IJCV manuscript No.
(will be inserted by the editor)

Sparse Adaptive Parameterization of Variability in Image Ensembles

Stanley Durrleman · **Stéphanie Allassonnière** · **Sarang Joshi**

Received: date / Accepted: date

Abstract This paper introduces a new parameterization of diffeomorphic deformations for the characterization of the variability in image ensembles. Dense diffeomorphic deformations are built by interpolating the motion of a finite set of control points that forms a Hamiltonian flow of self-interacting particles. The proposed approach estimates a template image representative of a given image set, an optimal set of control points that focuses on the most variable parts of the image, and template-to-image registrations that quantify the variability within the image set. The method automatically selects the most relevant control points for the characterization of the image variability and estimates their optimal positions in the template domain. The optimization in position is done during the estimation of the deformations without adding any computational cost at each step of the gradient descent. The selection of the control points is done by adding a L^1 prior to the objective function, which is optimized using the FISTA algorithm.

Keywords atlas construction · image variability · diffeomorphisms · sparsity · control points · FISTA

1 Introduction

1.1 The need to adapt generic parameterization of image variability

The statistical analysis of a set of images plays an important role in several fields, such as Computer Vision, Pattern Recognition, or Computational Anatomy. The goal of this analysis is to find the invariants across a given set of images and to characterize how these common features vary in appearance within the group. This mean and variance analysis is useful in several ways: (1) to measure how likely a new image may be considered as another observation of the same group of images; (2) to cluster the set of images into consistent subgroups; and (3) to understand what distinguishes two different sets of images. For instance, this can be used for classification purposes, to understand the variability of an anatomical structure observed in a normal population or to characterize what distinguishes normal versus pathological anatomical structures.

One way to approach this problem is to extract a set of features from the images and to perform statistics on feature vectors of small dimension. Usually, the definition of the features is specific to each problem and supposes that one already knows what features are interesting for a given application. By contrast, the generic Grenander's pattern theory for modeling objects (Grenander, 1994; Trouvé, 1998; Dupuis et al, 1998; Miller and Younes, 2001), which was later extended for population analysis (Lorenzen et al, 2005; Allassonnière et al, 2007; Durrleman et al, 2009), establishes a diffeomorphic map between each image in the data set and a common "template" image that is representative of the image ensemble. Both the template image and the deformations, together called an "atlas," need to be estimated. The former captures the invariants across the image ensemble and the latter describes how these invariants appear in

Stanley Durrleman · Sarang Joshi
Scientific Computing and Imaging (SCI) Institute, 72 S. Central Drive, Salt Lake City, UT 84112, USA
E-mail: stanley@sci.utah.edu (Stanley Durrleman) Phone: +1 801 585 1867, Fax: +1 801 585 6513

Stéphanie Allassonnière
Centre de Mathématiques Appliquées (CMAP), UMR CNRS 7641, Ecole Polytechnique
Route de Saclay, 91128 Palaiseau, France

individual observations. The atlas characterizes the variability of the set of images, without any prior knowledge of what is variable in the observations.

The distribution of the template-to-subjects deformations, seen as instances of random variables in the set of all possible deformations, gives a characterization of the variability of a given image ensemble. Intrinsic statistics on such deformations may be computed using the parameterization of the deformations in the LDDMM setting (Lei et al, 2007; Singh et al, 2010) or the displacement field of the grid of voxels using a log-Euclidean technique (Arsigny et al, 2006). In both cases, the mathematical objects used for statistics are of infinite dimension in theory, and of the order of the size of the images in practice. This very high dimensionality is an asset of the approach, in the sense that the model is flexible enough to capture a very wide range of possible variations across an image set. At the same time, this dimension is problematic, not only because the number of images are usually much smaller than the dimension of the descriptors, which is critical from a statistical point of view, but also because this very high dimension does not reflect the true number of degrees of freedom that are needed to describe the observed variability of the image set. For instance, if the images differ from rigid-body, affine transformations or such constrained deformations, then a small number of parameters is sufficient to describe this variability. Therefore, it would be beneficial to estimate which small subgroup of deformations the template-to-subjects deformations belong to, given an image ensemble. Then, statistics can be derived using the small dimensional parameterization of the deformations in this subgroup. In this paper, we address this issue by introducing a data-driven basis selection technique. We see the infinite parameterization of the diffeomorphisms as a dictionary of basis elements and we propose to find a small finite-dimensional subset of these basis elements which enables the description of the variability of a given image ensemble. The weights of the decomposition of the deformations on this basis will be used as a small-dimensional descriptor of the variability. The dimension of this descriptor will give an estimate of the 'true' number of degrees of freedom that underlie the variability of the image set.

A typical way of finding optimal basis is to estimate the deformations parameterized by a very large number of variables, and then apply a generic dimension-reduction technique to the set of descriptors. Techniques like Principal Component Analysis (PCA), Independent Component Analysis (ICA) or matching pursuit can be called upon. The problem is that such extrinsic dimension-reduction techniques try to minimize the approximation error in the parameter space, and not in the image space. These techniques do not make use of the input images to find the best reduction of dimension. By contrast, we propose here to estimate the deformations as the same time as their optimal parameterization. In this case, the reduction of the dimension of the parameterization can be balanced by adjusting the other parameters, so that the loss in the description of the variability is minimal. The whole optimization is driving by the minimization of a single criterion, which accounts for the balance between sparsity and matching accuracy. We will use a L^1 prior in this criterion to enforce the decomposition of the deformations to be as sparse as possible (i.e. with the most possible zero weights). The set of basis elements with non-zero weights defines the subgroup of deformations that is the more adapted for the description of the variability of a given image set.

1.2 Finite-dimensional parameterization of atlases

More precisely, we follow the statistical approach initiated in Allassonnière et al (2007), which considers that every observed image derives from an unknown template image plus identically distributed random white noise:

$$I_i = I_0 \circ \phi_i^{-1} + \varepsilon_i, \tag{1}$$

where I_i for $i = 1, \dots, N$ denote the original images, I_0 the template image, ϕ_i the N template-to-subject deformations, and ε_i the N images of white noise. $I_0 \circ \phi_i^{-1}$ is the usual action of the diffeomorphic deformation on images (seen as measures on the ambient space). The invariants within the image set are captured in the template image I_0. The variability is encoded in the deformations ϕ_i. The atlas consists in both the template (the photometric variable) and the set of deformations (the geometric variables). Both need to be estimated and are intrinsically of infinite dimension. To estimate such variables, we first introduce a generic finite-dimensional parameterization and then sparsity priors to adapt this parameterization to a particular set of observations.

The construction of the deformed image $I_0 \circ \phi_i^{-1}$ requires a computation of the gray level of the template image at potentially any location in the template image domain: the template image should have an infinite resolution. To introduce a finite-dimensional parameterization of the template image, we follow the approach proposed in Allassonnière et al (2007) and build a *continuous* template image by interpolating photometric weights located at a *discrete* set of photometric control points. The photometric control points define a basis for the parameterization of the template image.

In contrast to Allassonnière et al (2007), who used small deformations, we will use here large diffeomorphic deformations in the LDDMM setting for the deformations ϕ_i (Trouvé, 1998; Miller et al, 2002). In this framework, a large group of diffeomorphisms is seen as 'Riemannian manifold' of infinite dimension. The equivalent of the logarithm of a diffeomorphism is a continuous squared integrable velocity field.

The conjugate variable of the velocity field is the momenta, which is used to define intrinsic tangent-space statistics on deformations (Vaillant et al, 2004; Lei et al, 2007; Durrleman et al, 2009; Singh et al, 2010; Durrleman et al, 2011a). For image matching, the momenta is encoded by an image of infinite dimension, or numerically of the size of the input images. However, it has been shown in Durrleman et al (2009) that such continuous momenta maps can be efficiently approximated by a finite set of well-chosen Dirac delta momenta, where momenta stand for vectors attached to control points that are called geometric control points in this context. Therefore, we introduce a finite-dimensional parameterization of the momenta based on a finite set of vectors attached to control points, in the spirit of Joshi and Miller (2000).

The set of geometric control points, which may be located anywhere in the image domain, defines a potentially infinite-dimensional basis of the parameterization of the deformations. The vectors attached to them define the weights of the decomposition of a given deformations onto this basis. Defining an adapted basis for the description of the variability means finding the optimal positions of a finite number of control points: both the position and the number of the geometric control points should be optimized altogether, given an image ensemble. Indeed, an optimal set of geometric control points are unlikely to be equally distributed in the image domain; instead, they should be located at the most variable parts of the image. The optimal positions of the points are characteristic of the image ensemble and therefore shared by all the template-to-subjects deformations. The momentum vectors attached to these control points parameterize each of these deformations and are therefore specific to each observation. We will see that the optimization of the position of the control points could be done along with the estimation of the momentum vectors without introducing any additional cost in the computation of the gradient. The optimization of the number of control points will be done by introducing a L^1 sparsity priors on the set of momentum vectors, which will force the atlas estimation to use as few non-zero momentum vectors as possible by selecting the geometric control points that are most relevant for the description of the variability.

The proposed method follows the approach initiated in Durrleman et al (2011b), which introduced the control-point parameterization of large diffeomorphic deformations for image atlasing. However, in Durrleman et al (2011b), the whole time-varying parameterization of the template-to-subject deformations was optimized. Consequently, the deformations were geodesic, and therefore characterized by their initial momenta, only once the optimization algorithm has converged to a local minimum. The method was therefore more sensitive to numerical errors. By contrast, we propose here to take advantage of the geodesic shooting equations in order to guarantee that, at each step of the optimization process, the computed deformations are geodesic. We also propose here to use a convex L^1 penalty term instead of the non-convex log-L^1, which, in addition, needed an extra parameter. We also change the usual L^2 model of image for the parameteric image model introduced in Allassonnière et al (2007).

2 Formulation of parametric atlases

2.1 Parametric template models

The template I_0 should be a continuous image, which allows us to compute the gray level at any position x of the image domain, so that one can build the image $I_0(\phi^{-1}(x))$ and sample it at the pixels grid.

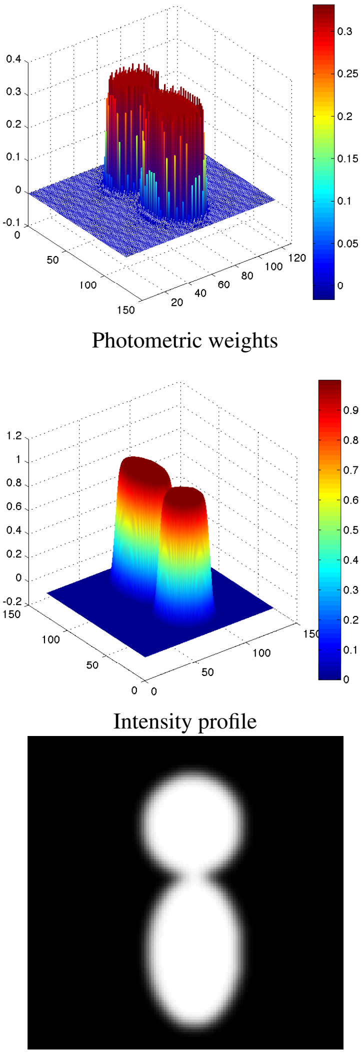
The parametric template model, which has been introduced in Allassonnière et al (2007), parameterize such continuous images with a discrete set of weights located at some "well-chosen" control points. Let's c_k^{ph} be a sparse set of N_{ph} control points in the image domain, called *photometric control points* in this context, and $K^{\mathrm{ph}}(x,y)$ an interpolating kernel. We define the parametric template at any location x in the image domain as the interpolation of photometric weights w_k located at the photometric control points (see Fig. 1):

$$I_0(x) = \sum_{k=1}^{n_p} K^{\mathrm{ph}}(x, c_k^{\mathrm{ph}}) w_k. \tag{2}$$

This template model has the advantage of a discrete parameterization, which can be easily handled. In particular, it facilitates the easy computation of a deformed template and its gradient without relying on finite-difference schemes.

Remark 1 (Comparison with templates defined as images) *Assume that one puts one photometric control point at each node of the pixels grid. Then, the template is represented by an image, of the same size of the observations, whose gray levels are given by the weights w_k. Moreover, assume that the kernel is the triangle function: $K^{ph}(x,y) = \frac{1}{\Delta}min(\Delta + (x-y), \Delta - (x-y))^+$, where Δ is the size of a pixel, then* (2) *is exactly the linear interpolation of the gray levels w_k at the sub-pixel level. This is the typical template model given as a digital image, which is linearly interpolated to compute gray values at any arbitrary locations in the image domain. This is one of the most popular template models in the literature. We proved here that this template model is the limit of our parametric template model.*

However, this limit suffers from two main limitations. First, it is encoded by an array whose size equals the number of the pixels in the image domain. This representation may *be highly redundant, especially for binary or highly contrasted images. In these cases, the information is localized in small areas of the image domain and large background areas are encoded in endless sequences of '0' in the template image. Second, the template image is sensitive to the sampling of the observations. In particular, it is difficult to use if observations have different sampling and different sizes. The parametric model addresses these two limitations.*



Photometric weights

Intensity profile

Image

Fig. 1 Parametric template model. A template image is defined by a set of signed photometric weights (left). An interpolating kernel builds a continuous intensity profile, at any resolution (middle), which is displayed as an image (right).

2.2 Parametric diffeomorphic deformation of images

For the deformations, our approach relies on the large diffeomorphic deformations introduced in Trouvé (1998); Dupuis et al (1998); Miller et al (2002). Diffeomorphisms are constructed by integrating infinitesimal splines transforms over time, which play the role of an instantaneous velocity field. Given a time-varying vector field $v_t(x)$ over the time interval $[0,1]$, one integrates the differential equation $\dot{\phi}_t(x) = v_t(\phi_t(x))$, with initial condition $\phi_0(x) = x$. The end-point of the path ϕ_1 is the diffeomorphism of interest. Under the conditions detailed in Beg et al (2005) and satisfied here, the resulting $(\phi_t)_{t\in[0,1]}$ is a flow of diffeomorphisms (for each time $t \in [0,1]$, ϕ_t is a diffeomorphic deformation). In particular, the vector fields v_t are supposed to belong to a Reproducible Kernel Hilbert Space (RKHS), namely the set of L^2 vector fields convolved with a regularizing kernel K, which plays the role of a low-pass filter and therefore controls the spatial smoothness of the vector fields.

In our approach, we assume a discrete parameterization of the driving velocity field v_t via a convolution operator:

$$v_t(x) = \sum_{k=1}^{n_g} K^{\mathrm{g}}(x, c_k^{\mathrm{g}}(t))\alpha_k(t), \tag{3}$$

where for each time t, $c_i^{\mathrm{g}}(t)$ denotes a set of n_g geometric control points, $\alpha_i(t)$ a set of n_g momentum vectors attached to them. K^{g} is a fixed positive definite kernel that defines a RKHS. In this work, we will use the Gaussian kernel $K(x,y) = \exp(-\|x-y\|^2/\sigma_{\mathrm{g}}^2)\mathrm{Id}$ (Id stands for the identity matrix) among other possible choices. It has been shown in Durrleman et al (2009) that such vector fields can approximate any vector field in the RKHS defined by the kernel K^{g}.

We denote $\mathbf{S}(t) = \{c_k^{\mathrm{g}}(t), \alpha_k(t)\}_{k=1,\ldots,n_g}$ (a $2dn_g$ vector, where $d = 2$ in 2D and 3 in 3D) the state of the system at time t. Knowing the state of the system at any time $t \in [0,1]$ defines a flow of diffeomorphisms. Indeed, any point x_0 in the ambient space follows the path $x(t) = \phi_t(x_0)$ which satisfies the ODE:

$$\begin{cases} \dot{x}(t) = v_t(x(t)) = \sum_{k=1}^{n_g} K^{\mathrm{g}}(x(t), c_k^{\mathrm{g}}(t))\alpha_k(t) \\ x(0) = x_0 \end{cases}. \tag{4}$$

The path $x(t)$ depends therefore only on the initial condition x_0 and the state of the driving system $\mathbf{S}(t)$. The final position $x(1)$ is by definition $\phi_1(x_0)$.

