Medical Image Analysis 55 (2019) 197–215

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

An algorithm for learning shape and appearance models without annotations

John Ashburner*, Mikael Brudfors, Kevin Bronik, Yaël Balbastre

Wellcome Centre for Human Neuroimaging UCL Queen Square Institute of Neurology 12 Queen Square, London, WC1N 3AR, UK

ARTICLE INFO

Article history:
Received 27 July 2018
Revised 19 February 2019
Accepted 17 April 2019
Available online 30 April 2019

Keywords:
Machine learning
Latent variables
Diffeomorphisms
Geodesic shooting
Shape model
Appearance model

ABSTRACT

This paper presents a framework for automatically learning shape and appearance models for medical (and certain other) images. The algorithm was developed with the aim of eventually enabling distributed privacy-preserving analysis of brain image data, such that shared information (shape and appearance basis functions) may be passed across sites, whereas latent variables that encode individual images remain secure within each site. These latent variables are proposed as features for privacy-preserving data mining applications.

The approach is demonstrated qualitatively on the KDEF dataset of 2D face images, showing that it can align images that traditionally require shape and appearance models trained using manually annotated data (manually defined landmarks etc.). It is applied to the MNIST dataset of handwritten digits to show its potential for machine learning applications, particularly when training data is limited. The model is able to handle "missing data", which allows it to be cross-validated according to how well it can predict left-out voxels. The suitability of the derived features for classifying individuals into patient groups was assessed by applying it to a dataset of over 1900 segmented T1-weighted MR images, which included images from the COBRE and ABIDE datasets.

1. Introduction

This paper introduces an algorithm for learning a model of shape and appearance variability from a collection of images, without relying on manual annotations. The shape part of the model concerns modelling variability with diffeomorphic deformations, which is essentially image registration. In contrast, the appearance part is about accounting for signal variability that is not well described by deformations, and is essentially about adapting a "template" to enable more precise registration.

The problem of image registration is sometimes viewed from a Bayesian perspective, whereby the aim is to determine the most probable deformation (ψ) given the fixed ($\mathbf{f}$) and moving ($\boldsymbol{\mu}$) images

$$\hat{\boldsymbol{\psi}} = \arg\max_{\psi} \log p(\psi|\mathbf{f}, \boldsymbol{\mu}) = \arg\max_{\psi} (\log p(\mathbf{f}|\psi, \boldsymbol{\mu}) + \log p(\psi)). \quad (1)$$

In practice, the regularisation term ($\log p(\psi)$) is not usually defined empirically, and simply involves a penalty based on some simple measure of deformation smoothness. One of the aims of this work is to try to improve on this simple model. By providing empirically derived priors for the allowable deformations, trained shape models have been shown to exhibit more robust image registration. An early example is Cootes and Taylor (1992), in which control point positions are constrained by their first few modes of variability. Training this model involved annotating images by manually placing a number of corresponding landmarks, computing the mean and covariance of the collection of landmarks, and then computing the eigenvectors of the covariance (Cootes et al., 1995). In neuroimaging, shape models have previously been used to increase the robustness of brain image segmentation (Babalola et al., 2009; Patenaude et al., 2011). The current work involves densely parameterised shape models within the diffeomorphic setting, and relates to previous work on diffeomorphic shape models (Cootes et al., 2008), as well as those using more densely parameterised deformations (Rueckert et al., 2003). Recently, Zhang and Fletcher (2015) developed their Principal Geodesic Analysis (PGA) framework for directly computing the main modes of shape variation within a diffeomorphic setting.

* Corresponding author.
E-mail address: j.ashburner@ucl.ac.uk (J. Ashburner).

https://doi.org/10.1016/j.media.2019.04.008

In addition to increasing the robustness of image registration tasks, shape models can also provide features that may be used for statistical shape analysis. This is related to approaches used in geometric morphometrics (Adams et al., 2004), where the aim is to understand shape differences among anatomies. Shape descriptors from the PGA framework have previously been found to be useful features for data mining (Zhang et al., 2017).

A number of works have investigated combining both shape and appearance variability into the same model (Cootes et al., 1995; 2001; Cootes and Taylor, 2001; Cootes et al., 2008; Belongie et al., 2002; Patenaude et al., 2011). These combined shape and appearance models have generally shown good performance in a number of medical imaging challenges (Litjens et al., 2014). While there is quite a lot written about learning appearance variability alone, the literature on automatically learning both shape and appearance together is fairly limited. Earlier approaches required annotated data for training, but there are now some works appearing that have looked into the possibility of using unsupervised or semi-supervised approaches for learning shape and appearance variability. Examples include Cootes et al. (2010), Alabort-i Medina and Zafeiriou (2014), Lindner et al. (2015) and Štern et al. (2016). The current work is about an unsupervised approach, but there is no reason why it could not be made semi-supervised by also incorporating some manually defined landmarks or other features.

This work was undertaken as a task in the Medical Informatics Platform of the EU Human Brain Project (HBP). The original aim of the Medical Informatics Platform was to develop a distributed knowledge discovery framework that enables data mining without violating patient confidentiality. The strategy was to involve a horizontally partitioned dataset, where data about different patients is stored in different hospital sites. Although this has not been done, the algorithm presented in this paper can be implemented (see Section 2.2) in a way that does not require patient-specific information to leave a site, and instead only shares aggregates, which reveal less about the individual subjects. Some leakage of information (potentially exploitable by those with malicious intent) is inevitable, particularly for sites holding data on only small numbers of individuals, but we leave this as a topic to be addressed elsewhere. Aggregated data may be weighted moments (e.g. $\Sigma_n r_n$, $\Sigma_n r_n \mathbf{z}_n$ or $\sum_n r_n \mathbf{z}_n \mathbf{z}_n^T$, where $\mathbf{z}_n$ is a vector of values for patient n, and r_n is a patient-specific weight generated by some rule), which could then be used for clustering or other forms of statistical analysis. Enabling this type of approach to be applied to images requires some form of dimensionality reduction, particularly if covariances need to be represented (such as for clustering into patient subgroups using Gaussian mixture models).

Our work takes a generative modelling approach. There is increasing interest in the use of generative approaches for machine learning, partly because they can be extended to work in a semi-supervised way. This enables unlabelled training data to contribute towards the model, potentially allowing more complex models to be learned from fewer labelled examples. Another motivation for generative modelling approaches is to enable missing data to be dealt with. Brain images – particularly hospital brain images – often have different fields of view from each other, with parts of the brain missing from some of the scans. Many machine learning approaches do not work well in the presence of missing data, so imputing missing information is an implicit part of the presented framework.

This work proposes a solution based on learning a form of shape and appearance model. The overall aim is to capture as much anatomical variability as possible using a relatively small number of latent variables. In addition to 3D brain image data, a number of other types of images will be used to illustrate other aspects of the very general framework that we present.

2. Methods

The proposed framework builds on many of the ideas presented in the principal geodesic analysis work of Zhang and Fletcher (2015). Modifications involve extending the framework to use a Gauss-Newton optimisation strategy, incorporating a variety of appearance noise models and also using a different overall form of regularisation. This section is divided into two main sections. The first of these describes the overall generative model, whereas the second describes the algorithm for fitting the model. Some of the notation used in this section is explained in Appendix A.

2.1. Generative model

The basic idea is that both shape and appearance may be modelled by linear combinations of spatial basis functions, and the objective is to automatically learn the best set of basis functions and latent variables from some collection of images. This is essentially a form of factorisation of the data. Each of the N images will be denoted by $\mathbf{f}_n \in \mathcal{R}^M$, where M is the number of pixels/voxels in an image, $1 \leq n \leq N$, and the entire collection of images by $\mathbf{F}$. An *appearance model* for the nth image is constructed from a linear combination of basis functions, such that

$$\mathbf{a}_n = \boldsymbol{\mu} + \mathbf{W}^a \mathbf{z}_n. \tag{2}$$

Here, $\mathbf{W}^a$ is a matrix containing K columns of appearance basis functions, and $\mathbf{z}_n$ is a vector of K latent variables for the nth image. The vector $\boldsymbol{\mu}$ is a mean image, with the same dimensions as a column of $\mathbf{W}^a$.

The *shape model* (used by Zhang and Fletcher (2015)) is encoded similarly, where initial velocity fields are computed by

$$\mathbf{v}_n = \mathbf{W}^v \mathbf{z}_n. \tag{3}$$

The Large-Deformation Diffeomorphic Metric Mapping (LDDMM) framework (Beg et al., 2005) is used, which allows images to be warped by smooth, invertible one-to-one mappings. Diffeomorphic deformations (ψ_n) are computed from each $\mathbf{v}_n$ by a procedure known as "geodesic shooting", which is presented in Algorithm 4 of Section 2.2.3.

From a probabilistic perspective, the likelihood can be summarised by

$$p(\mathbf{f}_n | \mathbf{z}_n, \boldsymbol{\mu}, \mathbf{W}^a, \mathbf{W}^v) = p(\mathbf{f}_n | \mathbf{a}_n(\psi_n)), \tag{4}$$

where $\mathbf{a}(\psi)$ denotes warping the entire $\mathbf{a}$ by the deformation ψ. Different forms of noise model are presented in Section 2.1.2, but for convenience, we use the generic definition

$$J(\mathbf{f}_n, \mathbf{z}_n, \boldsymbol{\mu}, \mathbf{W}^a, \mathbf{W}^v) = -\ln p(\mathbf{f}_n | \mathbf{z}_n, \boldsymbol{\mu}, \mathbf{W}^a, \mathbf{W}^v). \tag{5}$$

In practice, a small amount of regularisation is imposed on the mean ($\boldsymbol{\mu}$) by assuming it is drawn from a multivariate Gaussian distribution of precision $\mathbf{L}^\mu$ (see Section 2.1.3)

$$p(\boldsymbol{\mu}) = \mathcal{N}(\boldsymbol{\mu} | \mathbf{0}, (\mathbf{L}^\mu)^{-1}). \tag{6}$$

A weighted sum of two strategies for regularising estimates of the basis functions ($\mathbf{W}^a$ and $\mathbf{W}^v$) and latent variables ($\mathbf{z}_n$) is used, which are:

1. The first strategy involves separate priors on the basis functions, and on the latent variables. Each of the basis functions is assumed to be drawn from zero-mean highly multivariate Gaussian, parameterised by very large and sparse precision matrices. Possible forms of the matrices for regularising shape ($\mathbf{L}^v$) are described in Section 2.1.1, whereas those for appearance ($\mathbf{L}^a$) are described in Section 2.1.3. Priors for the basis functions (see Discussion section regarding scaling by N) are

$$p(\mathbf{W}^v) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{w}_k^v | \mathbf{0}, (N\mathbf{L}^v)^{-1}), \tag{7}$$

$$p(\mathbf{W}^a) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{w}_k^a|\mathbf{0}, (N\mathbf{L}^a)^{-1}). \tag{8}$$

The latent variables ($\mathbf{Z}$) are assumed to be drawn from zero-mean multivariate Gaussian distributions, parameterised by a precision matrix ($\mathbf{A}$) that is derived from the data.[1]

$$p(\mathbf{z}_n|\mathbf{A}) = \mathcal{N}(\mathbf{z}_n|\mathbf{0}, \mathbf{A}^{-1}). \tag{9}$$

The model assumes that matrix $\mathbf{A}$ is drawn from a Wishart distribution.

$$\begin{aligned} p(\mathbf{A}) &= \mathcal{W}_K(\mathbf{A}|\mathbf{\Lambda}_0, \nu_0) \\ &= \frac{|\mathbf{A}|^{(\nu_0-K-1)/2} \exp(-\frac{1}{2}Tr(\mathbf{\Lambda}_0^{-1}\mathbf{A}))}{2^{(\nu_0\ K)/2}|\mathbf{\Lambda}_0|^{\nu_0/2}\Gamma_K\left(\frac{\nu_0}{2}\right)}, \end{aligned} \tag{10}$$

where Γ_K is the multivariate gamma function. This prior can be made as uninformative as possible by using $\nu_0 = K$ and $\Lambda_0 = \mathbf{I}/\nu_0$, where $\mathbf{I}$ is an identity matrix. In general, $\mathbf{\Lambda}_0$ should be a positive definite symmetric matrix, with $\nu_0 \geq K$ so that the distribution can be normalised.

2. The second strategy (used by Zhang and Fletcher (2015)) is a pragmatic solution to ensuring that enough regularisation is used.

$$\begin{aligned} \ln p(\mathbf{Z}, \mathbf{W}^a, \mathbf{W}^v) = &-\frac{1}{2}Tr(\mathbf{Z}\mathbf{Z}^T((\mathbf{W}^a)^T\mathbf{L}^a\mathbf{W}^a \\ &+ (\mathbf{W}^v)^T\mathbf{L}^v\mathbf{W}^v)) + \text{const} \end{aligned} \tag{11}$$

This strategy imposes smoothness on the reconstructions by assuming penalties based on $\ln\mathcal{N}(\mathbf{W}^a\mathbf{z}_n|\mathbf{0}, \mathbf{L}^a)$ and $\ln\mathcal{N}(\mathbf{W}^v\mathbf{z}_n|\mathbf{0}, \mathbf{L}^v)$, in a similar way to more conventional regularisation approaches.

The weighting of the two strategies is controlled by user-specified weights λ_1 and λ_2. When everything is combined (see Fig. 1), the following joint log-probability is obtained

$$\begin{aligned} &\ln p(\mathbf{F}, \boldsymbol{\mu}, \mathbf{W}^a, \mathbf{W}^v, \mathbf{A}, \mathbf{Z}) \\ &= -\sum_{n=1}^{N} J(\mathbf{f}_n, \mathbf{z}_n, \boldsymbol{\mu}, \mathbf{W}^a, \mathbf{W}^v) - \frac{1}{2}\boldsymbol{\mu}^T\mathbf{L}^\mu\boldsymbol{\mu} \\ &\quad - \frac{\lambda_1 N}{2}\left(Tr((\mathbf{W}^a)^T\mathbf{L}^a\mathbf{W}^a) + Tr((\mathbf{W}^v)^T\mathbf{L}^v\mathbf{W}^v)\right) \\ &\quad + \frac{\lambda_1}{2}\left((N + \nu_0 - K - 1)\ln|\mathbf{A}| - Tr((\mathbf{Z}\mathbf{Z}^T + \mathbf{\Lambda}_0^{-1})\mathbf{A})\right) \\ &\quad - \frac{\lambda_2}{2}Tr(\mathbf{Z}\mathbf{Z}^T((\mathbf{W}^a)^T\mathbf{L}^a\mathbf{W}^a + (\mathbf{W}^v)^T\mathbf{L}^v\mathbf{W}^v)) + \text{const}. \end{aligned} \tag{12}$$

The model fitting procedure is described in Section 2.2. Ideally, the procedure would compute distributions for all variables, such that uncertainty was dealt with optimally. Unfortunately, this is computationally impractical for the size of the datasets involved. Instead, only point estimates are made for the latent variables ($\hat{\mathbf{z}}_n$) and various parameters ($\hat{\boldsymbol{\mu}}$, $\mathbf{W}^a$, $\mathbf{W}^v$), apart from $\mathbf{A}$, which is inferred within a variational Bayesian framework.

The approach also allows an alternative formulation, whereby shapes and appearances are modelled separately by having some of the latent variables control appearance, and others control shape. This may be denoted by

$$\mathbf{a}_n = \boldsymbol{\mu} + \sum_{k=1}^{K^a} \mathbf{w}_k^a z_{kn}, \tag{13}$$

[1] Note that the latent precision matrix $\mathbf{A}$ should not be confused with the appearance variables $\mathbf{a}_n$, which were introduced earlier. Hopefully, the context in which they are used should be enough to prevent any confusion.

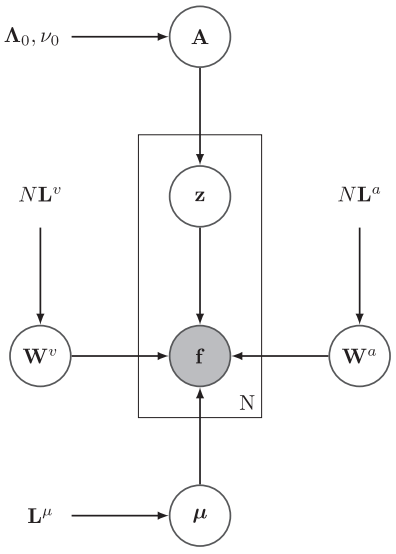

Fig. 1. A graphical representation of the model (showing only the 1st strategy). Gray circles indicate observed data, whereas white circles indicate variables that are either estimated ($\mathbf{W}^v$, $\mathbf{W}^a$, $\boldsymbol{\mu}$ and $\mathbf{z}$) or marginalised out ($\mathbf{A}$). The plate indicates replication over all images.

$$\mathbf{v}_n = \sum_{k=1}^{K^v} \mathbf{w}_k^v z_{mn}, \text{ where } m = K^a + k. \tag{14}$$

For simplicity, only the form where each latent variable controls both shape and appearance is described in detail. This is the form used in active appearance models (Cootes et al., 2001). Note however, that in the form where shape and appearance are controlled by separate latent variables, the precision matrix $\mathbf{A}$ still encodes covariance between the two types of variables. This means that latent variables controlling either shape or appearance are not estimated completely independently.

