Learning Structured Deformations using Diffeomorphic Registration

Julian Krebs[1,2], Tommaso Mansi[2], Boris Mailhé[2], Nicholas Ayache[1], and Hervé Delingette[1]

[1] Inria, Epione Team, Université Côte d'Azur, Sophia Antipolis, France
[2] Siemens Healthineers, Medical Imaging Technologies, Princeton, NJ, USA

Abstract. Studying organ motion or pathology progression is an important task in diagnosis and therapy of various diseases. Typically, this task is approached by deformable registration of successive images followed by the analysis of the resulting deformation field(s). Most registration methods require prior knowledge in the form of regularization of the image transformation which is often sensitive to tuneable parameters. Alternatively, we present a registration approach which learns a low-dimensional stochastic parametrization of the deformation – unsupervised, by looking at images. Hereby, spatial regularization is replaced by a constraint on this parameter space to follow a prescribed probabilistic distribution, by using a conditional variational autoencoder (CVAE). This leads to a generative model designed to be structured and more anatomy-invariant which makes the deformation encoding potentially useful for analysis tasks like the transport of deformations. We also constrain the deformations to be diffeomorphic using a new differentiable exponentiation layer. We used data sets of 330 cardiac and 1000 brain images and demonstrate accurate registration results comparable to two state-of-the-art methods. Besides, we evaluate the learned deformation encoding in two preliminary experiments: 1) We illustrate the model's anatomy-invariance by transporting the encoded deformations from one subject to another. 2) We evaluate the structure of the encoding space by clustering diseases.

1 Introduction

Analyzing geometric changes in pairs of successive images can give important clinical insights into organ function (e.g cardiovascular) and pathology (e.g. Alzheimer progression). In cardiac images, this analysis is a key task to assess the proper heart function. In neuro-degenerative diseases, it is important to quantify and predict disease evolution in long-term studies. These changes are typically studied by finding voxel correspondences – with deformable registration – and by examining the extracted deformations. Traditional deformable registration approaches aim to optimize a local similarity metric between deformed and target image, being regularized by various energies [9]. In order to retrieve invertible deformation fields, diffeomorphic registration was introduced. Among other parametrizations, one way to parametrize diffeomorphisms are stationary velocity fields (SVF) as proposed by Arsigny *et al.* [1].

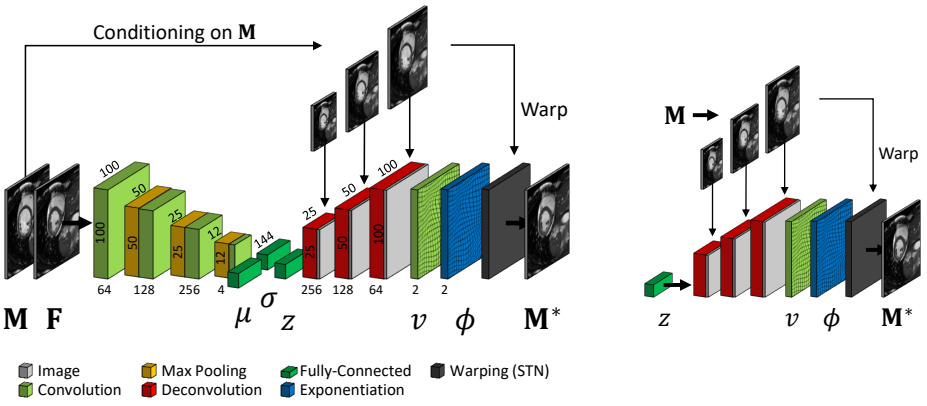
A major drawback of these methods is that spatial regularization like elastic, fluid or diffusion models are based on prior assumptions and are often sensitive to tunable parameters [9]. In recent years, other drawbacks of these approaches like high computational costs and risks of being stucked in local minima, have led to an increasing popularity of learning-based algorithms – notably deep learning (DL). One can classify these algorithms as supervised and unsupervised. Due to the difficulty of finding ground truth voxel mappings, supervised methods need to rely on predictions from existing algorithms [10], simulations or both [6]. These methods are either limited by the performance of the used existing algorithms or the realism of simulations. On the other hand, unsupervised approaches are based on an image similarity, often combined with a penalization or smoothing term (regularization). Many of such approaches, like [2,4], are realized using spatial transformer layers (STN [3]). However, the problem of choosing the right regularization and appropriate weighting parameters remains. Furthermore, important properties like transformation symmetry or diffeormorphisms[9] are still missing in DL-based approaches.