One could use the time-varying state of the system $\mathbf{S}(t)$ as the parameterization of the diffeomophism ϕ_1 as proposed in Durrleman et al (2011b). However, in this work, we will take advantage of the fact that among all paths $t \to \mathbf{S}(t)$ connecting ϕ_0 to ϕ_1 there is one which satisfies a minimum energy principle: the 'geodesic paths.' Indeed, the kernel K^{g} induces a metric on the space of velocity fields, and therefore on the space of diffeomorphisms (Miller et al, 2006). The distance between the diffeomorphism of interest ϕ_1 and the identity map ϕ_0 is the total kinetic energy needed to reach the former from the latter: $\int_0^1 \|v_t\|^2 dt = \int_0^1 \left\langle K^{-1}v_t, v_t \right\rangle dt$, which for the particular form of v_t given in (3) reduces to:

$$\int_0^1 \sum_{i=1}^{n_g} \sum_{j=1}^{n_g} \alpha_i(t)^t K^{\mathrm{g}}(c_i(t), c_j(t)) \alpha_j(t) dt. \tag{5}$$

This kinetic energy depends only on the time-varying state of the system $\mathbf{S}(t)$. Following mechanical principles, it has been shown in Miller et al (2006) that the extremal path connecting ϕ_0 and ϕ_1 is such that the state of the system $\mathbf{S}(t)$ satisfies the following set of ODEs:

$$\begin{cases} \dfrac{dc_k^{\mathrm{g}}(t)}{dt} = \displaystyle\sum_{l=1}^{n_g} K^{\mathrm{g}}(c_l^{\mathrm{g}}(t), c_k^{\mathrm{g}}(t)) \alpha_k(t) \\ \dfrac{d\alpha_k(t)}{dt} = -\displaystyle\sum_{l=1}^{n_g} \alpha_k(t)^t \alpha_l(t) \nabla_1 K^{\mathrm{g}}(c_l^{\mathrm{g}}(t), c_k^{\mathrm{g}}(t)) \end{cases}, \tag{6}$$

given initial conditions $\alpha_k(0) = \alpha_{0,k}$ and $c_k^{\mathrm{g}}(0) = c_{0,k}^{\mathrm{g}}$. Denoting $\mathbf{S}_0 = \{\alpha_{0,k}, c_{0,k}^{\mathrm{g}}\}_k$ the initial state of the system, (6) can be re-written in short as:

$$\begin{cases} \dot{\mathbf{S}}(t) = F(\mathbf{S}(t)) \\ \mathbf{S}(0) = \mathbf{S}_0 \end{cases}. \tag{7}$$

These differential equations can be interpreted as the motion of n_g self-interacting particles without external forces. The interaction between particles is given by the kernel K^{g}. The first equation in (6) gives the speed of the control points; the second one, its acceleration. Note that the first equation is consistent with the definition of the velocity field in (3), since it reads $\frac{dc_k^{\mathrm{g}}(t)}{dt} = v_t(c_k^{\mathrm{g}}(t))$.

These equations show that the whole flow of diffeomorphisms is entirely determined by the initial state of the system $\mathbf{S}_0$. Indeed, given $\mathbf{S}_0$, the integration of (6) gives the state of the system at any later time t: $\mathbf{S}(t)$ (the motion of the control points and the momentum vector over time). Then, the integration of (4) gives the motion of any point x_0 in the ambient space according to the flow of diffeomorphisms ϕ_t. The generation of diffeomorphisms ϕ can be fully controlled by the finite-dimensional vector $\mathbf{S}_0$. From a Riemannian perspective, $\mathbf{S}_0$ plays the role of the logarithm map (or tangent-space representation) of the diffeomorphism that it parameterizes. Such parameterizations are of paramount importance to define tangent-space statistics on diffeomorphisms (Vaillant et al, 2004; Pennec et al, 2006).

Accordingly, the inverse map ϕ_1 is also fully determined by $\mathbf{S}_0$. Given a point position y in the image domain Ω, the position given by the inverse flow $\phi_1^{-1}(y)$ can be computed by integrating the following ODE backward in time (where the velocity field has been reversed):

$$\frac{dy(t)}{dt} = -v_t(y(t)), \qquad y(1) = y. \tag{8}$$

Then, the final value at time $t = 0$ gives the mapped position: $y(0) = \phi_1^{-1}(y)$.

Let Y be an image of vectors, which gives the position of every voxel in the image domain. In the continuous setting, we have $Y(y) = y$ for any $y \in \Omega$, where Y is seen as a squared integrable map in $L^2(\Omega, \mathbb{R}^d)$. The domain Ω is deformed by the inverse diffeomorphism ϕ_1^{-1}: the inverse flow can be computed by integrating the following ODE:

$$\begin{cases} \dfrac{dY(t)}{dt} = G(Y(t), \mathbf{S}(t)) \\ Y(1) = \mathrm{Id}_{L^2} \end{cases}, \tag{9}$$

where

$$\begin{aligned} G(Y(t), \mathbf{S}(t)) &= -v_t(Y(t)) \\ &= -\sum_{k=1}^{n_g} K^{\mathrm{g}}(Y(t)(.), c_k^{\mathrm{g}}(t)) \alpha_k(t) \end{aligned} \tag{10}$$

maps an image of vectors in $L^2(\Omega, \mathbb{R}^3)$ and a $2dn_g$-dimensional vector to an image of vectors in $L^2(\Omega, \mathbb{R}^3)$[1].

Once integrated backward from time $t = 1$ to $t = 0$, the final image of vectors $Y(0)$ maps the domain Ω to $\phi_1^{-1}(\Omega)$. As a consequence, the deformation of the template image I_0 can be written as:

$$I_0(\phi_1^{-1}(y)) = I_0(Y(0)(y)). \tag{11}$$

Eventually, one can easily verify that the geodesic paths of the state of the system are energy conservative: for any time t, $\|v_t\|_V^2 = \|v_0\|_V^2$. Therefore, the total kinetic energy of a given path is given as:

$$L(\mathbf{S}_0) = \sum_{k=1}^{n_g} \sum_{l=1}^{n_g} \alpha_{0,k}^t K^{\mathrm{g}}(c_{0,k}^{\mathrm{g}}, c_{0,l}^{\mathrm{g}}) \alpha_{0,l}. \tag{12}$$

This is a function of only the initial state of the system, which will be used as a measure of regularity of the deformations in the objective function.

Remark 2 (Linearization of the deformation model) *This large deformation setting is built by the combination of infinitesimal transforms. The ODEs are integrated from $t = 0$ to $t = T$, where we fix T to 1 for large deformations. Now, we linearize the model in time, assuming T tends to zero. Then, at the first order, the flow of diffeomorphisms reduces to a single transform: $\phi(x) = x + v_0(x)$. v_0 plays the role of a displacement field, which has the form $v_0(x) = \sum_{k=1}^{n_g} K^g(x, c_k^g)\alpha_k$. This is typically a displacement field constructed by interpolation of radial basis functions: K^g plays the role of the radial basis function and α is the vectorial weight of the interpolation. Here, we build a diffeomorphism*

[1] If the image domain Ω is discretized into a regular lattice of N_{im} voxels, then $Y(t)$ could be seen as a dN_{im}-dimensional vectors of the voxels positions that are mapped back via the inverse deformation.

by composing several of such displacement fields, while ensuring via the ODE integration the diffeomorphic property of the final deformation. Similar constructions can be found in Rueckert et al (2006); Vercauteren et al (2009). In addition, our approach based on mechanical principles enables us to define a metric on the space of diffeomorphisms and then geodesic paths and tangent-space representation of the diffeomorphisms.

Remark 3 (Comparison with usual LDDMM methods) *This model of large diffeomorphic deformations has been used in the context of registration in mostly two occasions: for the registration of images (Miller et al, 2002; Beg et al, 2005) and the registration of point sets (Joshi and Miller, 2000; Miller et al, 2002; Vaillant and Glaunès, 2005; Glaunès et al, 2008). In both cases, one looks for the geodesic flows, which minimize a trade-off between a data fidelity term (sum of squared differences between gray values or between point locations) and a regularity term (the kinetic energy). It has been shown in Miller et al (2002, 2006) that the minimizing velocity fields have a particular structure in each case. For image registration, the minimizing velocity fields are parameterized by a time-varying continuous map of momenta $\alpha(t,x)$, such that the momentum vector $\alpha(t,x)\nabla_x I(t)$ is always pointing in the direction of the gradient of the moving template image: $v_t(x) = \int K(x,y)\alpha(t,y)\nabla_y I_t dy$, where $I(t) = I_0 \circ \phi_t^{-1}$. For point sets registration, it has been shown that the support of a momenta map reduces to the discrete set of points to be matched. The momenta map is then expressed as vectors attached to each of the points in the set, in a similar expression as in* (3) *where the control points are equal to the shape points.*

The proposed framework unifies both approaches, by using a set of control points that become independent of the objects to be matched. Both the position of the control points and the momentum vectors attached to them need to be estimated. The control points are not necessarily at the shape points. Momentum vectors are not constrained to be parallel to the image gradient. We will see that the optimization in the position of control points moves them toward the contours of the template image and are concentrated in the most variable areas of the image.

3 Atlas estimation via gradient descent

3.1 Objective function for atlas estimation

Our purpose is to estimate the whole atlas from a set of images: the template image and the template-to-subjects deformations. The parameters of the atlas to be optimized are: the photometric weights for the template image, a set of control points in the template image domain, and a collection of momentum vectors that, with the control points, parameterizes the template-to-subjects deformations. It is important to notice that the template image and the set of control points are shared by all the subjects: they parameterize the invariants in the population. By contrast, the momentum vectors are specific to each subject. They parameterize the variance of the image set around the mean template image. Note also that we do not optimize with respect to the position of the photometric control points, which is considered a fixed hyper-parameter.

Formally, the parameters are one vector of photometric weights $\mathbf{w}$, one vector of the position of the control points $\mathbf{c}_0^{\mathrm{g}}$, and N vectors of initial momenta $\alpha_{0,i}$, where N denotes the number of images in the data set. We denote $\mathbf{S}_i(t)$ the state of the system of the ith subject, which defines the ith template-to-subject deformations according to (7):

$$\begin{cases} \dot{\mathbf{S}}_i(t) = F(\mathbf{S}_i(t)) \\ \mathbf{S}_i(0) = (\mathbf{c}_0^{\mathrm{g}}, \alpha_{0,i}) \end{cases}, \tag{13}$$

where F is defined by (6).

The inverse deformations map $y \in \Omega$ to $\phi_{i,1}^{-1}(y)$, which equals $Y_i(0)(y)$, where the flow of images of vectors $Y_i(t)$ ($\in L^2(\Omega,\mathbb{R}^3)$) satisfy the ODEs given in (10):

$$\begin{cases} \dot{Y}_i(t) = G(Y_i(t), \mathbf{S}_i(t)) \\ Y_i(1) = \mathrm{Id}_{L^2} \end{cases}. \tag{14}$$

A Maximum A Posteriori estimation of these parameters in the same setting as in Allassonnière et al (2007) leads to the minimization of the following objective function:

$$E(\mathbf{w}, \mathbf{c}_0^{\mathrm{g}}, \{\alpha_{0,i}\}_{i=1,\dots,N}) = \frac{1}{2\sigma^2}\sum_{i=1}^{N} \|I_0 \circ Y_i(0) - I_i\|_{L^2}^2 + \|v_{0,i}\|_V^2, \tag{15}$$

where σ^2 is a scalar trade-off between fidelity-to-data and regularity.

The ith data term $A_i = \int_\Omega |I_0(Y_i(0)(x)) - I_i(x)|^2 dx$ depends on I_0 and $Y_i(0)$. The template image depends on the photometric weights $\mathbf{w}$ via (2). The image of vectors $Y_i(0)$ depends on positions of the set of particles $\mathbf{S}_i(t)$ via the ODE (14), which in turn depends on the initial state of the system $\mathbf{S}_{0,i}$ via the ODE (13). Therefore, the image of vectors $Y_i(0)$ depends on the initial state of the system $\mathbf{S}_{0,i}$ via the forward integration followed by a backward integration of ODEs. This set of ODE propagates back and forth the information from the template space at $t = 0$ to the subject space at $t = 1$.

The regularity term $\|v_{0,i}\|_V^2$ is the kinetic energy of the system of particles, which is conserved over time. Given (12), this term equals:

$$\begin{aligned} \|v_{0,i}\|_V^2 &= \sum_{p=1}^{n_g}\sum_{q=1}^{n_g} \alpha_{0,i,p}^t K^{\mathrm{g}}(c_{0,p}^{\mathrm{g}}, c_{0,q}^{\mathrm{g}})\alpha_{0,i,q} \\ &= \alpha_{0,i}^t \mathbf{K}^{\mathrm{g}}(\mathbf{c}_0^{\mathrm{g}}, \mathbf{c}_0^{\mathrm{g}})\alpha_{0,i}, \end{aligned} \tag{16}$$

where we denote $\mathbf{K}^{\mathrm{g}}(\mathbf{c}_0^{\mathrm{g}}, \mathbf{c}_0^{\mathrm{g}})$ the dn_g-by-dn_g symmetric block matrix, whose $(p,q)th$ block is given by $K^{\mathrm{g}}(c_{0,p}^{\mathrm{g}}, c_{0,q}^{\mathrm{g}})$. This term depends only on the initial state of the system of particles $\mathbf{S}_{0,i}$. Therefore, we write it as: $L_i(\mathbf{S}_{0,i}) = L_i(\mathbf{c}_0^{\mathrm{g}}, \alpha_{0,i})$.

Eventually, the criterion to be optimized writes:

$$E(\mathbf{w}, \mathbf{c}_0^{\mathrm{g}}, \{\alpha_{0,i}\}_{i=1,\dots,N}) = \sum_{i=1}^{N} A_i(\mathbf{w}, Y_i(0)) + L_i(\mathbf{c}_0^{\mathrm{g}}, \alpha_{0,i}). \quad (17)$$

Let us denote $E_i = A_i(\mathbf{w}, Y_i(0)) + L_i(\mathbf{c}_0^{\mathrm{g}}, \alpha_{0,i})$ the contribution of the *ith* subject to the objective function. Then, the gradient of E is given by:

$$\nabla_{\mathbf{w}} E = \sum_{i=1}^{N} \nabla_{\mathbf{w}} E_i, \quad \nabla_{\mathbf{c}_0^{\mathrm{g}}} E = \sum_{i=1}^{N} \nabla_{\mathbf{c}_0^{\mathrm{g}}} E_i, \quad \nabla_{\alpha_{0,i}} E = \nabla_{\alpha_{0,i}} E_i. \quad (18)$$

The gradient with respect to the photometric weights and control points positions involves a sum over the subjects: it reflects the fact that these variables are shared by all the subjects in the population in contrast to the momentum vectors that are specific to each subject. These equations also show that the gradient of the criterion can be computed by computing the gradient of each term of the sum in parallel. This is possible since there is no coupling between the parameters of different subjects.

