Diffeomorphic Autoencoders for LDDMM Atlas Building

Jacob Hinkle HINKLEJD@ORNL.GOV
David Womble WOMBLEDE@ORNL.GOV
Hong-Jun Yoon YOONH@ORNL.GOV
Oak Ridge National Laboratory, 1 Bethel Valley Road, Oak Ridge, Tennessee 37830

Editors: Under Review for MIDL 2019

Abstract

In this work, we present an example of the integration of conventional global and diffeomorphic image registration methods with deep learning. Our method employs a form of autoencoder in which the encoder network maps an image to a transformation and the decoder interpolates a deformable template to reconstruct the input. This enables image-based registration to occur simultaneously with training of deep neural networks, as opposed to current sequential optimization methods. We apply this approach to atlas creation, showing that a system that jointly estimates an atlas image while training the registration encoder network results in a high quality atlas despite drastic dimension reduction. In addition, the shared parametrization for deformations offered by the neural network enables training the atlas with stochastic gradient descent using minibatches on a single GPU. We demonstrate this approach using affine transformations and diffeomorphisms in the LDDMM vector momentum geodesic shooting formulation using the OASIS-3 dataset.
Keywords: image registration, atlas-building, deep learning, autoencoder, LDDMM

1. Introduction

Group-wise image registration is an important tool for the analysis of shapes within a population and enables successful downstream processing methods like atlas-based segmentation (Joshi et al., 2004; Rohlfing et al., 2004). As a result of its practical importance, the registration process has been studied extensively from a mathematical perspective in the large deformation diffeomorphic metric mapping (LDDMM) framework. In the LDDMM methodology, unbiased anatomical atlases are constructed for a given population by jointly estimating a collection of deformations along with a single representative image that serves as the atlas (Beg et al., 2005). Extensions to classic LDDMM atlas-building provide efficient parametrizations for these deformations (Singh et al., 2013), enable much faster integration of the necessary differential equations for registration (Zhang and Fletcher, 2015), or leverage Bayesian inference to provide robust distributional estimates for image registration (Zhang et al., 2013). Given an estimated atlas, more recent work applies deep learning in order to avoid a second optimization process, allowing single-shot registration of novel images to a precomputed atlas using a deep network to convert an image to a deformation field (Yang et al., 2017; Pathan and Hong, 2018).

In this work, we take a similar deep learning-based approach of estimating deformations from images and apply it to the atlas building process itself, jointly estimating the atlas image and the neural network that diffeomorphically deforms the atlas to match population images. We present results on a neuroimaging dataset showing that our diffeomorphic autoencoder framework results in meaningful deformations while minimizing the atlas building objective functional at least as well as the conventional method. Another key contribution of the current paper is an open source

Python library called *lagomorph* that extends the popular PyTorch deep learning module (Paszke et al., 2017) to enable image registration and computational anatomy[1] [2].

Our approach has numerous analogues in the deep learning literature. Capsule networks use multiple convolutional filters to map image patches into a matrix that is propagated via matrix multiplication through subsequent layers (Hinton et al., 2011; Sabour et al., 2017). This provides the capability for the capsule network to represent composition of 3D affine transforms through the network and reconstruct members of a given class based on an estimated pose. As a result, in some applications capsule networks are able to detect the 3D pose of objects, but lacking the ability to perform interpolation, they are limited in their capability to represent general free-form transformations common in image registration tasks.

The closest related methods to the present work are spatial transformer networks (STNs) (Jaderberg et al., 2015) and related approaches like deforming autoencoders (Shu et al., 2018). An STN is a neural network that includes layers that take a two-dimensional input image and generate either a free-form grid or an affine transform which is used to interpolate those images. STNs were developed in order to enable image classification networks to automatically align small patches within an image in order to aid recognition. Deforming autoencoders extend STNs by enforcing invertibility in the deformation field, but they do so at great cost in expressivity. By enforcing the constraint that the Jacobian matrix must be diagonal, they employ a deformation model which is not rotationally invariant and does not have the ability to represent local shear or rotation, restricting their deformations to be locally axis-aligned scalings. To our knowledge no existing related approaches enforce invertibility using diffeomorphism models based on vector field flows that are now common in medical image registration.