2.1.1. Differential operator for shape model

The precision matrix used in (Eq. (7)) has the form

$$\begin{aligned} \mathbf{v}^T\mathbf{L}^v\mathbf{v} = &\int_{x\in\Omega} \left(\omega_0^v\|v(x)\|^2 + \omega_1^v\|\nabla v(x)\|^2 + \omega_2^v\|\nabla^2 v(x)\|^2\right)dx \\ &+ \int_{x\in\Omega} \left(\frac{\omega_3^v}{4}\|Dv(x) + (Dv(x))^T\|_F^2 + \omega_4^v Tr(Dv(x))^2\right)dx \end{aligned} \tag{15}$$

where $\|\cdot\|_F$ denotes the Frobenius norm (the square root of the sum of squares of the matrix elements) and D denotes the operator computing Jacobian tensors. The above integral is defined in Sobolev space, which is a weighted Hilbert space where spatial derivatives, up to a certain degree, are accounted for. Five user-specified hyper-parameters are involved:

- ω_0^v controls absolute displacements, and is typically set to be a very small value.
- ω_1^v controls stretching, shearing and rotation.
- ω_2^v controls bending energy. This ensures that the resulting velocity fields have smooth spatial derivatives.
- ω_3^v controls stretching and shearing (but not rotation).
- ω_4^v controls the divergence, which in turn determines the amount of volumetric expansion and contraction.

Most of the regularisation in this work was based on a combination of the linear-elasticity (using *Lamé's constants* ω_3^v and ω_4^v)

and bending energy (ω_2^v) penalties. The effects of different forms of regularisation used for registration are illustrated in Ashburner and Ridgway (2012).

2.1.2. Noise models

A number of different choices for the noise model are available for (Eq. (4)), each suitable for modelling different types of image data. These models are based on $p(\mathbf{f}_n|\mathbf{a}'_n)$, which leads to an "energy" term (J) that drives the model fitting and is assumed to be independent across voxels

$$\mathbf{a}'_n = \boldsymbol{\Psi}_n(\boldsymbol{\mu} + \mathbf{W}^a\mathbf{z}_n) \tag{16}$$

$$J(\mathbf{a}'_n) = -\ln p(\mathbf{f}_n|\mathbf{a}'_n) = -\sum_{m=1}^{M} \ln p(f_{mn}|a'_{mn}). \tag{17}$$

Because the approach is generative, missing data are handled by simply ignoring those voxels where there is no information. By doing this, they do not contribute towards the objective function and play no role in driving the model fitting. A number of different energy functions have been implemented for modelling different types of data. These are listed next.

2.1.2.1. Gaussian noise model. Mean-squares difference is a widely used objective functions for image matching, which is based on the assumption of stationary Gaussian noise. For an image consisting of M pixels or voxels, the function would be

$$-J_{L_2}(\mathbf{a}') = \ln p(\mathbf{f}|\mathbf{a}', \sigma^2) = -\tfrac{M}{2}\ln(2\pi) - \tfrac{M}{2}\ln\sigma^2 - \tfrac{1}{2\sigma^2}||\mathbf{f} - \mathbf{a}'||_2^2, \tag{18}$$

where $||\cdot||_2$ denotes the Euclidean norm. The simplest approach to compute σ^2 is to make a maximum likelihood estimate from the variance by

$$\hat{\sigma^2} = \tfrac{1}{MN}\sum_{n=1}^{N} ||\mathbf{f}_n - \mathbf{a}'_n||_2^2. \tag{19}$$

2.1.2.2. Logistic function with Bernoulli noise model. When working with binary images, such as single tissue type maps having voxels of zeros and ones (or values very close to zero or one), it may be better to work under the assumption that voxels are drawn from a Bernoulli distribution, which is a special case of the binomial distribution. For a single voxel,

$$P(f|s) = s^f(1-s)^{1-f}. \tag{20}$$

The range $0 < s < 1$ must be satisfied, which is achieved using a logistic sigmoid function

$$s(a') = \frac{1}{1+\exp(-a')}. \tag{21}$$

Putting these together leads to the matching function

$$-J_{Bern}(\mathbf{a}') = \ln P(\mathbf{f}|\mathbf{a}') = \sum_{m=1}^{M}\left(f_m a'_m + \ln s(-a'_m)\right). \tag{22}$$

2.1.2.3. Softmax function with categorical noise model. If there are several binary maps to align simultaneously, for example maps of grey matter, white matter and background, then a categorical noise model is appropriate. A categorical distribution is a generalisation of the Bernoulli distribution, and also a special case of the multinomial distribution. The probability of a vector $\mathbf{f}$ of length C, such that $f_c \in \{0, 1\}$ and $\sum_{c=1}^{C} f_c = 1$, is given by

$$P(\mathbf{f}|\mathbf{s}) = \prod_{c=1}^{C} s_c^{f_c}, \tag{23}$$

where $s_c > 0$ and $\sum_{c=1}^{C} s_c = 1$. The constraint on $\mathbf{s}$ is enforced by using a softmax function.

$$s_c(\mathbf{a}') = \frac{\exp a'_c}{\sum_{c=1}^{C}\exp a'_c} \tag{24}$$

Using the "log-sum-exp trick", numerical overflow or underflow can be prevented by first subtracting the maximum of $\mathbf{a}$, so

$$s_c(\mathbf{a}') = \frac{\exp(a'_c - a^*)}{\sum_{c=1}^{C}\exp(a'_c - a^*)}, \text{ where } a^* = \max\{a'_1, \ldots, a'_C\} \tag{25}$$

Noting that each image is now a matrix of M voxels and C classes, the objective function can then be computed as

$$-J_{cat}(\mathbf{A}') = \ln P(\mathbf{F}|\mathbf{A}') = \sum_{m=1}^{M}\left(\sum_{c=1}^{C} a'_{mc}f_{mc} - a^* - \log\left(\sum_{c=1}^{C}\exp(a'_{mc} - a^*_m)\right)\right) \tag{26}$$

2.1.3. Differential operator for appearance model

Regularisation is required for the appearance variability, as it helps to prevent the appearance model from absorbing too much of the variance, at the expense of the shape model. This differential operator (again based on a Sobolev space) is used in Eqs. (6) and (8), and controlled by three hyper-parameters.

$$\mathbf{a}^T\mathbf{L}^a\mathbf{a} = \int_{x\in\Omega}\left(\omega_0^a\|a(x)\|^2 + \omega_1^a\|\nabla a(x)\|^2 + \omega_2^a\|\nabla^2 a(x)\|^2\right)dx \tag{27}$$

2.2. Algorithm for model fitting

A highly simplified version of what was implemented is shown in Algorithm 1. The model fitting approach involves alternating between computing the shape and appearance basis functions (plus a few other variables - *Step-1*), and re-estimating the latent variables (*Step-2*). For better convergence of the basis function updates, an orthogonalisation step is included in each iteration.

Step-1 relies on Gauss-Newton updates of three elements: the mean template ($\boldsymbol{\mu}$), shape subspace ($\mathbf{W}^a$) and appearance subspace ($\mathbf{W}^v$). These updates have the general form of $\mathbf{w} \leftarrow \mathbf{w} - (\mathbf{H} + \mathbf{L})^{-1}(\mathbf{g} + \mathbf{L}\mathbf{w})$, where $\mathbf{L}$ is a very sparse Toeplitz or circulant matrix encoding spatial regularisation, and $\mathbf{H}$ encodes a field of small matrices that are easy to invert. The full-multigrid method, described in Ashburner (2007), is particularly well suited to solving this type of problem.

Step-2 involves updating the latent variables ($\mathbf{Z}$) and Gaussian prior ($\mathbf{A}$). To break the initial symmetry, the latent variables are all initialised randomly, while ensuring that $\hat{\mathbf{Z}}\hat{\mathbf{Z}}^T = N\mathbf{I}$. Correspondingly, matrix $\mathbf{C}^z$ is initialised to $N\mathbf{I}$ and $\hat{\mathbf{A}}$ is initialised to $(N+\nu_0)(N\mathbf{I} + \boldsymbol{\Lambda}_0^{-1})^{-1}$. An initial estimate for $\boldsymbol{\mu}$ is computed from the unaligned data in a fairly straightforward way, whereas $\hat{\mathbf{W}}^a$ and $\hat{\mathbf{W}}^v$ are both initialised to zero.

Comments in Algorithm 1 saying "Dist" indicate which steps should be modified for running within a distributed privacy-preserving framework. The idea here is that the main procedure would be run on the "master" computer, whereas various functions would be run on the "worker" machines on which the data reside. These workers would only pass aggregate data back to the master, whereas the latent variables, which explicitly encode information about individuals, would remain on the workers. As the algorithm is described here, the images ($\mathbf{F}$) and estimated latent variables $\hat{\mathbf{Z}}$ are passed back and forth between the master and workers, but this need not be the case. If these data and variables were all to reside on the worker machines, the master machine would still be able to run using only the aggregate data.

Algorithm 1 Shape and appearance model.

Initialize variables ($\hat{\mathbf{Z}}$, $\hat{\boldsymbol{\mu}}$, $\hat{\mathbf{W}}^a$, $\hat{\mathbf{W}}^\nu$, $\mathbf{C}^z$ and $\hat{\mathbf{A}}$). ▷ Dist (some)
repeat
  $\mathbf{g}^\mu, \mathbf{H}^\mu \leftarrow \text{MeanDerivatives}(\mathbf{F}, \hat{\mathbf{Z}}, \hat{\boldsymbol{\mu}}, \hat{\mathbf{W}}^a, \hat{\mathbf{W}}^\nu)$ ▷ Dist
  $\hat{\boldsymbol{\mu}} \leftarrow \hat{\boldsymbol{\mu}} - (\mathbf{H}^\mu + \mathbf{L}^\mu)^{-1}(\mathbf{g}^\mu + \mathbf{L}^\mu \hat{\boldsymbol{\mu}})$

  $\mathbf{G}^\nu, \mathcal{H}^\nu \leftarrow \text{ShapeDerivatives}(\mathbf{F}, \hat{\mathbf{Z}}, \hat{\boldsymbol{\mu}}, \hat{\mathbf{W}}^a, \hat{\mathbf{W}}^\nu)$ ▷ Dist
  for $k = 1 \ldots K$ **do**
    $\hat{\mathbf{w}}^\nu_k \leftarrow \hat{\mathbf{w}}^\nu_k - (\mathbf{H}^\nu_{kk} + (\lambda_1 N + \lambda_2 c^z_{kk})\mathbf{L}^\nu)^{-1}(\mathbf{g}^\nu_k + (\lambda_1 N + \lambda_2 c^z_{kk})\mathbf{L}^\nu \hat{\mathbf{w}}^\nu_k)$
  end for

  $\mathbf{G}^a, \mathcal{H}^a \leftarrow \text{AppearanceDerivatives}(\mathbf{F}, \hat{\mathbf{Z}}, \hat{\boldsymbol{\mu}}, \hat{\mathbf{W}}^a, \hat{\mathbf{W}}^\nu)$ ▷ Dist
  for $k = 1 \ldots K$ **do**
    $\hat{\mathbf{w}}^a_k \leftarrow \hat{\mathbf{w}}^a_k - (\mathbf{H}^a_{kk} + (\lambda_1 N + \lambda_2 c^z_{kk})\mathbf{L}^a)^{-1}(\mathbf{g}^a_k + (\lambda_1 N + \lambda_2 c^z_{kk})\mathbf{L}^a \hat{\mathbf{w}}^a_k)$
  end for

  $\mathbf{C} \leftarrow (\hat{\mathbf{W}}^\nu)^T \mathbf{L}^\nu \hat{\mathbf{W}}^\nu + (\hat{\mathbf{W}}^a)^T \mathbf{L}^a \hat{\mathbf{W}}^a$
  $\hat{\mathbf{Z}}, \mathbf{S}, \mathbf{C}^z \leftarrow \text{UpdateLatentVariables}(\mathbf{F}, \hat{\mathbf{Z}}, \hat{\boldsymbol{\mu}}, \hat{\mathbf{W}}^a, \hat{\mathbf{W}}^\nu, \lambda_1 \hat{\mathbf{A}} + \lambda_2 \mathbf{C})$ ▷ Dist

  $\mathbf{T} \leftarrow \text{OrthogonalisationMatrix}(\mathbf{C}, \mathbf{C}^z, \mathbf{S}, N)$
  $\hat{\mathbf{W}}^a \leftarrow \hat{\mathbf{W}}^a \mathbf{T}^{-1}$
  $\hat{\mathbf{W}}^\nu \leftarrow \hat{\mathbf{W}}^\nu \mathbf{T}^{-1}$
  $\mathbf{C}^z \leftarrow \mathbf{T}\mathbf{C}^z\mathbf{T}^T$
  $\mathbf{S} \leftarrow \mathbf{T}\mathbf{S}\mathbf{T}^T$
  $\hat{\mathbf{Z}} \leftarrow \mathbf{T}\hat{\mathbf{Z}}$ ▷ Dist

  $\hat{\mathbf{A}} \leftarrow (N + \nu_0)(\mathbf{C}^z + \mathbf{S} + \boldsymbol{\Lambda}_0^{-1})^{-1}$
until convergence

For simplicity, Algorithm 1 does not include functions for computing variances (σ^2 used by the Gaussian noise model), etc., and these variables are not shown to be passed to the various functions that use them. However, it should be easy to see how these changes would be incorporated in practice. Also, the illustration does not show any steps requiring the objective function, which include various backtracking line-searches to ensure that parameter updates cause the objective function to improve each time. In practice, the algorithm is run for a fixed number of iterations, although the log-likelihood could be used to determine when to stop.

2.2.1. Updating the mean ($\hat{\boldsymbol{\mu}}$)

From (Eq. (12)), we see that a point estimate of the mean ($\boldsymbol{\mu}$) may be computed by

$$\hat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \left(\tfrac{1}{2}\boldsymbol{\mu}^T \mathbf{L}^\mu \boldsymbol{\mu} + \sum_{n=1}^{N} J(\mathbf{f}_n, \hat{\mathbf{z}}_n, \boldsymbol{\mu}, \hat{\mathbf{W}}^a, \hat{\mathbf{W}}^\nu) \right). \tag{28}$$

In practice, this log probability is not fully maximised with respect to $\boldsymbol{\mu}$ at each iteration. Instead, $\hat{\boldsymbol{\mu}}$ is updated by a single Gauss-Newton iteration. This requires gradients and Hessians computed as shown in Algorithm 2, which simply involves summing over those computed for the individual images. A small amount of regularisation is used for the estimate of the mean, which is important in situations where it can help to smooth over some of the effects of missing data.

2.2.2. Likelihood derivatives

The algorithm can be run using a number of different noise models, and the gradients and Hessians involved in the Gauss-Newton updates depend upon the one used.