In this paper, we suggest to replace the specification of a geometric regularization term with a statistical regularization term acting on a low-dimensional parameterization of deformations – learned from a training set. More precisely, the latent parameter set is constrained to follow a multi-variate unit Gaussian distribution. This implicitely regularizes the transformation depending on the training data. To this end, we use a conditional variational autoencoder (CVAE [5]) in a generative neural network where the decoder of the network is constrained on the moving image to enforce anatomy invariance of the deformation encoding. Furthermore, we propose a generic vector field exponentiation layer. This differentiable layer can be added to any neural network which predicts deformation or optical flow fields to generate diffeomorphic transformations. The deformation encoding can be potentially used to sample deformations that are similar to the ones seen in the training data or for the analysis of deformations. Our framework contains an STN and can be trained with a choice of similarity metrics (e.g. symmetric local cross correlation) and does not require any spatial regularization term. The main contributions of this paper are:

- A stochastic framework for learning a structured latent space for deformation modeling without a spatial regularization term.
- A novel (differentiable) exponentiation layer that ensures the outputs of neural networks to be diffeomorphic.
- As a proof of concept, first experiments on brain and cardiac images for deformation transport and disease clustering.

2 Methods

Learning a Structured Deformation Encoding The goal of image registration is to find the spatial transformation $\mathcal{T}_\theta$, parametrized by $\theta \in \mathbb{R}^D$ which best warps the moving image $\mathbf{M}$ to match the fixed image $\mathbf{F}$. Typically, this is done by minimizing an objective function of the form: $\arg\min_\theta \mathcal{F}(\theta, \mathbf{M}, \mathbf{F}) =$



(a) (b)

Fig. 1: (a) CVAE registration network during training and registration including diffeomorphic layer (exponentiation). Deformations are encoded in z from which velocities are decoded while being conditioned on the moving image. (b) Decoder network for sampling and deformation transport: Apply z-code conditioned on any image $\mathbf{M}$.

$\mathcal{D}(\mathbf{F}, \mathbf{M} \circ \mathcal{T}_\theta) + \mathcal{R}(\mathcal{T}_\theta)$ with the image similarity $\mathcal{D}$ and a spatial regularizer $\mathcal{R}$. Recent DL-based approaches (e.g. [2,4]), mimic the optimization of such an objective function by using spatial transformer layers (STN [3]) to warp the moving image in a differentiable way such that loss functions can operate on the warped image (similarity metric) and on the transformation itself (regularization). During training, the network parameters are updated through back-propagation of the gradients. Finally, registration is done in a single forward path.

Instead of regularizing the transformation, we propose to regularize the deformation encoding vector $z \in \theta$ such that it follows a multi-variate unit Gaussian distribution. To this end, we define the low-dimensional latent space of an encoder-decoder neural network as this structured z-space by using a stochastic CVAE [5]. The proposed network takes the moving and fixed image as input and includes a dense STN to output deformations and the warped image $\mathbf{M}^*$. A second major improvement, is a new exponentiation layer just in front of the dense STN which interprets the network's outputs as velocities (an SVF) and returns a diffeomorphism. The full network architecture can be seen in Fig. 1a. After training, the decoder can be also used for sampling as shown in Fig. 1b.