We aim to enable efficient use of proven image registration models in end-to-end training of neural networks for existing computational anatomy and medical imaging problems as opposed to improving performance in recognition tasks involving natural images. As such, we focus on using well-established diffeomorphic deformation models, implementing not only arbitrary interpolation grids but integration of the Euler-Poincaré equation on diffeomorphisms (EPDiff), foundational to the LDDMM methodology, as a differentiable neural network module.

2. Methods

In the LDDMM formulation, a smooth and invertible transformation φ of an image $I \in L^2(\mathbb{R}^3,\mathbb{R})$ is modeled as the group action of φ as an element of the diffeomorphism group $\mathrm{Diff}(\mathbb{R}^3)$ on I:

$$\varphi.I = I \circ \varphi^{-1}. \tag{1}$$

The diffeomorphism group is endowed with a right-invariant Riemannian metric, and distances between pairs of diffeomorphisms are given by the minimal lengths of paths connecting them (Joshi et al., 2004; Beg et al., 2005; Hinkle, 2015). Such minimizing paths are called geodesics and obey the EPDiff equation, which is a differential equation governing the evolution of the tangent vector field of a geodesic in the diffeomorphism group. A common method of registering images in LDDMM is to employ a shooting method to the initial value problem associated with the Euler-Poincaré equation integrated for unit time starting at the identity, parametrized by the initial velocity field of the evolving deformation, in order to find the closest match of the deformed image $\varphi.I$ to some target image J under the L^2 distance.

We implement LDDMM using the vector momentum formulation first presented in (Singh et al., 2013). In that method, a diffeomorphism is parametrized by a vector field m called the *initial*

1. Code: https://github.com/jacobhinkle/lagomorph
2. Experiments: https://github.com/jacobhinkle/diffeomorphic_autoencoders

momentum. Given m, a diffeomorphism is computed by initializing a $\varphi(0,x)=x$ and integrating until time one the following form of the EPDiff equation:

$$v(t,x)=K\big(D\varphi^{-1}(t,x)m(\varphi^{-1}(t,x))\big), \qquad \frac{\partial}{\partial t}\varphi(t,x)=v(t,\varphi(t,x)), \tag{2}$$

where K is the Green's function of the differential operator $(-\alpha\nabla^2-\beta\nabla\nabla\cdot+\gamma)^2$ and D is the Jacobian operator on vector fields. We used the parameters $\alpha=.1,\beta=0,\gamma=.01$ for all of the experiments in the present work.

Notice that φ is not needed to integrate the above equation or the action on the image I, so we compute $v(t,x)$ directly at each step and propagate the inverse displacement field $h(t,x)=\varphi^{-1}(t,x)-x$, using the following time-stepping scheme:

$$h(t+\Delta t,x)=-\Delta t v(t,x)+h(t,x-\Delta t v(x)). \tag{3}$$

Singh et al. (2013) provides details on obtaining the gradient with respect to the initial momentum m of an L^2 objective function subject to the continuous form of the vector EPDiff equation. Instead of discretizing the continuum solution as in that work, instead we implement the exact numeric gradient of each step of EPDiff.

Notice that integration of this differential equation requires fully differentiable implementations of the Jacobian of a vector field contracted with another vector field, as well as linear interpolation of the vector field m, and the operator K. We implemented the necessary primitives and their derivatives in an open source Python module we call *lagomorph*, using the PyTorch framework to handle automatic propagation of gradients via the chain rule. For the Jacobian operation we use central finite differences with extrapolation by clamping to the nearest boundary voxel. The K operator is implemented in the Fourier domain according to the derivation in (Beg et al., 2005). We also implement linear interpolation where derivatives are computed based on the derivative of the interpolation weights themselves, avoiding approximate finite difference derivatives in that case. The simple Euler time-stepping scheme derived above for this ordinary differential equation was then implemented using lagomorph. For the experiments presented here we integrated EPDiff using five time steps of the Euler integration scheme described above.