Algorithm 2 Computing gradients and Hessians for mean.

function MeanDerivatives($\mathbf{F}, \hat{\mathbf{Z}}, \hat{\boldsymbol{\mu}}, \hat{\mathbf{W}}^a, \hat{\mathbf{W}}^\nu$)
  $\mathbf{g}^\mu = 0$, $\mathbf{H}^\mu = 0$
  for $n = 1 \ldots N$ **do**
    $\mathbf{a} \leftarrow \hat{\boldsymbol{\mu}} + \hat{\mathbf{W}}^a \hat{\mathbf{z}}_n$
    $\boldsymbol{\Psi} \leftarrow \text{Shoot}(\hat{\mathbf{W}}^\nu \hat{\mathbf{z}}_n)$
    $\mathbf{g}', \mathbf{H}' \leftarrow \text{LikelihoodDerivatives}(\mathbf{f}_n, \mathbf{a}, \boldsymbol{\Psi})$
    $\mathbf{g}^\mu \leftarrow \mathbf{g}^\mu + \mathbf{g}'$
    $\mathbf{H}^\mu \leftarrow \mathbf{H}^\mu + \mathbf{H}'$
  end for
  return $\mathbf{g}^\mu, \mathbf{H}^\mu$
end function

2.2.2.1. Gaussian model. Algorithm 3 shows derivatives for the

Algorithm 3 Likelihood derivatives for Gaussian noise model.

function LikelihoodDerivatives($\mathbf{f}, \mathbf{a}, \boldsymbol{\Psi}$)
  $J' \leftarrow \frac{1}{2\sigma^2}||\boldsymbol{\Psi}\mathbf{a} - \mathbf{f}||^2 + \frac{M}{2}(\ln(\sigma^2) + \ln(2\pi))$ ▷ If needed
  $\mathbf{g}' \leftarrow \boldsymbol{\Psi}^T\left(\frac{1}{\sigma^2}(\boldsymbol{\Psi}\mathbf{a} - \mathbf{f})\right)$
  $\mathbf{H}' \leftarrow \text{diag}\left(\boldsymbol{\Psi}^T\left(\frac{1}{\sigma^2}\mathbf{1}\right)\right)$ ▷ where $\mathbf{1}$ is an array of ones
  return $J', \mathbf{g}', \mathbf{H}'$
end function

Gaussian noise model (Eq. (18)). For a single voxel, this is based on

$$\frac{dJ_{L_2}}{da'} = \frac{1}{\sigma^2}(a' - f) \text{ and } \frac{d^2 J_{L_2}}{da'^2} = \frac{1}{\sigma^2} \tag{29}$$

For voxels where data is missing, both J_{L_2} and $\frac{dJ_{L_2}}{da'}$ are assumed to be zero. Using matrix notation, the objective function for an image is therefore

$$J' = \frac{1}{2\sigma^2}(\boldsymbol{\Psi}\mathbf{a} - \mathbf{f})^T(\boldsymbol{\Psi}\mathbf{a} - \mathbf{f}) + \frac{M}{2}(\ln(\sigma^2) + \ln(2\pi)). \tag{30}$$

The gradients and Hessians, with respect to variations in $\mathbf{a}$, are

$$\mathbf{g}' = \boldsymbol{\Psi}^T\left(\frac{1}{\sigma^2}(\boldsymbol{\Psi}\mathbf{a} - \mathbf{f})\right) \tag{31}$$

$$\mathbf{H}' = \frac{1}{\sigma^2}\boldsymbol{\Psi}^T\boldsymbol{\Psi} \tag{32}$$

In practice, the Hessian ($\mathbf{H}'$) is approximated by a diagonal matrix

$$\mathbf{H}' \simeq \text{diag}\left(\boldsymbol{\Psi}^T\mathbf{1}\frac{1}{\sigma^2}\right), \tag{33}$$

where $\mathbf{1}$ is a vector of ones. This approximation works in the optimisation because all rows of $\boldsymbol{\Psi}$ sum to 1, so for any vector $\mathbf{d}$ of the right dimension, the rows of $\boldsymbol{\Psi}^T\text{diag}(\mathbf{d})\boldsymbol{\Psi}$ sum to $\boldsymbol{\Psi}^T\mathbf{d}$. Because (for trilinear interpolation) all elements of $\boldsymbol{\Psi}$ are greater than or equal to zero, so if all elements of $\mathbf{d}$ are non-negative, then all eigenvalues of $\text{diag}(\boldsymbol{\Psi}^T\mathbf{d}) - \boldsymbol{\Psi}^T\text{diag}(\mathbf{d})\boldsymbol{\Psi}$ are greater than or equal to zero.[2] These non-negative eigenvalues ensure that our approximation to the Hessian (Eq. (33)) is more positive semi-definite than (Eq. (32)).

2.2.2.2. Binary model. For the Bernoulli noise model with the sigmoidal squashing function (Eq. (22)), some modifications are made to the gradient and Hessian of Algorithm 3, based on the derivatives

$$\frac{dJ_{Bern}}{da'} = s(a') - f \text{ and } \frac{d^2 J_{Bern}}{da'^2} = s(a')(1 - s(a')). \tag{34}$$

[2] See https://en.wikipedia.org/wiki/Loewner_order.

Using matrix notation (where $\mathbf{s} \equiv s(\mathbf{a})$), the gradients and Hessians are

$$\mathbf{g}' = \boldsymbol{\Psi}^T(\boldsymbol{\Psi}\mathbf{s} - \mathbf{f}) \tag{35}$$

$$\mathbf{H}' = \boldsymbol{\Psi}^T \mathrm{diag}(\mathbf{s})\mathrm{diag}(1-\mathbf{s})\boldsymbol{\Psi} \simeq \mathrm{diag}\big(\boldsymbol{\Psi}^T \mathrm{diag}(\mathbf{s})(1-\mathbf{s})\big) \tag{36}$$

2.2.2.3. Categorical model. The categorical model with a softmax squashing function (Eq. (26)) would use the gradients and Hessians

$$\frac{dJ_{cat}}{da'_k} = s_k(\mathbf{a}') - f_k, \text{ where } s(\mathbf{a}') = \frac{\exp \mathbf{a}'}{\sum_{k=1}^{K} \exp a'_k} \tag{37}$$

$$\frac{d^2 J_{cat}}{da'_k a'_l} = s_k(\mathbf{a}')(\delta_{jk} - s_j(\mathbf{a}')), \tag{38}$$

where δ_{jk} is the Kronecker delta function. Computation of the gradients and the approximation of the Hessian follow similar lines to those for the binary and Gaussian models.

2.2.3. Geodesic shooting

Algorithm 4 shows how diffeomorphic deformations are computed from the initial velocities via a Geodesic shooting procedure. In the presented algorithm, $D\psi$ denotes the Jacobian tensor field of ψ, and $(D\psi)^T u$ indicates a pointwise multiplication with the transpose of the Jacobian. $|D\psi|$ denotes the field of Jacobian determinants. Lv in the continuous framework is equivalent to the matrix multiplication $\mathbf{L}^v\mathbf{v}$ in the discrete framework. The operation $L^g u$ denotes applying the inverse of L to u, such that $LL^g u = u$. In practice, this is a deconvolution, which is computed using fast Fourier transform (FFT) methods to obtain the Green's function (Bro-Nielsen and Gramkow, 1996). Because of this, the boundary conditions for the velocity fields (and other spatial basis functions) are assumed to be periodic. Much has already been written about the geodesic shooting procedure, so the reader is referred to Miller et al. (2006) and Ashburner and Friston (2011) for further information.

Algorithm 4 Geodesic shooting via Euler integration.

function SHOOT(v_0)
  $u_0 \leftarrow Lv_0$ ▷ $\mathbf{L}^v\mathbf{v} \equiv Lv$
  $\psi \leftarrow id$
  for $t = 1 \ldots T$ **do**
    $u \leftarrow |D\psi|\,(D\psi)^T u_0(\psi)$
    $v \leftarrow L^g u$ ▷ Convolution using FFT
    $\psi \leftarrow \psi(id - \frac{1}{T}v)$
  end for
  return ψ
end function

2.2.4. Updating appearance basis functions ($\hat{\mathbf{W}}^a$)

Appearance basis functions are optimised by

$$\hat{\mathbf{W}}^a = \arg\min_{\mathbf{W}^a}\Big(\tfrac{1}{2}Tr\big((\lambda_1 N\mathbf{I} + \lambda_2 \hat{\mathbf{Z}}\hat{\mathbf{Z}}^T)(\mathbf{W}^a)^T\mathbf{L}^a\mathbf{W}^a\big) + \sum_{n=1}^{N} J(\mathbf{f}_n, \hat{\mathbf{z}}_n, \hat{\boldsymbol{\mu}}, \mathbf{W}^a, \hat{\mathbf{W}}^v)\Big). \tag{39}$$

The first step involves computing the gradients and Hessians, which is shown in Algorithm 5. Note that this only shows the computation of gradients and Hessians for the Gaussian noise model, and that slight modifications are required when using other forms of noise model. Gradients and Hessians for updating these basis functions ($\mathbf{W}^a$) are similar to those for the mean updates, except for weighting based on the current estimates of the latent variables. Note that for this approach to work effectively, the rows of $\hat{\mathbf{Z}}$ should be orthogonal to each other, which is explained further in Section 2.2.8. Note that only a single Gauss-Newton step is performed in each iteration, so the objective function in (Eq. (39)) is not fully optimised, but merely improved over its previous value.

Algorithm 5 Computing gradients and Hessians for appearance.

function APPEARANCEDERIVATIVES($\mathbf{F}, \hat{\mathbf{Z}}, \hat{\boldsymbol{\mu}}, \hat{\mathbf{W}}^a, \hat{\mathbf{W}}^v$)
  for $k = 1 \ldots K$ **do**
    $\mathbf{g}^a_k \leftarrow \mathbf{0}$, $\mathbf{H}^a_{kk} \leftarrow \mathbf{0}$
  end for
  for $n = 1 \ldots N$ **do**
    $\mathbf{a} \leftarrow \hat{\boldsymbol{\mu}} + \hat{\mathbf{W}}^a \hat{\mathbf{z}}_n$
    $\boldsymbol{\Psi} \leftarrow \mathrm{Shoot}(\hat{\mathbf{W}}^v \hat{\mathbf{z}}_n)$
    $\mathbf{g}', \mathbf{H}' \leftarrow \mathrm{LikelihoodDerivatives}(\mathbf{f}_n, \mathbf{a}, \boldsymbol{\Psi})$
    for $k = 1 \ldots K$ **do**
      $\mathbf{g}^a_k \leftarrow \mathbf{g}^a_k + \hat{z}_{kn}\mathbf{g}'$
      $\mathbf{H}^a_{kk} \leftarrow \mathbf{H}^a_{kk} + \hat{z}^2_{kn}\mathbf{H}'$
    end for
  end for
  return $\mathbf{G}^a, \mathcal{H}^a$ ▷ Where $\mathbf{G}^a = \{\mathbf{g}^a_1, \mathbf{g}^a_2, \ldots, \mathbf{g}^a_K\}$
    ▷ $\mathcal{H}^a = \{\mathbf{H}^a_{1,1}, \mathbf{H}^a_{2,2}, \ldots, \mathbf{H}^a_{K,K}\}$
end function

2.2.5. Updating shape basis functions ($\hat{\mathbf{W}}^v$)

Shape basis functions are optimised by

$$\hat{\mathbf{W}}^v = \arg\min_{\mathbf{W}^v}\Big(\tfrac{1}{2}Tr\big((\lambda_1 N\mathbf{I} + \lambda_2 \hat{\mathbf{Z}}\hat{\mathbf{Z}}^T)(\mathbf{W}^v)^T\mathbf{L}^v\mathbf{W}^v\big) + \textstyle\sum_{n=1}^{N} J(\mathbf{f}_n, \hat{\mathbf{z}}_n, \hat{\boldsymbol{\mu}}, \hat{\mathbf{W}}^a, \mathbf{W}^v)\Big). \tag{40}$$

A single Gauss-Newton iteration is used to update the basis functions of the shape model ($\mathbf{W}^v$), which is done in such a way that changes to $\mathbf{W}^v$ improve the objective function with respect to its previous value, rather than fully optimise . (Eq. (40)). While most Gauss-Newton iterations improve the fit, a backtracking line search is included to ensure that they do not overshoot. As for updating $\mathbf{W}^a$, this requires the rows of $\hat{\mathbf{Z}}$ to be orthogonal to each other. The strategy for computing gradients and Hessians is shown in Algorithm 6.

Algorithm 6 Computing gradients and Hessians for shape.

function SHAPEDERIVATIVES($\mathbf{F}, \hat{\mathbf{Z}}, \hat{\boldsymbol{\mu}}, \hat{\mathbf{W}}^a, \hat{\mathbf{W}}^v$)
    ▷ Various settings (eg $\mathbf{L}^v$) are not passed as arguments
  for $k = 1 \ldots K$ **do**
    $\mathbf{g}^v_k \leftarrow \mathbf{0}$, $\mathbf{H}^v_{kk} \leftarrow \mathbf{0}$
  end for
  for $n = 1 \ldots N$ **do**
    $\mathbf{a} \leftarrow \hat{\boldsymbol{\mu}} + \hat{\mathbf{W}}^a \hat{\mathbf{z}}_n$
    $\boldsymbol{\Psi} \leftarrow \mathrm{Shoot}(\hat{\mathbf{W}}^v \hat{\mathbf{z}}_n)$
    $\mathbf{g}', \mathbf{H}' \leftarrow \mathrm{LikelihoodDerivatives}(\mathbf{f}_n, \mathbf{a}, \boldsymbol{\Psi})$
    $\mathbf{D} \leftarrow \big[\mathrm{diag}(\nabla_1 \mathbf{a}) \quad \mathrm{diag}(\nabla_2 \mathbf{a}) \quad \mathrm{diag}(\nabla_3 \mathbf{a})\big]$
    $\mathbf{g}' \leftarrow \mathbf{D}^T\mathbf{g}'$
    $\mathbf{H}' \leftarrow \mathbf{D}^T\mathbf{H}'\mathbf{D}$
    for $k = 1 \ldots K$ **do**
      $\mathbf{g}^v_k \leftarrow \mathbf{g}^v_k + \hat{z}_{kn}\mathbf{g}'_n$
      $\mathbf{H}^v_{kk} \leftarrow \mathbf{H}^v_{kk} + \hat{z}^2_{kn}\mathbf{H}'_n$
    end for
  end for
  return $\mathbf{G}^v, \mathcal{H}^v$
end function

2.2.6. Updating latent variables ($\hat{\mathbf{z}}_n$)

The modes of the latent variables are updated via a Gauss-Newton scheme (shown in Algorithm 7), similar to that used by Friston et al. (1995),Cootes et al. (2001) and Cootes and Taylor (2001).

Algorithm 7 Updating latent variables.

function UPDATELATENTVARIABLES($\mathbf{F}, \hat{\mathbf{Z}}, \hat{\boldsymbol{\mu}}, \hat{\mathbf{W}}^a, \hat{\mathbf{W}}^\nu, \mathbf{A}$)
  $\mathbf{S} \leftarrow \mathbf{0}$
  for $n = 1 \ldots N$ **do**
    $\mathbf{a} \leftarrow \hat{\boldsymbol{\mu}} + \hat{\mathbf{W}}^a \hat{\mathbf{z}}_n$
    $\boldsymbol{\Psi} \leftarrow \text{Shoot}(\hat{\mathbf{W}}^\nu \hat{\mathbf{z}}_n)$
    $\mathbf{g}', \mathbf{H}' \leftarrow \text{LikelihoodDerivatives}(\mathbf{f}_n, \mathbf{a}, \boldsymbol{\Psi})$
    $\mathbf{D} \leftarrow \begin{pmatrix}\text{diag}(\nabla_1 \mathbf{a}) & \text{diag}(\nabla_2 \mathbf{a}) & \text{diag}(\nabla_3 \mathbf{a})\end{pmatrix}$
    $\mathbf{B} \leftarrow \mathbf{D}^T \mathbf{W}^\nu + \mathbf{W}^a$
    $\mathbf{g} \leftarrow \mathbf{B}^T \mathbf{g}'$
    $\mathbf{H} \leftarrow \mathbf{B}^T \mathbf{H}' \mathbf{B}$
    $\hat{\mathbf{z}}_n \leftarrow \hat{\mathbf{z}}_n - (\mathbf{H} + \mathbf{A})^{-1}\left(\mathbf{g} + \mathbf{A}\hat{\mathbf{z}}_n\right)$
    $\mathbf{S} \leftarrow \mathbf{S} + (\mathbf{H} + \mathbf{A})^{-1}$
  end for
  $\mathbf{C}^z \leftarrow \hat{\mathbf{Z}}\hat{\mathbf{Z}}^T$
  return $\hat{\mathbf{Z}}, \mathbf{S}, \mathbf{C}^z$
end function

$$\hat{\mathbf{z}}_n = \arg\min_{\mathbf{z}_n} \Big(J(\mathbf{f}_n, \mathbf{z}_n, \boldsymbol{\mu}, \hat{\mathbf{W}}^a, \hat{\mathbf{W}}^\nu) + \tfrac{1}{2}\mathbf{z}_n^T \left(\lambda_1 \hat{\mathbf{A}} + \lambda_2 (\hat{\mathbf{W}}^a)^T \mathbf{L}^a \hat{\mathbf{W}}^a + \lambda_2 (\hat{\mathbf{W}}^\nu)^T \mathbf{L}^\nu \hat{\mathbf{W}}^\nu\right)\mathbf{z}_n \Big) \tag{41}$$

The inverse of the (approximate) Hessians allows a Gaussian approximation of the uncertainty with which the latent variables are updated to be computed ("Laplace approximation"). This is the $\mathbf{S}$ matrix, which is combined with $\hat{\mathbf{Z}}\hat{\mathbf{Z}}^T$ (returned as $\mathbf{C}^z$) and used to re-compute $\hat{\mathbf{A}}$.

