Learning low-dimensional representations of shape data sets with diffeomorphic autoencoders

Alexandre Bône, Maxime Louis, Olivier Colliot, Stanley Durrleman

Learning low-dimensional representations of shape data sets with diffeomorphic autoencoders

Alexandre Bône, Maxime Louis, Olivier Colliot, Stanley Durrleman, and the Alzheimer's Disease Neuroimaging Initiative

ARAMIS Lab, ICM, Inserm U1127, CNRS UMR 7225, Sorbonne University, Inria, Paris, France
{alexandre.bone, stanley.durrleman}@icm-institute.org

Abstract. Contemporary deformation-based morphometry offers parametric classes of diffeomorphisms that can be searched to compute the optimal transformation that warps a shape into another, thus defining a similarity metric for shape objects. Extending such classes to capture the geometrical variability in always more varied statistical situations represents an active research topic. This quest for genericity however leads to computationally-intensive estimation problems. Instead, we propose in this work to learn the best-adapted class of diffeomorphisms along with its parametrization, for a shape data set of interest. Optimization is carried out with an auto-encoding variational inference approach, offering in turn a coherent model-estimator pair that we name diffeomorphic auto-encoder. The main contributions are: (i) an original network-based method to construct diffeomorphisms, (ii) a current-splatting layer that allows neural network architectures to process meshes, (iii) illustrations on simulated and real data sets that show differences in the learned statistical distributions of shapes when compared to a standard approach.

1 Introduction

Medical imaging represents a unique challenge for statisticians: massive amounts of high-resolution data conceal high-stake information that, if correctly processed, could help describe and understand pathological conditions at the population level, or classify and predict clinical status at the individual level. In the case of anatomical imaging, information lies in the geometry of the imaged structures. When faced with such a data set, the most basic statistical questions are then: what is the typical geometry? How much does this specific individual deviate from this average?

Summarizing a data set of shapes in those terms consist in performing an adapted mean-variance analysis, that respects the intrinsic data structure. Pioneered two centuries ago by D'Arcy Thomson [12], deformation-based morphometry quantifies differences between shape objects – such as images or extracted meshes – via ambient-space deformations that warp one into the other. Contemporary approaches construct non-linear smooth invertible deformations, diffeomorphisms, by following streamlines of "velocity" vector fields which can

be either static (stationary velocity fields theory, SVF) [14] or dynamic (large deformation diffeomorphic metric mapping, LDDMM) [1, 9]. In any case, those approaches define large parametric classes of diffeomorphisms, which can be searched to compute the optimal transformation that warps a shape as close as possible to some target. The intensity of this deformation can then be used as a proxy to define a similarity metric, and finally learn the induced Fréchet mean shape and the associated variance [5, 11]

Recent efforts focused on proposing new parametric classes of diffeomorphisms. In [4], the authors propose a variation of the LDDMM construction where the parametrization of the diffeomorphisms is independent of the shapes on which they act, allowing unified handling of meshes – with or without point correspondence – and images. Even more recently, [6] generalizes the LDDMM framework by defining an extended class of diffeomorphisms parametrized by "modules" which encode local translations, scalings, or rotations. However, finding the structure of the deformations that will best capture shape variability is a very difficult task in practice, and learning it from the data often leads to intractable optimization problems. Coming from the deep learning research horizon, more and more contributions propose to change the optimization task into a prediction one: the optimal parameters coding for the desired diffeomorphism are directly predicted by a deep network, after its supervised or unsupervised training [2, 15]. The used deformation models are either SVF or LDDMM-based i.e. well-established generic approaches, fixed throughout the learning procedure.

This work proposes to *learn* the best-adapted class of diffeomorphisms along with its parametrization for a shape data set of interest, thanks to a network-based deformation method. Optimization is carried out with a auto-encoding variational inference approach [8], offering in turn a coherent model-estimator pair that we name "diffeomorphic auto-encoder" (DAE). The main contributions of this paper are: (i) an original method to construct diffeomorphisms by integrating dynamic velocity fields which are defined as the image of segments of $\mathbb{R}^n$ by a neural network; (ii) the introduction of the current-splatting layer that allows a network architecture to process mesh objects; (iii) the provided illustrations on both simulated and real data sets that show differences in the learned statistical distributions of shapes, when compared with a more standard LDDMM approach. A special care has been given to the scalability and versatility of the proposed method, which is designed to tackle statistical inference problems on high-dimensional shape data sets, with few requirements about the data structure. Section 2 details the method for constructing diffeomorphisms; Section 3 introduces the statistical atlas model; Section 4 presents the variational inference algorithm used for estimation. Section 5 gives experimental results, Section 6 discusses perspectives and concludes.