Conditional Variational Autoencoder for Registration In a standard variational autoencoder (VAE), an encoder maps an image to its z-code – an interpretable low-dimensional latent space, from which a decoder aims to reconstruct the original image. Typically, the encoder and decoder weights ω and γ are trained by maximizing a lower bound on the data likelihood. This results in a two-term loss function where the first term describes the reconstruction loss as the expected negative log-likelihood of the data given the latent representation z. In other words, the reconstruction loss represents a similarity metric between input and output image. The second term acts as a regularization term by forc-

ing the encoded distribution to be close to the desired probability distribution $p(z)$. Typically, the Kullback-Leibler divergence is used as a measure of how close the two distributions are and the target distribution $p(z)$ is mostly defined as multi-variate unit Gaussians, $p(z) = \mathcal{N}(0, I)$ with the identity matrix I. In this case, the encoder network predicts the mean $\mu \in \mathbb{R}^d$ and covariance $\sigma \in \mathbb{R}^d$, from which the decoder draws a d-dimensional sample as the z-code.

In CVAE, encoded and decoded distributions are conditioned on additional data like class information [5]. We propose to frame image registration as a CVAE where the moving image $\mathbf{M}$ acts as the conditioning data. In the encoding part, both fixed and moving images serve to estimate the probability on the z-code $q_\omega(z \mid \mathbf{F}, \mathbf{M})$. In the decoding p_γ, the goal is to reconstruct the fixed image $\mathbf{F}$ by warping $\mathbf{M}$. To help the reconstruction task and motivate anatomy-invariance in the latent space, we introduce a strong conditioning by involving $\mathbf{M}$ not only as the image to be warped in the last layer, but also in the first decoder layers. The network is expected to make use of the provided extra information of $\mathbf{M}$ such that less anatomic information need to be conveyed by the low-dimensional latent layer. The reconstruction term results in optimizing the expected negative log-likelihood of $p_\gamma(\mathbf{F} \mid z, \mathbf{M})$. Thus, the complete loss function has the form:

$$l(\omega, \gamma, \mathbf{F}, \mathbf{M}) = -E_{z \sim q_\omega(\cdot \mid \mathbf{F}, \mathbf{M})} \left[\log p_\gamma(\mathbf{F} \mid z, \mathbf{M})\right] + KL\left[q_\omega(z \mid \mathbf{F}, \mathbf{M}) \parallel p(z)\right]. \quad (1)$$

The KL-divergence can be computed in closed form [5]. Assuming a Gaussian log-likelihood term $\log p_\gamma$ is equivalent to minimizing a weighted SSD criterion (cf. [5]). We propose instead to assume a local cross-correlation (LCC) Boltzmann distribution for $p_\gamma(\mathbf{F} \mid z, \mathbf{M}) \sim \exp(-\lambda \mathcal{D}_{LCC}(\mathbf{F}, \mathbf{M}, v))$ where $\mathcal{D}_{LCC}$ is the LCC criterion as in [7]. With the weighting factor λ, velocities v and a small constant ϵ, which is added for numerical stability (as in[2]), we define:

$$\mathcal{D}_{LCC}(\mathbf{F}, \mathbf{M}, v) = -\lambda \frac{\overline{\mathbf{F} \circ \exp\left(-\frac{v}{2}\right)}\, \mathbf{M} \circ \exp\left(\frac{v}{2}\right)^2}{\left[\mathbf{F} \circ \exp\left(-\frac{v}{2}\right)\right]^2 \overline{\left[\mathbf{M} \circ \exp\left(\frac{v}{2}\right)\right]^2} + \epsilon}, \quad (2)$$

where $\overline{\mathbf{F}}$ symbolizes the local mean image derived by Gaussian smoothing with a strength of σ_G and kernel size k.

Exponentiation Layer: Generating Diffeomorphisms In the SVF setting, the transformation ϕ is defined as the Lie group exponential map with respect to the velocities v: $\phi(x) = \exp(v)$. For efficient computation, typically the scaling and squaring algorithm is used [1]. In order to generate diffeomorphic transformations ϕ in a neural network, we propose an exponentiation layer that implements the scaling and squaring algorithm in a fully differentiable way. To this end, the layer expects a vector field as input (the velocities v). In the first step, the field is scaled with a precomputable factor N: $v * 2^{-N}$ [1]. We do not learn N and precompute its value before training. In the second step, the approximated $\phi_0 \approx id + v * 2^{-N}$ (with id as a regular grid) is recursively squared, N-times, from $k = 1$ to N: $\phi_k = \phi_{k-1} \circ \phi_{k-1}$. This squaring step requires the composition of two vector fields on regular grids, which can be done with linear interpolation. In neural networks, differentiable linear interpolation is already used, e.g.

in STN [3]. This results in the diffeomorphism $\phi_N \equiv \phi$. These computations consist of standard operations that can be added to the computational graph and auto-differentiated in modern deep learning libraries.