2.1. LDDMM Atlas Building

The conventional approach to diffeomorphic atlas building follows the deformable template paradigm to register an estimated atlas image to each member in a population of images using smooth, invertible transformations (Joshi et al., 2004). Before building an atlas, the group G is endowed with a distance metric *a priori* defined by the squared geodesic distance. Given the vector momentum parametrization of diffeomorphisms described in the previous section, the squared distance from the identity transformation to the diffeomorphism $\varphi(1,\cdot)$ is the squared norm of the initial momentum:

$$d(e,\varphi(1,\cdot))^2=\|m\|_K^2=\langle Km,m\rangle_{L^2}^2, \tag{4}$$

where $e\in\mathrm{Diff}(\mathbb{R}^3)$ is the identity mapping. The atlas image $\hat{I}$ corresponding to a collection of images $\{J_i\}$ is then defined implicitly as the minimizer of the following sum:

$$\hat{I}=\operatorname*{argmin}_{I\in L^2(\mathbb{R}^3,\mathbb{R})}\sum_i \min_{m_i:\mathbb{R}^3\to\mathbb{R}^3}\big(\|\varphi_i(1,\cdot).I-J_i\|_{L^2}^2+\lambda\|m_i\|_K^2\big), \tag{5}$$

where $\varphi_i(t,\cdot)$ is the solution to the EPDiff equation given intial momentum m_i. We implement conventional atlas building by jointly estimating, using fixed step-size gradient descent, the atlas image I along with the momenta m_i. In order to standardize raw images, we also implemented affine atlas building, in which case transformations are directly represented by matrices $A_i\in\mathrm{GL}(3,\mathbb{R})$ and vectors $T_i\in\mathbb{R}^3$, with their action on images defined by

$$(A,T).I(x)=I(Ax+T), \tag{6}$$

and the squared norm of m_i in the above equations replaced by the squared Euclidean distance from (A,T) to the identity transformation $(id,0)$.

2.2. Atlas Creation Using Diffeomorphic Autoencoders

Notice that the LDDMM atlas building method resembles a *maximum a posteriori* (MAP) estimate in which the data are images J_i generated with Gaussian noise and momenta m_i,m_j that are independently generated from a prior Gaussian distribution on the space of momenta M. Related methods like Zhang et al. (2013) marginalize the momenta m_i instead of estimating their modes, representing a form of expectation maximization (EM) in which the transformations are considered missing data. In contrast to conventional atlas building (direct minimization of Eq. 5) and the EM approach of Zhang et al. (2013), in this work we consider an approximation of Eq. 5 where, given an image, a function performs minimization of the objective with respect to m non-iteratively. In our model, the momenta m_i are determined by applying a function $f:L^2(\mathbb{R}^3,\mathbb{R})\to M$ to the image J_i, which in turn determines a diffeomorphism:

$$m_i = f(J_i;\theta), \qquad \frac{\partial}{\partial t}\Big|_{t=0}\varphi_i(t,x) = Km_i(x), \tag{7}$$

where $\theta\in\Theta$ is a set of parameters for the function f and is independent of J_i and φ_i obeys the EPDiff equation. We then compute the following

$$\hat{I} = \underset{I\in L^2(\mathbb{R}^3,\mathbb{R})}{\operatorname{argmin}} \min_{\theta\in\Theta}\left(\sum_i \|\varphi_i(1,\cdot).I - J_i\|_{L^2}^2 + \lambda\|f(J_i;\theta)\|_K^2 + R(\theta)\right). \tag{8}$$

The function $f(J_i;\theta)$ is meant to mimic the result of iterative minimization of the loss to obtain a momentum vector field. One motivation for doing so is computational; since θ is shared among all the examples, Eq. 8 is readily optimized using stochastic gradient methods. Furthermore, as is also the case with related approaches (Yang et al., 2017; Pathan and Hong, 2018) when applying the atlas to a novel image, the registration procedure amounts to a single evaluation of the function f, avoiding another round of iterative image registration. Another motivation for this approach comes from a desire for the mapping from observed images to group elements to be as smooth as possible, a condition which may be violated in practice as the iterative minimization process constitutes a dynamical system that is susceptible to local minima and whose paths may bifurcate in which case the result depends overly on initialization.