The gradient with respect to the deformation parameters can be computed using the chain rule:

$$\begin{cases} \nabla_{\mathbf{c}_0^{\mathrm{g}}} E_i = \dfrac{1}{2\sigma^2} (d_{\mathbf{c}_0^{\mathrm{g}}} Y_i(0))^{\dagger} \nabla_{Y_i(0)} A_i + \nabla_{\mathbf{c}_0^{\mathrm{g}}} L_i \\ \nabla_{\alpha_{0,i}} E_i = \dfrac{1}{2\sigma^2} (d_{\alpha_{0,i}} Y_i(0))^{\dagger} \nabla_{Y_i(0)} A_i + \nabla_{\alpha_{0,i}} L_i \end{cases}.$$

We notice that these gradients are driven by the term: $\nabla_{Y_i(0)} A_i$, which, according to the definition of A_i, is the image of vectors whose value at position x is given by:

$$\nabla_{Y_i(0)} A_i(x) = 2\left(I_0(Y_i(0)(x)) - I_i\right) \nabla_{Y_i(0)(x)} I_0. \quad (19)$$

This is the usual image force, which drives the deformation of most intensity-based registration methods. This term contains all the interesting information about the data. The two Jacobian matrices $d_{\mathbf{c}_0^{\mathrm{g}}} Y_i(0)$ and $d_{\alpha_{0,i}} Y_i(0)$ show how to combine this image force with the underlying deformation model. In the next section, we will show a very efficient way to compute this gradient, which does not require the explicit computation of these Jacobian matrices. Instead, we will use a set of linearized ODEs to transport the image force back and forth between the template image domain and the subject image domain.

Remark 4 (Gradient in the small deformation setting) *The Jacobian matrices involved in the gradient are easy to compute in the small deformation setting, in which the integration of ODE are done using a unique step-size. In the setting of Remark 2, the flow of diffeomorphisms is reduced to the transform:* $\phi(x) = x + v(x) = x + \sum_{k=1}^{n_g} K(x, c_k^g)\alpha_k$*, which is parameterized by the fixed momenta* $(\mathbf{c}^g, \alpha)$*. The inverse deformation is approximated by* $\phi^{-1}(y_k) = y_k - v(y_k)$*. One term of the objective function is then reduced to (omitting the subject's index i):*

$$E(\mathbf{c}^g, \alpha) = \frac{1}{2\sigma^2} \left\| I_0 \circ \phi^{-1} - I \right\|^2 + \|v\|_V^2, \quad (20)$$

whose gradient can be computed straightforwardly as:

$$\nabla_{\alpha_k} E = -\frac{1}{\sigma^2} \sum_{l=1}^{N_{im}} K^g(c_k^g, y_l) \left(I_0(y_l - v(y_l)) - I(y_l)\right) \nabla_{y_l - v(y_l)} I_0 + 2 \sum_{p=1}^{n_g} K^g(c_k^g, c_p^g)\alpha_p,$$

$$\nabla_{c_k^g} E = \frac{1}{\sigma^2} \sum_{l=1}^{N_{im}} \frac{2}{\sigma_g^2} K^g(c_k^g, y_l) \times \left(I_0(y_l - v(y_l)) - I(y_l)\right) \left(\nabla_{y_l - v(y_l)} I_0\right)^t \alpha_k (c_k^g - y_l) - 2 \sum_{p=1}^{n_g} \frac{2}{\sigma_g^2} K^g(c_k^g, c_p^g) \alpha_k^t \alpha_p (c_k^g - c_p^g),$$

where for clarity purposes, we supposed the kernel of the form $K^g(x,y) = \exp\left(-\|x-y\|^2 / \sigma_g^2\right) I$*.*

The first equation consists of two terms: the first one is the convolution at the control points of the usual image force with the smoothing kernel, which tends to decrease the image discrepancy; the second one is a regularizer of the estimated momenta, which can be seen as a low-pass filter on the momenta. The second equation is the update rule for the control points positions. The first term shows that they are attracted by the voxels where the gradient of the image is large (i.e. the contours), provided that the momenta α_i *pushes in the 'right' direction, that of the image force (making the dot product negative). The second term is a repulsion term which moves away two control points which carry momenta pointing in the same direction (if* $\alpha_k^t \alpha_p > 0$*, then the opposite direction of the gradient points in the same direction as* $c_k^g - c_p^g$*, which tends to move* c_k^g *away from* c_p^g*). The effect of the term is to limit the redundancy of the parameterization at the scale of the kernel* σ_g*.*

The reader could verify that this gradient is precisely the linearization of the gradient, which will be given in the next section. The linearization is at order 0 for the first equation and at order 1 for the second one (the zeroth order vanishing).

3.2 Differentiation with respect to the position of the control points and momentum vectors

In this section, we show how to efficiently differentiate the gradient with respect to the deformation parameters. The following generic proposition shows that the gradient with respect with the deformation parameters can be computed by transporting the image force back and forth between the template and the subjects' image domain. Note that in the sequel we omit the subject's index i for clarity purposes.

Proposition 1 *Let us denote $\mathbf{S}_0 = (c_0^g, \alpha_0)$ be the $2dn_g$ parameters of a generic criterion E of the form:*

$$E(\mathbf{S}_0) = A(Y(0)) + L(\mathbf{S}_0),$$

where:

$$\begin{array}{ll} \dot{\mathbf{S}}(t) = F(\mathbf{S}(t)) & \mathbf{S}(0) = \mathbf{S}_0 \\ \dot{Y}(t) = G(Y(t), \mathbf{S}(t)) & Y(1) = Id_{L^2} \end{array}. \tag{21}$$

$Y(t)$ is an image of vectors in $L^2(\Omega, \mathbb{R}^d)$ for all t, A a differentiable map from $L^2(\Omega, \mathbb{R}^d)$ to $\mathbb{R}$ and F, G two differentiable maps.

Then, the gradient of E is given by:

$$\nabla_{\mathbf{S}_0} E = \xi(0) + \nabla_{\mathbf{S}_0} L, \tag{22}$$

where two auxiliary variables $\xi(t)$ (a vector of size $2dn_g$) and $\eta(t)$ (an image of vectors) satisfy the following linear ODEs:

$$\begin{cases} \dot{\eta}(t) = -(\partial_1 G(Y(t), \mathbf{S}(t)))^{\dagger} \eta(t) \\ \eta(0) = -\nabla_{Y(0)} A \end{cases}, \tag{23}$$

$$\begin{cases} \dot{\xi}(t) = -\partial_2 G(Y(t), \mathbf{S}(t))^{\dagger} \eta(t) - d_{\mathbf{S}(t)} F^t \xi(t) \\ \xi(1) = 0 \end{cases}. \tag{24}$$

The proposition is proven in Appendix A.

The first ODE in (23) shows that the auxiliary variable η transports the image force $\nabla_{Y(0)} A$ from the template ($t = 0$) to the subject ($t = 1$) space via a *linear* ODE. The second ODE in (24) is a linear ODE with source term, whose source is given by the result of the previous integration. This last ODE is integrated backward in time: the resulting value $\xi(0)$ is directly the gradient that we were looking for ($d_{\mathbf{S}_0} Y(0)^{\dagger} \nabla_{Y(0)} A$). This shows that the product between the Jacobian matrix and the image force can be efficiently computed via a forward and backward integration of *linear* ODEs.

It is also important to notice that the gradient is computed with respect to the whole state $\mathbf{S}_0$, which means that the gradients with respect to the position of the control points and the momentum vectors are computed altogether via a coupled system of ODEs. The optimization of the position of the control points does not involve any additional computational cost in the gradient computation!

Now, we can apply Proposition 1, with the expressions of F and G given in (6) and (10) to get the contribution of the *ith* subject to the gradient (18) (note that one needs one time-varying variable $\mathbf{S}(t)$, $\eta(t)$, and $\xi(t)$ per subject).

Decomposing the $2dn_g$ vectors into two dn_g vectors, $\mathbf{S}_0 = (\mathbf{c}_0^g, \alpha_0)$ and $\xi = (\xi^c, \xi^\alpha)$, we get:

$$\nabla_{c_{0,k}^g} E = \xi_k^c(0) + \nabla_{c_{0,k}^g} L,$$

$$\nabla_{\alpha_{0,k}} E = \xi_k^\alpha(0) + \nabla_{\alpha_{0,k}} L,$$

where we have:

$$\nabla_{\alpha_{0,k}} L = 2 \sum_{p=1}^{n_g} K^g(c_{0,k}^g, c_{0,p}^g) \alpha_{0,p},$$

$$\nabla_{c_{0,k}^g} L = 2 \sum_{p=1}^{n_g} \alpha_{0,p}{}^t \alpha_{0,k} \nabla_1 K^g(c_{0,k}^g, c_{0,p}^g).$$

The term $\partial_1 G(Y(t), \mathbf{S}(t))$ is an operator on $L^2(\Omega, \mathbb{R}^d)$:

$$\partial_1 G = -\sum_{p=1}^{n_g} \alpha_p(t) \nabla_1 K^g(Y(t), c_k(t))^t \mathrm{Id}_{L^2}$$

so that the image of vectors $\eta(t)$ is updated according to:

$$\dot{\eta}(t) = -\sum_{p=1}^{n_g} \eta(t)^t \alpha_p(s) \nabla_1 K^g(Y(s), c_p^g(t)). \tag{25}$$

The term $\partial_2 G(Y(t), \mathbf{S}(t))$ is a row vector of $2dn_g$ images of vectors. Decomposing it into two blocks of size (dn_g), we get $\partial_2 G = (d_{\mathbf{c}^g} G(Y(t), \mathbf{S}(t)) \quad d_\alpha G(Y(t), \mathbf{S}(t)))$. Therefore,

$$\begin{aligned} \partial_2 G(Y(t), \mathbf{S}(t))^{\dagger} \eta(t) &= \begin{pmatrix} \langle d_{c_k(t)} G, \eta(t) \rangle_{L^2} \\ \langle d_{\alpha_k(t)} G, \eta(t) \rangle_{L^2} \end{pmatrix} \\ &= \begin{pmatrix} \int_\Omega d_{c_k(t)} G^t \eta(t) \\ \int_\Omega d_{\alpha_k(t)} G^t \eta(t) \end{pmatrix}. \end{aligned} \tag{26}$$

Similarly, the function F can be divided into two blocks

$$F(\mathbf{S}(t)) = \begin{pmatrix} F^c(\mathbf{S}(t)) \\ F^\alpha(\mathbf{S}(t)) \end{pmatrix}, \tag{27}$$

where $F^c(\mathbf{S}(t))$ and $F^\alpha(\mathbf{S}(t))$ are respectively the first and second row in (6). Therefore, the differential of F is decomposed into 4 blocks as follows:

$$d_{\mathbf{S}(t)} F = \begin{pmatrix} \partial_{c^g(t)} F^c & \partial_{\alpha(t)} F^c \\ \partial_{c^g(t)} F^\alpha & \partial_{\alpha(t)} F^\alpha \end{pmatrix}. \tag{28}$$

Given the expressions of F and G given in (6) and (10) respectively, the update rule for the auxiliary variables $\xi^c(t)$ and $\xi^\alpha(t)$ are:

$$\begin{aligned} -\dot{\xi}_k^c(t) = &\int_\Omega \nabla_1 K^g(c_k^g(t), Y(t)(x)) \eta(t)(x)^t \alpha_k(t) dx \\ &+ (\partial_{c^g} F^c)^t \xi^c(t)_k + (\partial_{c^g} F^\alpha)^t \xi^\alpha(t)_k \end{aligned} \tag{29}$$

$$
\begin{aligned}
-\dot{\xi}_k^{\alpha}(t) = &\int_{\Omega} K^{\mathrm{g}}(c_k^{\mathrm{g}}(t), Y(s)(x))\eta(t)(x)dx \\
&+ (\partial_{\alpha}F^c)^t\xi^c(t)_k + (\partial_{\alpha}F^{\alpha})^t\xi^{\alpha}(t)_k
\end{aligned}
\tag{30}
$$

with

$$
\begin{aligned}
(\partial_c F^c)^t\xi^c(t)_k = &\sum_{p=1}^{n_g} \nabla_1 K^{\mathrm{g}}(c_k^{\mathrm{g}}(t), c_p^{\mathrm{g}}(t))\alpha_p(t)^t\xi_k^c(t) \\
&+ \sum_{p=1}^{n_g} \nabla_1 K^{\mathrm{g}}(c_k^{\mathrm{g}}(t), c_p^{\mathrm{g}}(t))\xi_p^c(t)^t\alpha_k(t) \\
(\partial_c F^{\alpha})^t\xi^{\alpha}(t)_k = &- \sum_{p=1}^{n_g} \alpha_k(t)^t\alpha_p(t)\nabla_{1,1}K^{\mathrm{g}}(c_k^{\mathrm{g}}(t), c_p^{\mathrm{g}}(t))^t\xi_k^{\alpha} \\
&+ \sum_{p=1}^{n_g} \nabla_{1,1}K^{\mathrm{g}}(c_k^{\mathrm{g}}(t), c_p^{\mathrm{g}}(t))^t\xi_p^{\alpha}(t)\alpha_p(t)^t\alpha_k(t) \\
(\partial_{\alpha}F^c)^t\xi^c(t)_k = &\sum_{p=1}^{n_g} K^{\mathrm{g}}(c_k^{\mathrm{g}}(t), c_p^{\mathrm{g}}(t))\xi_p^c(t) \\
(\partial_{\alpha}F^{\alpha})^t\xi^{\alpha}(t)_k = &\sum_{p=1}^{n_g} \nabla_1 K^{\mathrm{g}}(c_k^{\mathrm{g}}(t), c_p^{\mathrm{g}}(t))^t\xi_p^{\alpha}(t)^t\alpha_p(t) \\
&- \sum_{p=1}^{n_g} \alpha_p(t)\nabla_1 K^{\mathrm{g}}(c_k^{\mathrm{g}}(t), c_p^{\mathrm{g}}(t))^t\xi_k^{\alpha}(t)
\end{aligned}
$$

where the time-varying vectors $c_k^{\mathrm{g}}(t)$ and $\alpha_k(t)$ have been computed by integrating the ODE (6) from the initial conditions $c_{0,k}^{\mathrm{g}}$ and $\alpha_{0,k}$, and the time-varying images of vectors $Y(t)$ by integrating backward the ODE (9).

In these equations, we supposed the kernel symmetric: $K^{\mathrm{g}}(x,y) = K^{\mathrm{g}}(y,x)$. If the kernel is a scalar isotropic kernel of the form $K^{\mathrm{g}} = f(\|x-y\|^2)\mathrm{Id}$, then we have:

$$
\begin{aligned}
&\nabla_1 K^{\mathrm{g}}(x,y) = 2f'(\|x-y\|^2)(x-y), \\
&\nabla_{1,1}K^{\mathrm{g}}(x,y) = 4f''(\|x-y\|^2)(x-y)(x-y)^t + 2f'(\|x-y\|^2)\mathrm{Id}.
\end{aligned}
$$

3.3 Numerical implementation

The implementation of the above equations requires: to compute the integrals over the image domain Ω, to compute the sum over the control points, and to integrate the ODEs.

For this purpose, we discretize the image domain Ω into a regular lattice of voxels. The positions of the voxels are denoted $\{y_k\}_{k=1,\dots,N_{\mathrm{im}}}$. Their flow under the inverse deformation is given by the discretization of the ODE in (9):

$$
\begin{cases}
\dot{y}_k(t) = -v_t(y_k(t)) \\
y_k(1) = y_k
\end{cases},
\tag{31}
$$

which gives the practical way to compute $Y(t)(y_k) = y_k(t)$.

