega(\{\vec{x}_i(0)\}, \{\vec{x}_i(1)\})$ .
- Calculate the set of warp discrepancies:

$$d_{\text{disc}}(Y_0) = |\omega_{(N)}(Y_0) - Y_1|, \quad (49)$$

where  $|\cdot|$  is the Euclidean point-to-point distance.

- Find the point  $\vec{y}_0 \in Y_0$  for which  $d_{\text{disc}}$  is a maximum, and add this point as a new knotpoint, with  $\vec{x}_{N+1}(0) = \vec{y}_0, \vec{x}_{N+1}(1) = \vec{y}_1$ .
- Repeat until the desired accuracy is achieved.

It is obvious that we can always represent the desired warp exactly by choosing every point  $\vec{y}_0 \in Y_0$  as a knotpoint (any finite collection of non-coincident sample points with non-coincident displacements can be represented using some element of the diffeomorphism group [30]). However, in practice, we will have some limit to the accuracy required. For instance, for the case of warps derived from some registration algorithm, it may be appropriate to use the criterion that we have an accurate enough representation of the warp when the resampled approximately-warped image exactly matches the

resampled exactly-warped image. This will, of course, depend on the particular resampling algorithm chosen.

In Fig. 5, we give an explicit example of a warp approximant. The warp that we wish to approximate is generated by concatenating 20 random, localised Cauchy warps; each warp acts only within a defined ellipsoidal region, which is itself restricted to lie wholly within the unit circle. For the purposes of illustration, we here define the unit circle as the inscribed circle of the image. Each Cauchy warp is a parameterised, strictly diffeomorphic warp of an ellipsoidal region, which uses a 2D flat-space version of the cumulative-density-function-based methods described in [31]. Within and on the boundary of each ellipsoidal region, the Cauchy warp is a  $C^1$ -differentiable warp, where all points within the region translate in the same direction, with the magnitude of the translation controlled by the motion of the central point. The description of a single Cauchy-based warp hence requires  $2 + 2 + 2 = 6$  degrees of freedom, with  $20 \times 6 = 120$  degrees of freedom for the concatenated warps.

This warp is then applied to a  $190 \times 190$  pixel image. The warped image is *not* resampled, but plotted as a shaded surface with deformed faces/pixels. We then approximate this warp using the greedy algorithm described above. The result for the case of 30 knotpoints (equal to  $30 \times 2 \times 2 = 120$  degrees of freedom) is given. Note that the Cauchy-based representation is inherently local, and mathematically unrelated to the GIS representation. However, for the same number of degrees of freedom, we see that the approximating warp is visually extremely close to the Cauchy warp. The distribution of discrepancies (49) for pixels inside the unit circle is given in the figure, with 50% of the pixels having a discrepancy of less than 0.6 pixel units, and the maximum discrepancy being 2.25 pixels. A further example, that of approximating 3D warps for shapes is presented in a Section IV-B.

So far, we have just considered approximating single warps – in practice, we might wish to approximate a group of warps using a common spatial reference. That is to say, the set of initial knotpoints and initial knotpoint positions is common to all the warps. The greedy algorithm given above can be generalised to include this case, by computing the sum of discrepancies (49) across the group.

### III. COMPARISON WITH OTHER SPLINE-BASED METHODS.

As noted previously, there are other landmark/knotpoint based methods of constructing exact or approximate diffeomorphisms. When comparing such methods, there are two important points to consider – the mathematical statement of the algorithm, and the details of the numerical implementation of the algorithm.

In terms of splines, we have two choices – to consider the deformation as a single discrete displacement event, or as the endpoint of some smooth, continuous flow. Considering the deformation field approach, splines of the form given in (1) have the problem that they are only guaranteed one-to-one for small enough displacements, and that they are not symmetric under interchange of the landmark/knotpoint initial and final positions. Johnson and Christensen addressed this

problem in the context of registration [32] with the Consistent Landmark Thin-Plate Spline (CLTPS). They added a term to the Lagrangian that depended on the consistency of the transformation and its inverse. However, we emphasize that adding another weighted term to the Lagrangian is *not* the same mathematically as imposing an exact constraint, but is instead a trade-off between the bending energy of the deformation field and the inverse consistency criterion. More importantly, such a formulation does not give a metric on the space of transformations. However, it should be noted that their aim was to produce an algorithm suitable for registration, where CPU time is more of an issue, rather than the inverse problem, of finding knotpoints to represent an arbitrary diffeomorphism, which is the main focus of this paper.

For the flow-of-splines case, we will consider three different Lagrangian formulations. We define the common smoothing term:

$$E[\vec{v}] = \int_0^1 dt \int d\vec{x} \vec{v}(t, \vec{x}) \cdot (L^\dagger L) \vec{v}(t, \vec{x}). \quad (50)$$

The paths of the landmarks/knotpoints are denoted by  $\{\vec{x}_i(t) : i = 1, \dots, N\}$ , where  $\vec{x}_i(t) \in \mathbb{R}^n$ , with Cartesian components  $\{(\vec{x}_i(t))_\mu, \mu = 1, \dots, n\}$ .

A) Joshi & Miller [12] consider inexact landmark matching:

$$\mathcal{L}_A = E[\vec{v}] + \sum_{i=1}^N (\vec{y}_i - \vec{x}_i(1))_\mu (\Sigma^{(i)})_{\mu\nu}^{-1} (\vec{y}_i - \vec{x}_i(1))_\nu,$$

where  $\{\vec{y}_i\}$  are the targets for the landmarks, and  $\{(\Sigma^{(i)})_{\mu\nu}^{-1} : i = 1, \dots, N, \mu, \nu = 1, \dots, n\}$  the parameters that determine the degree of end-point matching,  $\Sigma^{(i)} \in \mathbb{R}^n \times \mathbb{R}^n$  being the  $n \times n$  error covariance matrix for the  $i^{\text{th}}$  landmark.