The function f must be flexible enough to represent the optimal transformation for each example in the dataset, which makes neural network architectures a good fit due to their universal approximation property. This perspective essentially converts the atlas building problem into an unsupervised deep learning problem resembling an autoencoder approach (Kingma and Welling, 2014). When the neural network f contains a particularly low-dimensional hidden layer, the space of activations in that layer is commonly called the *latent space* and the subnetwork mapping from the input to that point is called the *encoder*. In that case, notice that a diffeomorphic autoencoder amounts to a regular image encoder along with a decoder that maps from the latent space to a momentum vector field that is integrated via EPDiff to produce a diffeomorphism that in turn is applied to the template image, which can be thought of as another parameter of the autoencoder network. Following the autoencoder analogy, in this work we employ standard neural network architectures and training methodology for training the autoencoder, using mini-batch stochastic gradient descent to estimate the base image I jointly with the neural network parameters θ.

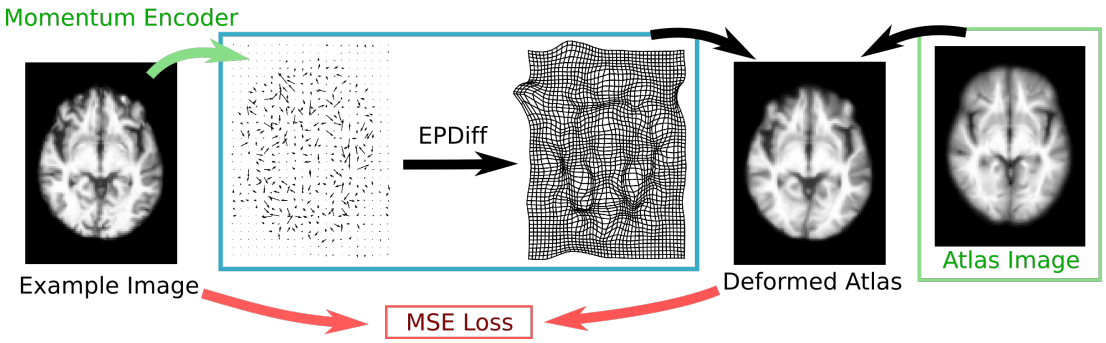

Figure 1: Diagram of a diffeomorphic autoencoder network. The momentum encoder neural network maps each input image to a pose (blue), in this case a diffeomorphism parametrized by a momentum vector field. The EPDiff equation is integrated to form a deformation field which is applied to the atlas image, which is then compared to the input via mean squared error. Estimated components of the network are highlighted in green with fixed components in black and the image match term of the objective function in red.

3. Results

The following experiment was performed on a single Nvidia Tesla V100 GPU, but multi-GPU and multi-node parallelism are facilitated in a straightforward fashion by PyTorch's `distributed` library.

3.1. Data

The OASIS-3 dataset consists of 1983 3D T1-weighted brain MRI images from 990 subjects (Marcus et al., 2007). We used a random selection of 25 of the provided skull-stripped images, corresponding to 25 distinct subjects, cropped from a resolution of $256\times256\times256$ to $165\times235\times219$ voxels, which was the smallest uniform crop we could perform on the dataset while maintaining a border of at least one zero voxel in every direction for every image. We then applied average pooling within $2\times2\times2$ windows to downscale the images to the size of $83\times118\times110$ voxels before further processing.

3.2. Affine Autoencoder Results

We built affine atlases using the OASIS-3 data preprocessed as was described in the previous section, using both conventional atlas building using affine transforms and an affine autoencoder. For the affine encoder network, we used a convolutional neural network with six convolutional layers followed by a multilayer perceptron with hidden layers of size 256 then 64, followed by the output of size 12, representing a 3×3 matrix and three-dimensional translation vector. A ReLU activation was used at each hidden layer, $2\times2\times2$ local max pooling was used after the first two convolutional layers, and the convolutions had the following sizes and numbers of channels: (3,2), (3,4), (3,8), (3,8), (3,8), (3,8). We initialized the network using the Kaiming initialization method built into PyTorch (He et al., 2015). One of our goals was to maintain simplicity in the network architecture, and we did not observe a need for further regularization methods like dropout, or heuristics like batch normalization.