2 Deformation mechanics

Similarly to LDDMM, diffeomorphisms are constructed by integrating dynamic "velocity" fields over a unit interval. Those velocity fields path are taken as the

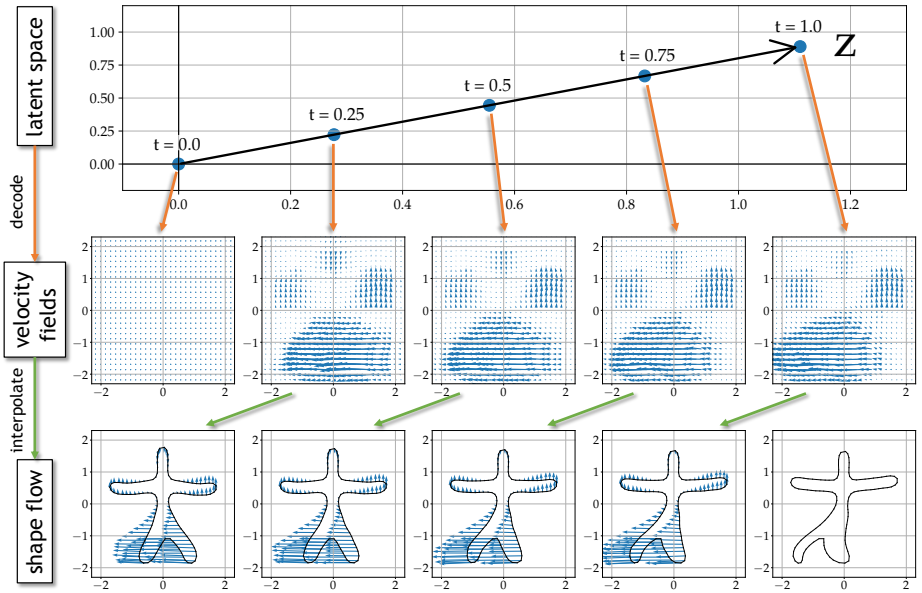

Fig. 1. Deformation mechanics: diffeomorphism are obtained by following the streamlines of dynamic velocity fields, which are themselves defined as the image of latent-space segments by a non-linear mapping represented in practice by a network. The parameters of this decoding mapping will be optimized for each new application, therefore adapting the core deformation mechanics to the considered data set.

image of an abstract "latent-space" segment through a neural network. Once learned, the parameters of this "decoding" neural network determine a non-linear parametrization of the obtained diffeomorphisms by the "latent" representations.

Let Ω be an open and bounded set of the ambient space $\mathbb{R}^d$ with $d \in \{2, 3\}$. Let $n, s \in \mathbb{N}^\star$ and $D_\theta : \mathbb{R}^n \to C_0^s(\Omega, \mathbb{R}^d)$ an infinitely differentiable mapping that associate to any $z \in \mathbb{R}^n$ a s-smooth vanishing vector field v on Ω. This mapping is called "decoder", and is taken under the form of a neural network with parameters θ. For the rest of this paper, the decoder D_θ is structured with three fully connected layers followed by four deconvolutional layers, with *tanh* activation functions for all layers except the last one. For any $z \in \mathbb{R}^n$, assuming that the path $t \in [0, 1] \to v_t = D_\theta(z \cdot t)$ is s-absolutely integrable, i.e. that $\int_0^1 \|v_t\|_{s,\infty} < \infty$ with $\|v\|_{s,\infty} = \sum_{k=0}^{s} \left\|\nabla^k v\right\|_\infty$, implies that there exist an unique flow of diffeomorphisms $t \to \phi_t$ such that $\partial_t \phi_t = v_t \circ \phi_t$ and $\phi_0 = \mathrm{Id}_\Omega$ [16]. For such paths, we note $\Phi_\theta : z \to \phi_1$ the mapping that associates the diffeomorphism reached at unit time when integrating the "velocity" vector field path decoded from the segment $t \to z \cdot t$. The integrated vector fields are called "velocity" fields by analogy with fluid mechanics, where particles $x \in \mathbb{R}^d$ follow the streamlines of a dynamic flow $t \to v_t$. Under integrability conditions of this flow, we have defined a θ-parametric class of diffeomorphisms, indexed by the Euclidian vector space

$\mathbb{R}^n$ which we call the "latent" space. In practice, the integrability condition will be explicitely enforced by adding a dedicated regularity term to the optimized loss function, with the introduction of a corresponding Lagrange multiplier λ.

Figure 1 illustrates the discrete version of those deformation mechanics: a single latent-space parameter $z \in \mathbb{R}^n$ with n typically small (here $n = 2$) encodes for a flow of diffeomorphisms of the ambient space $\mathbb{R}^d$ (here $d = 2$) that can in turn deform any shape object. A fixed number T of uniformly distributed samples of the latent-space segment $t \to z \cdot t$ are decoded by the same neural network D_θ into a set of T corresponding velocity fields discretized on a fixed and regular "deformation" grid G_d. Those velocity fields are then successively linearly interpolated on the shape to deform, and integrated according to a forward Euler scheme. We further impose that all layers of the decoder are without bias, ensuring that $\Phi_\theta(0_{\mathbb{R}^n}) = \mathrm{Id}_\Omega$. Note finally that D_θ is infinitely differentiable, enforcing some temporal smoothness of the decoded velocity fields $t \to D_\theta(z \cdot t)$.