This allows us to compute the image force and the data term using a sum of squared differences. Indeed, the image force $\nabla_{Y(0)}A(x) = 2(I_0(Y(0)(x)) - I(y_k))\nabla_{Y(0)(x)}I_0$ in (25) involve the computation of $I_0(Y(0))$ and $\nabla_{Y(0)}I_0$. According to the parametric image model, these two terms can be sampled at the positions y_k to build discrete images:

$$
\begin{aligned}
&I_0(y_k(0)) = \sum_{p=1}^{n_p} K^{\mathrm{ph}}(y_k(0), c_p^{\mathrm{ph}})w_p, \\
&\nabla_{y_k(0)}I_0 = \sum_{p=1}^{n_p} w_p\nabla_1 K^{\mathrm{ph}}(y_k(0), c_p^{\mathrm{ph}}).
\end{aligned}
\tag{32}
$$

Then, the data term is computed as the sum of squared differences between the images: $A(Y(0)) = \sum_{k=1}^{N_{\mathrm{im}}}(I_0(y_k(0)) - I(y_k))^2$.

For the numerical implementation, we suppose the kernel K^{g} translation-invariant ($K^{\mathrm{g}}(x,y) = f(x-y)$). In this case, all the integrals over the image domain Ω in (29) and (30) are convolutions. The kernel K^{g}, its gradient $\nabla_1 K^{\mathrm{g}}$, and the Jacobian matrix $\nabla_{1,1}K^{\mathrm{g}}$ are all translation-invariant, therefore one samples them at the nodes of the lattice and their FFTs are pre-computed. At a given time t, the voxels have moved to the position given in $y_k(t)$, which carry a vector $\eta(t)(y_k)$. Then one employs a splatting algorithm (also called Partial Volume Projection in Durrleman (2010)) to project the vectors $\eta(t)(y_k)$ at the neighbor voxels around the position $y_k(t)$. This is the numerical implementation of the change of variable $x = Y(t)(y)$ within the integrals. Then, one computes the convolution using the FFT of this image of vectors and the pre-computed FFT of the kernel. The output at the positions $c_k(t)$ are computed using a linear interpolation (also called Partial Volume Interpolation in Durrleman (2010)).

If the number of control points is small enough, then the sum over the number of control points can be implemented 'as is.' Otherwise, one may use the same approximation tool (Partial Volume Projection, followed by convolution between discrete images, followed by Partial Volume Interpolation) to compute the discrete convolutions. This is also called particle-mesh approximation in this context and is explained in-depth in Durrleman (2010)[Chap. 2]. In particular, the approximation error is controlled by the ratio between the grid size and the rate of decay of the kernel.

The ODEs are integrated by using a Euler scheme with prediction/correction scheme. This has the same accuracy as a Runge Kutta scheme of order 2.

3.4 Differentiation with respect to the photometric weights

To complete the computation of the gradient of the objective function, we need to differentiate it with respect to the photometric weights. The part of the criterion that depends on these weights is:

$$
\sum_{i=1}^{N} \int_{\Omega} (I_0(Y_i(0)) - I_i)^2,
$$

where, according to the parametric template model (2):

$$I_0(Y_i(0)(y)) = \sum_{p=1}^{n_p} K^{\mathrm{ph}}(Y_i(0)(y), c_p^{\mathrm{ph}}) w_p.$$

Therefore, the gradient with respect to the photometric weight is given by:

$$\nabla_{w_p} E = 2\sum_{i=1}^{N} \int_{\Omega} K^{\mathrm{ph}}(Y_i(0)(y), c_p^{\mathrm{ph}}) \left(I_0(Y_i(0)(y)) - I_i(y)\right) dy, \tag{33}$$

which is discretized as:

$$\nabla_{w_p} E = 2\sum_{i=1}^{N} \sum_{k=1}^{N_{\mathrm{im}}} K^{\mathrm{ph}}(y_{i,k}(0), c_p^{\mathrm{ph}}) \left(I_0(y_{i,k}(0)) - I_i(y_k)\right). \tag{34}$$

This gradient is also a convolution, which is implemented by projecting the current *ith* residual image $(I_0 \circ \phi_i^{-1} - I_i)$ at the neighboring voxels around positions $y_{i,k}(0)$, computing the convolution using FFTs and interpolating the output image at the positions of the photometric control points c_p^{ph}.

Eventually, the overall gradient minimization procedure is summarized in Algorithm 1, where we wrote the ODEs in integral forms and use the discrete version of the equations.

4 Adjusting the number of control points with sparsity priors

4.1 L^1-sparsity priors on geometric parameters

The dimension of the parameterization of the deformations is determined by the number of geometric control points. In this section, we would like to adjust the dimension of this parameterization, so that it better reflects the true number of degrees of freedom that is needed to describe the variability of the image set. An optimal set of geometric control points would be concentrated near the contours of the template image, where the need of deformation is the most important.

The kinetic energy is used as a L^2 regularity term used in the criterion. The effect of this term is to spread the 'total amount of momentum' that is needed over the whole set of control points. Indeed, it is always less energetic to have two momentum vectors pushing in the same direction with the same weight, than only one with a doubled weight. This is in contradiction with our goal to select a small amount of relevant control points to describe the variability of the image set. To enforce the distribution of momenta to be concentrated on a small set of control points, we introduce an additional L^1 penalty term in the spirit of elastic nets (Zou and Hastie, 2005)[2]:

$$E(\mathbf{w}, \mathbf{c}_0^{\mathrm{g}}, \{\alpha_{0,i}\}_{i=1,\dots,N}) = \sum_{i=1}^{N} \left\{ A_i(\mathbf{w}, \mathbf{y}_i(0)) + L_i(\mathbf{c}_0^{\mathrm{g}}, \alpha_{0,i}) + \gamma_{\mathrm{g}} \sum_{p=1}^{n_g} \|\alpha_{0,i,p}\| \right\}, \tag{35}$$

where $\|.\|$ denotes the Euclidean norm in the ambient 2D or 3D space.

As we will see, the effect of this prior is to enforce momentum vectors of small magnitude to vanish. Therefore, this will enforce the deformations to be encoded in a small number of non-zero parameters. We will say that a given geometric control point is c_p^{ph} *active*, if the momentum vector $\alpha_{0,i,p}$ is non-zero for at least one subject i. The effect of the sparsity prior is to minimize the number of active control points.

4.2 Optimization with F/ISTA

To optimize this new criterion, we rely on the adaptation of the gradient method called *Iterative Shrinkage Thresholding Algorithm* (ISTA) (Beck and Teboulle, 2009) and its faster version called FISTA for Fast-ISTA. The idea is to use the previous gradient of the least square criterion (i.e. without the L^1 penalty) and then to threshold the update of the momentum vectors if their magnitude is not large enough. Therefore, at any time of the optimization procedure, a given momentum vector can be set to zero if the gradient does not push strongly enough, or, on the contrary, can make active an inactive control point if the gradient has a large enough magnitude. The F/ISTA method enables to set the threshold given the sparsity weight γ_{g} and the current step-size of the gradient descent. The fast version adds the ideas of Nesterov (1983) to speed-up the optimization procedure.

To be more precise, let us write the new criterion as:

$$E(\{\alpha_{0,i}\}_i, \mathbf{w}) = E^{LS}(\{\alpha_{0,i}\}_i, \mathbf{w}) + g_\alpha(\{\alpha_{0,i}\}_i),$$

where E^{LS} denotes the previous least-square criterion (17) and $g_\alpha(\{\alpha_{0,i}\}_i) = \gamma_{\mathrm{g}} \sum_{i=1}^{N} \|\alpha_{0,i}\|_{\mathbb{R}^{n_g}}$.

F/ISTA is built on the quadratic approximation of the criterion as:

$$\begin{aligned} Q_{L_{\mathrm{ph}},L_{\mathrm{g}}}(\{\alpha_{0,i}\}_i, \{\alpha'_{0,i}\}_i, \mathbf{w}, \mathbf{w}') &= E^{LS}(\{\alpha'_{0,i}\}_i, \mathbf{w}') \\ &+ \sum_{i=1}^{N} (\alpha_{0,i} - \alpha'_{0,i})^t \nabla_{\alpha'_{0,i}} E^{LS} + (\mathbf{w} - \mathbf{w}') \nabla_{\mathbf{w}'} E^{LS} \\ &+ \frac{1}{2L_{\mathrm{g}}} \|\alpha_{0,i} - \alpha'_{0,i}\|^2 + \frac{1}{2L_{\mathrm{ph}}} \|\mathbf{w} - \mathbf{w}'\|^2 + g_\alpha(\alpha_{0,i}), \end{aligned} \tag{36}$$

[2] Note that this is not exactly the elastic net paradigm, since we do not use the usual Euclidean norm on the momentum vectors for the L^2 penalty $(\alpha_{0,i}{}^t \alpha_{0,i})$ but the L^2 metric induced by the metric $\mathbf{K}^{\mathrm{g}}$ instead $(\alpha_{0,i}{}^t \mathbf{K}^{\mathrm{g}} \alpha_{0,i})$

Algorithm 1 Atlas Estimation with Adaptive Parameterization

```
**Input/Initialization:**
set of images $I_i$ for $i = 1, \ldots, N$
array of positions of the pixels in the image domain $\mathbf{y} = \{y_k\}_k$
photometric kernel $K^{\mathrm{ph}}$, geometric kernel $K^{\mathrm{g}}$
array of positions of photometric control points $\mathbf{c}^{\mathrm{ph}}$
array of positions of geometric control points $\mathbf{c}_0^{\mathrm{g}}$
trade-off regularity/fidelity-to-data $\sigma^2$
vector of photometric weights $\mathbf{w} \leftarrow 0$
momentum vectors $\alpha_{0,i} \leftarrow 0$ for all subjects $i$

**repeat** {Gradient descent}
    $\nabla_{\mathbf{w}} E \leftarrow 0$, $\nabla_{c_0^{\mathrm{g}}} E \leftarrow 0$, $\nabla_{\alpha_{0,i}} E \leftarrow 0$
    **for** $i = 1, \ldots, N$ **do**
        {Generate deformation as in (6) (forward integration)}
        $c_k^{\mathrm{g}}(t) = c_{0,k}^{\mathrm{g}} + \int_0^t \sum_{p=1}^{n_g} K^{\mathrm{g}}(c_k^{\mathrm{g}}(s), c_p^{\mathrm{g}}(s)) \alpha_{i,p}(s) ds$
        $\alpha_{i,k}(t) = \alpha_{0,i,k} - \int_0^1 \sum_{p=1}^{n_g} \alpha_{i,k}(s)^t \alpha_{i,p}(s) \nabla_1 K^{\mathrm{g}}(c_p^{\mathrm{g}}(s), c_k^{\mathrm{g}}(s)) ds$
        {Deform the image domain with $\phi_i^{-1}$ (backward integration)}
        $y_k(t) = y_k - \int_t^1 \sum_{p=1}^{N_{\mathrm{im}}} K(y_k(s), c_p(s)) \alpha_{i,j}(s) ds$
        {Compute image force}
        $I_0(y_k(0)) = \sum_{p=1}^{n_p} K^{\mathrm{ph}}(y_k(0), c_p^{\mathrm{ph}}) w_p$ {deformed template image}
        $\nabla_{y_k(0)} I_0 = \frac{1}{2\sigma^2} \sum_{p=1}^{n_p} w_p \nabla_1 K^{\mathrm{ph}}(y_k(0), c_p^{\mathrm{ph}})$ {deformed gradient template image}
        $\nabla_{y_k(0)} A = \frac{1}{\sigma^2} (I_0(y_k(0)) - I_i(y_k)) \nabla_{y_k(0)} I_0$ {Image Force}
        {Compute auxiliary variable $\eta$ as in (25) (forward integration)}
        $\eta_k(t) = -\nabla_{y_k(0)} A - \int_0^t \sum_{p=1}^{n_g} \eta_k(s)^t \alpha_p(s) \nabla_1 K^{\mathrm{g}}(y_k(s), c_p^{\mathrm{g}}(s)) ds$
        {Compute auxiliary variables $\xi^c$ and $\xi^\alpha$ as in (29) and (30) (backward integration)}
        $\eta_k^c(t) = \int_t^1 \dot{\xi}_k^c(s) ds$ as in (29)
        $\eta_k^\alpha(t) = \int_t^1 \dot{\xi}_k^\alpha(s) ds$ as in (30)
        {Compute gradient}
        $\nabla_{c_{0,k}^{\mathrm{g}}} E \leftarrow \nabla_{c_{0,k}^{\mathrm{g}}} E + \xi_k^c(0) + 2 \sum_{p=1}^{n_g} K^{\mathrm{g}}(c_{0,k}^{\mathrm{g}}, c_{0,p}^{\mathrm{g}}) \alpha_{0,i,p}$
        $\nabla_{\alpha_{0,i,k}} E \leftarrow \xi_k^\alpha(0) + 2 \sum_{p=1}^{n_g} {\alpha_{0,i,p}}^t \alpha_{0,i,k} \nabla_1 K^{\mathrm{g}}(c_{0,k}^{\mathrm{g}}, c_{0,p}^{\mathrm{g}})$
        $\nabla_{w_p} E \leftarrow \nabla_{w_p} E + 2 \sum_{k=1}^{N_{\mathrm{im}}} K^{\mathrm{ph}}(y_{i,k}(0), c_p^{\mathrm{ph}})(I_0(y_{i,k}(0)) - I_i(y_k))$
    **end for**
    {Update parameters}
    $\mathbf{w} \leftarrow \mathbf{w} - \tau \nabla_{\mathbf{w}} E$ {Update photometric weights}
    $\mathbf{c}_0^{\mathrm{g}} \leftarrow \mathbf{c}_0^{\mathrm{g}} - \tau' \nabla_{\mathbf{c}_0^{\mathrm{g}}} E$ {Update positions of geometric control points}
    $\alpha_{0,i} \leftarrow \alpha_{0,i} - \tau' \nabla_{\alpha_{0,i}} E$ for $i = 1, \ldots, N$ {Update the momentum vectors of each subject}
**until** Convergence

**Output:**
Template image $I_0 = \sum_{k=1}^{n_p} K^{\mathrm{ph}}(., c_k^{\mathrm{ph}}) w_k$
Set of optimal control points in the template image domain: $\mathbf{c}_0^{\mathrm{g}}$
Parameterization of template-to-subject deformations by momentum vectors $\alpha_{0,i}$
```

where $L_{\mathrm{g}}, L_{\mathrm{ph}}$ are two positive constants, which will play the role of two step-sizes in the adapted gradient descent scheme.

The key tool of F/ISTA is the minimization of this quadratic approximation:

$$\underset{\alpha_{0,1},\ldots,\alpha_{0,N},\mathbf{w}}{\arg\min} \; Q_{L_{\mathrm{ph}},L_{\mathrm{g}}}(\{\alpha_{0,i}\}_i, \{\alpha'_{0,i}\}_i, \mathbf{w}, \mathbf{w}')$$

as a function of the $\alpha'_{0,i}$'s and $\mathbf{w}'$.