B) Camion & Younes [11] also consider inexact landmark matching:

$$\mathcal{L}_B = E[\vec{v}] + \sigma \sum_{i=1}^N \int_0^1 dt \left\| \frac{d\vec{x}_i(t)}{dt} - \vec{v}(t, \vec{x}_i(t)) \right\|^2,$$

where the fixed parameter  $\sigma$  determines the mis-match between the velocity field and the knotpoint velocities.

C) In the current paper (from (20)) finds knotpoints to represent an arbitrary diffeomorphism:

$$\mathcal{L}_C = E[\vec{v}] + \sum_{i=1}^N \int_0^1 dt \vec{\lambda}_i(t) \cdot \left( \vec{v}(t, \vec{x}_i) - \frac{d\vec{x}_i(t)}{dt} \right).$$

The first difference to note is that (A) and (B) are solved for specific values of the parameters  $\{(\Sigma^{(i)})_{\mu\nu}^{-1}\}$  and  $\sigma$  respectively, whereas the  $\{\vec{\lambda}_i(t)\}$  in (C) are Lagrange multipliers – this means that (A) and (B) are both cases of a trade-off between the energy of the velocity field  $E[\vec{v}]$  and a mis-match term, whereas (C) *exactly* imposes an *exact* matching constraint. This is equivalent to saying that (A) and (B) are both based on *smoothing* splines, whereas our implementation is based on an *interpolating* spline formalism. It could be argued that we can approach the exact matching by letting the

coefficient(s) of the matching term tend to infinity. However, from an implementation point of view, this is a poor choice – it is not clear *a priori* how large ‘large’ is in terms of the value of a parameter. Joshi & Miller [12] noted this point, saying that care had to be taken in constructing the  $\Sigma \rightarrow 0$  limit – their approach was to construct an inexact matching to within some  $\epsilon$ -ball of the targets, and then perform an interpolation to get exact matching. This, however, will only ever give an approximation to the true exact match. Given that an exact match is what is required for the case of representing diffeomorphisms, it seems sensible to include this *a priori* as in (C), rather than as the limit of some inexact matching process.

The second difference is that (A) does not lead to a metric, whereas (B) and (C) (since they are both functionals of the Eulerian velocity field) both lead to right-invariant metrics on the space of knotpoint positions. The two aims of the current paper are to construct representation(s) of arbitrary diffeomorphisms and to construct a symmetric metric on the group of Sobolev diffeomorphisms – (B) could theoretically construct such a representation, but at the cost of introducing a further arbitrary parameter (i.e., the exact value of  $\sigma$  chosen). Attempting to numerically construct the limit of  $\sigma \rightarrow \infty$  would suffer from the problems noted above.

The numerical problems with the implementations used in (A) and (B) were discussed earlier (section II-E.2), where it was shown that a non-time-symmetric implementation of a time-symmetric continuum algorithm can lead to appreciable errors, in that in general, it will neither converge to the correct exact solution, nor indeed to a time-symmetric solution.

#### IV. REPRESENTING DIFFEOMORPHISMS

##### A. Representing 2D Diffeomorphisms

When considering warps of 2D biological images, it is obviously important that the generated warps are not only diffeomorphic, but also make some sense biologically. To investigate this, we considered a set of 2D T1-weighted MR axial slices of brains, where the slices chosen show the lateral ventricles. For each image, the positions of the lateral ventricles and the skull were annotated by a radiologist using a set of 163 points. We took a subset of 66 of these points to be the positions of our knots (see Fig. 6). Given a pair of images, the knotpoint positions on the images give us the initial and final positions of our knotpoint paths. We then calculated the geodesic interpolating spline warp corresponding to these positions using the 2D biharmonic clamped-plate spline as Green’s function. The bounding circle for the spline is shown in Fig. 7.

Note that we did not affinely align the knotpoints before calculating the warp; hence the algorithm has to deal with a non-trivial pseudo-affine part. Including the affine part would have made the task easier. Example results are shown in Fig. 8.

It can be seen that the warps are indeed diffeomorphic, and appear to be very smooth (see, for example, the detail shown in Fig. 9), and biologically plausible, which would not be true had a simpler scheme been used. The warped images

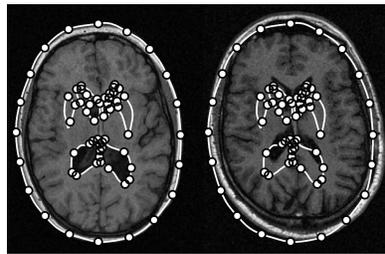


Fig. 6. *Left:* Annotation (white line) and knotpoints (white circles) on the original brain slice.

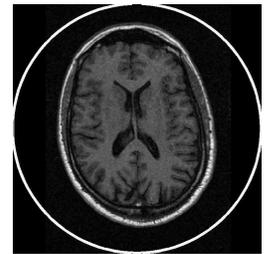


Fig. 7. An example 2D brain slice with the bounding circle.