3 Experiments

We demonstrate our framework on the two intra-patient tasks of cardiac MRI cine and longitudinal brain MRI registration. In the cardiac case, end-diastole frames are registered to end-systole frames, a very large deformation. In the brain case, the baseline frame is registered to the last available time frame. Furthermore, we show preliminary experiments evaluating the learned deformation encoding: its potentials for transporting encoded deformations from one patient to another and showing the structure of the space. All experiments are in 2-D. 3-D extension will be explored in future work.

Data In the cardiac experiments, we used 184 short-axis datasets acquired from different hospitals and the 150 cases from the Automatic Cardiac Diagnosis Challenge (ACDC) at STACOM 2017[3], mixing congenital heart diseases with images from adults. We used 234 cases for training and for testing the remaining 100 cases from ACDC, that contain segmentation and disease class information. In the brain experiments, we used 1042 T1 cases from the ADNI database [8], bias-corrected and rigidly aligned with the follow-up image. 100 randomly picked cases were used as the test dataset. Disease information were available and segmentation masks for evaluation were generated by SPM[4] with standard parameters. In our network, the whole brain masks were only used during training as an additional loss function to punish deformations outside the brain. During testing, masks were not required. For both use cases, 7-11 axial slices around the center slice of each volume have been used for training and testing. The images were down-sampled and cropped to a size of 100x100 pixels with a spacing of 2mm. These dimensions were chosen to save computation time and are not a limitation of the framework (validated on different image sizes).

Implementation Details The encoder of our neural network consisted of 4 convolutional and 3 max-pooling layers (Fig. 1a). The bottleneck layers (μ, σ, z) were fully-connected. The decoder had 3 deconvolutional layers, where the outputs at each layer were concatenated with sub-sampled versions of the moving image $\mathbf{M}$ (conditioning). At the end, a final convolution layer was placed in front of the exponentiation and transformer layer. The latent code size d was set to 144 as a trade off between registration quality and generalizability. This leads to a total of ∼950k trainable parameters in the network. Batch normalization and L2 weight decay with a factor of 0.0001 were applied. The numbers of iterations in the exponentiation layer was set to $N = 4$ in all experiments. In training,

[3] https://www.creatis.insa-lyon.fr/Challenge/acdc/index.html
[4] http://www.fil.ion.ucl.ac.uk/spm/

Table 1: Registration results on the brain and cardiac datasets, comparing the baseline (BL) with the LCC-Demons[7] (DEM), VoxelMorph-2 (VM)[2] and our approach.

Cardiac	BL	DEM	VM	**our**	**Brain**	BL	DEM	VM	**our**
RMSE	.0495	.0374	**.0244**	.0386	RMSE	.0396	.0342	**.0340**	.0348
Neg. Det. Jac. (%)	-	**0.0**	3.5	**0.0**	Neg. Det. Jac.	-	**0.0**	1.2	**0.0**
Velo. Mag. 10^{-4}	-	9.87	-	**7.43**	Velo. Mag.	-	.42	-	**.37**
Velo. Grad. 10^{-5}	-	**1.3**	-	3.9	Velo. Grad.	-	**.1**	-	.3
LH DICE	.901	.925	.917	**.931**	CSF DICE	.888	.912	**.919**	.904
LV DICE	.548	**.773**	.767	.771	GM DICE	.856	.872	**.898**	.865
BP DICE	.681	**.848**	.772	.819	WM DICE	.936	.950	**.959**	.943
LH HD 95%	11.08	10.44	20.22	**9.50**	Vent. DICE	.913	.943	**.953**	.934
LV HD 95%	17.54	14.76	13.31	**12.02**					
BP HD 95%	18.7	16.8	19.87	**14.50**					