Since Q is a sum of positive terms involving only either the variables $\alpha_{0,i}$ or each of the coordinates w_k, the minimum is reached for w_k and $\alpha_{0,i}$ being equal to:

$$p_{L_{\mathrm{ph}}}(\mathbf{w}'_k) = \underset{w \in \mathbb{R}}{\arg\min} \left(\frac{1}{2L_{\mathrm{ph}}} \left| w - (w'_k - L_{\mathrm{ph}} \nabla_{w'_k} E^{LS}) \right|^2 \right)$$

$$p_{L_{\mathrm{g}}}(\alpha'_{0,i}) = \underset{\alpha \in \mathbb{R}^{n_g}}{\arg\min} \left(\gamma_{\mathrm{g}} \|\alpha\| + \frac{1}{2L_{\mathrm{g}}} \left\| \alpha - (\alpha'_{0,i} - L_{\mathrm{g}} \nabla_{\alpha'_{0,i}} E^{LS}) \right\|^2 \right).$$

The first minimizer is given by the usual update of the gradient descent:

$$p_{L_{\mathrm{ph}}}(\mathbf{w}'_k) = w'_k - L_{\mathrm{ph}} \nabla_{w'_k} E^{LS}. \tag{37}$$

Algorithm 2 Sparse Atlas Estimation with FISTA

```
**Initialization:**
$k \leftarrow 0$
geometric control points $\mathbf{c}_0^{\mathrm{g}}(k)$ as nodes of a regular lattice
momentum vectors $\alpha_{0,i}(k) \leftarrow 0$ for all subjects $i$
vector of photometric weights $\mathbf{w}(k) \leftarrow 0$
Take step-sizes $L_{\mathrm{ph}} > 0$, $L_{\mathrm{g}} > 0$ and $\eta > 1$
Set $\mathbf{c}_0^{\mathrm{g}\prime}(k+1) = \mathbf{c}_0^{\mathrm{g}}(k)$, $\alpha'_{0,i}(k+1) = \alpha_{0,i}(k)$ and $\mathbf{w}'(k+1) = \mathbf{w}(k)$
Set $t_1 = 1$, $k = 1$
**repeat**
    $k \leftarrow k+1$
    Compute $\nabla_{\mathbf{c}_0^{\mathrm{g}\prime}(k)} E^{LS}$, $\nabla_{\alpha'_{0,i}(k)} E^{LS}$ and $\nabla_{\mathbf{w}'(k)} E^{LS}$ as in Algorithm 1
    Find the smallest pair of nonnegative integers $i_k, j_k$ (e.g. for the lexicographic order) such that with $\overline{L_{\mathrm{ph}}} = L_{\mathrm{ph}}/\eta^{i_k}$ and $\overline{L_{\mathrm{g}}} = L_{\mathrm{g}}/\eta^{j_k}$
        $$E(\{p_{\overline{L_{\mathrm{g}}}}(\alpha'_{0,i}(k))\}, p_{\overline{L_{\mathrm{ph}}}}(\mathbf{w}'(k))) \leq Q_{\overline{L_{\mathrm{ph}}},\overline{L_{\mathrm{g}}}}(\{\alpha_{0,i}(k)\}_i, \{p_{\overline{L_{\mathrm{g}}}}(\alpha'_{0,i}(k))\}_i, \mathbf{w}, p_{\overline{L_{\mathrm{ph}}}}(\mathbf{w}'(k)))$$
    $L_{\mathrm{ph}} \leftarrow \overline{L_{\mathrm{ph}}}$, $L_{\mathrm{g}} \leftarrow \overline{L_{\mathrm{g}}}$
    $\mathbf{c}_0^{\mathrm{g}}(k) = \mathbf{c}_0^{\mathrm{g}\prime}(k) - L_{\mathrm{g}} \nabla_{\mathbf{c}_0^{\mathrm{g}\prime}(k)} E^{LS}$
    $\alpha_{0,i}(k) = p_{L_{\mathrm{g}}}(\alpha'_{0,i}(k))$
    $\mathbf{w}(k) = p_{L_{\mathrm{ph}}}(\mathbf{w}'(k))$
    $t_{k+1} = \left(1 + \sqrt{1 + 4t_k^2}\right)/2$
    $\mathbf{c}_0^{\mathrm{g}\prime}(k+1) = \mathbf{c}_0^{\mathrm{g}}(k) + \left(\frac{t_k - 1}{t_{k+1}}\right)(\mathbf{c}_0^{\mathrm{g}}(k) - \mathbf{c}_0^{\mathrm{g}}(k-1))$
    $\alpha'_{0,i}(k+1) = \alpha_{0,i}(k) + \left(\frac{t_k - 1}{t_{k+1}}\right)(\alpha_{0,i}(k) - \alpha_{0,i}(k-1))$
    $\mathbf{w}'(k+1) = \mathbf{w}(k) + \left(\frac{t_k - 1}{t_{k+1}}\right)(\mathbf{w}(k) - \mathbf{w}(k-1))$
**until** Convergence

**Output:**
Set of optimally placed control points $\mathbf{c}_0^{\mathrm{g}}(k)$
Sparse set of non-zero momentum vectors $\alpha_{0,i}(k)$
Sparse set of non-zero photometric weights $\mathbf{w}(k)$
```

Applying Lemma 2 in Appendix B shows that the second minimizer is given by:

$$p_{L_{\mathrm{g}}}(\alpha'_{0,i}) = S_{\gamma_{\mathrm{g}} L_{\mathrm{g}}}\left(\left\|\alpha'_{0,i} - L_{\mathrm{g}} \nabla_{\alpha'_{0,i}} E^{LS}\right\|\right) \frac{\alpha'_{0,i} - L_{\mathrm{g}} \nabla_{\alpha'_{0,i}} E^{LS}}{\left\|\alpha'_{0,i} - L_{\mathrm{g}} \nabla_{\alpha'_{0,i}} E^{LS}\right\|}, \tag{38}$$

where the S_γ denotes the usual soft-thresholding function $S_\gamma(x) = \max(x - \gamma, 0) - \min(x + \gamma, 0)$, where the threshold $\gamma_{\mathrm{g}} L_{\mathrm{g}}$ is the product of the sparsity weight and the current step-size. This soft-thresholding function has mostly two effects. First, it tends to penalize the update $(\alpha'_{0,i} - L_{\mathrm{g}} \nabla_{\alpha'_{0,i}} E^{LS})$ of high magnitude by adding or subtracting the quantity $\gamma_{\mathrm{g}} L_{\mathrm{g}}$. Second, it thresholds to zero any update whose magnitude is below $\gamma_{\mathrm{g}} L_{\mathrm{g}}$, thus enforcing the sparsity.

According to Beck and Teboulle (2009), these updates are combined within a gradient descent called FISTA, as shown in Algorithm 2. Note that the gradient with respect to the position of the control points is not affected by the sparsity prior.

Remark 5 (Non-convex optimization) *F/ISTA is proven to converge to the minimum of the criterion if the least-square criterion is a convex function. In our case, the criterion is convex with respect to the photometric weights, but is* not *convex with respect to the momentum vectors. This has the same drawback as using the gradient descent scheme for non-convex optimization: only local minima can be reached. However, the conditions, under which the F/ISTA algorithm converges to a local minimum of the criterion, still need to be rigorously established. Experimentally, our results will show very stable output if the sparsity weight γ_g is varied, which suggests good convergence properties of the optimization procedure.*

5 Experiments

5.1 The importance of optimally placed control points

In this section, we illustrate the discrete parameterization of deformations with the registration between a pair of images. We compute the registration between two simulated images and use a regular lattice as the initial set of geometric control points, with a spacing equal to the standard deviation of the geometric kernel (σ_{g}). This gives a set of 25 control points, which is the maximum number of degrees of freedom allowed by the deformation model. The registration consists

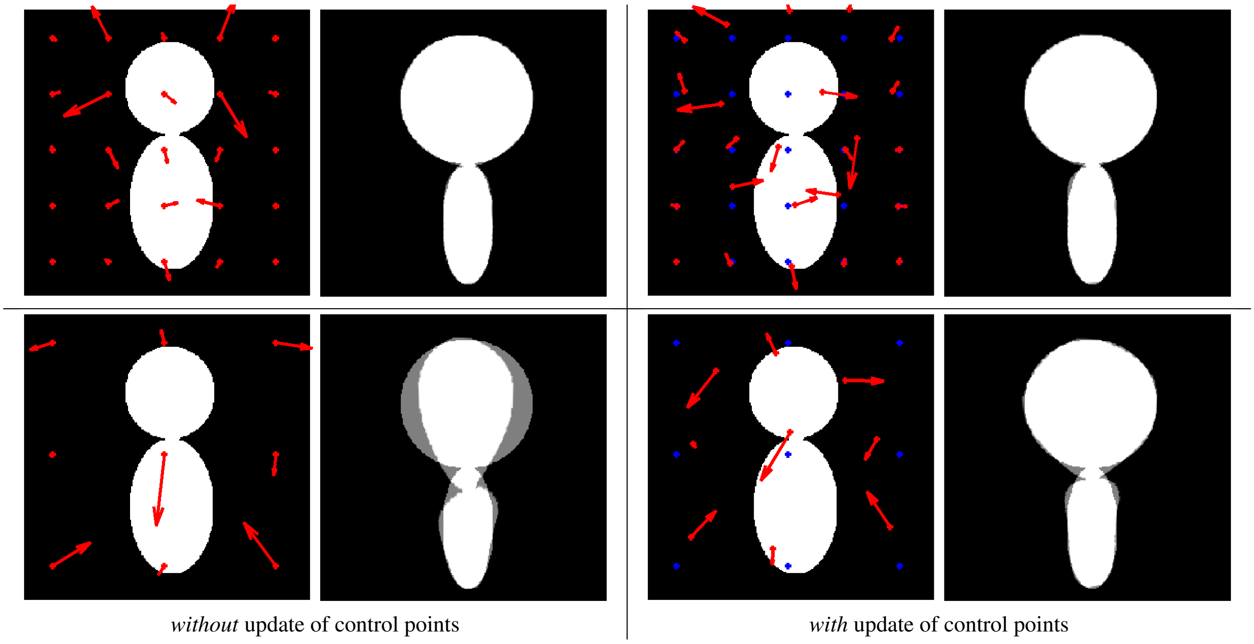

without update of control points
with update of control points

Fig. 2 Registration between a pair of simulated images. A discrete parameterization of the deformation is estimated using 25 (top) or 9 (bottom) control points. On the left of each panel: the source image with the initial momenta (red arrows). On the right the superimposition of the deformed source and target image. First row shows that a *discrete* parameterization is sufficient for a perfect matching. Second row shows that moving the control points to their optimal positions gives a much better representation of the shape differences for a fixed number of parameters. Note that the optimal position of the control points tends to be close to the boundary of the shape (or the areas of high gradients). Standard deviation of the geometric kernel $\sigma = 50$ pixels, trade-off between regularity and fidelity to data $\sigma^2 = 10^{-2}$. Images are 128×128.

in optimizing the momenta to get the best matching possible using the gradient descent. In Fig. 2 (top left), we show the results obtained by optimizing only the momentum vectors (in magnitude and direction), whereas in the top right panel, we show the results when both the position of the control points and the momentum vectors attached to them are optimized.

We see that both approaches lead to an accurate matching between both images, as would do the usual image matching methods. This means that the discrete parameterization of the deformation could efficiently replace the usual continuous parameterization with a continuous momenta map, providing that one is able to accurately determine the number and the positions of the control points. In the former, the number of parameters that encode the deformation is $2*25 = 50$, whereas a continuous momenta map would involve as many parameters as the total number of pixels of the images: $128^2 = 16884$. With the parameterization we proposed, we achieved a compression ratio of 99.6% in the parameterization of the deformation, with minimal sacrifice to the matching accuracy.

However, it is likely that the number of parameters needed to describe the difference between these two images is much smaller than $2*25$. In this experiment, we manually select a subset of nine regularly spaced control points to drive the registration. In Fig. 2 (bottom left), this set of control points does not allow an accurate match of the two images. But, if one optimizes the position of the control points during the registration (Fig 2, bottom right), then an accurate matching can be obtained with only nine control points. This shows that a sparse parameterization of the deformation could not be obtained without an optimal placement of the control points in the image domain.

In this experiment, the number of control points was fixed. In the next experiments, we will use the sparsity prior introduced in Sec. 4 to automatically select the most relevant control points and therefore determine the optimal number of them.

5.2 Atlas of 3 simulated images

We use a set of three simulated images to illustrate our method (Fig. 3). The image dimensions are 128×128 pixels, and we fixed the standard deviation of the geometric kernel to $\sigma_g = 25$, the standard deviation of the photometric kernel to $\sigma_{ph} = 5$, and the trade-off between regularity and fidelity-to-data to $\sigma^2 = 5\,10^{-3}$, which is small enough to allow almost perfect matching between images. We initialize the algorithm with a regular lattice of 25 geometric control points with a spacing equal to σ_g and with a regular lattice of 676 photometric control points with a spacing equal to σ_{ph}, which represents only 4% of the 128^2 pixels in the image.

For a fixed value of the sparsity prior $\gamma_g = 250$, the atlas is given in Fig. 3. We assess the impact of the sparsity

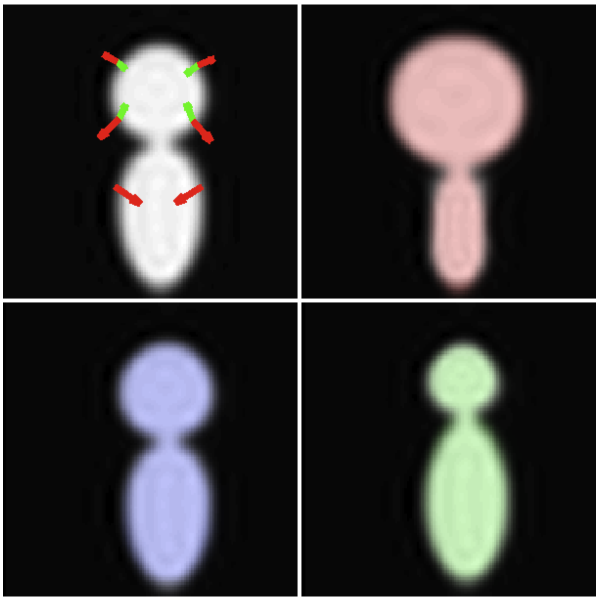


Fig. 3 Atlas estimation from a set of 3 simulated images. Top left: the template image superimposed with the initial momentum vectors. The color of the vectors corresponds to that of the image. Top-right and bottom: the original image superimposed with the warped template image. The superimposition shows a matching with a high accuracy.

prior on the atlas estimation by varying the sparsity parameter γ_g between 0 and 900. Significant samples are shown in Fig. 4. As expected, a small value of the sparsity parameter leads to very accurate matching and sharp atlas, but at the cost of a redundant parameterization of the deformations: only a few control points are not active (Fig. 4 first row). By increasing the sparsity parameter, the representation becomes sparser and sparser (Fig. 4 second and third row), while keeping nearly the same atlas sharpness. However, if the sparsity prior is too strong, the counterpart of the sparsity is a less and less accurate matching and therefore less and less sharp template image (Fig. 4 fourth row). This suggests that there is an optimal value of the sparsity prior for which the representation is as sharp as possible with minimal sacrifice to the atlas sharpness. The corresponding number of control points would give an estimate of an optimal number of degrees of freedom needed to capture the variability in the image ensemble.

To give a more quantitative evaluation of this assumption, we plot the evolution of the number of control points and the norm of the residual matching errors (as a measure of the atlas sharpness) versus the sparsity prior in Fig. 5. This experiment shows that the number of active control points can be decreased from 25 to 8 with minimal sacrifice to the atlas sharpness by increasing the sparsity parameter from 0 to 250. Beyond this value, the number of control points is stable before decreasing again, but at the cost of an exponentially increasing residual data term, meaning a less and less accurate description of the variability of the image ensemble. This suggests an optimal value of the parameter near $\gamma_g \sim 250$.

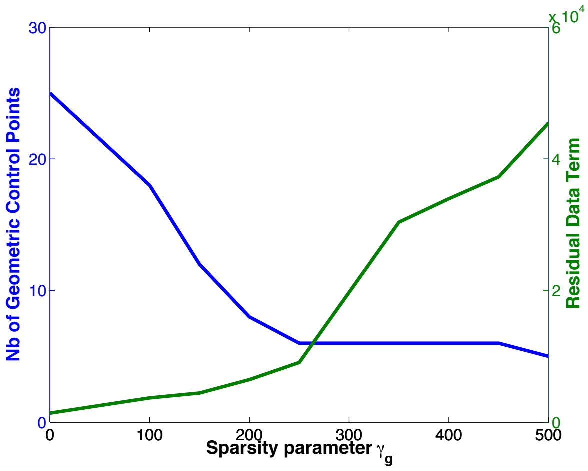


Fig. 5 Impact of the sparsity parameters γ_g on the atlas sharpness. The greater γ_g, the fewer the number of active geometric control points, the less sharp the atlas. The 'L'-shape of the curves shows that the number of geometric control points can be reduced from 25 to 8 with minimal sacrifice to the atlas sharpness. The 'optimal' value of $\gamma_g = 250$ selects the minimal number of degree of freedom to capture most of the variability in the image ensemble.