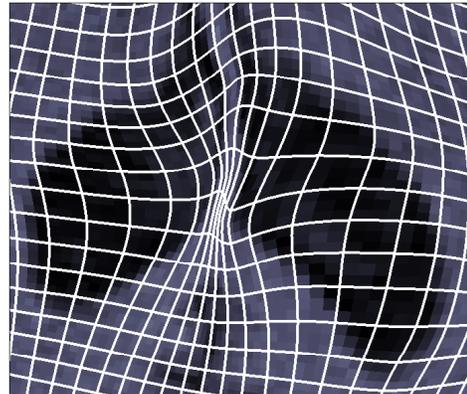


Fig. 9. Close-up of a warped source image from Fig. 8 (last column). The deformed pixel grid is shown (white lines), plotted at every 3 pixels.

are not resampled – the images are instead plotted as faceted grayscale surfaces, so that the size and position of each warped pixel is retained. The pairs of images were chosen to illustrate cases where the required deformation was considerable, both in terms of the change in shape of the ventricles and skull, and in terms of the difference in scale and orientation of the slice as a whole. The resultant warped images do indeed appear to be biologically plausible, despite the relatively low dimensionality of the representation used – structures other than the labelled ones have been brought into approximate alignment. This suggests that a dense correspondence (for instance, one given by a non-rigid registration using maximisation of mutual information) could also be represented by these warps without an inordinate increase in the dimensionality of the representation (i.e., the number of knotpoints required). It was shown earlier that the GIS representation can be used to represent mathematically unrelated diffeomorphic warps to reasonable accuracy given the same dimensionality of representation. Testing this assertion for warps from dense non-rigid registrations is the subject of future work.

##### B. Representing 3D Diffeomorphisms

We have shown that this geodesic interpolating spline basis can generate biologically plausible warps in 2D; we now proceed to show that we can also do the same in 3D. We take as our dataset a set of segmented hippocampi. Each example consists of a triangulated surface with 268 vertices, where the vertices for each example have been manipulated to give

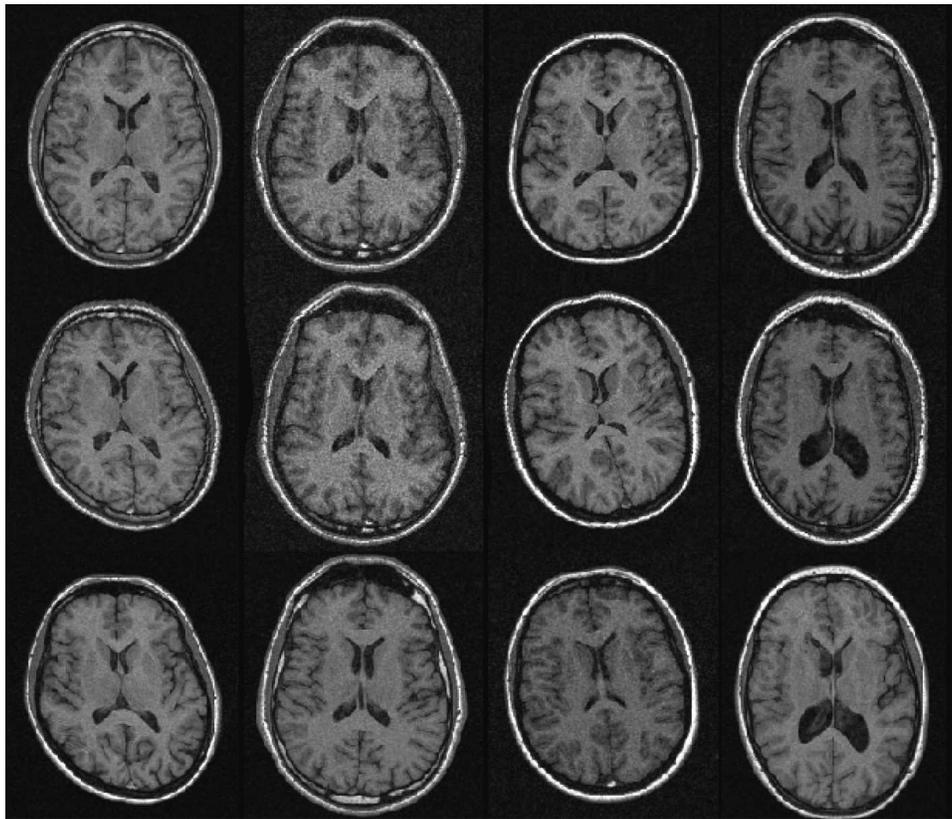


Fig. 8. Four examples of warp interpolation using the clamped-plate spline. Pixel intensity is unchanged, but note that the image structures are approximately aligned. *Top row*: Unwarped source image, *Middle row*: Warped source image, *Bottom row*: Target image.

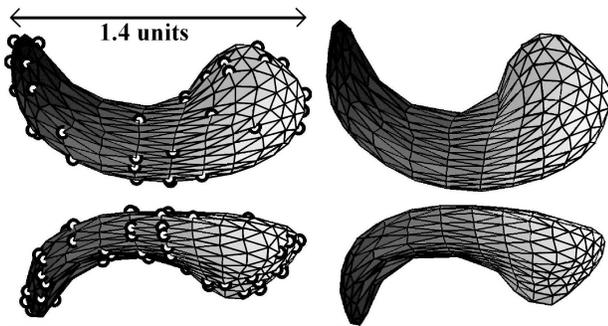


Fig. 10. Two views of a pair of source (left) and target (right) hippocampi with knotpoints (circles), all to the same scale. The correspondence between the shapes is indicated by the shading.

the optimised correspondence [31]. Examples are shown in Fig. 10.

Pairs of hippocampi were chosen at random, and the shapes aligned using generalised Procrustes analysis. We used the triharmonic clamped-plate spline (see Table I) as our GIS basis, with each hippocampus being scaled to fit within the unit sphere. The warp between source and target was approximated using the greedy algorithm described previously.