the strength of the Gaussians for computing the LCC was set to $\sigma_G = 2$ with a kernel size $k = 9$. The loss balancing factor λ was empirically chosen such that encoded training samples roughly had zero means and variances of 1 and the reconstruction loss was optimized (cardiac: $\lambda = 82$, brain: $\lambda = 55$). We used the Adam optimizer with a learning rate of 0.0005 and a mini-batch size of 50. The framework has been implemented in *Tensorflow*, using *Keras*[5]. Training took less than 12 hours on a single GPU (*NVIDIA GTX TITAN X*).

Registration Results We compare our registration algorithm with the LCC-demons [7] with manually tuned parameters and a 2-D version of the non-diffeomorphic DL-based method VoxelMorph-2 [2] (VM) with a smoothness weighting parameter of 1.5, as proposed in [2]. In the brain experiments, whole brain masks were used in the LCC-Demons for masking, while no masks were used in VM. As a surrogate measure of registration performance we used the intensity root mean square error (RMSE), mean DICE score and 95%-tile Hausdorff distance (HD) in mm on the left ventricle (LV), left bloodpool (BP) and left heart (LH, including LV and BP) in cardiac images and on the cerebrospinal fluid (CSF), ventricles (Vent), grey (GM) and white matter (WM) in brain images. We do not report HD measures in the brain case, since the available probability maps could not reliably used for the HD computation. In the cardiac case, our algorithm shows similar DICE scores compared to the LCC-demons, which is only better in BP with .848 to .819 mean DICE scores. However, our method reached significantly better HD scores. The VoxelMorph algorithm reached a very low RSME of .0244 but could not reach the other algortihms in terms of DICE and especially HD scores. On brain images, all results are very close to each other. VoxelMorph results in the best DICE and RSME scores, closely followed by the demons and our results with the biggest difference on the GM of .898 (VM) compared to ours of .865 (cf. Table 1). Although, VM shows good results on the brain, the method produces highly non-diffeomorphic deformation fields since 1.2% of the displacements have a negative determinant of the

[5] https://keras.io/

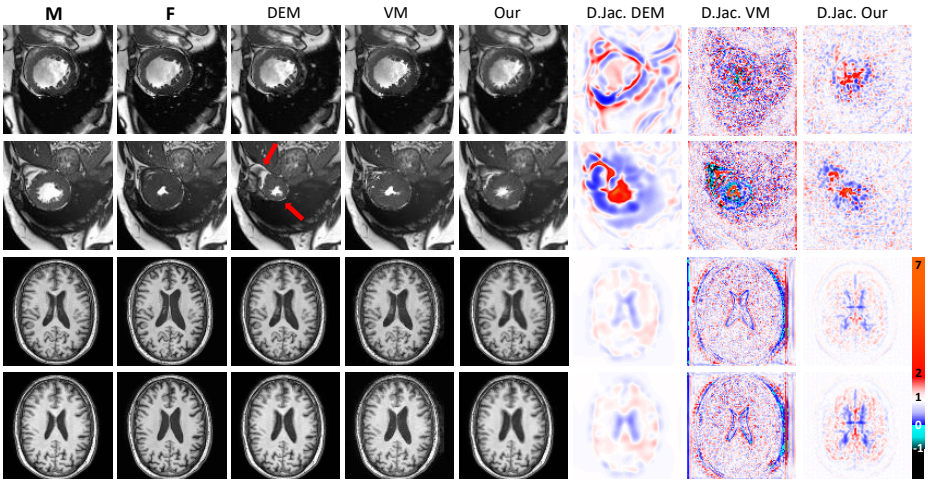


Fig. 2: Registration results for two brain and cardiac examples of our method versus the LCC-demons (DEM) and VoxelMorph (VM). (Column 1-2) moving/fixed image. (Column 3-5) warped moving image of DEM, VM and our algorithm. (Column 6-8) Jacobian determinant of the deformation field.