3 Atlas model

3.1 Generative statistical model

Let $y = (y_i)_i$ be a data set of N shapes. For $i = 1, ..., N$, we model the observations y_i as a random deformation of a template shape y_0:

$$y_i \overset{\text{iid}}{\sim} \mathcal{N}_\mathcal{E}\left(\Phi_\theta(z_i) \cdot y_0, \sigma_\epsilon^2\right) \quad \text{with} \quad z_i \overset{\text{iid}}{\sim} \mathcal{N}(0, I_n) \tag{1}$$

under the constraint that the $\Phi_\theta(z_i)$ are diffeomorphisms. The latent individual variables $z_i \in \mathbb{R}^n$ encode the deformations that warp the template y_0 into each observed shape y_i. Note that the template is encoded by $z_0 = 0$, by construction.

Scalability and versatility concerns are at the core of the proposed method: note that no particular assumption on the nature of shapes has been made so far. The density function of the "normal" distribution $\mathcal{N}_\mathcal{E}$ assumed on the observed y_i can be generically noted: $p(y_i|z_i; \theta, y_0) \propto \exp\left(-d_\mathcal{E}[y_i, \Phi_\theta(z_i) \cdot y_0]^2/2 \cdot \sigma_\epsilon^2\right)$ where $d_\mathcal{E}$ is an extrinsic distance measure on shapes. If the considered shapes are images – of fixed dimension – or meshes with point-to-point correspondence, the simple ℓ^2 metric is a natural and convenient choice: $d_\mathcal{E}(y^\alpha, y^\beta)^2 = \left\|y^\beta - y^\alpha\right\|_{\ell^2}^2$. In the case of mesh data without point correspondence, the current [13] representation can be used to construct a well-defined distance metric between shapes, at the expense of the characteristic scale hyper-parameter $\sigma_\mathcal{E}$. Noting respectively $(c_k)_{k=1,...,K}$ and $(n_k)_{k=1,...,K}$ the centers and normals of the segments or triangles forming the connectivity of the manipulated meshes, we then define:

$$d_\mathcal{E}(y^\alpha, y^\beta)^2 = \sum_{k=1}^{K^\alpha} \sum_{l=1}^{K^\beta} \exp\left[-\left\|c_l^\beta - c_k^\alpha\right\|^2 / \sigma_\mathcal{E}^2\right] \cdot (n_k^\alpha)^\top \cdot n_l^\beta. \tag{2}$$

3.2 Comparison with LDDMM-based approaches

From a generative point of view, the proposed model associates to any latent-space z_i a deformation and, in turn, a shape. From a learning perspective, esti-

mating the shared parameters θ and y_0 learns a new n-dimensional representation of shapes. This global approach could straightforwardly be followed within the already-established LDDMM framework. Using intuitive notations, LDDMM diffeomorphisms could be noted $\Phi(m_i)$ where the "momentum" parameter m_i is of imposed dimensions – typically large. Note that the mapping Φ is not indexed by some θ: deformation mechanics are fixed. In order to represent the geometrical variability in a more compact way, the momenta can be constrained to span a vector space of chosen dimension by specifying $m_i = A \cdot z_i$ where $z_i \sim \mathcal{N}(0, I_n)$ is a n-dimensional vector and A is a matrix parameter to learn. This model is named "principal geodesic analysis" (PGA) in [17], in reference to the PCA-like prior on the momenta covariance. Our approach goes a step further by breaking the linear relationship between the latent-space representations and the associated velocity fields. The learned representations when introducing a non-linear network are evaluated in Section 5 by comparison with the PGA approach.

4 Network-based variational inference

4.1 Rationale

Our goal is to estimate both the template shape y_0 and the parameters θ of the decoder which parametrize the geometry of the learned n-dimensional manifold of deformations, under the constraint of diffeomorphic $\Phi_\theta(z_i)$. In the ideal case, we would also like to determine the posterior distribution $p(z_i|y_i; \theta, y_0)$, which would give us low-dimensional latent-space representations of the individual registrations of y_i on y_0, in a probabilistic sense. Knowing this posterior would also allow to instantly register any new shape y_{N+1} to y_0. Being intractable, we approximate it by a parametric distribution $q(z_i|y_i; \vartheta)$ that we model as an uncorrelated Gaussian of $\mathbb{R}^n$. We estimate ϑ jointly with θ and y_0. We note E_ϑ the parametric "encoding" function that associates to any y_i the mean and diagonal covariance of the approximate posterior $q(z_i|y_i; \vartheta)$. This mapping is taken under the form of a neural network, composed of four convolution and one fully-connected layers, with *tanh* activation functions for all layers except the last.

This global optimization approach is know as variational Bayes [7]. The idea of introducing a network encoding function comes from [8]. Figure 2 presents the final "diffeomorphic auto-encoder" (DAE) model-estimator pair.