5.3 Atlas of 20 images from the US postal database

In this section, we used the US postal database to estimate the variability in hand-written digits (Hastie et al, 2009). The size of the images is 16×16. We set the standard deviation to $\sigma_g = 3$ for the geometric kernel, to $\sigma_{ph} = 1.1$ for the photometric kernel and the trade-off between regularity and fidelity-to-data to $\sigma^2 = 10^{-3}$.

For each digit (from 0 to 9), we estimated an atlas from a training set of 20 images. Then, we registered the estimated template image to a set of 10 test images (different from the training samples) using the set of control points that has been selected and placed during the atlas construction. We repeated the experiment for 26 different training sets with no intersection between the training sets. We also randomized the test sets in a similar fashion. Eventually, we had 26 different atlases and 260 registrations to test data for each digit. We repeated the whole cross-validation procedure for a value of the sparsity parameter γ_g varying between 0 and 1000 by a step of 50. In Fig. 6, we show the decrease profile of the number of control points in the atlas with respect to the sparsity parameter γ_g. It shows in particular the relative low variance of this number when the training samples are varied, thus showing the robustness of the atlas construction method. We used the residual data term after registration to the test samples as a measure of capability of the atlas to capture the variability of the shapes of the digits. The variation of this measure with respect to the sparsity parameter γ_g (Fig. 6) shows a sigmoid-like curve for digits 2, 4, 5 and 8 or an exponential-like curve for digit 0, 1, 3, and to a lesser extent for digits 6, 7 and 9. In the most obvious cases, the graph shows that there is likely to be an optimal value of the sparsity parameter for which the number of control points is significantly decreased and the capability of the atlas to capture shape variability has not been dramatically altered. This is confirmed by computing the Wilcoxon test between distribution of the residual data term at two consecutive values of the sparsity parameters (red segments in Fig. 6 denote intervals of statistically significant increase, p-value < 1%). In almost every case, there is an interval from $\gamma_g = 100$ onwards, for which the residual data term does not significantly increase (no red segments in Fig. 6): this is the range of values for which one can decrease the number of control points, without significantly altering the variability captured by the model. Once one reaches the red zone, there is a risk that we loose significant information. Note that the extent of the red zone depends on the threshold used for the test, here 1%.

For the largest sparsity priors, the template image is very fuzzy (it is the mean image) and there are no control points to capture the variability. In this case, the residual term measures the variance of the image set, and this measure itself has a large variance across the cross-validation tests (Fig. 6). For the smallest values of the sparsity parameter, the variance of the residual term is smaller, thus suggesting that the atlas captured most of the image ensemble variability and that the residual term captures mostly noise that does not vary much when randomizing the training and test sets. This is also confirmed by the Wilcoxon tests that take into account both the median and the variance of the distribution of the residual data term.

The images in Fig. 6 show a template image and the corresponding distribution of control points for the sparsity parameter that seems to be a good balance between sparsity and atlas sharpness. Fig. 7 and 8 show atlases for other values of the sparsity parameters for the digit 0 and 2 respectively.

Note that even if we need to remove the sparsity prior ($\gamma_g = 0$) to have a sharp atlas, the total number of degrees of freedom in the parameterization of the deformations is only of $2 * 36 = 72$, which is significantly smaller than the total number of pixels in the image $16 * 16 = 256$, which would be the size of the parameterization of the deformation in the usual atlas construction method. From a statistical point of view, this means that we could expect better estimates of the mean and covariance, since we manage to decrease the ratio between the dimension of the variables (p) and the number of observations (N) for a given data set. Even in this least favorable case ($p = 72$ without L^1 prior), the adaptive finite-dimensional parameterization of the deformation that we introduced should help to increase the power of the statistical estimations derived from the atlas.

Eventually, we simulated new images according to the variability captured by the atlas. To this end, we performed a Principal Component Analysis of the initial momentum vectors as follows. For each atlas shown in Fig. 6 (left panels that show the atlas for a given value of the sparsity parameter), we compute the sample mean and centered covariance matrix of the set of 20 initial momentum vectors. With the notations of the previous sections, the empirical mean writes:

$$\overline{\alpha}_0 = \frac{1}{20}\sum_{i=1}^{20}\alpha_{0,i}$$

and the $(i,j)th$ term of centered covariance matrix Σ:

$$\Sigma_{i,j} = \frac{1}{20}\sum_{p=1}^{n_g}\sum_{q=1}^{n_g}(\alpha_{0,i,p}-\overline{\alpha}_{0,p})^t K^{\mathrm{g}}(c^{\mathrm{g}}_{0,p},c^{\mathrm{g}}_{0,q})(\alpha_{0,j,q}-\overline{\alpha}_{0,q})$$

where we used the metric induced by the kernel K^{g} to compute the inner-product between the set of momentum vectors of two subjects.

Given V_m and λ_m the 19 eigenvectors and non-zero eigenvalues of the matrix Σ, the mth direction of the eigenmode is given as:

$$\tilde{\alpha}_m = \sqrt{\lambda_m}\frac{\sum_{i=1}^{20}V_{m,i}\alpha_{0,i}}{\left\|\sum_{i=1}^{20}V_{m,i}\alpha_{0,i}\right\|_V} = \sqrt{\frac{1}{20}}\sum_{i=1}^{20}V_{m,i}\alpha_{0,i}$$

since $\left\|\sum_{i=1}^{20}V_{m,i}\alpha_{0,i}\right\|_V^2 = 20\lambda_m$. Therefore, we simulate a new set of initial momentum vectors as:

$$\tilde{\alpha} = \overline{\alpha} \pm \sum_{m=1}^{19}\gamma_m\tilde{\alpha}_m \tag{39}$$

where γ_m are independent and identically distributed normal variables. For each sampling of the γ_m variables, we simulate two images according the sign in (39), which corresponds to the mean ± one standard deviation. To create the image, one finds the geodesic deformation corresponding to the simulated set of momemtum vectors $\tilde{\alpha}$ by solving (6), and then deform the template image solving (10).

Results of the simulations are shown in Fig. 9. The simulated images show that the atlas is able to reproduce a large part of the variability of the image ensemble. Note that we used a Gaussian model in this simulation, which is a symmetric distribution around the mean, whereas the true distribution of the observations is not. Therefore, we observed sometimes an unrealistic image, whereas the image generated along the opposite direction resembles one in the dataset.

6 Discussion and Conclusion

In this paper, we proposed a control point parameterization of large deformation diffeomorphisms to drive template-to-subject image registrations. Given an image ensemble, the proposed method moves the position of the control points toward the most variable parts of the images. The optimization in control point positions opens up the possibility to drastically reduce the number of control points by selecting those that are the most relevant for the description of the variability in the image set. This is done by introducing a L^1 prior in the spirit of the now *in vogue* sparsity methods. The decomposition of the template-to-subject registrations onto these control points gives a compact and adapted descriptor of the image variability. This descriptor of small dimension reflects the constrained nature of the variability of a given image set. To the very best of our knowledge, this is the first time that sparsity methods are used for deformations learning and statistical image analysis, in the context of Grenander's group action approach for modeling objects (Grenander and Miller, 1998). This is in contrast to methods that focus on the decomposition of the image intensity in a sparse dictionary like in Yu et al (2010).

The proposed parameterization of diffeomorphisms for image matching differs from LDDMM image registration, for which the deformation is parameterized by a continuous map of momenta that are always parallel to the image gradient (Miller et al, 2006). Here, we proposed to use a finite set of momenta, which are not constrained in their direction. Control points techniques have been widely used for small deformation transformations, for instance in Glasbey and Mardia (2001), whereas its use for large deformation matching of images is much more challenging. In Rueckert et al (2006), diffeomorphisms were built by a composition of small B-splines transforms without a comprehensive variational formulation and without optimizing the positions of the control points. In Allassonnière et al (2005), diffeomorphisms were characterized via a finite set of initial momenta located at the vertices of a "texture mesh," but no attempt was made to estimate an optimal mesh describing a whole population of images. The parameterization of diffeomorphisms by seed points in Grenander et al (2007) does not fall in a Riemannian framework and therefore is also difficult to use for template estimation and statistical analysis. The inherent difficulty is to find an efficient way to transport information back and forth from the template space to the space of each subject. Indeed, control points flow from template to subject's space (via the deformation ϕ), whereas the information contained in the image set needs to be pulled back to the source to build the template image ($I_0 \circ \phi^{-1}$). Similarly, for the computation of the gradient, small variations of the data term need to be transported back and forth to compute the induced variations of the deformations parameters. In this work, we address this issue by using an *explicit* dynamical system to drive the deformation. Then, a derivation borrowed from optimal control theory allows us to show that the gradient of the criterion can be efficiently computed by integrating the *linearized* dynamical system (Prop 1). One of the striking results of this formulation is that optimizing the positions of the control points in the template space can be done *at no additional computational cost* at each iteration. Another advantage of using an explicit dynamical system formulation is that the deformations are fully characterized by the initial conditions of the ODEs, thus giving a very efficient way to define intrinsic statistics in the space of deformations. The initial conditions are used as descriptors of the variability.

In our approach, the motion of the control points to their optimal place is driven by the gradient of the objective function. This is in contrast to Marsland and McLachlan (2007); Hansen et al (2008) where control point positions are assessed heuristically. We also tried heuristic rules to add control points where the residual image forces were the most important and to remove control points so that the orthogonal projection of the velocity field on the space spanned by the rest of the control points was maximized. However, we noticed that such heuristics were not competitive as they lack reproducibility, robustness and do not allow minimizing the cost function as efficiently as with the L^1 prior optimized with FISTA. In the FISTA optimization also, inactive control points could become active at any iteration, and vice versa. The only drawback of FISTA is to fix the maximum number of control points that could become active, namely the number of control points in the initial set. However, in our case, we know that the maximum number of control points is the number of patches of radius σ_g (i.e. the standard deviation of the deformation kernel) that is needed to cover the whole image domain (Durrleman et al, 2009). This is confirmed empirically since we always observed a decrease of the number of active control points as soon as the L^1 prior weight was not zero, meaning that we always overestimated the number of control points needed. Constraining the control points to be initially at the nodes of a regular lattice does not seem to be a strong constraint either, since, as shown in Fig. 2, control points could move up to half the distance to their closest neighbors.

In our model equation (1), we supposed the noise normally distributed. However, it is clear that the residual difference image after registration has some spatial structure on it. Therefore, spatially structured noise would be more relevant, but this would make the derivation of the criterion in the Maximum A Posteriori sense much more challenging. A workaround could be to perform a Principal Component Analysis on the residuals to discover the spatial correlations, as done in Cootes et al (2001) for images or in Durrleman et al (2009) for geometric structures. Changes in texture or appearance could be a confounding effect in diffeomorphic registration. It could be beneficial therefore to use more intrinsic models of texture like in Meyer (2001); Trouvé and Younes (2005).

The methods in Risser et al (2011); Sommer et al (2012b) propose a multi-scale parameterization of diffeomorphic deformations for landmark or image matching. These ideas could be included in our statistical framework for group studies, so that each control point carries a set of momenta associated with different kernel sizes. In Sommer et al (2012a), the authors proposed to also extend the dictionary of basis elements by adding differentials of the kernel, thus modeling torques as differentials of the Delta Dirac currents (Durrleman, 2010). It is clear that making the dictionary of basis elements the largest possible will increase the compactness of the parameterization of a given deformation and will ease the statistical processing and the interpretability of the results. It is also clear that more work has to be done in this direction.

We expect to show in the future that the resulting compact and adapted descriptors increase the power of the statistical estimations derived from them, such as hypothesis tests or classification errors. One practical limitation of the method is the estimation of the best trade-offs γ between sparsity and fidelity-to-data and σ^2 between regularity and fidelity-to-data. These parameters could be estimated in a Bayesian framework by adding a Laplace prior for the former and a Gaussian prior in the latter in the spirit of Allassonnière et al (2007, 2010).

7 Acknowledgments

We would like to thank Timothy O'Keefe and Paul Sanders for their kind proofreading of the manuscript. This work has been supported by ANR grant IRMGroup and NIH grants NIBIB (5R01EB007688) and NCRR (2P41 RR0112553-12).

A Proof of Proposition 1

Let $\delta\mathbf{S}_0$ be a small perturbation of the deformation parameters. This perturbation induces a perturbation of the system of particles $\delta\mathbf{S}(t)$, which induces a perturbation of the position of the pixels mapped back by the inverse deformation $\delta\mathbf{y}(0)$, which in turn induces a perturbation of the criterion δE:

$$\delta E = \left(\nabla_{\mathbf{y}(0)}A\right)^t \delta\mathbf{y}(0) + \left(\nabla_{\mathbf{S}_0}L\right)^t \delta\mathbf{S}_0. \tag{40}$$

According to (21), the perturbations of the state of the system of particles $\delta\mathbf{S}(t)$ and the pixel positions $\delta\mathbf{y}(t)$ satisfy the linearized ODEs:

$$\begin{array}{ll} \dot{\delta\mathbf{S}}(t) = d_{\mathbf{S}(t)}F\delta\mathbf{S}(t) & \delta\mathbf{S}(0) = \delta\mathbf{S}_0 \\ \dot{\delta\mathbf{y}}(t) = \partial_1 G\delta\mathbf{y}(t) + \partial_2 G\delta\mathbf{S}(t) & \delta\mathbf{y}(1) = 0 \end{array}.$$

The first ODE is linear. Its solution is given by:

$$\delta\mathbf{S}(t) = \exp\left(\int_0^t d_{\mathbf{S}(u)}F du\right)\delta\mathbf{S}_0. \tag{41}$$

The second ODE is linear with source term. Its solution is given by:

$$\delta\mathbf{y}(0) = -\int_0^1 \exp\left(-\int_0^s \partial_1 G(u)du\right)\partial_2 G(s)\delta\mathbf{S}(s)ds. \tag{42}$$

Plugging (41) into (42) and then into (40) leads to:

$$\nabla_{\mathbf{S}_0}E = -\int_0^1 {R_{0s}}^t \partial_2 G(s)^t {V_{s0}}^t \nabla_{\mathbf{y}(0)}A ds + \nabla_{\mathbf{S}_0}L, \tag{43}$$

where $R_{st} = \exp\left(\int_s^t d_{\mathbf{S}(u)}F du\right)$ and $V_{st} = \exp\left(-\int_s^t \partial_1 G(u)du\right)$.

Let us denote $\eta(s) = -{V_{s0}}^t\nabla_{\mathbf{y}(0)}A$, $g(s) = \partial_2 G(s)^t \eta(s)$ and $\xi(t) = \int_t^1 {R_{0s}}^t g(s)ds$, so that the gradient (43) can be re-written as:

$$\nabla_{\mathbf{S}_0}E = \int_0^1 {R_{0s}}^t g(s)ds + \nabla_{\mathbf{S}_0}L = \xi(0) + \nabla_{\mathbf{S}_0}L.$$

Now, we need to make explicit the computation of the auxiliary variables $\eta(t)$ and $\xi(t)$. By definition of V_{s0}, we have $V_{00} = \mathrm{Id}$ and $dV_{s0}/ds = V_{s0}\partial_1 G(s)$, which implies that $\eta(0) = -\nabla_{\mathbf{y}(0)}A$ and $\dot{\eta}(t) = -\partial_1 G(t)^t \eta(t)$.