Fig. 11 shows the distribution of the discrepancies between the aligned source and target, and the final warped source and target, for a set of 70 knotpoints. It can be seen that the distribution of discrepancies as a whole has been shifted towards smaller values. In Fig. 12, we show the maximum,

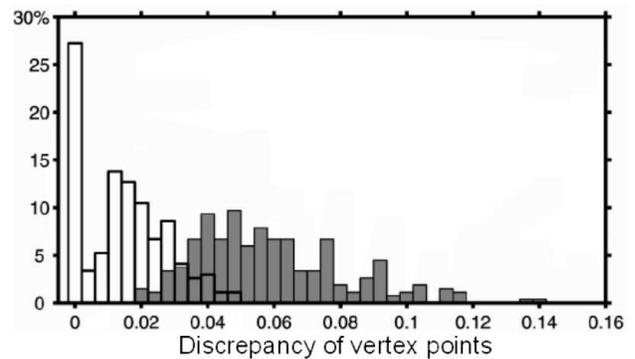


Fig. 11. Distribution of vertex discrepancy between source and target (grey bars), and warped source and target (white bars). Units as in Fig. 10.

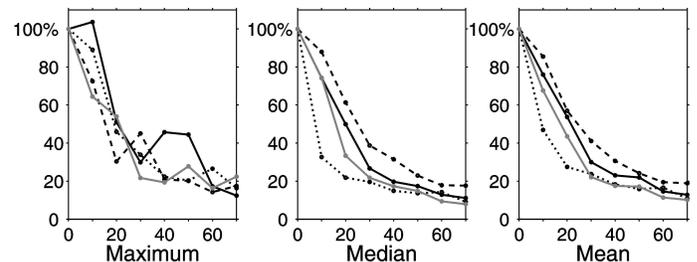


Fig. 12. The maximum, median and mean discrepancies (in units of their respective initial values, so that all graphs start at 100% by definition), for non-knot points only, as a function of the number of knots. Data is shown from 4 randomly selected pairs of hippocampi.

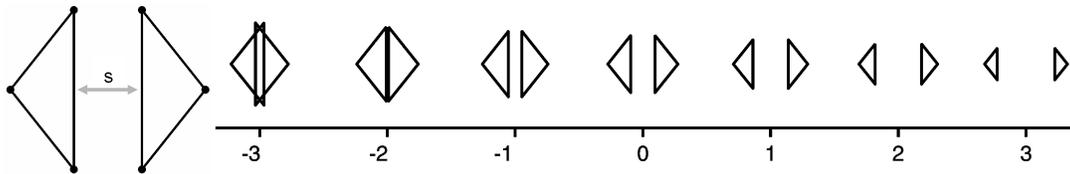


Fig. 13. *Left*: The distance  $s$  between the discrete structures varies. *Right*: Example shapes generated from the Statistical Shape Model of the discrete structures dataset for varying model parameter (Mahalanobis distance). The change in scale as the model parameter varies is caused by the fact that the points have been Procrustes aligned.

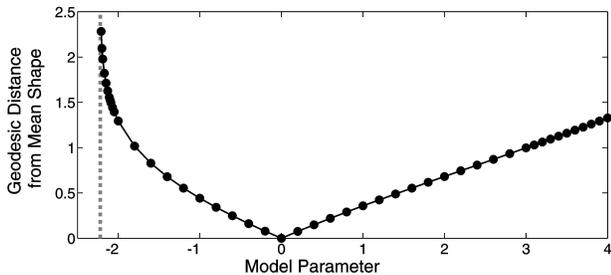


Fig. 14. Geodesic distance versus model parameter for the discrete structure dataset of Figure 13.

median and mean discrepancies for non-knot points only as a function of the number of knotpoints for 4 random pairs of hippocampi. Note that the nature of our greedy algorithm for selecting knotpoints means that the maximum discrepancy is not guaranteed to decrease monotonically. However, all three graphs show that the algorithm quickly reaches a reasonable representation of the required warp, for a number of knotpoints that is approximately 25% of the number of vertices. The total warp approximation (that is, up to 70 knotpoints) for a pair of hippocampi typically took a few hours to compute, using a non-compiled MATLAB implementation.

## V. MODELLING DISCRETE STRUCTURES

### A. A Simple Example

There are numerous examples in biological and medical images of cases where a pair of structures remain discrete, although the spacing between them varies considerably across a population, for example, the lateral ventricles in the brain. We constructed a simple dataset to investigate the problems associated with modelling such a variation; the basic idea is shown on the left of Fig. 13. Elements of the dataset comprise 6 points defining two triangles, with the separation  $s$  between the two triangles constrained to remain positive. A training set of 100 examples was generated, with the separation being chosen at random. A standard method for analyzing such a dataset is to build a statistical shape model (SSM) [16]; the training set was Procrustes aligned, and a linear SSM constructed.

The model correctly displayed only one mode of variation; example shapes generated by the model are shown on the right of Fig. 13. As was expected, this simple linear model can generate illegal shapes – when the model parameter went below a threshold of approximately  $-2.22$  standard deviations, the triangles intersected, so that the structures were no longer

discrete. This was not seen in the training set, and therefore should not be allowed. We then calculated the CPS geodesic warps generated by taking the 6 points that define each pair of triangles as knotpoints, and warping between examples generated by the model for varying model parameter and the model mean shape. The CPS was a suitable choice of Green’s function because the triangles are discrete objects. The variation of the geodesic distance from the mean shape, plotted against the model parameter (which is equivalent to the Mahalanobis distance, a distance defined from a Euclidean metric on the space of point positions) is shown in Fig. 14.

Positive values of the model parameter, which correspond to increasing separation of the triangles, show a relationship between model parameter and geodesic distance that is very close to linear, but for negative values of the parameter (decreasing separation), the geodesic distance diverges at the precise value of the parameter that corresponds to zero separation. The limiting case of zero separation corresponds to coincident displacements of originally non-coincident points, which, by definition, cannot be represented by a diffeomorphic transformation. Hence, as we approach this non-diffeomorphic limit, the distance of the required diffeomorphic transformation from the identity element of the diffeomorphism group diverges. This theoretical divergence is reflected in the numerical implementation by the asymptotic behaviour of the computed distance as shown in the Figure.