4.2 Encoding image and mesh shapes

In the case of image data, the encoding network E_ϑ acts on the pixel values directly. In the case of mesh objects, a preliminary "current-splatting" [3] operation is performed on a regular grid G_s in order to represent this mesh as a d-channels image, which is then fed to the encoder network (see Figure 2). With coherent notations with respect to equation (2), the d-channel splatting intensity $\mathcal{S}_{y_i}(x)$ at any physical location $x \in \mathbb{R}^d$ for the mesh y_i is given as:

$$\mathcal{S}_{y_i}(x) = \sum_{k=1}^{K} \exp\left[-\|x - c_k\|^2 / \sigma_{\mathcal{S}}^2\right] \cdot n_k \tag{3}$$

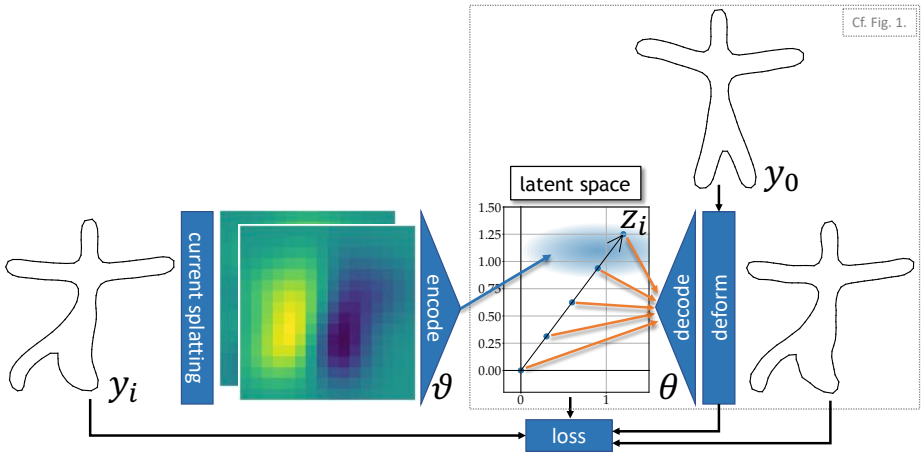

Fig. 2. Global architecture of the diffeomorphic auto-encoder (DAE). An observation y_i is encoded as a normal probability distribution, from which is sampled a latent representation $z_i \in \mathbb{R}^n$. The latent-space segment $[0, z_i]$ is then decoded into a dynamic velocity field, which is integrated into a diffeomorphism of the ambient space $\mathbb{R}^d$. This deformation is applied to a template shape y_0 to produce a reconstruction of the original shape y_i. The parameters of the encoder ϑ, of the decoder θ, and the template shape y_0 are estimated by stochastic gradient descent. To encode meshes, a preliminary current-splatting is performed before feeding y_i to the network.

where $\sigma_{\mathcal{S}}$ is a characteristic length hyper-parameter [3, 5].

4.3 Diffeomorphic constraint as a regularity term

As discussed in Section 2, preventing the integral over $[0, 1]$ of the s-Sobolev norm of the decoded velocity field paths $t \to D_\theta(z_i \cdot t)$ from going to infinity is enough to ensure diffeomorphic deformations $\Phi_\theta(z_i)$. This constraint is therefore simply transformed into a regularity term. Rather than penalizing the s-Sobolev norm, we choose to penalize an equivalent norm, introduced in [18]. For any $v, w \in D_\theta(\mathbb{R}^n)$, let $\langle ., . \rangle_S$ the Sobolev metric such that $\langle v, w \rangle_S = \int_\Omega S(v)^\top \cdot w$ with $S = (\mathrm{Id} - \alpha \cdot \Delta)^s$, noting $(.)^\top$ the transposition operator and $\Delta(.)$ the Laplacian one. S is a symmetric positive-definite differential operator when the scale parameter verifies $\alpha > 0$. We note $\|.\|_S$ the induced norm. For faster computation, this norm will in practice be evaluated in the Fourier domain (see [18]).

Introducing the Lagrange multiplier λ, we define the Sobolev regularity loss:

$$\mathcal{R}_s(\theta, \vartheta; y_i) = \lambda \cdot \int_{t \in [0,1]} \int_{z_i \in \mathbb{R}^n} \left\| D_\theta(z_i \cdot t) \right\|_S^2 \cdot q(z_i | y_i; \vartheta) \cdot dz_i \cdot dt \tag{4}$$

$$\approx \frac{\lambda}{T \cdot L} \sum_{t=1}^{T} \sum_{l=1}^{L} \left\| D\left(z_i^{(l)} \cdot \tfrac{t-1}{T-1}\right) \right\|_S^2 = \mathcal{R}'_s(\theta, \vartheta; y_i) \tag{5}$$

where $z_i^{(l)} \overset{\text{iid}}{\sim} q(.|y_i;\vartheta)$ for $l = 1, ..., L$, and T is the number of Euler time-steps.