2.2.7. Expectation of the precision matrix ($\hat{\mathbf{A}}$)

This work uses a variational Bayesian approach for approximating the distribution of $\mathbf{A}$, which is a method described in more detail by textbooks, such as Bishop et al. (2006) or Murphy (2012). Briefly, it involves taking the joint probability of (Eq. (12)), discarding terms that do not involve $\mathbf{A}$, and substituting the expectations of the other parameters into the expression. This leads to the following approximating distribution, which can be recognised as Wishart.

$$\begin{aligned}\ln q(\mathbf{A}) &= \tfrac{1}{2}(N + \nu_0 - K - 1)\ln\det|\mathbf{A}| \\ &\quad - \tfrac{1}{2}Tr\left((\mathbb{E}[\mathbf{Z}\mathbf{Z}^T] + \boldsymbol{\Lambda}_0^{-1})\mathbf{A}\right) + \text{const} \\ &= \ln \mathcal{W}_K(\mathbf{A}|\boldsymbol{\Lambda}, \nu),\end{aligned} \tag{42}$$

where $\boldsymbol{\Lambda} = (\mathbb{E}[\mathbf{Z}\mathbf{Z}^T] + \boldsymbol{\Lambda}_0^{-1})^{-1}$ and $\nu = \nu_0 + N$. In practice, $\mathbb{E}[\mathbf{Z}\mathbf{Z}^T]$ is approximated by $\mathbf{C}^z + \mathbf{S}$, described previously. Other steps in the algorithm use the expectation of $\mathbf{A}$, which (see Appendix B of Bishop et al. (2006)) is

$$\hat{\mathbf{A}} = \mathbb{E}[\mathbf{A}] = \nu\boldsymbol{\Lambda}. \tag{43}$$

2.2.8. Orthogonalisation

The strategy for updating $\hat{\mathbf{W}}^a$ and $\hat{\mathbf{W}}^\nu$ involves some approximations, which are needed in order to save memory and computation. This approximation is related to the Jacobi iterative method for determining the solutions to linear equations, which is only guaranteed to converge for diagonally dominant matrices. Rather than work with the Hessian for the entire $\mathbf{W}$ matrix together, only the Hessians for each column of $\mathbf{W}$ are computed by Algorithms 5 and 6. This corresponds with a block diagonal Hessian matrix for the entire $\mathbf{W}$, which has the form

$$\mathbf{H} = \begin{pmatrix} \mathbf{H}_{11} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_{22} & \ldots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{H}_{KK} \end{pmatrix}. \tag{44}$$

More stable convergence can be achieved by transforming the basis functions and latent variables in order to minimise the amount of signal that would be in the off-diagonal blocks, thus increasing the diagonal dominance of the system of equations. In situations where diagonal dominance is violated, convergence can still be achieved by decreasing the update step size. This is analogous to using a weighted Jacobi iteration, where in practice the weights are found using a backtracking line-search.

Signal in the off-diagonal blocks is reduced by orthogonalising the rows of $\hat{\mathbf{Z}}$. This is achieved by finding a transformation, $\mathbf{T}$, such that $\mathbf{T}\hat{\mathbf{Z}}(\mathbf{T}\hat{\mathbf{Z}})^T$ and $(\hat{\mathbf{W}}^\nu\mathbf{T}^{-1})^T\mathbf{L}^\nu\hat{\mathbf{W}}^\nu\mathbf{T}^{-1} + (\hat{\mathbf{W}}^a\mathbf{T}^{-1})^T\mathbf{L}^a\hat{\mathbf{W}}^a\mathbf{T}^{-1}$ are both diagonal matrices. Transformation $\mathbf{T}$ is derived from an eigen-decomposition of the sufficient statistics, whereby the symmetric positive definite matrices are decomposed into diagonal ($\mathbf{D}^z$ and $\mathbf{D}^w$) and orthonormal ($\mathbf{V}^z$ and $\mathbf{V}^w$) matrices, such that

$$\mathbf{V}^z\mathbf{D}^z(\mathbf{V}^z)^T = \mathbf{C}^z, \tag{45}$$

$$\mathbf{V}^w\mathbf{D}^w(\mathbf{V}^w)^T = \mathbf{C}, \tag{46}$$

where $\mathbf{C}^z = \hat{\mathbf{Z}}\hat{\mathbf{Z}}^T$ and $\mathbf{C} = (\hat{\mathbf{W}}^\nu)^T\mathbf{L}^\nu\hat{\mathbf{W}}^\nu + (\hat{\mathbf{W}}^a)^T\mathbf{L}^a\hat{\mathbf{W}}^a$.

A further singular value decomposition is then used, giving

$$\mathbf{U}\mathbf{D}\mathbf{V}^T = (\mathbf{D}^w)^{\frac{1}{2}}(\mathbf{V}^w)^T\mathbf{V}^z(\mathbf{D}^z)^{\frac{1}{2}}. \tag{47}$$

The combination of various matrices is used to give an initial estimate of the transform

$$\mathbf{T} = \mathbf{D}\mathbf{V}^T(\mathbf{D}^z)^{-\frac{1}{2}}(\mathbf{V}^z)^T. \tag{48}$$

The above $\mathbf{T}$ matrix could be used to render the matrices orthogonal, but their relative scalings would not be optimal. The remainder of the orthogonalisation procedure involves an iterative strategy similar to expectation maximisation, where the aim is to estimate some diagonal scaling matrix $\mathbf{Q}$ with which to multiply $\mathbf{T}$. This matrix is parameterised by a set of parameters $\mathbf{q}$, such that

$$\mathbf{Q} = \text{diag}(\exp\mathbf{q}). \tag{49}$$

The first step of the iterative scheme involves re-computing $\hat{\mathbf{A}}$, as described in Section 2.2.7, but incorporating the current estimates of $\mathbf{Q}\mathbf{T}$.

$$\hat{\mathbf{A}} = \nu\boldsymbol{\Lambda} = (N + \nu_0)(\mathbf{Q}\mathbf{T}(\mathbf{C}^z + \mathbf{S})(\mathbf{Q}\mathbf{T})^T + \boldsymbol{\Lambda}_0^{-1})^{-1}. \tag{50}$$

The next step in the iterative scheme is to re-estimate $\mathbf{q}$, such that

$$\hat{\mathbf{q}} = \arg\min_{\mathbf{q}} (Tr\left(\text{diag}(\exp(-\mathbf{q}))(\mathbf{T}^{-1})^T\mathbf{C}\mathbf{T}^{-1}\text{diag}(\exp(-\mathbf{q}))\right) + Tr\left(\text{diag}(\exp\mathbf{q})\mathbf{T}\mathbf{C}^z\mathbf{T}^T\text{diag}(\exp\mathbf{q})\hat{\mathbf{A}}\right)). \tag{51}$$

This is achieved via a Gauss-Newton update, which uses first and second derivatives with respect to $\mathbf{q}$. The overall strategy is illustrated in Algorithm 8, which empirically is found to converge well.

3. Results

To show the general applicability of the approach, evaluations were performed with a number of datasets of varying characteristics. Our implementation[3] is written in a mixture of MATLAB and C code (MATLAB "mex" files for the computationally expensive parts).

[3] https://github.com/WTCN-computational-anatomy-group/Shape-Appearance-Model.

Algorithm 8 Orthogonalising the variables.

function OrthogonalisationMatrix($\mathbf{C}, \mathbf{C}^z, \mathbf{S}, N$)
  $\mathbf{V}^z, \mathbf{D}^z \leftarrow \text{eig}(\mathbf{C}^z)$
  $\mathbf{V}^w, \mathbf{D}^w \leftarrow \text{eig}(\mathbf{C})$
  $\mathbf{U}, \mathbf{D}, \mathbf{V} \leftarrow \text{svd}((\mathbf{D}^w)^{\frac{1}{2}}(\mathbf{V}^w)^T\mathbf{V}^z(\mathbf{D}^z)^{\frac{1}{2}})$
  $\mathbf{T} \leftarrow \mathbf{D}\mathbf{V}^T(\mathbf{D}^z)^{-\frac{1}{2}}(\mathbf{V}^z)^T$
  $\mathbf{q} \leftarrow \mathbf{0}$
  $\mathbf{Q} \leftarrow \text{diag}(\exp\mathbf{q})$
  repeat
    $\hat{\mathbf{A}} \leftarrow (N+\nu_0)(\mathbf{QT}(\mathbf{C}^z+\mathbf{S})(\mathbf{QT})^T + \mathbf{\Lambda}_0^{-1})^{-1}$ ▷ See Eq. (43).
    $\mathbf{R} \leftarrow 2\hat{\mathbf{A}} \odot (\mathbf{T}\mathbf{C}^z\mathbf{T}^T)^T$ ▷ "$\odot$" denotes a Hadamard product
    $\mathbf{g} \leftarrow \mathbf{QR}\text{diag}(\mathbf{Q}) - 2\mathbf{Q}^{-2}\text{diag}((\mathbf{T}^{-1})^T\mathbf{C}\mathbf{T}^{-1})$ ▷ Gradient
    $\mathbf{H} \leftarrow \mathbf{QRQ} + \text{diag}(\mathbf{QR}\text{diag}(\mathbf{Q})) + 4\mathbf{Q}^{-2}(\mathbf{T}^{-1})^T\mathbf{C}\mathbf{T}^{-1}$ ▷ Hessian
    $\mathbf{q} \leftarrow \mathbf{q} - \mathbf{H}^{-1}\mathbf{g}$
    $\mathbf{Q} \leftarrow \text{diag}(\exp\mathbf{q})$
  until Convergence
  $\mathbf{T} \leftarrow \mathbf{QT}$
  return T
end function

3.1. Qualitative 2D experiments with faces

After years of exposure to faces, most people can identify whether an image of a face is plausible or not, so images of human faces provide a good qualitative test of how well the algorithm can model biological variability.

The straight on views from the Karolinska Directed Emotional Faces (KDEF) data-set (Lundqvist et al., 1998) were used to make a visual assessment of how well the algorithm performs. This data-set consisted of photographs of 70 participants, holding seven different facial expressions, which was repeated twice. Some of the images were excluded because they were systematically brighter (47 images) or had different dimensions (one image), leaving a final dataset consisting of 932 colour images, which were downsampled to a size of 282×382. The original intensities were in the range of 0 to 255, but these values were re-scaled by 1/255.

A 64 eigenmode model was used ($K = 64$), which assumed Gaussian noise. Model fitting (i.e., learning the shape and appearance basis functions, etc.) was run for 20 iterations, with $\nu_0 = 1000$, $\lambda = [15.2\ 0.8]$, $\omega^a = [4\ 512\ 64]$, $\omega^\mu = N[10^{-4}\ 0.1\ 0.1]$ and $\omega^\nu = [10^{-3}\ 0\ 16\ 1\ 1]$. It was fit to the entire field of view of the images, rather than focusing only on the faces, and some of the resulting fits are shown in Fig. 2. The first set of images are a random selection of the original data, with the full shape and appearance model fits shown immediately below. As can be seen, the fit is reasonably good - especially given that only 64 modes of variability were used, and that these have to account for a lot of variability of hair etc. Below these are the shape model fits, generated by warping the mean according to the estimated deformations ($\boldsymbol{\mu}(\psi_n)$). The appearance fits are shown at the bottom ($\mathbf{a}_n$ from (Eq. (4))). Ideally, these reconstructions of appearance should be in perfect alignment with each other, which is not quite achieved in certain parts of the images. In particular, the thickness of the neck varies according to whether or not the people in the images have short or long hair. When looked at separately, the shape and appearance parts of the model do not behave quite so well, but when combined, they give quite a good fit. Fig. 3 shows a simple 64-mode principal component analysis (PCA) fit to the same data, which clearly does not capture variability quite as well as the shape and appearance model.

For these examples, there should really have been a distinction between inter-subject variability and intra-subject variability, using some form of hierarchical model for the latent variables. This type of hierarchical mixed-effects model is widely used for analysing multi-subject data within the neuroimaging field (Friston et al., 2002), and a number of works have applied mixed effects modeling to image registration (Datar et al., 2012; Allassonnière et al., 2015).

3.1.1. Simulating faces

Once the model is learned, it becomes possible to generate random faces from the estimated distribution. This involves drawing a random vector of latent variables $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \hat{\mathbf{A}}^{-1})$, and using these to reconstruct a face. Fig. 4 shows two sets of randomly generated faces, where the lower set used the same latent variables as the upper set, except that they were multiplied by -1. Although some of the random faces are not entirely plausible, they are much more realistic than faces generated from a simple 64-mode PCA model (shown in Fig. 3).

3.1.2. Vector arithmetic

In many machine learning applications, it is useful to be able to model certain non-linearities in the data in an approximately linear way, allowing more interpretable linear methods to be used while still achieving a good fit. Following Radford et al. (2015), this section shows that simple arithmetic on the latent variables can give intuitive results. The first three columns of Fig. 5 show the full shape and appearance model fits to various faces. Images in the right hand column of Fig. 5 were generated by making linear combinations of the latent variables that encode the images in the first three columns, and then reconstructing from these. Unlike arithmetic computed in pixel space (not shown), performing arithmetic on the vectors encoding the images gives reasonably plausible results.

3.2. 2D experiments with MNIST

In this section, the behaviour of the approach using "big data" is assessed, which gives more of an idea of how this type of method may behave with some of the very large image datasets currently being collected. Instead of testing on a large collection of medical images, the approach was applied to a large set of tiny images of hand-written digits. MNIST[4] (LeCun et al., 1998) is a modified version of the handwritten digits from the National Institute of Standards and Technology (NIST) Special Database 19. The dataset consists of a training set of 60,000 28×28 pixel images of the digits 0 to 9, along with a testing set of 10,000 digits. MNIST has been widely used for assessing the accuracy of machine learning approaches, and is used here as it allows behaviour of the current approach to be compared against the state-of-the-art pattern recognition methods.

In recent years, the medical imaging community has seen many of the established "old-school" approaches replaced by deep learning, but in doing so, "have we thrown the baby out with the bath water?".[5] There may still be widely used concepts from orthodox medical imaging (i.e., not deep learning) that are still useful. In particular, geometric transformations of images are now finding their way into various machine learning approaches (e.g. Hinton et al., 2011; Taigman et al., 2014; Jaderberg et al., 2015). Much of the early work on deep learning was performed using MNIST. Although good accuracies were achieved, the computer vision community did not take such work seriously because the images were so small. This, however, was the early days of deep learning (i.e.,

[4] http://yann.lecun.com/exdb/mnist/.

[5] This was said by the late David MacKay (MacKay, 2003) in relation to the success of kernel methods, such as support-vector machines or Gaussian processes, which, at the time, were replacing neural networks in practical applications.

Fig. 2. Shape and appearance fit shown for a randomly selected sample of the KDEF face images. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

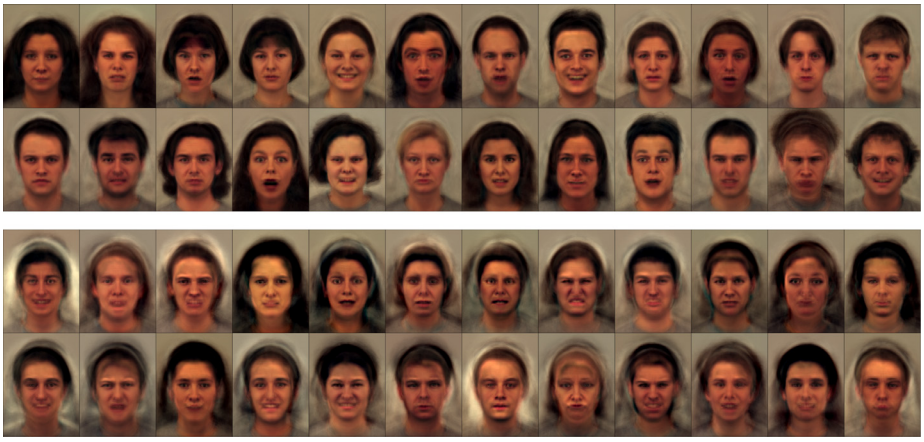

Fig. 3. Fits using a simple 64-mode principal component analysis model are shown above (cf. Fig. 2), and random faces generated from the same PCA model are shown below (cf. Fig. 4). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

before 2012), and was a sign of things to come. This section describes an attempt to begin to reclaim some of the territory lost to deep learning.