For $\xi(t)$, we notice that $R_{ts} = \mathrm{Id} - \int_t^s \frac{dR_{us}}{du}du = \mathrm{Id} + \int_t^s R_{us}d_{\mathbf{S}(u)}F(u)du$. Therefore, using Fubini's theorem, we get:

$$\begin{aligned} \xi(t) &= \int_t^1 {R_{ts}}^t g(s)ds \\ &= \int_t^1 g(s) + d_{\mathbf{S}(s)}F^t \int_s^1 {R_{su}}^t g(u)duds \\ &= \int_t^1 g(s) + d_{\mathbf{S}(s)}F^t \xi(s)ds. \end{aligned}$$

This last equation is nothing but the integral form of the ODE given in (24).

B Lemma: soft thresholding in $\mathbb{R}^N$

Lemma 2 (Soft-thresholding in $\mathbb{R}^N$) *Let X and $X_0 \neq 0$ two vectors in $\mathbb{R}^N$ and F the criterion:*

$$F(X) = \|X\| + \frac{1}{2\beta}\|X - X_0\|^2.$$

Then F achieves its minimum for

$$X = S_\beta\left(\|X_0\|\right)\frac{X_0}{\|X_0\|},$$

where $S_\beta(x) = \max(x-\beta,0) - \min(x+\beta,0)$.

Proof If $X \neq 0$, F is differentiable and $\nabla_X F = \frac{X}{\|X\|} + (X - X_0)/\beta$. This shows that if the minimum of F is reached for $X \neq 0$ then X is parallel to X_0 and we can write $X = \lambda X_0/\|X_0\|$ where λ satisfies:

$$\lambda/|\lambda| + (\lambda - \|X_0\|)/\beta = 0.$$

If $\lambda > 0$, then the minimum is reached for $\lambda_+ = \|X_0\| - \beta$, which occurs only if $\|X_0\| > \beta$. If $\lambda < 0$, then the minimum is reached for $\lambda_- = \|X_0\| + \beta$, which occurs only if $\|X_0\| < -\beta$. In the other cases, namely $\|X_0\| \in [-\beta,\beta]$, $X = 0$.

Since $F(\lambda_+ X_0/\|X_0\|) - F(0) < 0$, then the minimum of F is reached for $X = (\|X_0\| - \beta)X_0/\|X_0\|$ if $\|X_0\| > \beta$. Since $F(\lambda_- X_0/\|X_0\|) - F(0) > 0$, then the minimum of F is reached for $X = (\|X_0\| + \beta)X_0/\|X_0\|$ if $\|X_0\| < -\beta$. If $\|X_0\| \in [-\beta,\beta]$, the minimum of F is reached for $X = 0$. These three cases are combined in a single expression using the soft-thresholding function S_β.

□

References

Allassonnière S, Trouvé A, Younes L (2005) Geodesic shooting and diffeomorphic matching via textured meshes. In: Proc. of EMM-CVPR, pp 365–381

Allassonnière S, Amit Y, Trouvé A (2007) Towards a coherent statistical framework for dense deformable template estimation. Journal of the Royal Statistical Society Series B 69(1):3–29

Allassonnière S, Kuhn E, Trouvé A (2010) Construction of bayesian deformable models via a stochastic approximation algorithm: A convergence study. Bernoulli Journal 16(3):641–678

Arsigny V, Commowick O, Pennec X, Ayache N (2006) A log-euclidean framework for statistics on diffeomorphisms. In: Proc. MICCAI, Springer, no. 4190 in LNCS, pp 924–931

Beck A, Teboulle M (2009) A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences 2(1):183–202

Beg MF, Miller MI, Trouvé A, Younes L (2005) Computing large deformation metric mappings via geodesic flows of diffeomorphisms. IJCV 61:139–157

Cootes TF, Edwards GJ, Taylor CJ (2001) Active appearance models. Trans Pattern Anal and Machine Intell 23(6):681–685

Dupuis P, Grenander U, Miller M (1998) Variational problems on flows of diffeomorphisms for image matching. Quaterly of Applied Mathematics 56(3):587–600

Durrleman S (2010) Statistical models of currents for measuring the variability of anatomical curves, surfaces and their evolution. Thèse de sciences (phd thesis), Université de Nice-Sophia Antipolis

Durrleman S, Pennec X, Trouvé A, Ayache N (2009) Statistical models of sets of curves and surfaces based on currents. Medical Image Analysis 13(5):793–808

Durrleman S, Fillard P, Pennec X, Trouvé A, Ayache N (2011a) Registration, atlas estimation and variability analysis of white matter fiber bundles modeled as currents. NeuroImage 55(3):1073 – 1090

Durrleman S, Prastawa M, Gerig G, Joshi S (2011b) Optimal data-driven sparse parameterization of diffeomorphisms for population analysis. In: Székely G, Hahn H (eds) Information Processing in Medical Imaging (IPMI), LNCS, vol 6801, pp 123–134

Glasbey CA, Mardia KV (2001) A penalised likelihood approach to image warping. Journal of the Royal Statistical Society, Series B 63:465–492

Glaunès J, Qiu A, Miller M, Younes L (2008) Large deformation diffeomorphic metric curve mapping. International Journal of Computer Vision 80(3):317–336, DOI 10.1007/s11263-008-0141-9

Grenander U (1994) General Pattern Theory: a Mathematical Theory of Regular Structures. Oxford University Press

Grenander U, Miller MI (1998) Computational anatomy: An emerging discipline. Quarterly of Applied Mathematics LVI(4):617–694

Grenander U, Srivastava A, Saini S (2007) A pattern-theoretic characterization of biological growth. Transactions on Medical Imaging 26(5):648–659

Hansen MS, Larsen R, Glocker B, Navab N (2008) Adaptive parametrization of multivariate b-splines for image registration. In: Computer Vision and Pattern Recognition, IEEE, pp 1–8

Hastie T, Tibshirani R, Friedman J (2009) The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, second Edition

Joshi S, Miller M (2000) Landmark matching via large deformation diffeomorphisms. IEEE Transactions on Image Processing 9(8):1357–1370

Lei W, Beg F, Ratnanather T, Ceritoglu C, Younes L, Morris J, Csernansky J, Miller M (2007) Large deformation diffeomorphism and momentum based hippocampal shape discrimination in dementia of the alzheimer type. IEEE Trans on Medical Imaging 26:462–470

Lorenzen P, Davis B, Joshi SC (2005) Unbiased atlas formation via large deformations metric mapping. In: Medical Image Computing and Computer-Assisted Intervention-MICCAI, Springer, Lecture Notes in Computer Science, vol 3750, pp 411–418

Marsland S, McLachlan RI (2007) A Hamiltonian particle method for diffeomorphic image registration. In: Proceedings of Information Processing in Medical Imaging (IPMI), Springer, LNCS, vol 4548, pp 396–407

Meyer Y (2001) Oscillating patterns in image processing and nonlinear evolution equations, University Lecture Series, vol 22. American Mathematical Society, Providence, RI, the fifteenth Dean Jacqueline B. Lewis memorial lectures

Miller I M, Trouvé A, Younes L (2002) On the metrics and euler-lagrange equations of computational anatomy. Annual Review of Biomedical Engineering 4:375–405

Miller M, Younes L (2001) Group actions, homeomorphisms, and matching: A general framework. International Journal of Computer Vision 41:61–84

Miller M, Trouvé A, Younes L (2006) Geodesic shooting for computational anatomy. Journal of Mathematical Imaging and Vision 24(2):209–228

Nesterov YE (1983) A method of solving a convex programming problem with convergence rate $o(1/k^2)$. Soviet Math Dokl 27(2), translation by A. Rosa

Pennec X, Fillard P, Ayache N (2006) A Riemannian framework for tensor computing. International Journal of Computer Vision 66(1):41–66

Risser L, Vialard FX, Wolz R, Murgasova M, Holm DD, Rueckert D (2011) Simultaneous multi-scale registration using large deformation diffeomorphic metric mapping. Trans Med Imaging (30):1746–1759

Rueckert D, Aljabar P, Heckemann RA, Hajnal J, Hammers A (2006) Diffeomorphic Registration using B-Splines. In: Proc. MICCAI, pp 702 – 709

Singh N, Fletcher P, Preston J, Ha L, King R, Marron J, Wiener M, Joshi S (2010) Multivariate statistical analysis of deformation momenta relating anatomical shape to neuropsychological measures. In: Jiang T, Navab N, Pluim J, Viergever M (eds) Proc. MICCAI'10, Springer, Lecture Notes in Computer Science, vol 6363, pp 529–537

Sommer S, Nielsen M, Darkner S, Pennec X (2012a) Higher order kernels and locally affine lddmm registration ArXiv:1112.3166v1

Sommer S, Nielsen M, Pennec X (2012b) Sparsity and Scale: Compact Representations of Deformation for Diffeomorphic Registration. In: IEEE Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA 2012), Breckenridge, Colorado, USA, URL `http://hal.inria.fr/hal-00641357/en/`

Trouvé A (1998) Diffeomorphisms groups and pattern matching in image analysis. International Journal of Computer Vision 28(3):213–221

Trouvé A, Younes L (2005) Metamorphoses through lie group action. Foundations of Computational Mathematics 5(2):173–198

Vaillant M, Glaunès J (2005) Surface matching via currents. In: Proceedings of Information Processing in Medical Imaging, Springer, Lecture Notes in Computer Science, vol 3565, pp 381–392

Vaillant M, Miller M, Younes L, Trouvé A (2004) Statistics on diffeomorphisms via tangent space representations. NeuroImage 23:161–169

Vercauteren T, Pennec X, Perchant A, Ayache N (2009) Diffeomorphic demons: Efficient non-parametric image registration. NeuroImage 45(1, Supp.1):S61–S72, DOI 10.1016/j.neuroimage.2008.10.040

Yu G, Sapiro G, Mallat S (2010) Image modeling and enhancement via structured sparse model selection. In: Proceedings of the International Conference on Image Processing (ICIP), IEEE, pp 1641–1644

Zou H, Hastie T (2005) Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society, Series B 67(2):301–320

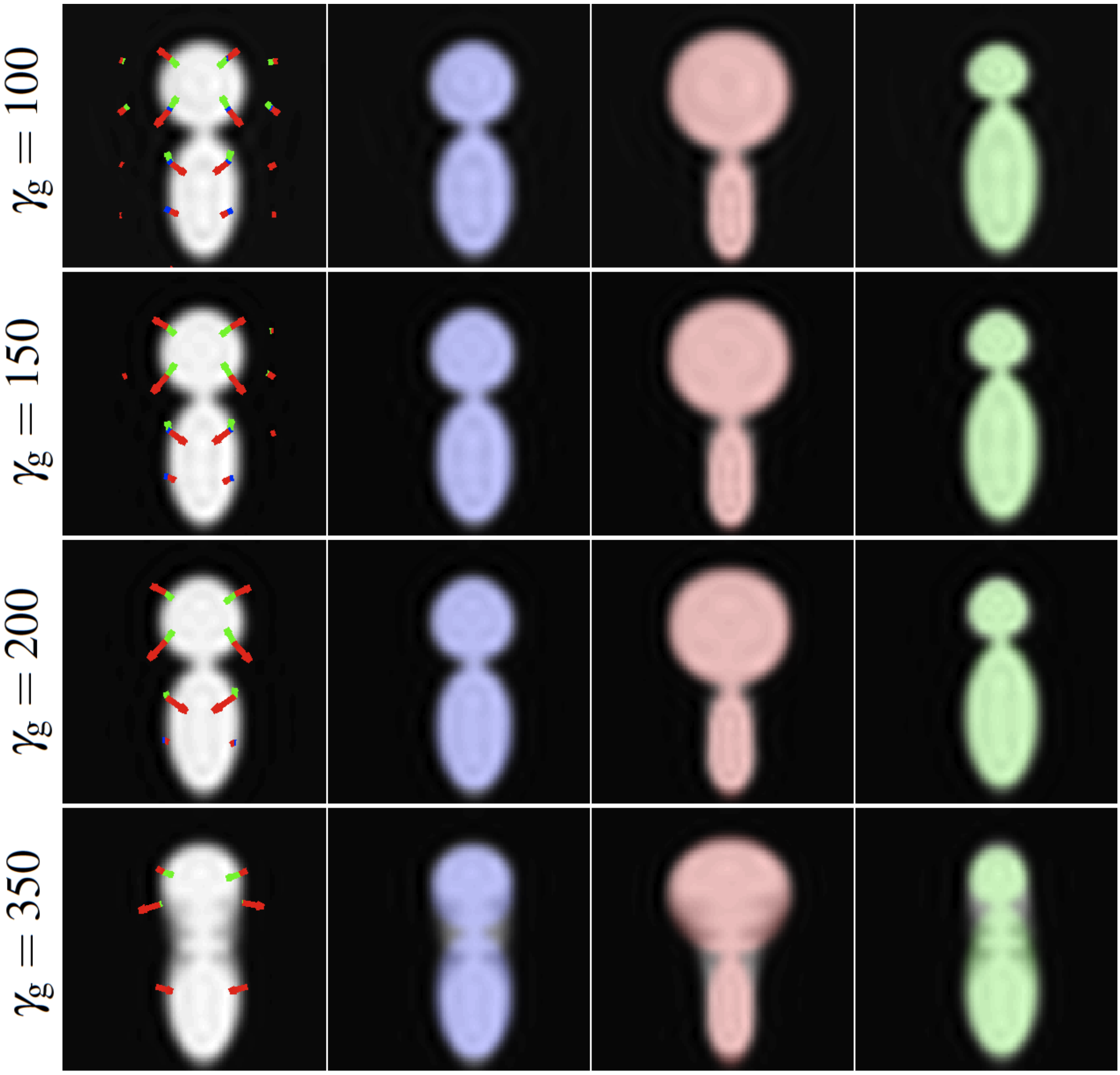


Fig. 4 Impact of the geometric sparsity prior γ_g on the atlas estimation. The larger γ_g, the more penalized the initial momentum, the smaller the number of active control points, the less sharp the atlas. If the sparsity penalty term is too strong, then the template-to-subject matchings are not accurate (large residual errors), which eventually affects the sharpness of the template image. From the initial grid of 5x5 control points, 16 were selected for $\gamma_g = 100$ (the closest control points to the edge of the image has been pruned), 8 for $\gamma_g = 200$ and 6 for $\gamma_g = 350$.