4.4 Loss function

The loss writes $\mathcal{L}(y_i;\theta,\vartheta,y_0) = \mathcal{A}(y_i;\theta,\vartheta,y_0) + \mathcal{R}_{kl}(y_i;\theta,\vartheta) + \mathcal{R}_s(y_i;\theta,\vartheta)$, with:

$$\mathcal{A}(y_i;\theta,\vartheta,y_0) = -\int \log p(y_i|z_i;\theta,y_0)\cdot q(z_i|y_i;\vartheta)\cdot dz_i \approx -\frac{1}{L}\sum_{l=1}^{L} \log p(y_i|z_i^{(l)};\theta,y_0) \tag{6}$$

where $z_i^{(l)} \overset{\text{iid}}{\sim} q(.|y_i;\vartheta)$, and $\mathcal{R}_{kl}(y_i;\theta,\vartheta) = \mathrm{KL}\big[q(z_i|y_i;\vartheta) \,||\, p(z_i)\big]$, KL(.||.) denoting the Kullback-Leibler divergence operator.

Noting $\mathcal{A}'$ the Monte-Carlo approximation of the attachment term $\mathcal{A}$ given by equation (6), the discrete loss function that is actually minimized writes $\mathcal{L}'(y_i;\theta,\vartheta,y_0) = \mathcal{A}'(y_i;\theta,\vartheta,y_0) + \mathcal{R}_{kl}(y_i;\theta,\vartheta) + \mathcal{R}'_s(y_i;\theta,\vartheta)$. The Kullback-Leibler regularity term $\mathcal{R}_{kl}$ can be analytically derived as a function of the mean and variance of the approximate posterior distribution $q(z_i|y_i;\vartheta)$ [8].

4.5 Optimization details

Minimization of $\mathcal{L}'(y_i;\theta,\vartheta,y_0)$ is performed by stochastic gradient descent. Gradients with respect to the parameters θ, ϑ, y_0 are automatically computed thanks to the auto-differentiation library from the PyTorch project [10]. The numerical gradient of the loss $\mathcal{L}'$ with respect to the template shape y_0 is spatially smoothed with a Gaussian kernel of standard deviation σ_y before being applied by the gradient-based method. This operation is highly beneficial in practice when dealing with noisy data, ensuring that the original topology of the template shape is conserved [4]. The so-called reparametrization trick detailed in [8] ensures that gradients with respect to the encoder parameters ϑ are computable across the sampling procedure. In this same article, the authors report that drawing only $L = 1$ sample per data point is reasonable as long as the Adam batches are large enough; the same strategy will be adopted in this paper, with batches of size 32. Code will be made available upon publication.

5 Experiments

For all subsequent experiments, the parameters of the Sobolev metric are $\alpha = 0.5$ and $s = 3$. The corresponding Lagrange multiplier is fixed to $\lambda = 1$. Finally, forward Euler integration is numerically carried out with $T = 11$ steps. Our DAE is compared to the LDDMM-based PGA model, briefly introduced in Section 3.2.

5.1 Learned latent space with simulated rectangle meshes

A data set of $N = 441$ rectangle meshes of $\mathbb{R}^2$ is simulated: all are centered on the origin, but vary in their length and width which are independently and

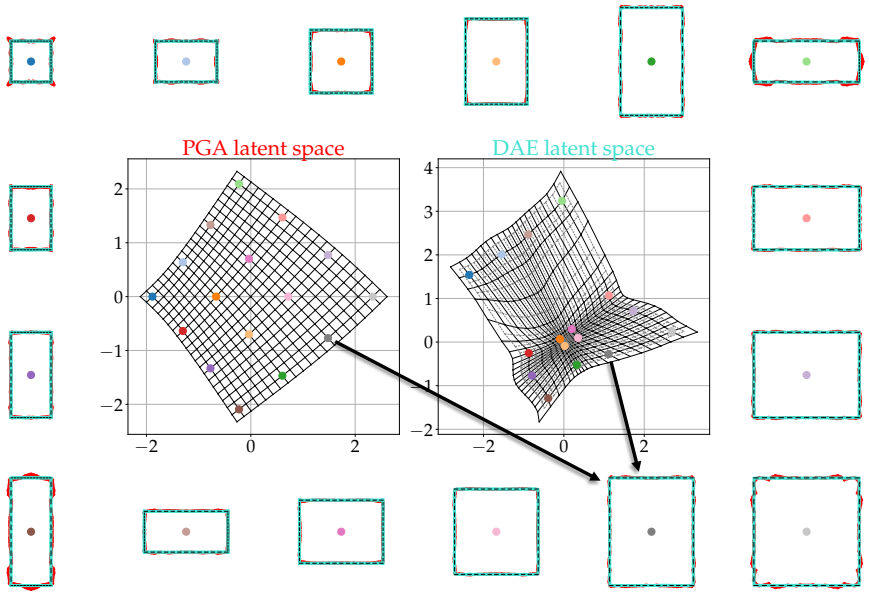

Fig. 3. [Center] Latent spaces learned by PGA and DAE. Each node of the plotted grids in solid black correspond to one of the 441 simulated rectangles. The interleaved dotted grey grid in the DAE space corresponds to the additional 400 test rectangles. [Outer] A subsample of 16 training rectangles are plotted in dotted black lines; a color code allows their identification in the latent spaces. The PGA and DAE reconstructions are plotted in red and light blue respectively.