The geodesic distance therefore allows us to differentiate between physical and non-physical variations in a way that naïve linear models cannot.

### B. Using the Geodesic Distance to Classify Variations

We now consider the role of the geodesic distance in classifying legal and illegal variations in real biological data. We take as our dataset the annotated outlines of the anterior lateral ventricles as used in section IV-A in the axial brain slices. Each example consists of 40 knotpoints (see Fig. 15). The set of training examples was Procrustes aligned and then scaled to fit inside the unit circle. A linear SSM was built from this training set in the usual way. We then used this SSM to generate random example shapes. These examples were classified as legal if the outlines of the ventricles did not intersect either themselves or each other, and illegal otherwise. The training set of shapes were all defined to be legal.

We then computed the GIS warps with  $T = 10$  timesteps, using the biharmonic CPS basis, between the classified set of shapes and the mean shape from the model. Computation times were typically between twenty minutes and an hour

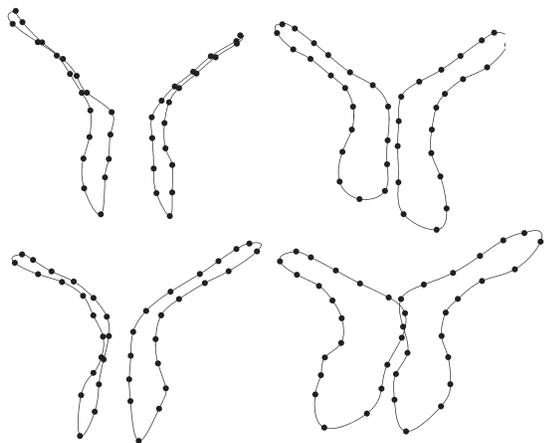


Fig. 15. *Top*: Two examples from the training set. *Bottom*: Legal (*left*) and illegal (*right*) examples generated by the SSM. Knotpoints are indicated by black circles; lines are for the purposes of illustration.

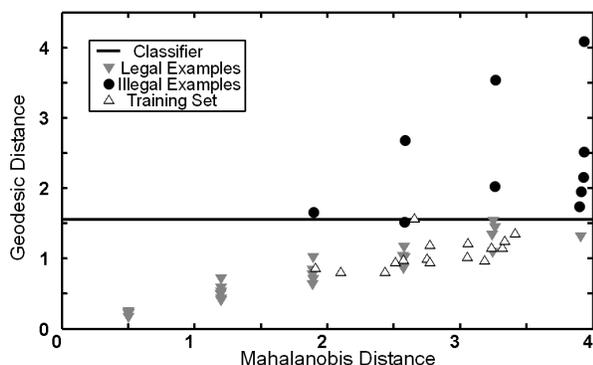


Fig. 16. Mahalanobis versus geodesic distances from the mean shape for *black circles*: illegal shapes generated by the SSM, *grey triangles*: legal shapes generated by the SSM, *white triangles*: the training set.

per example, depending on the degree of deformation. The geodesic distance from the mean is compared with the Mahalanobis distance from the mean in Fig. 16. It is immediately obvious that we cannot separate the legal and illegal shapes by using the Mahalanobis distance. However, using the geodesic distance, it is possible to construct a simple classifier (see Fig. 16) that separates the two groups, with only one example shape being misclassified (the black circle just below the line, labelled (ii) in Fig. 17). If we look at the misclassified point in detail (Fig. 17), we see that the training shape (i) has actually been mis-annotated – the outline contains a self-intersection – and is hence illegal. If we adjust the classifier in light of this, then shapes (i) and (ii) will be correctly classified, but shape (iii) will be incorrectly classified as illegal, although it can be seen that it is very close to being illegal, since the two ventricular outlines nearly meet.

It should be noted that, unlike the simpler example considered earlier, these knotpoint motions can actually be represented by a diffeomorphic mapping – all that ‘intersects’ here are the smooth outlines drawn between the knotpoints, which were for the purposes of illustration, and to provide a simple visual method of classification. What the geodesic distance is actually detecting here is a difference in the ‘topology’

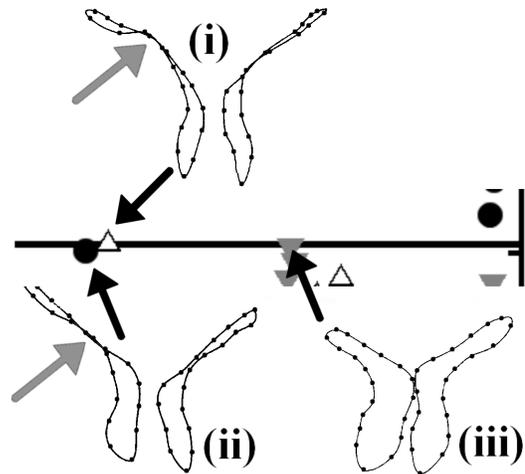


Fig. 17. Detail of Fig. 16. (i) A training shape that appears misclassified, but actually contains an annotation error and self-intersection (grey arrow), (ii) an illegal shape generated by the model, with self-intersection (grey arrow), (iii) a legal shape generated by the model, correctly classified.

of the knotpoint configurations between the illegal and legal examples.

The correspondences that we have used in this example are a subset of the correspondences that we would expect to be generated by a successful non-rigid registration of the images. Increasing the density of points in the training shapes would have left the Mahalanobis distance essentially unchanged, but the result for the GIS warp would have improved, giving a greater separation between the two sets of shapes. This is because, in the limit where the lines become infinitely densely sampled, what was formerly only a ‘topological’ change would become the difference between a diffeomorphic and a non-diffeomorphic mapping, as was the case in the first simple example. Then, as previously, the geodesic distance for the illegal shapes would diverge, approaching infinity as the sampling density increased. We can now extend this result to the case of modelling the deformation fields for the non-rigid registration; a linear model of such deformation fields would suffer the same problem as the linear SSM, where now the overlapping structures would correspond to a folding of the warp. The GIS cannot, by definition, generate such a folding since it is guaranteed to be diffeomorphic.