Unlike most conventional pattern recognition approaches, the strategy adopted here is generative. Training involves learning independent models of the ten different digits in the training set, while testing involves fitting each model in turn to each image in the test set, and performing model comparison to assess which of the ten models better explains the data. The training stage involved learning $\hat{\boldsymbol{\mu}}$, $\hat{\mathbf{W}}^{a}$, $\hat{\mathbf{W}}^{v}$ and $\hat{\mathbf{A}}$ for each digit class. A similar strategy was previously adopted by Revow et al. (1996). From a probabilistic perspective, the probability of the kth label given an image ($\mathbf{f}$)

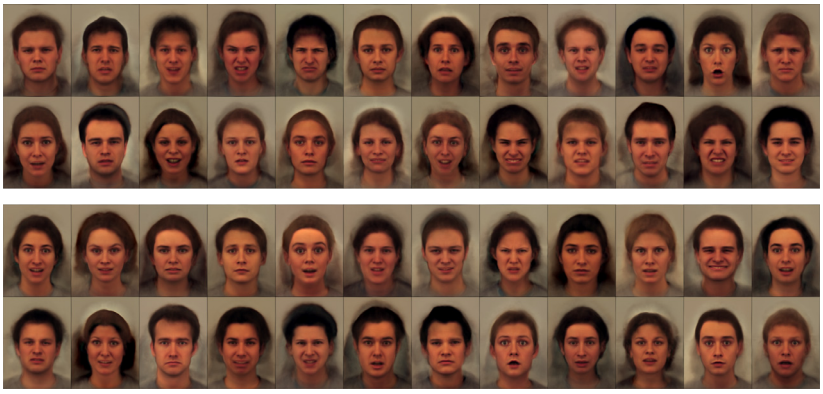

Fig. 4. Random faces generated from the shape and appearance model. The lower set of faces were generated with the same latent variables as those shown in the upper set, except the values were multiplied by −1 and thus show a sort of "opposite" face. For example, if a face in the top set has a wide open mouth, then the mouth should be tightly closed in the corresponding image of the bottom set. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

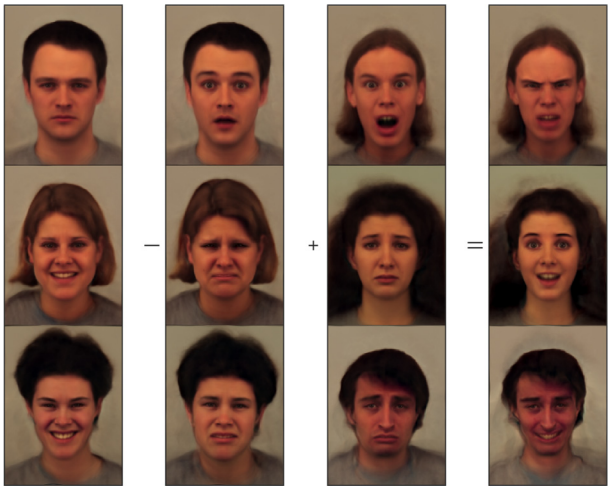

Fig. 5. An example of simple linear additions and subtractions applied to the latent variables. The first three columns show the full shape and appearance model fits to various faces. Images in the right hand column were generated by making linear combinations of the latent variables that encode the images in the first three columns, and then reconstructing from these linear combinations. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

is

$$P(\mathcal{M}_k|\mathbf{f}) = \frac{P(\mathbf{f},\mathcal{M}_k)}{P(\mathbf{f})} = \frac{\int_{\mathbf{z}} P(\mathbf{f}|\mathbf{z},\mathcal{M}_k)p(\mathbf{z}|\mathcal{M}_k)d\mathbf{z}P(\mathcal{M}_k)}{\sum_{l=0}^{9}\int_{\mathbf{z}} P(\mathbf{f}|\mathbf{z},\mathcal{M}_l)p(\mathbf{z}|\mathcal{M}_l)d\mathbf{z}P(\mathcal{M}_l)} \quad (52)$$

The above integrals are intractable, so are approximated. This was done by a "Laplace approximation"[6] whereby the approximate distribution of $\mathbf{z}$ is given by

$$q(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\hat{\mathbf{z}},\mathbf{S}^{-1}) \quad (53)$$

From this approximation, we can compute

$$\begin{aligned}\int_{\mathbf{z}} P(\mathbf{f},\mathbf{z}|\mathcal{M})d\mathbf{z} &\simeq P(\mathbf{f},\hat{\mathbf{z}}|\mathcal{M})\int_{\mathbf{z}}\exp\left(-\tfrac{1}{2}(\mathbf{z}-\hat{\mathbf{z}})^T\mathbf{S}(\mathbf{z}-\hat{\mathbf{z}})\right)d\mathbf{z}\\ &= P(\mathbf{f},\hat{\mathbf{z}}|\mathcal{M})\,|\mathbf{S}/(2\pi)|^{1/2}\end{aligned} \quad (54)$$

For each image ($\mathbf{f}$), the mode ($\hat{\mathbf{z}}$) of $p(\mathbf{f},\mathbf{z}|\mathcal{M}_k)$ was computed (see Section 2.2.6) by

$$\begin{aligned}\hat{\mathbf{z}} = \arg\min_{\mathbf{z}}\Big(&J(\mathbf{f},\mathbf{z},\boldsymbol{\mu},\hat{\mathbf{W}}^a,\hat{\mathbf{W}}^v)\\ &+\tfrac{1}{2}\mathbf{z}^T\left(\lambda_1\hat{\mathbf{A}}+\lambda_2(\hat{\mathbf{W}}^a)^T\mathbf{L}^a\hat{\mathbf{W}}^a+\lambda_2(\hat{\mathbf{W}}^v)^T\mathbf{L}^v\hat{\mathbf{W}}^v\right)\mathbf{z}\Big).\end{aligned} \quad (55)$$

The Hessian of the objective function around this mode (2.2.6) was used to approximate the uncertainty ($\mathbf{S}^{-1}$).

Training was done with different sized subsets (300, 500, 1000, 3000, 5000, 10,000, and all 60,000) of the MNIST training data, whereas testing was always done using the 10,000 test images. In each of the training subsets, the first of the images were always used, which generally leads to slightly different sized training sets for each of the digits. Example images, along with the fit from the models trained using the first 10,000 images, are shown in Fig. 6. Model fitting was run for 20 iterations, using a Bernoulli likelihood with $K = 16$, $\nu_0 = 16$, $\lambda = [0.95\ 0.05]$, $\omega^a = [0.002\ 0.2\ 0]$, $\omega^\mu = N[10^{-7}\ 10^{-5}\ 0]$ and $\omega^\nu = [0.002\ 0.02\ 2\ 0.2\ 0.2]$.

When applied to medical images, machine learning can suffer from the curse of dimensionality. The number of pixels or voxels in each image (M) is often much greater than the number of labelled images (N) available for training. For MNIST, there are 60,000 training images, each containing 784 pixels, giving $N/M \simeq 75$. In contrast, even after down-sampling to a lower resolution, a 3D MRI scan contains in the order of 20,000,000 voxels. Achieving a similar N/M as for MNIST would require about 1.5 billion labelled images, which clearly is not feasible. For this reason, this section focuses on classification methods trained using smaller subsets of the MNIST training data. Accuracies are compared against those reported by Lee et al. (2015) for their Deeply Supervised Nets, which is a deep learning approach that performs close to state-of-the-art (for 2015), particularly for smaller training sets. Invariant scattering convolutional networks are also known to work well for smaller training sets, so some accuracies taken from Bruna and Mallat (2013) are also included in the comparison. We are not aware of more recent papers that assess the accuracy of deep learning using smaller training sets.

Plots of error rate against training set size are shown in Fig. 7, along with the approximate error rates from Lee et al. (2015) and Bruna and Mallat (2013). The plot shows the proposed method to be more accurate than deep learning for smaller training sets, but it is less accurate when using the full training set, as the error rate plateaus to a value of about 0.85% for training set sizes of around 5000 onward. Visual assessment of the fits to the misclassified digits (Fig. 7) suggests that relatively few of the failures can be attributed to registration errors.

These experiments with MNIST suggest that one avenue of further work could be to elaborate on the simple multivariate Gaus-

[6] For a textbook explanation of Bayesian approaches, including the Laplace approximation, see MacKay (2003), Bishop et al. (2006) or Murphy (2012).

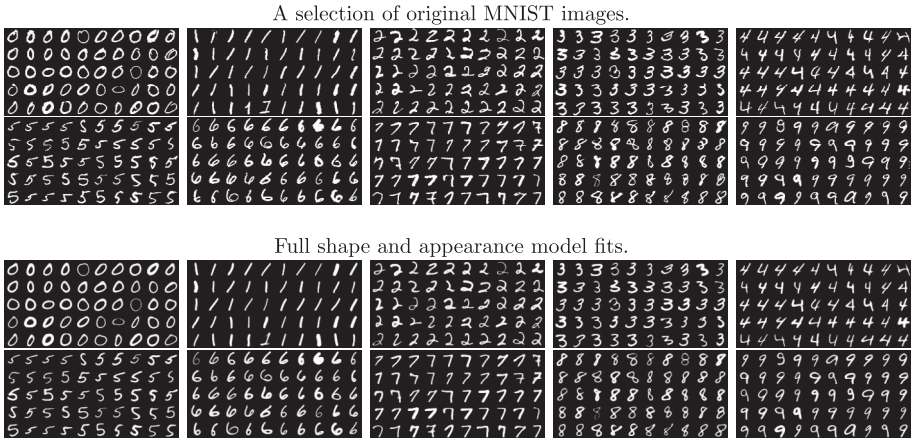

Fig. 6. A random selection of digits from the first 10,000 MNIST training images, along with the model fit. In general, good alignment is achieved.

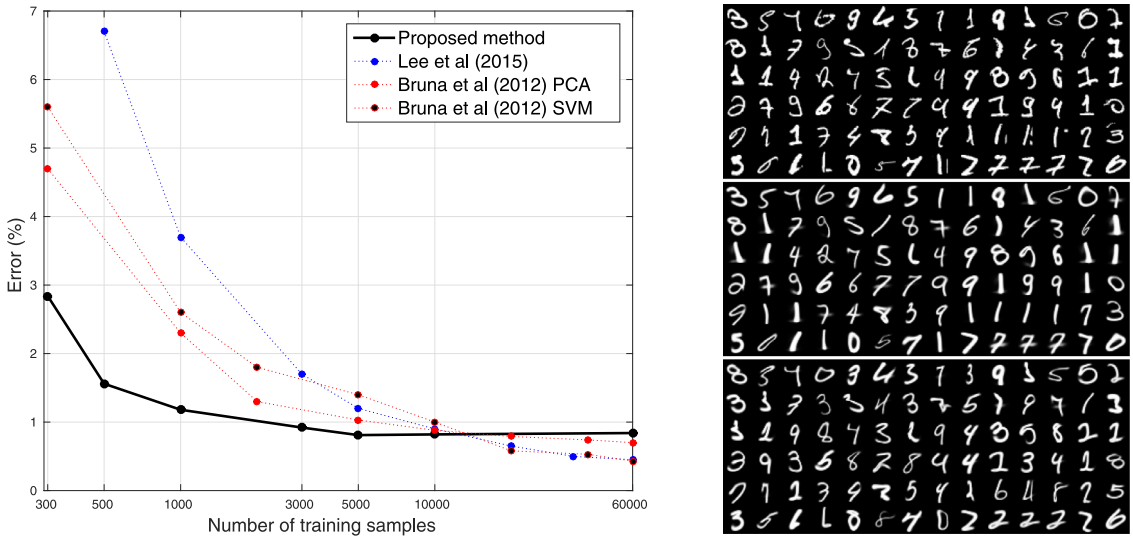

Fig. 7. Left: Test errors from training the method using different sized subsets of the MNIST data (the error rate from random guessing would be 90%). Right: All the MNIST digits the method failed to correctly identify (after training with the full 60,000) are shown above. These are followed by the model fits for the true digit, and then the model fits for the incorrect guess (i.e., the one with the most model evidence). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

sian model for the distribution of latent variables. Although accuracies were relatively good for smaller training sets, the Gaussian assumptions meant that increasing the amount of training data beyond about 5000 examples did not bring any additional accuracy. One example of where the Gaussian distribution fails is when attempting to deal with sevens written either with or without a bar through them, which clearly requires some form of bimodal distribution to describe (see Fig. 8). One approach to achieving a more flexible model of the latent variable probability density would to use a Gaussian Mixture Model (GMM) (Cootes and Taylor, 1999). One of the aims of the Medical Informatics Platform of the HBP was to cluster patients into different sub-groups. In addition to possibly achieving greater accuracy, incorporating a GMM over the latent variables could also lead to this clustering goal being achieved.

3.3. Experiments with segmented MRI

Experiments were performed using 1913 T1-weighted MR images from the following datasets.

- The *IXI* dataset, which is available under the Creative Commons CC BY-SA 3.0 license from http://brain-development.org/ixi-dataset/. Information about scanner parameters and subject demographics are also available from the web site. Scans were collected on three different scanners using a variety of MR sequences. This work used only the 581 T1-weighted scans.
- The *OASIS Longitudinal* dataset is described in Marcus et al. (2010). The dataset contains longitudinal T1-weighted MRI scans of elderly subjects, some of whom had dementia. Only data from the first 82 subjects of this dataset were downloaded from http://www.oasis-brains.org/, and averages of the scans acquired at the first time point were used.
- The *COBRE* (Centre for Biomedical Research Excellence) dataset are available for download from http://fcon_1000.projects.nitrc.org/indi/retro/cobre.html under the Creative Commons CC BY-NC license. The dataset includes fMRI and T1-weighted scans of 72 patients with Schizophrenia and 74 healthy controls. Only the T1-weighted scans were used. Information about scanner parameters and subject demographics is available from the web site.
- The *ABIDE I* (Autism Brain Imaging Date Exchange) dataset was downloaded via http://fcon_1000.projects.nitrc.org/indi/abide/abide_I.html and is available under the Creative Commons CC BY-NC-SA license. There were scans from 1102 subjects, where 531 were individuals on the Autism Spectrum. Subjects were

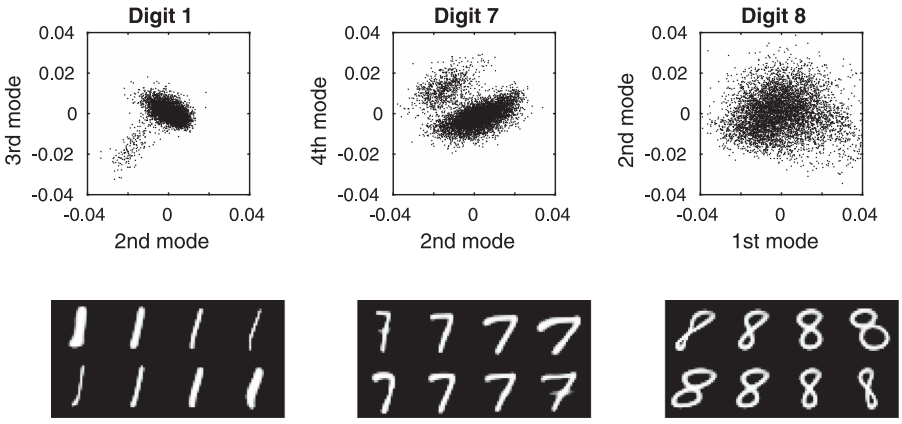

Fig. 8. Illustration of the non-Gaussian distributions of the latent variables for some of the MNIST digits. Plots of selected latent variables are shown above, with the corresponding modes of variation shown below. Gaussian mixture models are likely to provide better models of variability than the current assumption of a single Gaussian distribution.

drawn from a wide age range and were scanned at 17 different sites around the world. All the T1-weighted scans were used, and these had a very wide range of image properties, resolutions and fields of view. For example, many of the scans did not cover the cerebellum.

The images were segmented using the algorithm in SPM12, which uses the approach described in Ashburner and Friston (2005), but with some additional modifications that are described in the appendices of Weiskopf et al. (2011) and Malone et al. (2015). Binary maps of grey and white matter were approximately aligned into ICBM152 space using a rigid-body transform obtained from a weighted Procrustes analysis (Gower, 1975) of the deformations estimated by the segmentation algorithm. These approximately aligned images have an isotropic resolution of 2 mm.

3.3.1. 2D experiments with segmented MRI

It is generally easier to visualise how an algorithm is working when it is run in 2D, rather than 3D. The examples here will be used to illustrate the behaviour of the algorithm under topological changes, when variability can not be modelled only via diffeomorphic deformations.

A single slice was extracted from the grey and white matter images of each of the 1913 subjects, and the joint shape and appearance model was fit to the data using the settings for categorical image data. This assumed that each voxel was a categorical variable indicating one of three tissue classes (grey and white matter, as well as background). Each 2D image was encoded by 100 latent variables (i.e. $K = 100$). Eight iterations of the algorithm were used, with $\lambda = [0.9\ \ 0.1]$, $\omega^a = [0.1\ \ 16\ \ 128]$, $\omega^\mu = N[0.0001\ \ 0.01\ \ 0.1]$, $\omega^\nu = [0.001\ \ 0\ \ 32\ \ 0.25\ \ 0.5]$ and $\nu_0 = 100$.