Figure 3.2 (left panel) shows the convergence traces of both affine atlas building methods. Interestingly, the conventional affine atlas building algorithm is unable to reach the same mean squared error

loss as the affine autoencoder. A plausible explanation for this is that the poses for some examples are not well-optimized and may have fallen into local minima. In the affine autoencoder model, similar images map to similar transformations, meaning it is much less likely for "normal" images to be caught in local minima. Additionally, the ability to use minibatch stochastic gradient descent as opposed to direct gradient descent may play a role in the improved convergence of affine autoencoders.

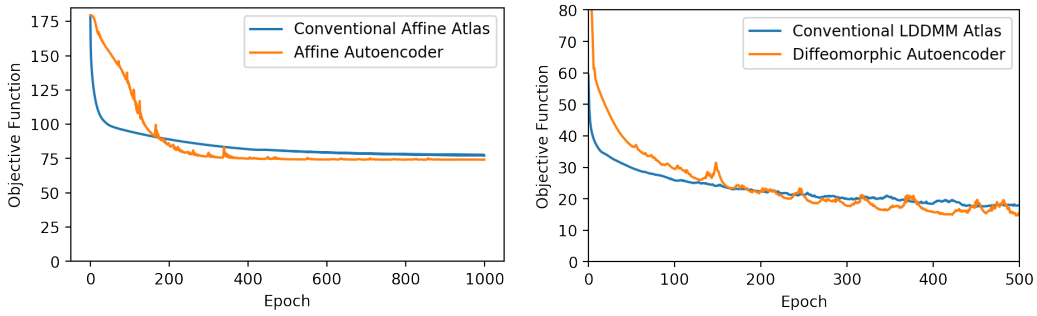

Figure 2: Reduction in the objective function per epoch for both conventional affine atlas and affine autoencoder (left) and LDDMM (right) atlas building algorithms on 25 T1w MRI images from the OASIS-3 dataset. Both LDDMM algorithms used data that were standardized with affine transforms to align according to a prebuilt affine autoencoder. The diffeomorphic autoencoder converges to a loss slightly below that of the conventional atlas building method, despite slower early convergence when features are being learned.

Figure 3.3 (left-most column) shows the results of voxel averaging without transforming the images, as well as estimated affine autoencoder image (third column), showing considerable increase in clarity in the affine atlas.

3.3. Diffeomorphic Autoencoder Results

For the LDDMM momentum encoder network, we used a convolutional neural network with 4 convolutional layers followed by a multilayer perceptron with hidden layers of size 256 then 64, followed by the output vector field of the network of size $3\times83\times118\times110$. A ReLU activation is used at each layer, $2\times2\times2$ local max pooling is used after each of the first three convolutional layers, and the convolutions have the following sizes and numbers of channels: (5,4), (5,16), (5,32), (3,32). As in the affine study, we initialized the network using Kaiming initialization and did not use dropout or batch normalization.

We first trained an affine autoencoder and used it to pre-align the images, then trained both conventional atlas building and a diffeomorphic autoencoder in separate steps. For each of those steps, the same atlas-building objective function was minimized. Figure 3.2 (right panel) shows that, as in the affine case, the diffeomorphic autoencoder converges to a slightly lower objective function value, despite slower convergence in an early phase of training, presumably during which the network is finding relevant features for encoding before proceeding to tune the pose outputs (decoder) to best match the data.

Clearly, the diffeomorphic atlases show improvement over the affine atlas in each case, with a considerable amount of structure visible in the LDDMM atlas that is not present in the affine atlas, indicating that the predictive diffeomorphism network has encoded meaningful transformations.