## VI. CONCLUSIONS

The modelling of the dense deformation fields that are produced by non-rigid registration algorithms is an important topic that has not yet been studied sufficiently. This paper contributes to the topic by describing a suitable representation of the deformation fields that is diffeomorphic and has an inherent non-Euclidean metric. The method is based on Geodesic Interpolating Splines, which are defined by a set of knotpoints, with the spline basis of the representation defined by a Green’s function and boundary conditions. These can be altered to suit the task in hand. We give examples of some suitable Green’s functions, and focus particularly on the polyharmonic clamped-plate spline, which has the boundary conditions that

it is zero on and outside the unit ball, and vanishes smoothly at the boundary. We contend that these are suitable boundary conditions for many biological tasks because many biological structures are discrete objects, and the background to those objects should not be warped.

We choose to use Geodesic Interpolating Splines because they generate general diffeomorphic (i.e., smooth and one-to-one) functions, as well as having an inherent metric. The assumption of diffeomorphic mappings is important because, as mentioned in the introduction, conventional approaches to modelling assume a dense correspondence between the set of examples in the model training set. The assumption of a *meaningful* dense correspondence between every example in a group of images will not always be true. But the target we are working towards is a groupwise registration strategy that is capable of taking a group of images and separating them into subset(s) where the meaningful dense correspondence assumption holds across each subset, and a set of outliers, where the assumption fails. The failure of this assumption will be indicated either by atypical values of the warp parameters, or by atypical values of the pixel-value deformation fields, or a combination of the two. However, we cannot quantitatively define what we mean by atypical unless we have a way of quantifying what we mean by typical, which means some sort of modelling strategy has to be applied to the mutually corresponding subset. Part of that strategy will be a definition of distance on the space of smooth, diffeomorphic warps.

We have demonstrated how to represent diffeomorphic warps in both two and three dimensions, and have shown that we need a relatively low-dimensional representation in order to capture most of the variation between pairs of images – using just 70 knotpoints chosen from a set of 268 vertices enabled us to represent the warp between 2 hippocampi in three dimensions to reasonable accuracy. This suggests that the representation can accurately represent real biological variations with a reasonably small number of knotpoints. One benefit of our approach is that using the greedy algorithm described in the paper, the knotpoints are not positioned equally densely across the image plane, as in conventional grid-based approaches, but where they are needed, so that the effective local dimensionality of the warps can be adjusted.

Finally, we have compared the metric inherent to the representation with the Mahalanobis distance – a Euclidean metric on the space of parameters that is used by conventional linear modelling strategies. Our experiments have shown that the geodesic distance can reliably separate out legal and illegal variations from a set of two dimensional ventricle shapes generated by a shape model. By comparison, the Mahalanobis distance is completely unsuited to this non-linear task.

In our future work, we plan to consider using the method to fully analyse groups of non-rigid registrations of a variety of medical images, including the construction of non-linear models on the relevant subspace of the diffeomorphism group.

## APPENDIX

### EXACT SOLUTIONS FOR A SINGLE KNOTPOINT

For the polyharmonic clamped-plate GIS with one knotpoint ( $N = 1$ ), (29) and (30) reduce to:

$$E[\vec{x}(t)] = \int_0^1 dt \frac{\left(\frac{d\vec{x}}{dt}\right)^2}{G(\vec{x}(t), \vec{x}(t))}, \quad (51)$$

where  $\vec{x}(t) \in \mathbb{D}^n$  is the path of the knotpoint, with the constraint that the endpoints  $\vec{x}(0)$  and  $\vec{x}(1)$  are held fixed. Exact solutions to this energy-minimisation can then be computed, for both varying dimensionality  $n$ , and varying order of clamped-plate spline. We give here as examples the biharmonic case in one and two dimensions.

For the biharmonic CPS in one dimension, the Green's function is [25]:

$$G(x, y) = \frac{1}{3}(1 - xy)^3 - (1 - xy)(x - y)^2 + \frac{2}{3}|x - y|^3. \quad (52)$$

The unit ball in one dimension  $\mathbb{D}^1$  is just the line segment  $|x| \leq 1$ . That is, the points  $|x| = 1$  are fixed under the action of the flow. Substituting into (51):

$$E[x(t)] = \int_0^1 dt \frac{3}{(1 - x^2(t))^3} \left(\frac{dx(t)}{dt}\right)^2. \quad (53)$$

The Euler-Lagrange equations are:

$$\begin{aligned} \frac{\partial E}{\partial x(t)} - \frac{d}{dt} \left( \frac{\partial E}{\partial \dot{x}(t)} \right) &= 0, \quad \dot{x}(t) \doteq \frac{dx(t)}{dt}, \\ \Rightarrow \frac{d^2 x(t)}{dt^2} &= -\frac{3x(t)}{(1 - x^2(t))} \left(\frac{dx(t)}{dt}\right)^2, \end{aligned} \quad (54)$$

which has the solution:

$$x(t) = \sin(\tan^{-1}(At + B)), \quad (55)$$

$$B = \tan(\sin^{-1}(x(0))),$$

$$A = \tan(\sin^{-1}(x(1))) - B,$$

$$\text{with: } E_{\text{opt}} = 3A^2. \quad (56)$$

For the case of the biharmonic clamped-plate GIS in two dimensions ( $n = 2$ ) we use the expression for the Green's function given in Table I. The energy to be optimised is then given by:

$$E[\vec{x}(t)] = \int_0^1 dt \left(\frac{d\vec{x}}{dt}\right)^2 (1 - \vec{x}^2)^{-2}, \quad (57)$$

with, as before, the constraint that the endpoints  $\vec{x}(0)$  and  $\vec{x}(1)$  are held fixed. This is the energy for paths in the Poincaré disc model of the hyperbolic plane [33]; the geodesics are arcs of circles of varying centre and radius that intersect the unit circle at right angles. In polar coordinates, we take the point to start at  $(r_0, \theta_0)$  at  $t = 0$ , and finish at  $(r_1, \theta_1)$  at  $t = 1$ . For the case  $\theta_0 \neq \theta_1$ :

$$r(\theta) = k \cos(\theta - \psi) - \sqrt{k^2 \cos^2(\theta - \psi) - 1}, \quad (58)$$

$$t(\theta) = A + B \tanh^{-1} \left[ \left( \sqrt{k^2 - 1} \cot(\theta - \psi) \right)^{\pm 1} \right], \quad (59)$$

where the polar angle  $\theta$  (rather than  $t$ ) is taken as the dependent variable. The sign in the exponent is chosen such that  $t$  is real. The parameters  $k, \psi, A$  and  $B$  are determined by fitting to the endpoints. That is:

$$R_0 \doteq \frac{1+r_0^2}{2r_0}, R_1 \doteq \frac{1+r_1^2}{2r_1}, \quad (60)$$

$$\tan(\psi) = \frac{R_0 \cos \theta_1 - R_1 \cos \theta_0}{R_1 \sin \theta_0 - R_0 \sin \theta_1} \quad (61)$$

$$k = \frac{R_0}{\cos(\theta_0 - \psi)} = \frac{R_1}{\cos(\theta_1 - \psi)}, \quad (62)$$

$$B = \left[ \tanh^{-1} \left( \frac{R_1 \sqrt{k^2 - 1}}{k} \right)^{\pm 1} - \tanh^{-1} \left( \frac{R_0 \sqrt{k^2 - 1}}{k} \right)^{\pm 1} \right]^{-1}, \quad (63)$$

$$A = -B \tanh^{-1} \left( \frac{R_0 \sqrt{k^2 - 1}}{k} \right)^{\pm 1}. \quad (64)$$

The remaining special case is when  $\theta_0 = \theta_1$ . Then:

$$\theta(t) = \theta_0 = \theta_1, \quad (65)$$

$$r(t) = \tanh(at + b), \quad (66)$$

$$\text{where: } b = \tanh^{-1}(r_0), a = \tanh^{-1}(r_1) - b. \quad (67)$$

#### ACKNOWLEDGEMENTS

Our thanks to G. Gerig, M. Styner and co-workers from the University of North Carolina, Chapel Hill, for the hippocampus dataset; the dataset formed part of a schizophrenia study funded by the Stanley Foundation. Our thanks also to Rhodri Davies for providing us with his optimised correspondences for this dataset.

This paper is an extension of the paper ‘‘Constructing Diffeomorphic Representations of Non-Rigid Registrations of Medical Images’’ [34] published in IPMI 2003.