Some model fits are shown in Fig. 9, and the principal modes of variability are shown in Fig. 10, which shows that these images are reasonably well modelled. Note that the topology of the images may differ, which (by definition[7]) is not something that can be modelled by diffeomorphisms alone. The inclusion of the appearance model allows these topology differences to be better captured.

3.3.2. Imputing missing data

The ability to elegantly handle missing data is a useful requirement for mining hospital scans. These often have limited fields of view, and may miss out parts of the brain that are present in other images. The objective here is to demonstrate that a reasonable image factorisation can be learned, even when some images in the dataset may not have full organ coverage.

This experiment used the same slice through the data as above, and a rectangle covering 25% of the area of the images was placed randomly in each and every image of the training set (wrapping around at the edge of the field of view), and the intensities within these rectangles set to NaN ("not a number" in the IEEE 754 floating-point standard). The algorithm was trained, using the same settings as described previously, on the these modified images. Although imputed missing values may not be explicitly required, they do provide a useful illustration of how well the model works in less than ideal situations. Fig. 12 shows a selection of the images with regions set to NaN, and the same images with the missing values predicted by the algorithm.

The ability to handle missing data allows cross-validation to be used to determine the accuracy of a model, and how well it generalises. In addition to the joint shape and appearance model, this work also allows simplified versions to be fitted that involve only shape (i.e., not using $\mathbf{W}^a$, as in Zhang and Fletcher (2015)) or in a form that varies only the appearance (i.e. not using $\mathbf{W}^v$). This work also includes a version where different sets of latent variables control the shape and appearance. Here, there were 30 variables to control appearance ($K^a = 30$ in (Eq. (13)), and 70 to control shape ($K^v = 70$ in (Eq. (14))). The aim was to compare the four models by assessing how well they are able to predict data that was unavailable to the model during fitting. This gives us ground truth with which to compare the models' predictions, and is essentially a form of cross-validation procedure. Accuracy was measured by the log-likelihood of the ground truth data, which was computed only for pixels that the models did not have access to during training.

The results of the cross-validation are shown in Fig. 13, and show that the two models that combine both shape and appearance have greater predictive validity than either the shape or appearance models alone. To clarify the general pattern, the log-likelihoods of each patch were also plotted after subtracting their mean log-likelihood over all model configurations. Although the difference was small, the best results were from the model where each latent variable controls both shape and appearance, rather than when they are controlled separately ($p < 10^{-5}$ from a paired t-test).

Changes to hyper-parameter settings, etc. may improve accuracies further. The effects of changing ω^a and ω^v were assessed by running a similar comparison using the model where the same

[7] Topology is concerned with properties that are preserved following diffeomorphic deformations (see https://en.wikipedia.org/wiki/Topology).

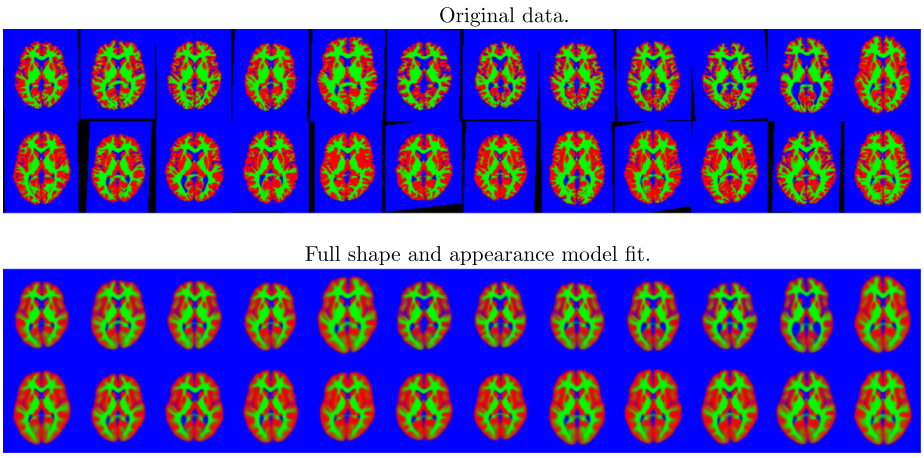

Fig. 9. A random selection of the 2D brain image data, showing grey matter (red), white matter (green) and other (blue). Black regions indicate missing data. Below these is the model fit to the images. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

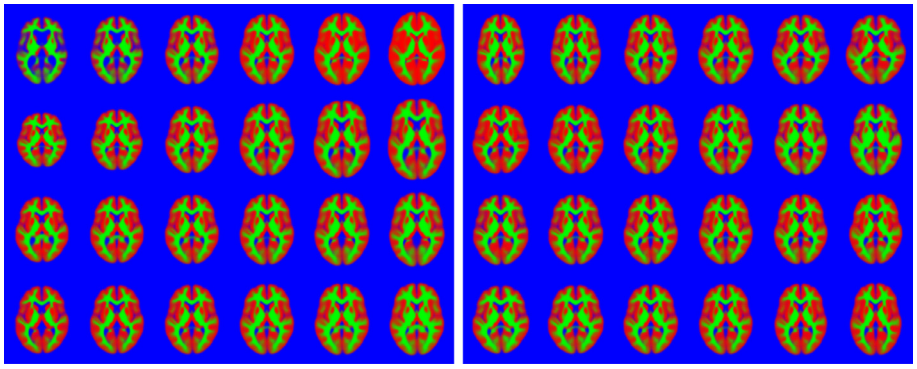

Fig. 10. First eight (out of a total of 100) modes of variability found from the 2D brain image dataset, shown at −5, −3, −1, +1, +3 & +5 standard deviations. Note that these modes encode some topological changes, in addition to changes in shape. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

latent variables control both shape and appearance. The hyper-parameter settings were varied over two orders of magnitude by scaling the previously used settings by 0.1, 1 and 10. In addition, the settings for ω^{μ} were decreased by a factor of 100. Results are shown in Fig. 14, and gave the best accuracies with $\omega^{a} = [0.01\ 1.6\ 12.8]$ and $\omega^{v} = [0.001\ 0\ 32\ 0.25\ 0.5]$. Using the smaller ω^{μ} made an insignificant difference to the average log likelihoods (result not shown). Paired t tests between all pairs of comparisons showed that the choice of hyperparameter settings plays an important role. A similar comparison could also be made by varying other hyper-parameter settings.

3.3.3. 3D experiments with segmented MRI

The aim of this section was to apply the method to a large set of 3D images, and use the resulting latent variables as features for pattern recognition. For this, a version of the model was used whereby some latent variables controlled appearance, whereas others controlled shape. The motivation for this was that it allows the different types of features to be differentially weighted when they are used to make predictions.

The algorithm was run on the full 3D dataset, using 70 variables to control shape ($K^{v} = 70$) and 30 to control appearance ($K^{a} = 30$). Eight iterations were used, with $\lambda = [1\ 1]$, $\omega^{a} = [0.01\ 1\ 50]$, $\omega^{\mu} = N[0.00001\ 0.01\ 0.1]$ and $\omega^{v} = [0.001\ 0\ 10\ 0.1\ 0.2]$. Slice 40 of the resulting mean image is shown in Fig. 15, alongside the mean from one of the 2D experiments. Note that the mean from the 2D model is slightly crisper than that from the one in 3D. The main reason for this is simply that it is a 3D fit, so that there is a great deal more variability to explain. Achieving a similar quality of fit for the full 3D data, as was achieved for the 2D data, would require in the order of 1000 ($100^{3/2}$) variables.

The main objective of this work is to extract a small number of features from sets of anatomical medical images, which are effective for machine learning applications. Here, a five-fold cross-validation is used to assess the effectiveness of these features. Machine learning used a linear Gaussian process classification procedure, which is essentially equivalent to a Bayesian approach to logistic regression. The implementation was based on the method for binary classification using expectation propagation described in Rasmussen and Williams (2006). For the COBRE dataset, classification involved separating controls from patients with schizophrenia. Similarly, the analysis of the ABIDE dataset involved identifying those subjects on the autism spectrum, with features orthogonalised with respect to the different sites. Classification involved three hyper-parameters, which weighted the contributions from shape features, appearance features and a constant offset. Resulting ROC curves are shown in Fig. 16.

For ABIDE, the accuracy and 95% confidence interval was $57.6 \pm 2.9\%$. While this is not especially high, it is close to the accuracy reported by others who have applied machine learning to the T1-weighted scans. Most previous works (Haar et al., 2014; Katuwal et al., 2015; Ghiassian et al., 2016) have reported their best classification accuracies of around 60% when using the same dataset. Results are roughly comparable with some of the accuracies obtained by Monté-Rubio et al. (2018) or Demirhan (2018). Those papers reported multiple accuracies, so it would be difficult to choose a single accuracy with which to compare.

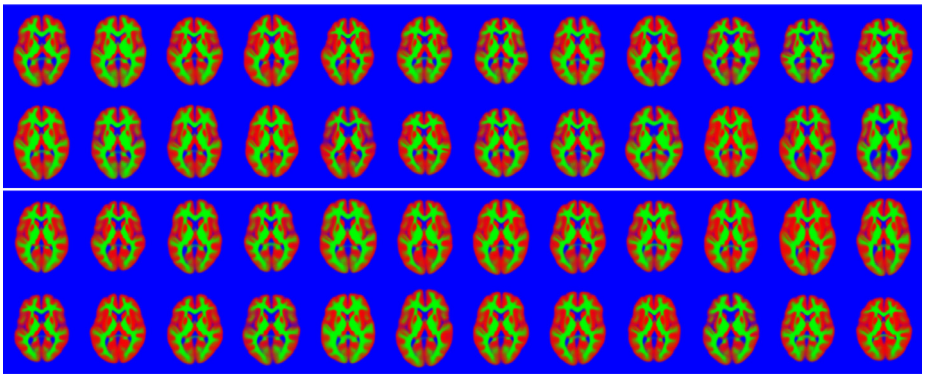

Fig. 11. Randomly generated slice through brain images. These images were constructed by using randomly assigned latent variables. Note that the top set of images uses the same random variables as the bottom set, except they are multiplied by −1. This means that one set is a sort of "opposite" of the other. For example, if a brain in the upper set has large ventricles, then the corresponding brain in the lower set will have small ventricles. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

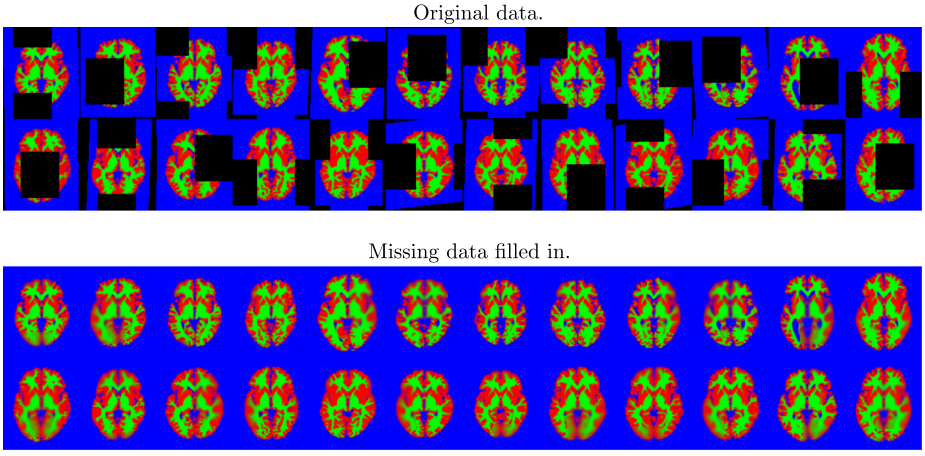

Fig. 12. A random selection of the 2D brain image data showing the location of missing data. The attempt to fill in the missing information is shown below. These may be compared against the original images shown in Fig. 9. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

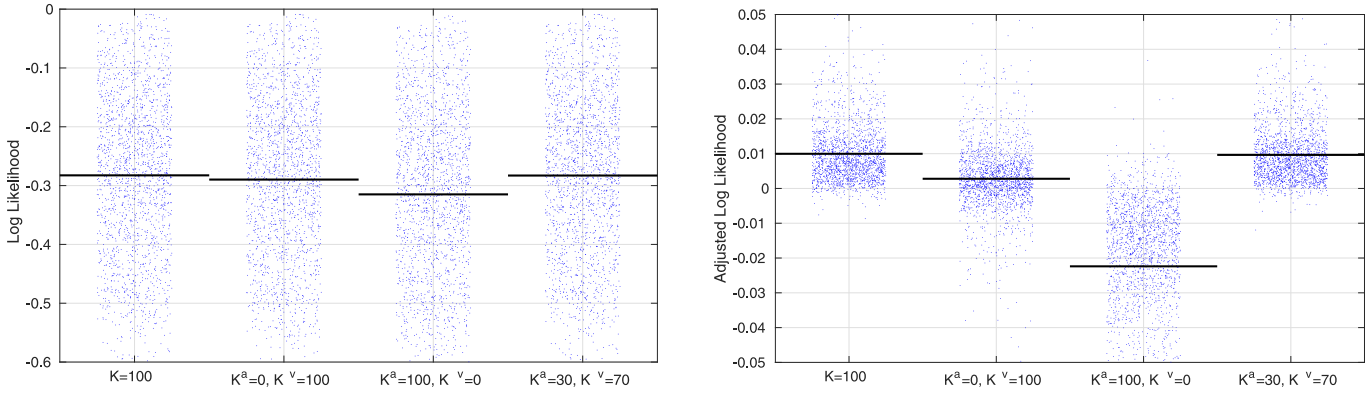

Fig. 13. Cross-validation accuracy measures based on predicting the left-out patches of the images using different model configurations. The blue dots show the mean value for each of the 1913 images, whereas the horizontal bars show the mean values overall. The plot on the left shows mean log-likelihoods over the pixels in each patch, wheres the plot on the right shows the log-likelihoods after subtracting the mean – over model configurations – for each patch.

The accuracy achieved for the COBRE dataset was $74.7 \pm 7.1\%$, which is similar to the 69.7% accuracy reported by Cabral et al. (2016) using COBRE, and was roughly comparable with many of the accuracies obtained by Monté-Rubio et al. (2018) or Demirhan (2018). Others have used other datasets of T1-weighted scans for identifying patients with schizophrenia. Nieuwenhuis et al. (2012) achieved 71.4% and Ma et al. (2018) achieved 75.8% accuracy for separating controls from subjects with schizophrenia, but using larger datasets.

Anatomical T1-weighted MRI is unlikely to be the most useful type of data for assessing psychiatric disorders, and better classification accuracies have been achieved using other modalities, such as fMRI (Silva et al., 2014). We note that some other papers have reported much higher accuracies using the COBRE dataset,

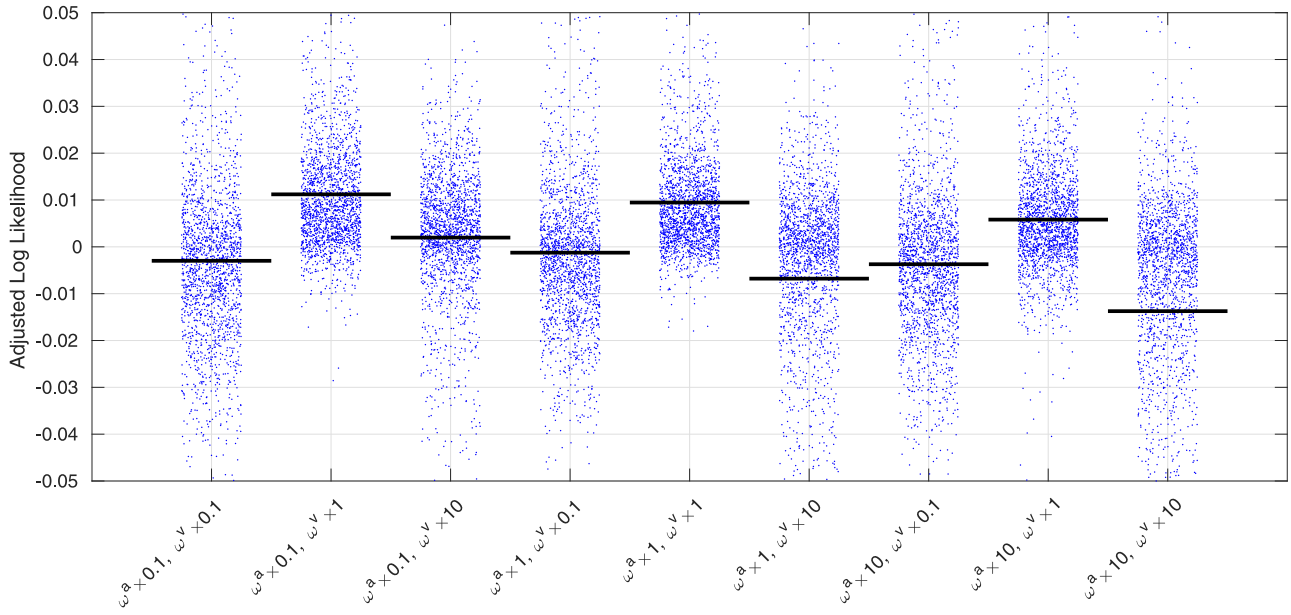

Fig. 14. Cross-validation accuracy measures based on predicting the left-out patches of the images using different hyper-parameter settings. The blue dots show the mean value for each of the 1913 images, whereas the horizontal bars show the mean values overall. Accuracy measures are mean log-likelihoods (over voxels), after adjustment.