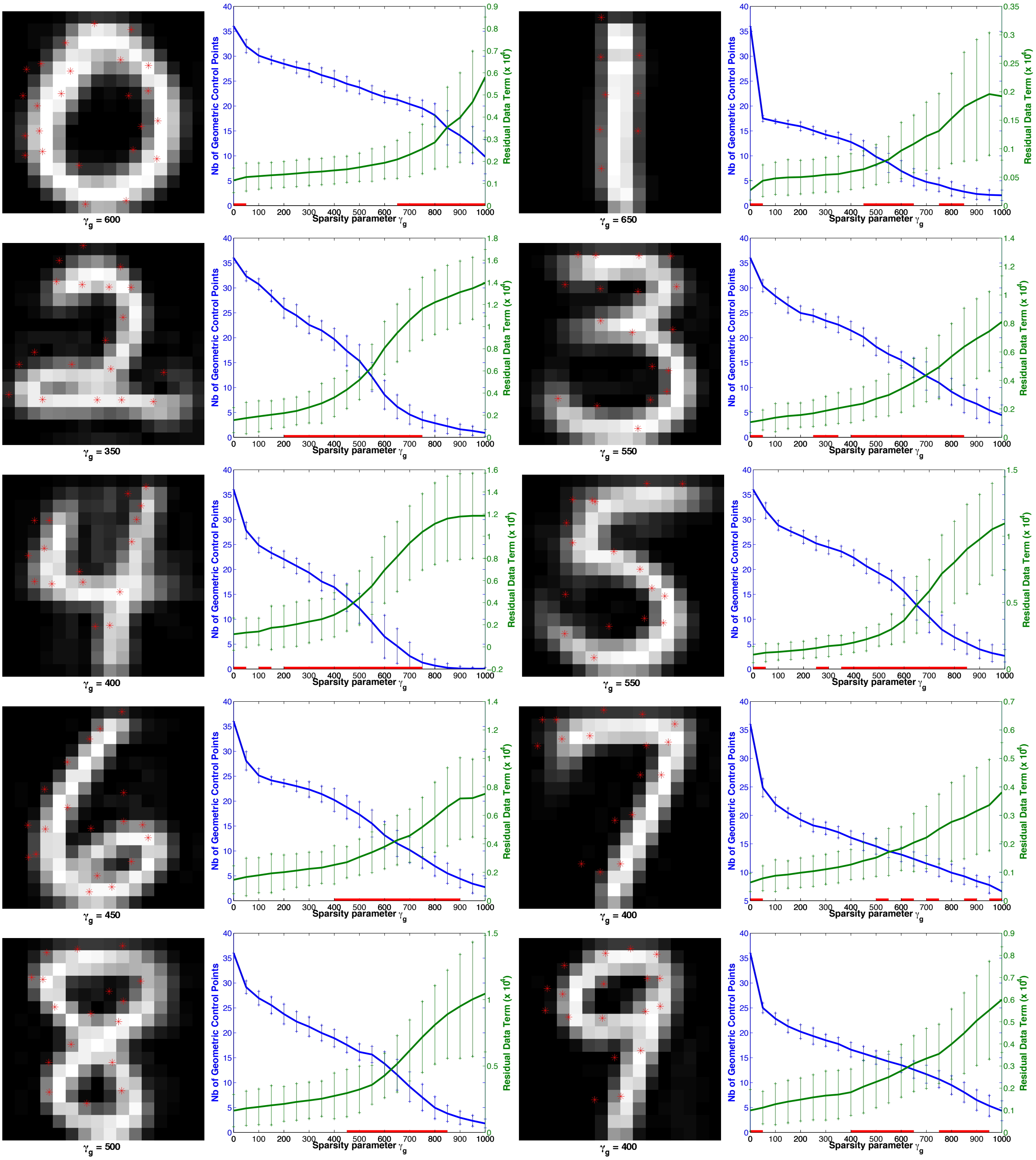
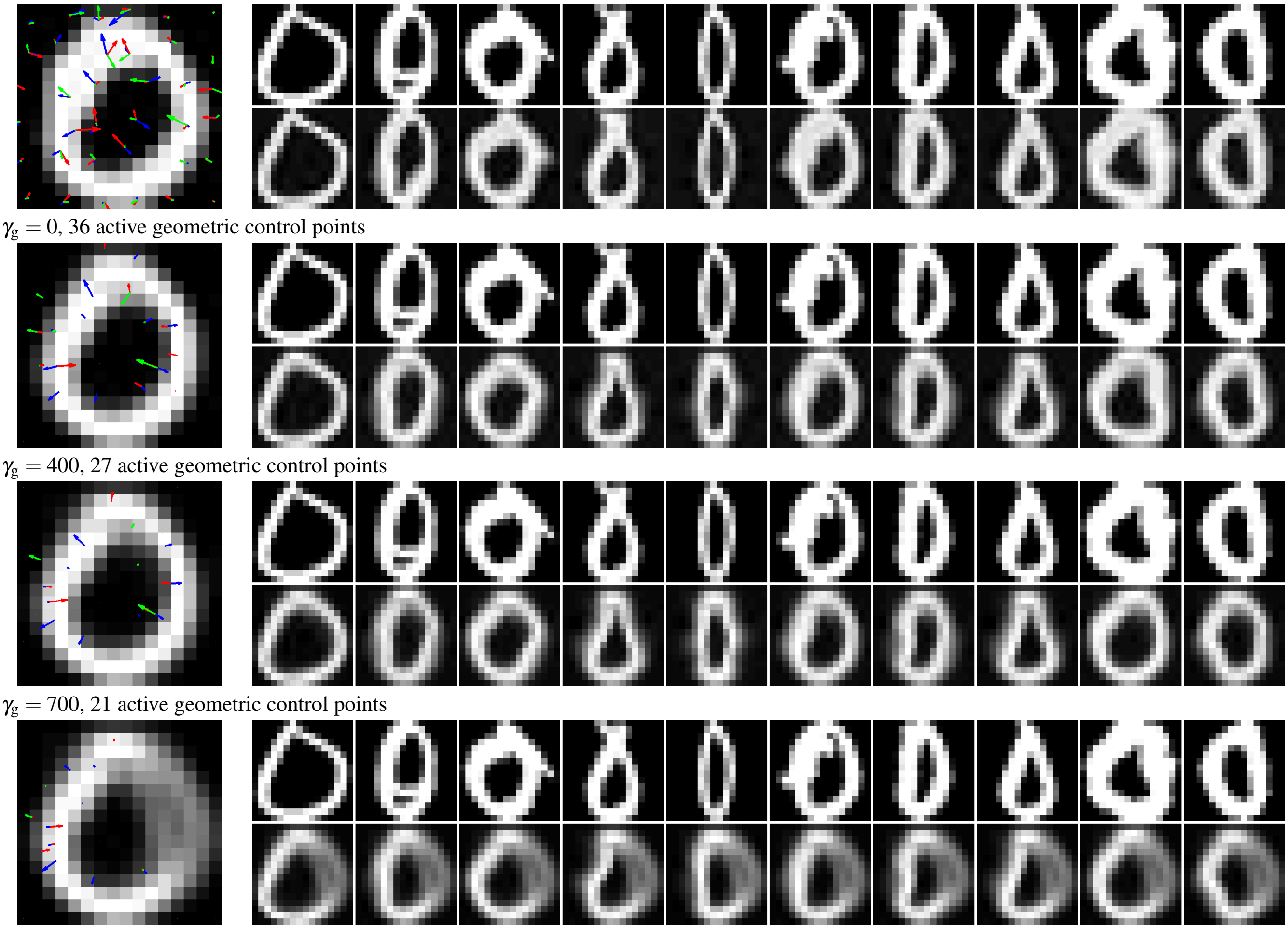


Fig. 6 Atlas of digits from the US postal database. Blue curves plot the number of geometric control points versus the sparsity prior γ_g. Mean and standard deviation is indicated when randomizing the training dataset of 20 images (26 training sets without intersection). Green curves plot the residual data term measured when registering the atlas to one test sample. Mean and standard deviation is shown for 260 of such tests for each value of the sparsity parameter γ_g. This shows that the atlas sharpness decreases with the dimension of its parameterization while the sparsity prior is increased. The shape of the green curves (a plateau phase followed by rapid increase) suggests that there is an optimal value of the sparsity parameter γ_g where the dimension of the atlas could be reduced without sacrificing much of the atlas sharpness. The red intervals indicate when the residual data term is significantly increased between two consecutive values of the sparsity parameter (Wilcoxon test with p-value $< 1\%$) The left panels shows a selected template image for a given value of the sparsity parameter along with the position of the geometric control points (red asterisks).



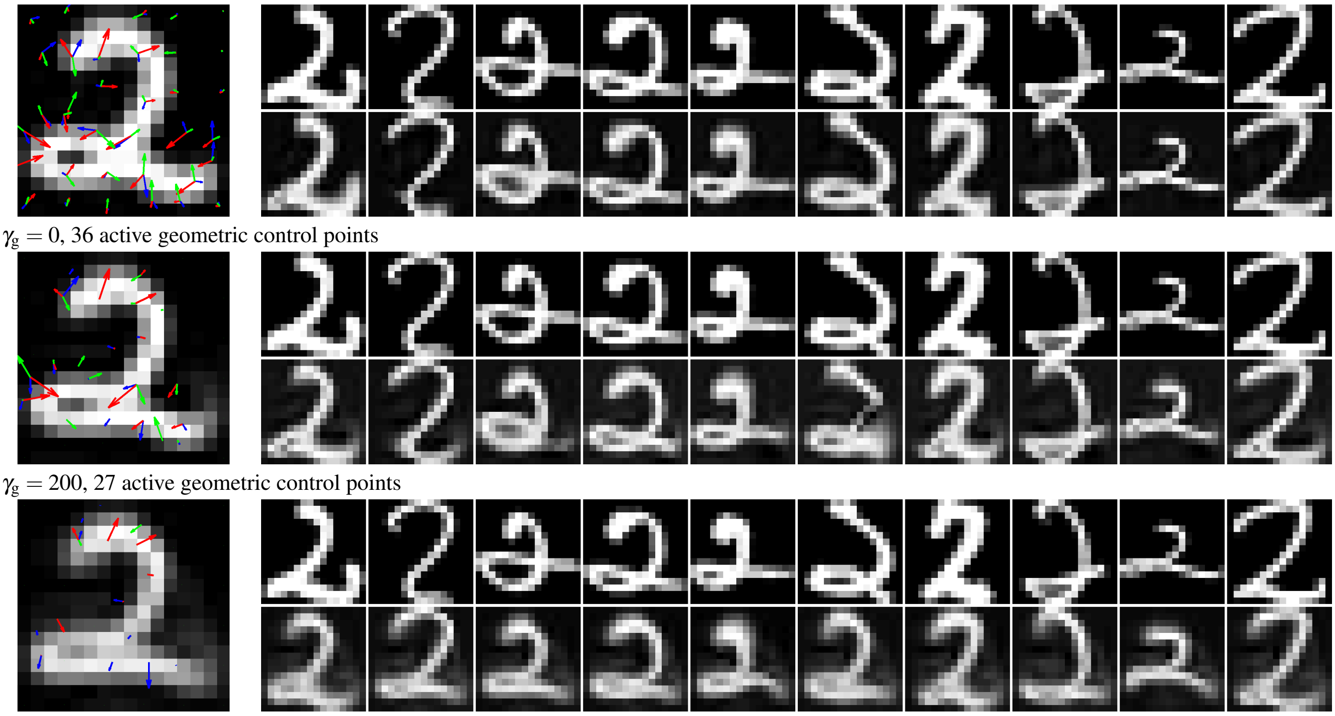
$\gamma_g = 0$, 36 active geometric control points

$\gamma_g = 400$, 27 active geometric control points

$\gamma_g = 700$, 21 active geometric control points

$\gamma_g = 800$, 13 active geometric control points

Fig. 7 Atlas of digit ‘0’ for different values of the sparsity parameter γ_g. Left: the template image with the set of momentum vectors of the first three images in the training data set superimposed (in red, green and blue). Right, top row: the first ten training images (among 20); bottom row: the template image warped to each training image using the sparse parameterization. The more degrees of freedom in the deformation parameterization, the more variations in shape the deformations capture.



$\gamma_g = 0$, 36 active geometric control points

$\gamma_g = 200$, 27 active geometric control points

$\gamma_g = 400$, 18 active geometric control points

Fig. 8 Atlas of digit ‘2’ for different values of the sparsity parameter γ_g. Left: the template image with the set of momentum vectors of the first three images in the training data set superimposed (in red, green and blue). Right, top row: the first ten training images (among 20); bottom row: the template image warped to each training image using the sparse parameterization. The more degrees of freedom in the deformation parameterization, the more variations in shape the deformations capture. At the intermediate level ($\gamma_g = 200$, middle row), some features are captured (presence or absence of the loop, curvature of the shape) while others are missed (extremities of the digit are not matched as accurately as in the first row). In the last row, the limited number of degrees of freedom allows to only match the height and width of the shape. As a consequence, the deformed template looks closer to the template image rather than the observations and the template image is blurrier.

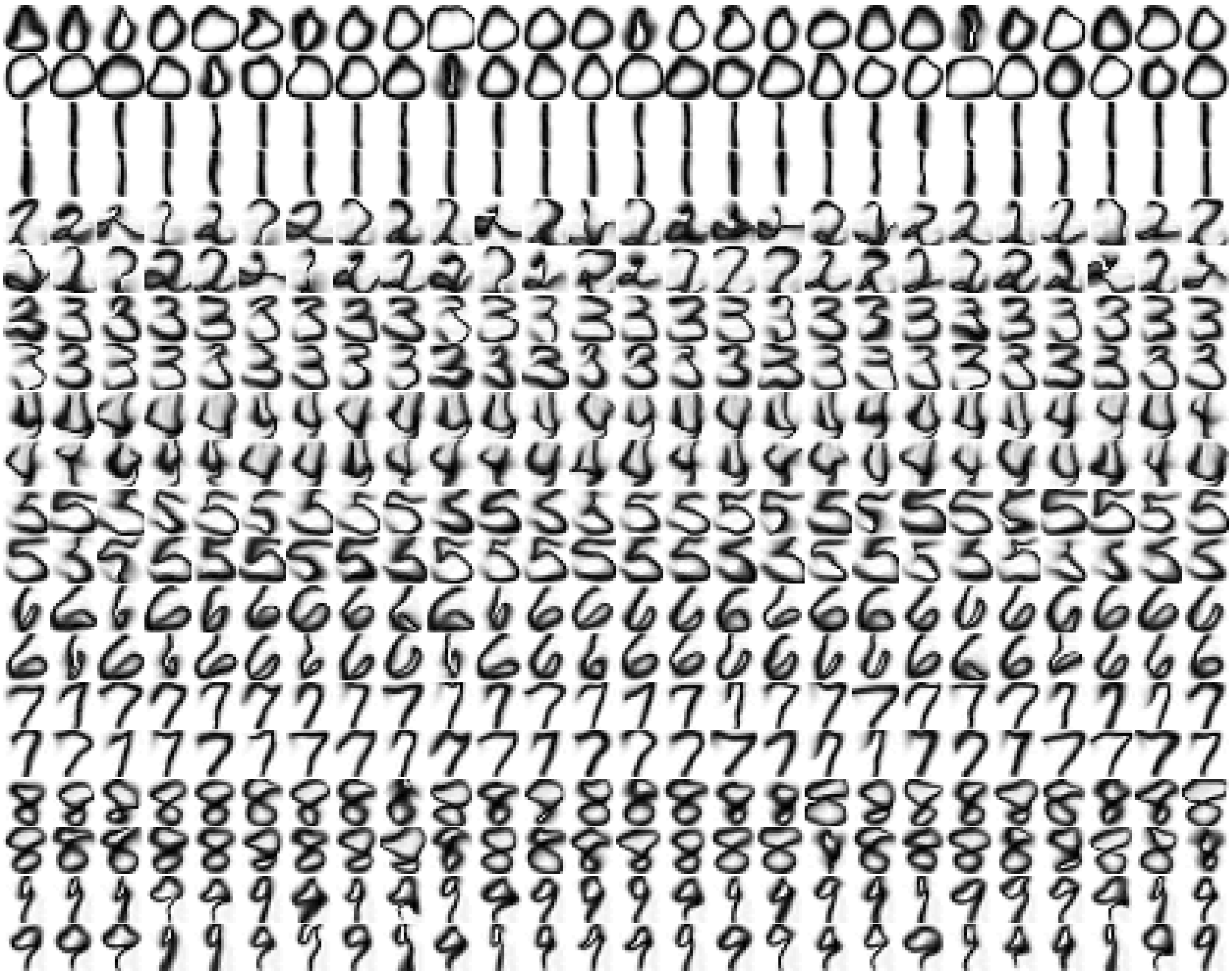


Fig. 9 For each digit, we simulate new images based on the atlas that has been selected in Fig. 6 (left panels). Deformations of the template were generated based on the Principal Component Analysis of the initial momentum vectors. For each digit, simulations on both opposite directions are shown in the first and second row (see text for details). The variety of the simulated shapes shows that the atlas was able to capture most of the variability in the training image set.