regularly distributed between 0.5 and 1.5. Those meshes are simulated with point correspondence: the noise model is therefore simply based on the ℓ^2 metric, with $\sigma_\epsilon = 0.01$. The remaining chosen parameters are $\sigma_{\mathcal{S}} = \sigma_y = 0.2$, respectively for the splatting and template gradient smoothing operations. We first learn a PGA model with $n = 2$ components with a deformation scale fixed to 0.1, and then learn our DAE model, initialized on the PGA results.

Figure 3 represents the learned latent spaces. Both methods have correctly learned the variations in length and width of the dataset. The PGA latent space is quite regular, when the DAE one seems to feature more complex spatial relationships. DAE seems to create more curvature in the latent space. We observe also that the DAE reconstructions match more tightly the training points: the mean square errors are 1.55×10^{-4} $(\pm 1.22 \times 10^{-4})$ and 3.43×10^{-6} $(\pm 2.17 \times 10^{-6})$ in the PGA and DAE cases respectively. The ability to better match observations is certainly the consequence of allowing many more degrees of freedom in the parametrization of the diffeomorphisms. Whether the induced curvature in the latent space is a also a consequence of this construction still need to be understood. A second data set of $N = 400$ rectangles of length and width independently and regularly distributed between 0.525 and 1.475 is simulated and

encoded in the DAE latent space (see Figure 3). Note that this operation is virtually instantaneous. After subsequent decoding, the reconstruction error amounts to 3.40×10^{-6} $(\pm 1.69 \times 10^{-6})$, indicating a very good generalization performance.

5.2 Generalization to test data with hippocampi meshes

A total of 324 right hippocampi meshes are segmented from baseline T1-weighted magnetic resonance (MR) images of the ADNI database, after standard alignment preprocessing. The obtained meshes are without point correspondence: the current noise model will be used, with a kernel width $\sigma_{\mathcal{E}} = 5\,mm$ and an uncertainty parameter of $\sigma_\epsilon = 0.1$. Other spatial parameters are chosen equal: $\sigma_{\mathcal{S}} = \sigma_y = 5\,mm$. We learn the PGA and DAE models in dimension $n = 10$, on $N = 162$ training meshes, and then personalize them to the second testing half. The deformation scale parameter for the PGA is taken equal to $10\,mm$, and the same current metric is used for both methods.

Table 1 gives the obtained reconstruction and generalization errors. The DAE model better fits the training data, when the PGA model better generalizes. Note however that the personalization of the PGA model to a new hippocampus requires to solve an optimization problem, which is done with a gradient method, when the learned encoder gives quasi-instantaneous results in the case of the DAE. Refining this initial guess with a gradient method, the so-called DAE+ performance improves, and gives generalization residuals smaller on average the intrinsic uncertainty on the data – which is indicated in the first column. This uncertainty has been computed by preprocessing the secondary MR images (same subject, same visit, same machine) available in the ADNI database into hippocampi meshes, and computing the current-metric residual with the primary measurements. A statistical meaning can be given to this DAE versus DAE+ distinction: recalling the encoder is probabilistic i.e. outputs the normal density distribution $q(z_i|y_i;\vartheta)$, the reported DAE generalization performance directly evaluates the decoded average $E[q(z_i|y_i;\vartheta)]$ when the DAE+ computes a MAP estimate against the full q distribution, so the comparison with PGA which also seeks for the MAP is more fair.

Figure 4 plots the deformation of the PGA and DAE templates onto a test hippocampus. The residuals values are $68.4\,mm^2$, $126.7\,mm^2$ (not plotted) and $75.4\,mm^2$ for the PGA, DAE and DAE+ methods respectively. The rightmost

	Data noise	PGA	DAE	DAE+
Reconstruction	85.2 ± 40.1	66.7 ± 11.5	$\mathbf{32.6 \pm 6.0}$	-
Generalization	85.2 ± 40.1	$\mathbf{67.7 \pm 12.6}$	116.8 ± 20.0	74.7 ± 16.1

Table 1. Reconstruction and generalization residuals (in mm^2), measured with the current metric (scale parameter of $5\,mm$). The data noise is evaluated by leveraging the secondary MR images available in the ADNI database. The DAE+ column refers to a gradient-descent-based refinement of the encoded test data points.