#### REFERENCES

- [1] M. Bro-Nielsen and C. Gramkow, ‘‘Fast fluid registration of medical images,’’ in *Proceedings of Visualization in Biomedical Computing (VBC)*, 1996, pp. 267–276.
- [2] C. Chefd’Hotel, G. Hermosillo, and O. Faugeras, ‘‘Variational methods for multimodal image matching,’’ *International Journal of Computer Vision*, vol. 50, no. 3, pp. 329–343, 2002.
- [3] J. Gee, M. Reivich, and R. Bajcsy, ‘‘Elastically deforming 3D atlas to match anatomical brain images,’’ *Journal of Computer Assisted Tomography*, vol. 17, no. 2, pp. 225–236, 1993.
- [4] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes, ‘‘Non-rigid registration using free-form deformations: Application to breast MR images,’’ *IEEE Transactions on Medical Imaging*, vol. 18, no. 8, pp. 712–721, 1999.
- [5] S. Marsland and C. J. Twining, ‘‘Constructing data-driven optimal representations for iterative pairwise non-rigid registration,’’ in *Proceedings of the Second International Workshop on Biomedical Image Registration (WBIR)*, ser. Lecture Notes in Computer Science, J. C. Gee, J. A. Maintz, and M. W. Vannier, Eds., vol. 2717. Springer-Verlag, 2003, pp. 50–60.
- [6] S. Marsland, C. J. Twining, and C. J. Taylor, ‘‘Groupwise non-rigid registration using polyharmonic clamped-plate splines,’’ in *Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, ser. Lecture Notes in Computer Science, R. E. Ellis and T. M. Peters, Eds., vol. 2879. Springer-Verlag, 2003, pp. 771–779.
- [7] A. Guimond, J. Meunier, and J.-P. Thirion, ‘‘Average brain models: A convergence study,’’ INRIA, Sophia Antipolis, Tech. Rep. RR-3731, 1999.
- [8] L. LeBriquer and J. Gee, ‘‘Design of a statistical model of brain shape,’’ in *Proceedings of Information Processing in Medical Imaging (IPMI)*, ser. Lecture Notes in Computer Science, J. Duncan and G. Gindi, Eds., vol. 1230. Springer-Verlag, 1997, pp. 477–482.
- [9] D. Rueckert, A. F. Frangi, and J. A. Schnabel, ‘‘Automatic construction of 3D statistical deformation models using non-rigid registration,’’ in *Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, ser. Lecture Notes in Computer Science, vol. 2208. Springer-Verlag, 2001, pp. 77–84.
- [10] Y. Choi and S. Lee, ‘‘Injectivity conditions of 2D and 3D uniform cubic B spline functions,’’ *Graphical Models*, vol. 62, no. 6, pp. 411–427, 2000.
- [11] V. Camion and L. Younes, ‘‘Geodesic interpolating splines,’’ in *Proceedings of EMMCVPR’01*, ser. Lecture Notes in Computer Science, M. Figueiredo, J. Zerubia, and A. K. Jain, Eds., vol. 2134. Springer-Verlag, 2001, pp. 513–527.
- [12] S. C. Joshi and M. M. Miller, ‘‘Landmark matching via large deformation diffeomorphisms,’’ *IEEE Transactions on Image Processing*, vol. 9, no. 8, pp. 1357–1370, 2000.
- [13] S. Marsland and C. J. Twining, ‘‘Clamped-plate splines and the optimal flow of bounded diffeomorphisms,’’ in *Statistics of Large Datasets, Proceedings of Leeds Annual Statistical Research Workshop (LASR)*, 2002, pp. 91–95.
- [14] R. Schmid, ‘‘Infinite dimensional Lie groups with applications to mathematical physics,’’ *Journal of Geometry and Symmetry in Physics*, vol. 1, pp. 1–67, 2004.
- [15] A. Kriegl and P. W. Michor, *The convenient setting of global analysis*, ser. Mathematical Surveys and Monographs, 53. American Mathematical Society, Providence, RI, 1997.
- [16] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, ‘‘Active shape models – their training and application,’’ *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.
- [17] F. L. Bookstein, ‘‘Principal Warps: Thin-Plate Splines and the Decomposition of Deformations,’’ *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 6, pp. 567–585, 1989.
- [18] L. H. Staib and J. S. Duncan, ‘‘Boundary finding with parametrically deformable models,’’ *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, pp. 1061–1075, 1992.
- [19] C. Brechbühler, G. Gerig, and O. Kübler, ‘‘Parametrization of closed surfaces for 3D shape description,’’ *Computer Vision and Image Understanding*, vol. 61, no. 2, pp. 154–170, 1995.
- [20] S. M. Pizer, D. S. Fritsch, P. Yushkevich, V. Johnson, and E. Chaney, ‘‘Segmentation, registration, and measurement of shape variation via image object shape,’’ *IEEE Transactions on Medical Imaging*, vol. 18, no. 10, pp. 851–865, 1999.
- [21] M. Styner and G. Gerig, ‘‘Hybrid boundary-medial shape description for biologically variable shapes,’’ in *Proceedings of IEEE Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA) 2000*, June 2000, pp. 235–242.
- [22] R. H. Davies, C. J. Twining, T. F. Cootes, J. C. Waterton, and C. J. Taylor, ‘‘A minimum description length approach to statistical shape modeling,’’ *IEEE Transactions on Medical Imaging*, vol. 21, no. 5, pp. 525–537, 2002.
- [23] G. Wahba, *Spline Models for observational data*, ser. CBMS-NSF regional conference series in applied mathematics. Society for industrial and applied mathematics (SIAM), 1992, vol. 59.
- [24] J. Duchon, ‘‘Interpolation des fonctions de deux variables suivant le principe de la flexion des plaques minces,’’ *Revue Française d’Automatique, Informatique, Recherche Opérationnelle (RAIRO) Analyse Numerique*, vol. 10, pp. 5–12, 1976.
- [25] T. Boggio, ‘‘Sulle funzioni di green d’ordine m,’’ *Rendiconti - Circolo Matematico di Palermo*, vol. 20, pp. 97–135, 1905.
- [26] C. J. Twining, S. Marsland, and C. J. Taylor, ‘‘Measuring geodesic distances on the space of bounded diffeomorphisms,’’ in *Proceedings of the British Machine Vision Conference (BMVC)*, vol. 2. BMVA Press, 2002, pp. 847–856.
- [27] P. Dupuis, U. Grenander, and M. I. Miller, ‘‘Variational problems on flows of diffeomorphisms for image matching,’’ *Quarterly of Applied Mathematics*, vol. 56, no. 3, pp. 587–600, 1998.
- [28] A. Trounev, ‘‘Diffeomorphisms groups and pattern matching in image analysis,’’ *International Journal of Computer Vision*, vol. 28, no. 3, pp. 213–221, 1998.
- [29] A. Mills, ‘‘Optimizing the paths for geodesic interpolating splines,’’ Master’s thesis, University of Manchester, 2003.

- [30] P. W. Michor and C. Vizman, “n-transitivity of certain diffeomorphism groups,” *Acta Mathematica Universitatis Comeniana*, vol. 63, no. 2, pp. 221–225, 1994.
- [31] R. Davies, C. J. Twining, T. F. Cootes, J. C. Waterton, and C. J. Taylor, “3D statistical shape models using direct optimisation of description length,” in *Proceedings of the 7<sup>th</sup> European Conference on Computer Vision (ECCV)*, ser. Lecture Notes in Computer Science, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds., vol. 2352. Springer-Verlag, 2002, pp. 3–20.
- [32] H. Johnson and G. E. Christensen, “Consistent landmark and intensity-based image registration,” *IEEE Transactions on Medical Imaging*, vol. 21, no. 5, pp. 450–461, 2002.
- [33] J. W. Anderson, *Hyperbolic Geometry*, ser. Springer Undergraduate Mathematics Series. New York: Springer-Verlag, 1999, ch. 4.1, pp. 95–104.
- [34] C. J. Twining and S. Marsland, “Constructing diffeomorphic representations of non-rigid registrations of medical images,” in *Proceedings of Information Processing in Medical Imaging (IPMI)*, ser. Lecture Notes in Computer Science. Springer-Verlag, 2003.