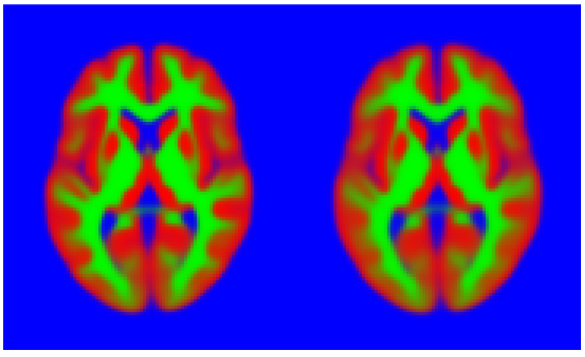

Fig. 15. An illustration of the mean images from the 2D and 3D experiments (after Softmax). Left: The mean image from the 2D experiments (c.f. Figs. 9 and 10). Right: Slice 40 of the mean image from the 3D experiment. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

but many of these works made use of manual annotations or may not have kept a strict separation between testing and training data.

3.4. Experiments with head and neck

Most conventional image registration algorithms involve some form of local optimisation, and are therefore susceptible to getting caught in local optima. Good initialisation can help avoid such optima. This is often achieved by registering via a rigid or affine transform, which captures some of the main modes of shape variability. However, this does not capture the main ways that biological structures may vary in shape, and it may be possible to do better. In this section, we examine how suited the proposed model is to this task by comparing "groupwise" registrations initialised with affine transforms versus those initialised using the proposed method. The Ants software[8] (Avants et al., 2014) was used for this, as it is widely accepted to be an effective image registration package.

The data were the 581 T1-weighted scans from the IXI dataset, which were approximately rigidly aligned and downsampled to an isotropic resolution of 1.75 mm. The resulting images all had dimensions of $103 \times 150 \times 155$ with a field of view that covered both head and neck, and were scaled to have maximum value of 1.0. Approximately binary masks of the brains within the original T1-weighted scans were extracted using the segmentation module (Ashburner and Friston, 2005) of the SPM12 software[9], and these were also transformed in the same way.

1. For the case where Ants was initialised via affine transforms, registration was run serially in 3D using one of the scripts released with the software (Avants et al., 2010; 2011). The script first corrected the images for smooth variations in intensity nonuniformity using N4 (Tustison et al., 2010), and the actual registration minimised the local correlation coefficients via a greedy gradient descent.
   `antsMultivariateTemplateConstruction.sh -d 3 -c 0 -o ants *.nii`
   The warps generated by Ants were applied to all the brain masks to bring them into a common space.
2. The proposed method was also run on the data, using 20 iterations with the Gaussian noise model, $K^a = 4$, $K^v = 60$, $\omega^v = [0.01\ 0\ 10\ 1\ 2]$, $\omega^a = [100\ 1000\ 0]$, $\omega^\mu = [0.01\ 10\ 0]$, $\nu_0 = 140$ and $\lambda = [9.5\ 0.5]$. The resulting parameter estimates were then used to warp all the images to approximately match the mean, before the alignment was refined further by applying Ants to these warped images. Warps generated by the proposed model were composed with those generated by Ants, and the result was used to warp all the brain masks into a common space.

The mean ($\boldsymbol{\mu}$) of all the binarised aligned mask images was computed and the following Jaccard and binomial log-likelihood overlap measure derived for each ($\mathbf{b}$) of them.

$$J(\boldsymbol{\mu}, \mathbf{b}) = \frac{\sum_{m=1}^{M}((\mu_m > \frac{1}{2}) \wedge b_m)}{\sum_{m=1}^{M}((\mu_m > \frac{1}{2}) \vee b_m)}$$

[8] https://github.com/ANTsX/ANTs.

[9] https://www.fil.ion.ucl.ac.uk/spm/software/spm12/.

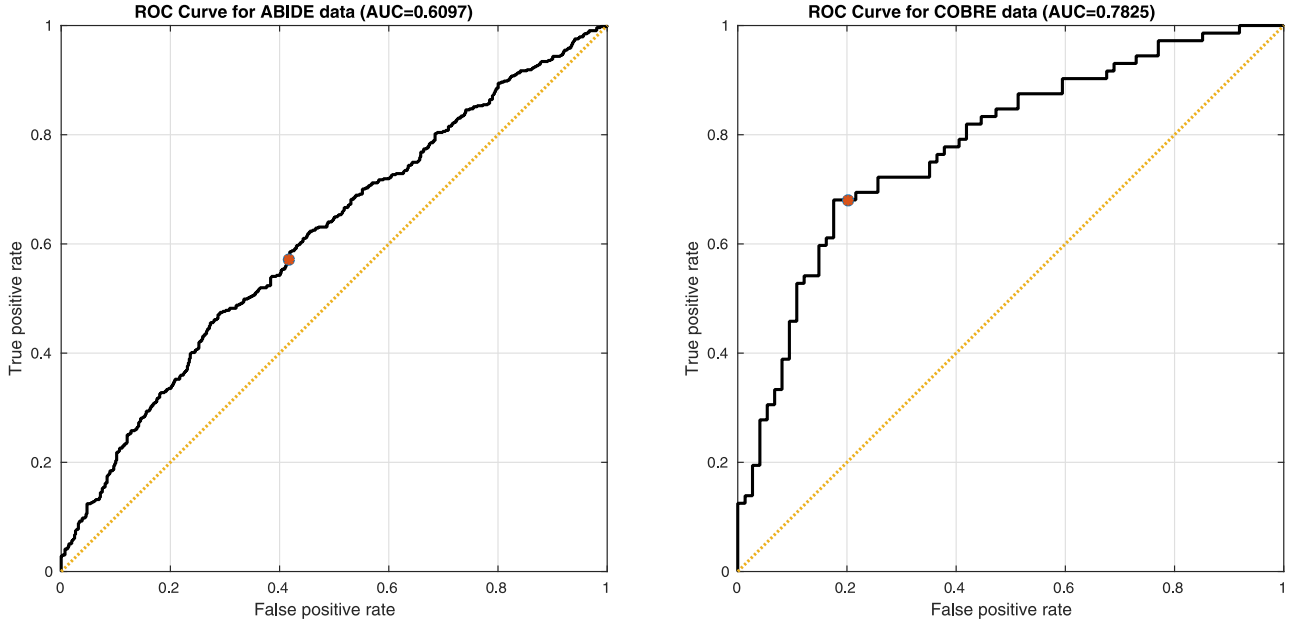

Fig. 16. ROC curves from five-fold cross-validation accuracies from the ABIDE and COBRE data. Red dots show the point on the curve where the classification gives probabilities of 0.5.

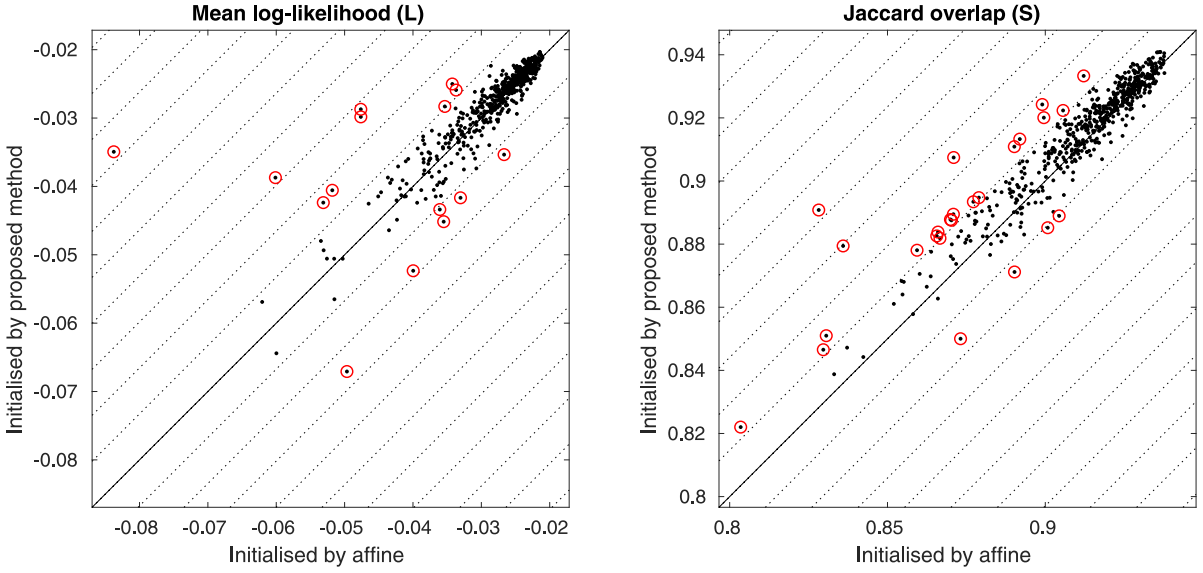

Fig. 17. Overlap measures from the two registration approaches. Diagonal lines are spaced two standard deviations apart. Circled points indicate outliers of more than two standard deviations.

$$L(\boldsymbol{\mu}, \mathbf{b}) = \frac{1}{M}\sum_{m=1}^{M}(b_m \log_2 \mu_m + (1-b_m)\log_2(1-\mu_m)) \tag{56}$$

We note that these measures reflect overlap of "spatially normalised" images, which is what typically interests many users of registration software.[10]The resulting overlap measures are shown in Fig. 17, and are mostly similar between the two approaches. However, the pattern of outliers (more outliers in the top left than in the bottom right) suggests that using the proposed approach to initialise registration leads to slightly more robust alignment. An analysis based on the Jacard overlap, counting outliers beyond 2 standard deviations, would show a clear benefit of the proposed method, but the pattern is less certain when the log-likelihood measures are also considered. Because the numbers of outliers are relatively small, it is difficult to draw firm statistical conclusions.

Fig. 18 shows the mid-sagittal slice through a selection of the basis functions estimated by the proposed model. The four appearance basis functions were intended to capture variability across scanners, plus a few other sources of signal intensity variability such as that of bone marrow in the skull. Rather than the individual components of the shape basis functions, their divergence is shown instead in Fig. 18. These divergence maps encode expansion or contraction within the diffeomorphic deformations. The first of these is mostly concerned with overall head size (and suggest that larger heads are associated with greater bulk at the back of the neck), whereas the second and third components appear to mostly capture variability related to the amount of body fat – particularly in the neck. Other shape components encode neck angulation

[10] From a modelling perspective, the overlaps would have been better computed by warping the mean to match each individual image.

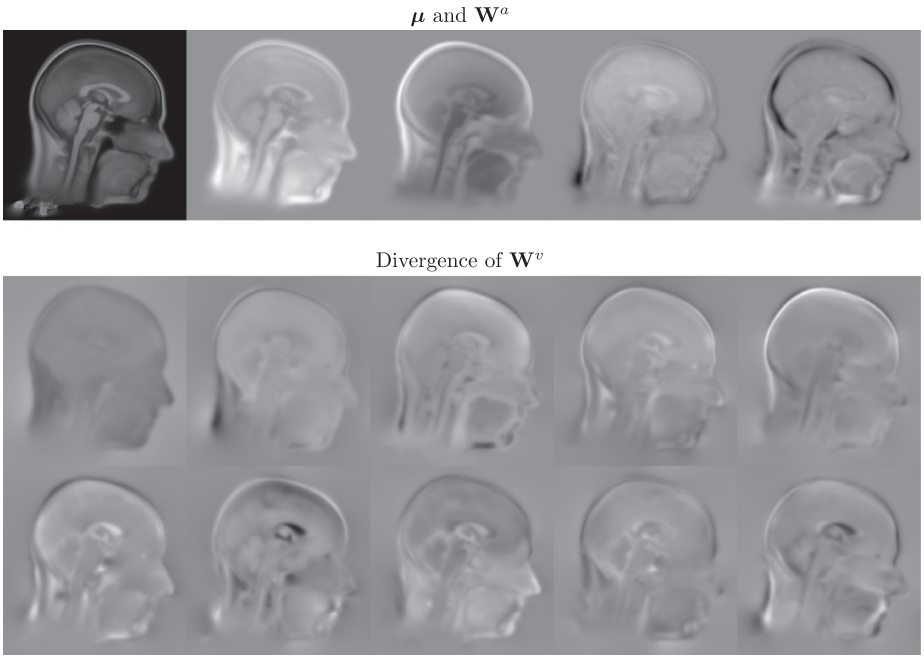

Fig. 18. Mid-sagittal slice through the basis functions. The mean ($\boldsymbol{\mu}$) and four appearance basis functions ($\mathbf{W}^a$) are shown above, while the divergences of the first 10 shape basis functions ($\mathbf{W}^v$) are shown below.

and various other aspects of head shape variability. The proposed model was run with only 60 shape components because the intention was to assess its utility for capturing the main modes of variability, as a precursor to the finer alignment.

4. Discussion

This work presents a very general generative framework that may have widespread use within the medical imaging community, particularly for those situations where conventional image registration approaches are more likely to fail. Because of its generality, the model we presented should provide a good starting point for a number of avenues of further development.

Most image analysis applications have a number of settings to be tuned, and the current approach is no exception. Although this tuning is rarely discussed in papers, the settings can have quite a large impact on any results. We propose that a cross-validation strategy, as shown in Section 3.3.2, could be used for this. The approach taken in this work is simply to treat the construct as a model of the data, and to assess it according to how well it describes and predicts the observations. This work does not consider identifiability issues relating to how well it can separately estimate shape information versus appearance information.

Additional attention is the setting of λ_1 and λ_2 may be needed. From the perspective of the underlying generative model used, these settings should ideally sum to 1. In practice however, greater regularisation ($\lambda_1 + \lambda_2 > 1$) is required in order to achieve good results. A plausible explanation for this would be that assumptions of i.i.d. noise are not generally met, so a "virtual decimation factor", which accounts for correlations among residuals, may need to be accounted for Groves et al. (2011). The fact that the approach is not fully Bayesian (i.e., it only makes point estimates of many parameters and latent variables, rather than properly accounting for their uncertainty) may be another reason why additional regularisation is needed.

One aspect of the presented approach that is slightly unconventional is the scaling by N of $\mathbf{L}^v$ and $\mathbf{L}^a$ in (Eqs. (7)) and (8). Normally when constructing probabilistic generative models, the priors should not be adjusted according to how much data is available. An exception was made here because it has the effect of pushing the solution towards the basis functions encoding unit variance, rather than a variance that scales with N, with a corresponding decrease in the variance of the latent variables. In terms of the overall model fit, this only influences the behaviour of the prior $p(\mathbf{A}) = \mathcal{W}_K(\mathbf{A}|\mathbf{I}/\nu_0, \nu_0)$, which in turn influences the variance of the latent variables. Without this Wishart prior, the scaling by N could have been omitted without affecting the overall model fits. An alternative strategy could have involved constraining the basis functions such that $(\mathbf{W}^v)^T\mathbf{L}^v\mathbf{W}^v = \mathbf{I}$.

Another limitation of our proposed shape and appearance model is that it assumes that appearance and shape evolve separately, such that the appearance changes are added to the mean, and then the results are deformed to match the individual images. It may be possible to achieve slightly improved results by incorporating a metamorphosis approach (Trouvé and Younes, 2005), which considers that shape and appearance evolve simultaneously. It is currently unclear whether the benefits from this type of elegant approach could bring enough practical benefit to make it worthwhile. Appearance changes and deformations are both typically relatively small, so an improvement in how the interaction between the two types of variability are handled seems unlikely to make an easily discernible difference.

There are a number of directions in which the current work could be extended. One avenue would be to allow some shape variability beyond what can be encoded by the first few eigenmodes. For example, Balbastre et al. (2018) combined the eigenmode representation with a model of additional shape variability, giving a framework that is conceptually related to that of Allassonnière et al. (2007), as this allows a covariance matrix over velocity fields to be defined and optimised.