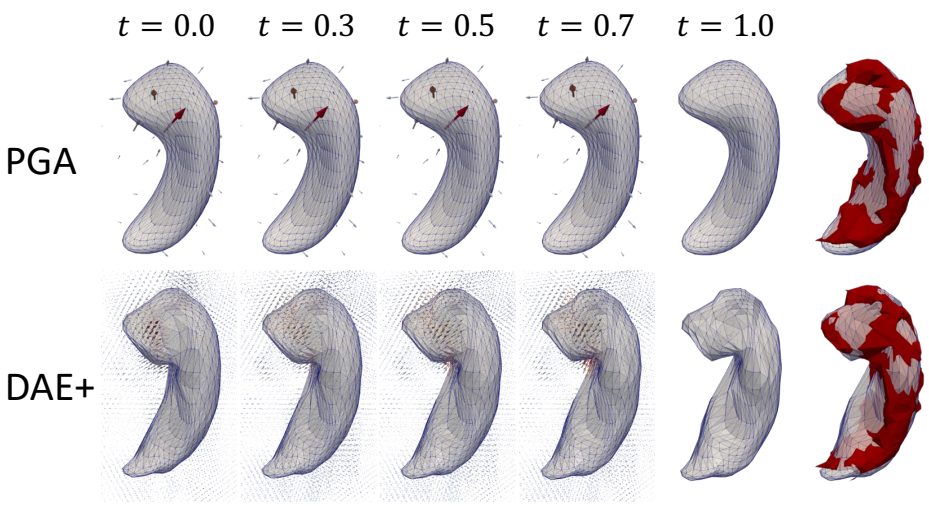

Fig. 4. Estimated diffeomorphic deformation of the PGA and DAE templates (leftmost meshes) onto a test hippocampus (rightmost red meshes). The dynamic PGA momenta and the DAE velocity fields are indicated by arrows, colored according to their magnitude. The current-metric PGA residual is $68.4\,mm^2$, the DAE+ $75.4\,mm^2$.

figures superimpose the target mesh with the fully-deformed templates. Both templates are globally similar, the PGA one being however quite smoother. The deformation fields, represented by the arrows, seem to mainly act in the same "neck" region of the hippocampi. The final registration quality is difficult to evaluate by eye, due to the noise on the original meshes.

5.3 Modes of variability and classification with brain MR images

We now consider a data set of $N = 160$ brain T1-weighted MR images from the ADNI project: 54 are from control subjects (CN), 53 from subjects presenting mild cognitive impairments (MCI), and the last 53 from patients diagnosed with Alzheimer's disease (AD). The images are aligned with an affine transformation in a preprocessing step. The simple ℓ^2 metric is used on the voxel values for the noise model, with an uncertainty parameter $\sigma_\epsilon = 1/255$. The template update smoothing is done with $\sigma_y = 1\,mm$. The PGA and DAE representations are learned in dimension $n = 3$, the PGA scale parameter being fixed to $1\,cm$.

Figure 5 plots the components of a PCA fitted a posteriori on the latent representations of the training images for the DAE, as well as the first axis of variability computed with the PGA. The first components of variability (top rows) explain respectively 59.0% and 56.7% of the captured variance by the PGA and DAE models, and clearly correspond to the ventricle size variability, which is known to be a marker of Alzheimer's disease. Table 2 gives the classification scores obtained by a 11-nearest-neighbors classifier, evaluated in a leave-one-out fashion. Note that 11 is a prime number, thus avoiding ties in the voting

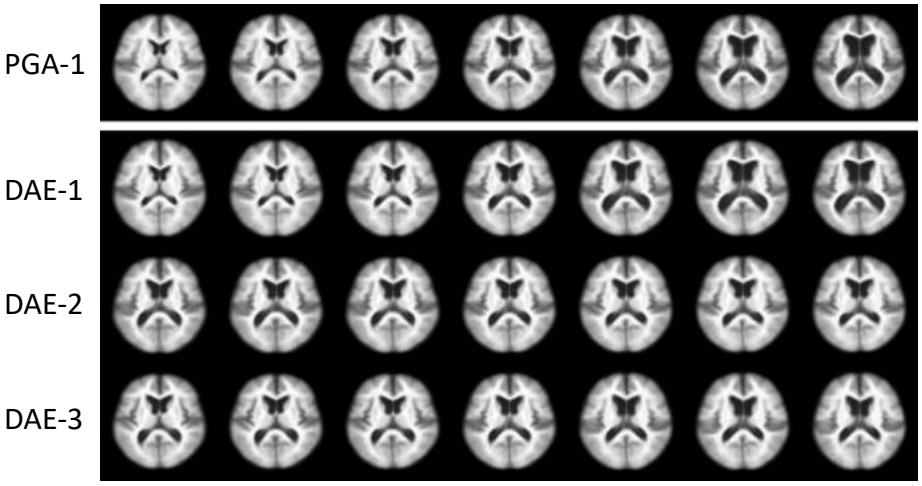

Fig. 5. First principal axis determined with the PGA approach, and all three principal axes computed with the DAE model. For each axis are plotted the shapes deviating by -1.5, -1, -0.5, 0, 0.5, 1, 1.5 times the standard deviation from the template y_0. Note in particular that the central column plots the learned PGA and DAE templates y_0.

process. Scores are in all cases above the chance threshold, and even reach an accuracy of 85.0 % in the CN versus AD classification task, based on the DAE latent representations. If both representations are pooled together, the 3-classes performance slightly increases to 62.5 %, suggesting some complementarity.