Figure 3.3 shows the effect of interpolation in the latent space of the trained diffeomorphic autoencoder. In this case, we use the width 64 fully-connected hidden layer in our network as the latent space,

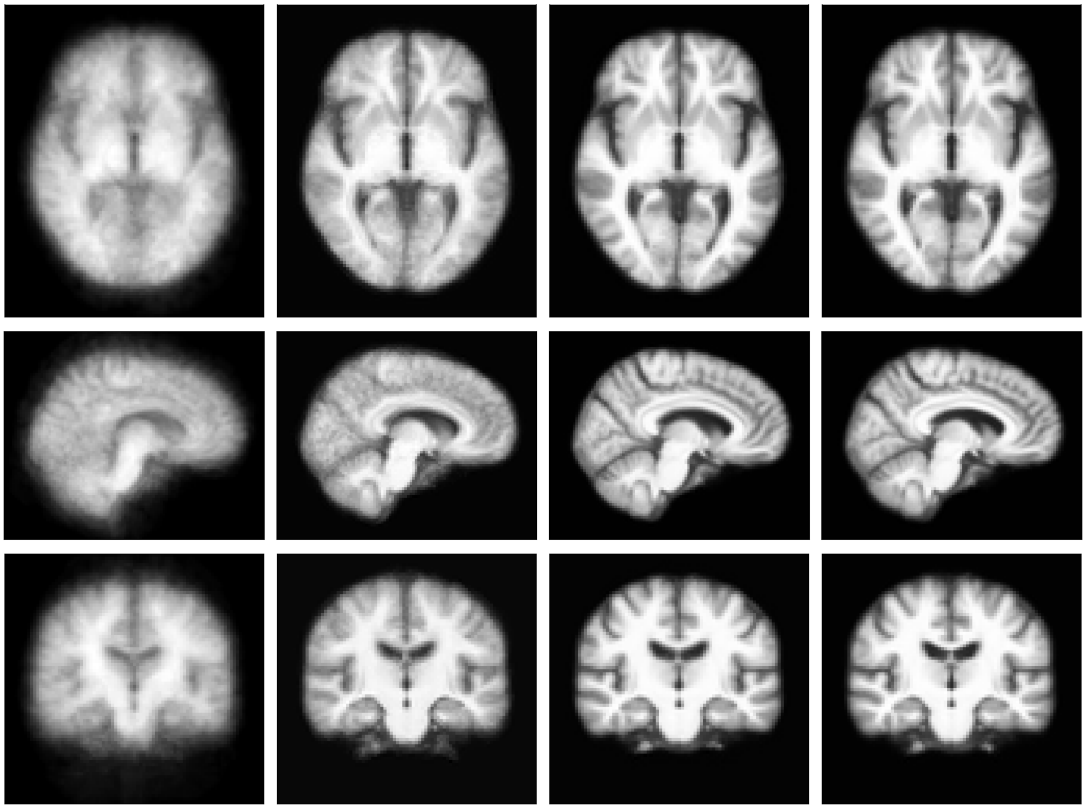

Figure 3: Comparison of estimated atlases. (left to right) Voxel average, affine autoencoder template, conventional LDDMM atlas, and diffeomorphic autoencoder atlas of 25 T1w MRI images from the OASIS-3 dataset. The diffeomorphic autoencoder atlas closely resembles the conventional atlas, and both of those show considerable increased detail compared to the affine atlas.

and interpolate between the corresponding encoded positions of two images in that space. The result is a collection of decoded continuously-varying smooth transformations (Fig. 3.3, bottom row) that continuously transform the atlas image from one configuration to the other. Unlike a conventional autoencoder or Euclidean interpolation of the input images (Fig. 3.3, top row), the diffeomorphic autoencoder provides canonical point correspondences along the interpolation path and preserves fine details from the template. This is remarkable considering that the autoencoder has reduced the dimension from the image size of over a million voxels to just 64 dimensions in the latent space.

4. Conclusion

Previously, Yang et al. (2017) showed that deep neural networks provide sufficient flexibility to effectively perform image registration within a given population. In this work we go further, showing that deep learning provides utility in the atlas building process itself. More broadly, this work represents, to our knowledge, the first example of end-to-end training of a deep learning-enabled LDDMM registration algorithm.