The framework would also generalise further for handling paired or multi-view data, which could add a degree of supervision to the method. There have been a number of publications on generating age- or gender-specific templates, or on geodesic regression approaches (Niethammer et al., 2011; Fletcher, 2013) for modelling trajectories of ageing. Concepts from joint matrix fac-

torisation approaches, such as canonical correlation analysis (Bach and Jordan, 2005; Klami et al., 2013), could be integrated into the current work, and these could be used to allow the model fitting to be informed by age, gender, disease status etc.

Declaration of competing interest

The authors have no conflicts of interest to declare.

Acknowledgments

This project has received funding from the European Unions Horizon 2020 Research and Innovation Programme under Grant Agreement No. 720270 (HBP SGA1). YB has been supported by the MRC and Spinal Research Charity through the ERA-NET Neuron joint call (MR/R000050/1). The Wellcome Centre for Human Neuroimaging is supported by core funding from the Wellcome Trust [grant number 203147/Z/16/Z]. Funding for the OASIS dataset came from the following grants: P50 AG05681, P01 AG03991, R01 AG021910, P20 MH071616, U24 RR021382. Funding for the ABIDE I dataset came from a variety of sources, which include NIMH (K23MH087770 and R03MH096321), the Leon Levy Foundation, Joseph P. Healy and the Stavros Niarchos Foundation to the Child Mind Institute. The imaging data and phenotypic information of the COBRE dataset was collected and shared by the Mind Research Network and the University of New Mexico funded by a National Institute of Health Center of Biomedical Research Excellence (COBRE) grant 1P20RR021938-01A2.

Appendix A. Notation

In most of this paper, matrices are written in bold upper-case (e.g., $\mathbf{W}^a$, $\mathbf{Z}$, etc). In the computations, images are treated as vectors. These are written as lower-case bold, which includes the notation for individual columns of various matrices (e.g., $\mathbf{w}_k^a$ denotes the kth column of $\mathbf{W}^a$, $\mathbf{z}_n$ denotes the nth column of $\mathbf{Z}$, etc). Scalars are written in italic, with dimensions in upper-case. Estimates or expectations of parameters are written with a circumflex (e.g., $\hat{\mathbf{Z}}$). Collections of vectors may be conceptualised as matrices, so are written in bold-upper-case (e.g., $\mathbf{G}^a$, where individual vectors are $\mathbf{g}_k^a$). Collections of matrices are written in "mathcal" font (e.g., $\mathcal{H}^a$, where individual matrices are $\mathbf{H}_{kk}^a$). The matrix transpose operation is denoted by the "^{T}" superscript (as in $\mathbf{\Psi}^T$). Creating a diagonal matrix from a vector (as in diag(exp $\mathbf{q}$)), as well as treating the diagonal elements of a matrix as a vector (as in diag($\mathbf{Q}$)) are both denoted by "diag". The trace of a matrix (sum of diagonal elements) is denoted by "Tr".

This paper mixes both discrete and continuous representations of the same objects. For the discrete case, where a velocity field is treated as a vector, it is denoted by $\mathbf{v}_n$. Alternatively, the same object may be treated as a continuous 3D vector field, where it is denoted by v_n.

In addition, deformations may be treated as discrete or continuous. Within the continuous setting, warping an entire image by a diffeomorphism ψ may be denoted by $a' = a(\psi)$. In the discrete setting, this resampling may be conceptualised as a matrix multiplication, where a very large sparse matrix $\mathbf{\Psi}$ encodes the same deformation (and associated trilinear interpolation), such that $\mathbf{a}' = \mathbf{\Psi}\mathbf{a}$. The transpose of this matrix can be used to perform a push-forward operation, which is frequently used in this work and which we denote by $\mathbf{f}' = \mathbf{\Psi}^T\mathbf{f}$.

Sometimes, gradients of an image are required. In 3D, the three components of the spatial gradient of $\mathbf{a}$ are denoted by $\nabla_1\mathbf{a}$, $\nabla_2\mathbf{a}$ and $\nabla_3\mathbf{a}$.

References

Adams, D., Rohlf, F., Slice, D., 2004. Geometric morphometrics: ten years of progress following the 'revolution'. Ital. J. Zool. 71 (1), 5–16.
Allassonnière, S., Amit, Y., Trouve, A., 2007. Towards a coherent statistical framework for dense deformable template estimation. J. R. Stat. Soc. Ser. B (Stat. Methodol.) 69 (1), 3–29.
Allassonnière, S., Durrleman, S., Kuhn, E., 2015. Bayesian mixed effect atlas estimation with a diffeomorphic deformation model. SIAM J. Imaging Sci. 8 (3), 1367–1395.
Ashburner, J., 2007. A fast diffeomorphic image registration algorithm. Neuroimage 38 (1), 95–113.
Ashburner, J., Friston, K., 2005. Unified segmentation. Neuroimage 26 (3), 839–851.
Ashburner, J., Friston, K., 2011. Diffeomorphic registration using geodesic shooting and gauss-newton optimisation. Neuroimage 55, 954–967.
Ashburner, J., Ridgway, G.R., 2012. Symmetric diffeomorphic modeling of longitudinal structural MRI. Front Neurosci. 6.
Avants, B.B., Tustison, N.J., Song, G., Cook, P.A., Klein, A., Gee, J.C., 2011. A reproducible evaluation of ANTs similarity metric performance in brain image registration. Neuroimage 54 (3), 2033–2044.
Avants, B.B., Tustison, N.J., Stauffer, M., Song, G., Wu, B., Gee, J.C., 2014. The insight toolkit image registration framework. Front Neuroinf. 8, 44.
Avants, B.B., Yushkevich, P., Pluta, J., Minkoff, D., Korczykowski, M., Detre, J., Gee, J.C., 2010. The optimal template effect in hippocampus studies of diseased populations. Neuroimage 49 (3), 2457–2466.
Babalola, K.O., Patenaude, B., Aljabar, P., Schnabel, J., Kennedy, D., Crum, W., Smith, S., Cootes, T., Jenkinson, M., Rueckert, D., 2009. An evaluation of four automatic methods of segmenting the subcortical structures in the brain. Neuroimage 47 (4), 1435–1447.
Bach, F., Jordan, M., 2005. A Probabilistic Interpretation of Canonical Correlation Analysis, 688. Department Statistics University California, Berkeley, CA.
Balbastre, Y., Brudfors, M., Bronik, K., Ashburner, J., 2018. Diffeomorphic brain shape modelling using gauss-newton optimisation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, Cham, pp. 862–870.
Beg, M., Miller, M., Trouvé, A., Younes, L., 2005. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. Int. J. Comput. Vis. 61 (2), 139–157.
Belongie, S., Malik, J., Puzicha, J., 2002. Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Anal. Mach. Intell. 24 (4), 509–522.
Bishop, C., et al., 2006. Pattern Recognition and Machine Learning. Springer New York:.
Bro-Nielsen, M., Gramkow, C., 1996. Fast fluid registration of medical images. In: Visualization in Biomedical Computing. Springer, pp. 265–276.
Bruna, J., Mallat, S., 2013. Invariant scattering convolution networks. IEEE Trans. Pattern Anal. Mach. Intell. 35 (8), 1872–1886.
Cabral, C., Kambeitz-Ilankovic, L., Kambeitz, J., Calhoun, V.D., Dwyer, D.B., Von Saldern, S., Urquijo, M.F., Falkai, P., Koutsouleris, N., 2016. Classifying schizophrenia using multimodal multivariate pattern recognition analysis: evaluating the impact of individual clinical profiles on the neurodiagnostic performance. Schizophr. Bull. 42 (suppl_1), S110–S117.
Cootes, T., Twining, C., Babalola, K., Taylor, C., 2008. Diffeomorphic statistical shape models. Image Vis. Comput. 26 (3), 326–332.
Cootes, T.F., Edwards, G.J., Taylor, C.J., 2001. Active appearance models. IEEE Trans. Pattern Anal. Mach. Intell. 23 (6), 681–685.
Cootes, T.F., Taylor, C.J., 1992. Active shape models – 'Smart Snakes'. In: Proceedings of the BMVC92. Springer, pp. 266–275.
Cootes, T.F., Taylor, C.J., 1999. A mixture model for representing shape variation. Image Vis. Comput. 17 (8), 567–573.
Cootes, T.F., Taylor, C.J., 2001. Statistical models of appearance for medical image analysis and computer vision. In: Proceedings of the Medical Imaging. International Society for Optics and Photonics, pp. 236–248.
Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J., 1995. Active shape models – their training and application. Comput. Vis. Image Understand. 61 (1), 38–59.
Cootes, T.F., Twining, C.J., Petrovic, V.S., Babalola, K.O., Taylor, C.J., 2010. Computing accurate correspondences across groups of images. IEEE Trans. Pattern Anal. Mach. Intell. 32 (11), 1994–2005.
Datar, M., Muralidharan, P., Kumar, A., Gouttard, S., Piven, J., Gerig, G., Whitaker, R., Fletcher, P.T., 2012. Mixed-effects shape models for estimating longitudinal changes in anatomy. In: Proceedings of the International Workshop on Spatio-temporal Image Analysis for Longitudinal and Time-Series Image Data. Springer, pp. 76–87.
Demirhan, A., 2018. The effect of feature selection on multivariate pattern analysis of structural brain MR images. Phys. Med. 47, 103–111.
Fletcher, P.T., 2013. Geodesic regression and the theory of least squares on rlemannian manifolds. Int. J. Comput. Vis. 105 (2), 171–185.
Friston, K., Ashburner, J., Frith, C.D., Poline, J.-B., Heather, J.D., Frackowiak, R.S., et al., 1995. Spatial registration and normalization of images. Hum Brain Mapp. 3 (3), 165–189.
Friston, K.J., Penny, W., Phillips, C., Kiebel, S., Hinton, G., Ashburner, J., 2002. Classical and bayesian inference in neuroimaging: theory. Neuroimage 16 (2), 465–483.
Ghiassian, S., Greiner, R., Jin, P., Brown, M.R., 2016. Using functional or structural magnetic resonance images and personal characteristic data to identify ADHD and autism. PLoS ONE 11 (12), e0166934.
Gower, J.C., 1975. Generalized procrustes analysis. Psychometrika 40 (1), 33–51.

Groves, A.R., Beckmann, C.F., Smith, S.M., Woolrich, M.W., 2011. Linked independent component analysis for multimodal data fusion. Neuroimage 54 (3), 2198–2217.
Haar, S., Berman, S., Behrmann, M., Dinstein, I., 2014. Anatomical abnormalities in autism? Cerebral Cortex 26 (4), 1440–1452.
Hinton, G.E., Krizhevsky, A., Wang, S.D., 2011. Transforming auto-encoders. In: Proceedings of the International Conference on Artificial Neural Networks. Springer, pp. 44–51.
Jaderberg, M., Simonyan, K., Zisserman, A., et al., 2015. Spatial transformer networks. In: Proceedings of the Advances in Neural Information Processing Systems, pp. 2017–2025.
Katuwal, G.J., Cahill, N.D., Baum, S.A., Michael, A.M., 2015. The predictive power of structural MRI in autism diagnosis. In: Proceedings of the 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, pp. 4270–4273.
Klami, A., Virtanen, S., Kaski, S., 2013. Bayesian canonical correlation analysis. J. Mach. Learn. Res. 14 (Apr), 965–1003.
LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86 (11), 2278–2324.
Lee, C.-Y., Xie, S., Gallagher, P., Zhang, Z., Tu, Z., 2015. Deeply-supervised nets. In: Proceedings of the Artificial Intelligence and Statistics, pp. 562–570.
Lindner, C., Thomson, J., Cootes, T.F., arcOGEN Consortium, et al., 2015. Learning-based shape model matching: training accurate models with minimal manual input. In: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 580–587.
Litjens, G., Toth, R., van de Ven, W., Hoeks, C., Kerkstra, S., van Ginneken, B., Vincent, G., Guillard, G., Birbeck, N., Zhang, J., et al., 2014. Evaluation of prostate segmentation algorithms for MRI: the PROMISE12 challenge. Med. Image Anal. 18 (2), 359–373.
Lundqvist, D., Flykt, A., Öhman, A., 1998. The Karolinska Directed Emotional Faces (KDEF). CD ROM from Department of Clinical Neuroscience, Psychology Section, Karolinska Institutet, pp. 91–630.
Ma, Q., Zhang, T., Zanetti, M.V., Shen, H., Satterthwaite, T.D., Wolf, D.H., Gur, R.E., Fan, Y., Hu, D., Busatto, G.F., et al., 2018. Classification of multi-site MR images in the presence of heterogeneity using multi-task learning. NeuroImage: Clin. 19, 476–486.
MacKay, D.J., 2003. Information Theory, Inference and Learning Algorithms. Cambridge University Press.
Malone, I.B., Leung, K.K., Clegg, S., Barnes, J., Whitwell, J.L., Ashburner, J., Fox, N.C., Ridgway, G.R., 2015. Accurate automatic estimation of total intracranial volume: a nuisance variable with less nuisance. Neuroimage 104, 366–372.
Marcus, D., Fotenos, A., Csernansky, J., Morris, J., Buckner, R., 2010. Open access series of imaging studies: longitudinal MRI data in nondemented and demented older adults. J. Cogn. Neurosci. 22 (12), 2677–2684.
Alabort-i Medina, J., Zafeiriou, S., 2014. Bayesian active appearance models. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, pp. 3438–3445.
Miller, M., Trouvé, A., Younes, L., 2006. Geodesic shooting for computational anatomy. J. Math. Imaging Vis. 24 (2), 209–228.
Monté-Rubio, G.C., Falcón, C., Pomarol-Clotet, E., Ashburner, J., 2018. A comparison of various MRI feature types for characterizing whole brain anatomical differences using linear pattern recognition methods. Neuroimage 178, 753–768.
Murphy, K.P., 2012. Machine Learning: A Probabilistic Perspective. MIT Press.
Niethammer, M., Huang, Y., Vialard, F., 2011. Geodesic regression for image time-series. In: Proceedings of the Medical Image Computing and Computer-Assisted Intervention–MICCAI 2011. Springer, pp. 655–662.
Nieuwenhuis, M., van Haren, N.E., Pol, H.E.H., Cahn, W., Kahn, R.S., Schnack, H.G., 2012. Classification of schizophrenia patients and healthy controls from structural MRI scans in two large independent samples. Neuroimage 61 (3), 606–612.
Patenaude, B., Smith, S.M., Kennedy, D.N., Jenkinson, M., 2011. A Bayesian model of shape and appearance for subcortical brain segmentation. Neuroimage 56 (3), 907–922.
Radford, A., Metz, L., Chintala, S., 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv:1511.06434.
Rasmussen, C., Williams, C., 2006. Gaussian Processes for Machine Learning. Springer.
Revow, M., Williams, C.K., Hinton, G.E., 1996. Using generative models for handwritten digit recognition. IEEE Trans. Pattern Anal. Mach. Intell. 18 (6), 592–606.
Rueckert, D., Frangi, A.F., Schnabel, J., et al., 2003. Automatic construction of 3-d statistical deformation models of the brain using nonrigid registration. IEEE Trans. Med. Imaging 22 (8), 1014–1025.
Silva, R.F., Castro, E., Gupta, C.N., Cetin, M., Arbabshirani, M., Potluru, V.K., Plis, S.M., Calhoun, V.D., 2014. The tenth annual MLSP competition: Schizophrenia classification challenge. In: Proceedings of the IEEE International Workshop on Machine Learning for Signal Processing (MLSP). IEEE, pp. 1–6.
Štern, D., Ebner, T., Urschler, M., 2016. From local to global random regression forests: Exploring anatomical landmark localization. In: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 221–229.
Taigman, Y., Yang, M., Ranzato, M., Wolf, L., 2014. Deepface: Closing the gap to human-level performance in face verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701–1708.
Trouvé, A., Younes, L., 2005. Metamorphoses through lie group action. Found. Comput. Math. 5 (2), 173–198.
Tustison, N.J., Avants, B.B., Cook, P.A., Zheng, Y., Egan, A., Yushkevich, P.A., Gee, J.C., 2010. N4ITK: Improved N3 bias correction. IEEE Trans. Med. Imaging 29 (6), 1310–1320.
Weiskopf, N., Lutti, A., Helms, G., Novak, M., Ashburner, J., Hutton, C., 2011. Unified segmentation based correction of R1 brain maps for RF transmit field inhomogeneities (UNICORT). Neuroimage 54 (3), 2116–2124.
Zhang, M., Fletcher, P.T., 2015. Bayesian principal geodesic analysis for estimating intrinsic diffeomorphic image variability. Med. Image Anal. 25 (1), 37–44.
Zhang, M., Wells, W.M., Golland, P., 2017. Probabilistic modeling of anatomical variability using a low dimensional parameterization of diffeomorphisms. Med. Image Anal. 41, 55–62.