6 Discussion and perspectives

We have presented and illustrated a method that jointly learns an atlas model and a class of diffeomorphisms from a data set of shapes. Diffeomorphisms are then parametrized by low-dimensional latent-space parameters. Similarly to LDDMM, those diffeomorphisms are constructed by integrating dynamic velocity fields. Unlike LDDMM, the relationship between latent-space parameters and the velocity fields is (highly) non-linear. A network does this mapping, and only little assumptions are made: infinite differentiability and absence of bias.

	CN/MCI/AD	CN/AD	CN/MCI	MCI/AD
PGA	58.8 %	84.1 %	67.3 %	**71.7 %**
DAE	**61.3 %**	**85.0 %**	67.3 %	68.9 %

Table 2. Leave-one-out classification scores obtained with a 11-nearest-neighbors classifier, taking the learned PGA or DAE latent representations as input.

A theoretical perspective would be to determine conditions under which the image of the latent-space by this mapping defines a manifold. The decoded velocity field paths would then be geodesics for the push-forwarded Euclidian metric. In a second step, if an equivalence relationship could be established between the pushforward and the Sobolev metrics, the somewhat extrinsic Sobolev regularity term might not be needed anymore to ensure the construction of diffeomorphisms. Practical perspectives are numerous, and include speed benchmarks against contemporary statistical shape analysis softwares, network architecture refinement for better overfitting prevention, joint training on classification tasks.

The use of neural networks for generating diffeomorphisms is a promising avenue to learn the metric of shape spaces from the data itself, while raising several challenging theoretical questions.

References

1. Beg, F., Miller, M., Trouvé, A., Younes, L.: Computing large deformation metric mappings via geodesic flows of diffeomorphisms. IJCV (2005)
2. Dalca, A.V., Balakrishnan, G., Guttag, J., Sabuncu, M.R.: Unsupervised learning for fast probabilistic diffeomorphic registration. arXiv preprint:1805.04605 (2018)
3. Durrleman, S.: Statistical models of currents for measuring the variability of anatomical curves, surfaces and their evolution. Ph.D. thesis (2010)
4. Durrleman, S., Prastawa, M., Charon, N., Korenberg, J.R., Joshi, S., Gerig, G., Trouvé, A.: Morphometry of anatomical shape complexes with dense deformations and sparse parameters. NeuroImage (2014)
5. Gori, P., Colliot, O., Marrakchi-Kacem, L., Worbe, Y., Poupon, C., Hartmann, A., Ayache, N., Durrleman, S.: A bayesian framework for joint morphometry of surface and curve meshes in multi-object complexes. Medical Image Analysis **35** (2017)
6. Gris, B., Durrleman, S., Trouvé, A.: A sub-riemannian modular framework for diffeomorphism-based analysis of shape ensembles. SIAM Journal on Imaging Sciences **11**(1), 802–833 (2018)
7. Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., Saul, L.K.: An introduction to variational methods for graphical models. Machine learning **37**(2), 183–233 (1999)
8. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. stat **1050**, 10 (2014)
9. Miller, M.I., Trouvé, A., Younes, L.: Geodesic shooting for computational anatomy. Journal of Mathematical Imaging and Vision **24**(2), 209–228 (2006)
10. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)
11. Pennec, X.: Intrinsic statistics on riemannian manifolds: Basic tools for geometric measurements. Journal of Mathematical Imaging and Vision **25**(1), 127–154 (2006)
12. Thompson, D.W., et al.: On growth and form. On growth and form. (1942)
13. Vaillant, M., Glaunès, J.: Surface matching via currents. In: Information processing in medical imaging. pp. 1–5. Springer (2005)
14. Vercauteren, T., Pennec, X., Perchant, A., Ayache, N.: Symmetric log-domain diffeomorphic registration: A demons-based approach. In: MICCAI. Springer (2008)
15. Yang, X., Kwitt, R., Styner, M., Niethammer, M.: Quicksilver: Fast predictive image registration–a deep learning approach. NeuroImage **158**, 378–396 (2017)
16. Younes, L.: Shapes and Diffeomorphisms. Applied Mathematical Sciences, Springer Berlin Heidelberg (2010), https://books.google.fr/books?id=SdTBtMGgeAUC

17. Zhang, M., Fletcher, P.T.: Bayesian principal geodesic analysis in diffeomorphic image registration. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 121–128. Springer (2014)
18. Zhang, M., Fletcher, P.T.: Fast diffeomorphic image registration via fourier-approximated lie algebras. Int. Journal of Computer Vision pp. 1–13 (2018)