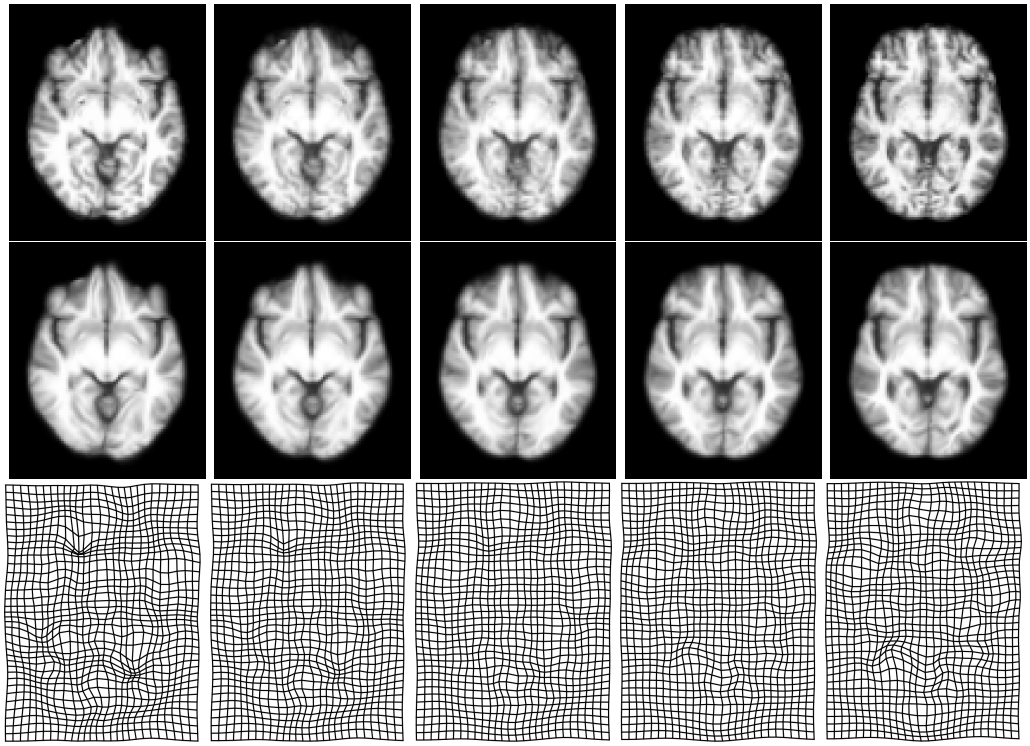

Figure 4: Diffeomorphic interpolation using the autoencoder (axial slices). (top row) Euclidean interpolation between two affine standardized images. (middle row) Deformed atlas image resulting from interpolation in latent space of diffeomorphic autoencoder. (bottom row) Deformation grids corresponding to interpolation in latent space of diffemorphic autoencoder.

Like with conventional autoencoders, a possible pitfall to this approach is that the momentum encoder network may be too simplistic to faithfully represent the variability in deformations that are present in a population. Care should be taken then to compare against conventional atlas building where possible and to design the autoencoder network to balance efficiency with expressivity. More intricate neural momentum encoder architectures than the rudimentary networks used in the present work are certainly conceivable, and are readily implemented with reasonable performance using our lagomorph library. Intuitively, it appears likely that existing architectures like capsule networks and STN architectures in particular will be useful for building these momentum encoder networks.

Integration of deep learning into the atlas creation methodology promises to enable creative new approaches to statistical shape analysis in neuroimaging and other fields. Inherent to this approach is the view that the compression that occurs in a deep neural network applied to neuroimages is most effectively represented as dimension reduction *within the diffeomorphism group* representing the expected anatomic variability, as opposed to conventional autoencoder approaches that are primarily designed to represent appearance and the presence or absence of image components directly. In the future we plan to leverage PyTorch-based frameworks including probabilistic programming libraries like Pyro, in order to perform efficient Bayesian inference in image registration using a variational autoencoder approach. We also plan to use diffeomorphic autoencoders to integrate deep learning approaches to segmentation like fully convolutional networks (Long et al., 2015) and U-nets (Ronneberger et al., 2015) with successful atlas-based approaches.

Acknowledgments

This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725.

The study was supported by the Laboratory Directed Research and Development (LDRD) program of Oak Ridge National Laboratory, under LDRD projects No. 9325 and No. 9613.

Data were provided in part by OASIS Principal Investigators: T. Benzinger, D. Marcus, J. Morris; NIH P50AG00561, P30NS09857781, P01AG026276, P01AG003991, R01AG043434, UL1TR000448, R01EB009352. AV-45 doses were provided by Avid Radiopharmaceuticals, a wholly owned subsidiary of Eli Lilly.

References

M. Faisal Beg, Michael I. Miller, Alain Trouvé, and Laurent Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International Journal of Computer Vision*, 61(2):139–157, Feb 2005. ISSN 1573-1405.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, 2015.

Jacob D Hinkle. *The Quantitative Study of Changes in Anatomy*. PhD thesis, The University of Utah, 2015.

Geoffrey E. Hinton, Alex Krizhevsky, and Sida D. Wang. Transforming auto-encoders. In Timo Honkela, Włodzisław Duch, Mark Girolami, and Samuel Kaski, editors, *Artificial Neural Networks and Machine Learning – ICANN 2011*, pages 44–51, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ISBN 978-3-642-21735-7.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, pages 2017–2025, Cambridge, MA, USA, 2015. MIT Press.

S. Joshi, Brad Davis, Matthieu Jomier, and Guido Gerig. Unbiased diffeomorphic atlas construction for computational anatomy. *NeuroImage*, 23:S151 – S160, 2004. ISSN 1053-8119. doi: https://doi.org/10.1016/j.neuroimage.2004.07.068.

Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *The 2nd International Conference on Learning Representations (ICLR)*, 2014.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 3431–3440, 2015.

Daniel Marcus, Tracy Wang, Jamie Parker, John G Csernansky, John C Morris, and Randy Buckner. Open access series of imaging studies (OASIS): Cross-sectional MRI data in young, middle aged, nondemented, and demented older adults. *Journal of cognitive neuroscience*, 19:1498–507, 10 2007.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

Sharmin Pathan and Yi Hong. Predictive image regression for longitudinal studies with missing data. In *International Conference on Medical Imaging with Deep Learning (MIDL)*, 2018. URL https://openreview.net/forum?id=BJPMK-3jM.

Torsten Rohlfing, Robert Brandt, Randolf Menzel, and Calvin R. Maurer. Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains. *NeuroImage*, 21(4):1428 – 1442, 2004. ISSN 1053-8119.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24574-4.

Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 3856–3866. Curran Associates, Inc., 2017.

Zhixin Shu, Mihir Sahasrabudhe, Riza Alp Guler, Dimitris Samaras, Nikos Paragios, and Iasonas Kokkinos. Deforming autoencoders: Unsupervised disentangling of shape and appearance. In *The European Conference on Computer Vision (ECCV)*, September 2018.

Nikhil Singh, Jacob Hinkle, Sarang Joshi, and P. Thomas Fletcher. A vector momenta formulation of diffeomorphisms for improved geodesic regression and atlas construction. *Proceedings. IEEE International Symposium on Biomedical Imaging*, 2013:1219–1222, April 2013. ISSN 1945-7928. doi: 10.1109/ISBI.2013.6556700.

Xiao Yang, Roland Kwitt, Martin Styner, and Marc Niethammer. Quicksilver: Fast predictive image registration - a deep learning approach. *NeuroImage*, 158:378 – 396, 2017. ISSN 1053-8119.

Miaomiao Zhang and P. Thomas Fletcher. Finite-Dimensional Lie Algebras for Fast Diffeomorphic Image Registration. In Sebastien Ourselin, Daniel C. Alexander, Carl-Fredrik Westin, and M. Jorge Cardoso, editors, *Information Processing in Medical Imaging*, pages 249–260, Cham, 2015. Springer International Publishing. ISBN 978-3-319-19992-4.

Miaomiao Zhang, Nikhil Singh, and P. Thomas Fletcher. Bayesian estimation of regularization and atlas building in diffeomorphic image registration. In James C. Gee, Sarang Joshi, Kilian M. Pohl, William M. Wells, and Lilla Zöllei, editors, *Information Processing in Medical Imaging*, pages 37–48, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-38868-2.