Jacobian, in the cardiac case even 3.5% (cf. Jacobian determinants in Fig. 2). In general, our approach leads to velocity fields with smaller amplitude but less smoothness than the demons algorithm. Visual results are shown in Fig. 2 and in the supplementary material.

Deformation Encoding For evaluating the learned deformation encoding, we show the anatomy-invariance by transporting a deformation from one subject to another. Therefore, we take a z-code from one patient and condition the decoder on the image of another patient (Fig. 1b). More precisely, in Fig. 4 we transported a pathological deformation (heart: cardiomyopathy DCM, brain: dementia) to two healthy cases. In both cases, the deformations had similar properties (disease-specific: enlargement of brain ventricles or reduced cardiac contraction) which are different to the real (healthy) transformations. The resulting deformation fields are adapted to the anatomy of the conditioning image.

In a second experiment, we tried to use the encoded z-features and disease information of our cardiac test set to visualize the structure of the learned space. Therefore, we linearly projected the features to a 3-D space by using the 3 most discriminative CCA components (canonical correlation analysis). We used the ACDC classes: dilated cardiomyopathy DCM, hypertrophic cardiomyopathy HCM, myocardial infarction MNF,

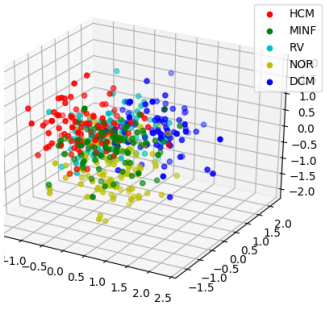


Fig. 3: Shows distributions of cardiac diseases after projecting the z-codes of 100 test images on 3 CCA components.

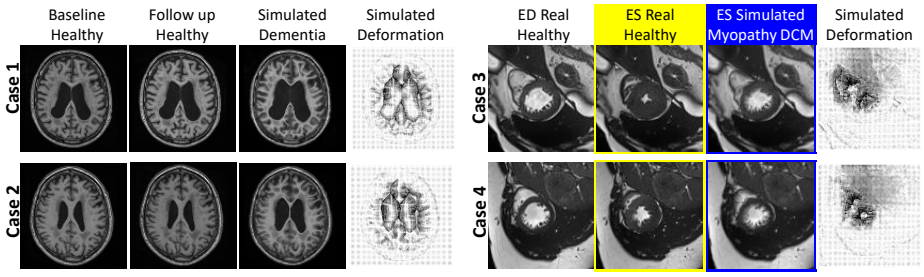


Fig. 4: Transport the z-code of a pathological deformation (cardiomyopathy DCM, dementia) to a healthy subject.

abnormal right ventricle RV and normal NOR. In Fig. 3, one can see that the classes of the 100 test sets are clustered in the projected space. By using the 5 most discriminative CCA components and applying SVM on-top, the 5 class classification accuracy reaches 68% with 10-fold cross-validation. These results (only based on deformation z-codes of 2-D images) suggest, that the deformation encoding learned a structured space in which similar deformations are close to each other.

4 Conclusion

In this work, we presented a deformable registration framework that learns a structured deformation encoding. Since the encoding is constrained, the framework does not require any explicit spatial regularization. Furthermore, an exponentiation layer has been introduced that creates diffeomorphic transformations. We have shown robust and accurate registration results on two big datasets with comparable performance to two state-of-the-art algorithms. On these datasets, our approach produces more regular deformation fields than a DL-based non-diffeomorphic algorithm. First results show, that the encoding could potentially be used for deformation transport and clustering tasks. In future work, we plan to further explore the deformation encoding to evaluate these tasks more deeply.

Acknowledgements: Data used in preparation of this article were obtained from the EU FP7-funded project MD-Paedigree, the ACDC STACOM challenge 2017 and the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). As such, the investigators within the ADNI contributed to the design and implementation of ADNI and/or provided data but did not participate in analysis or writing of this report. A complete listing of ADNI investigators can be found at: `http://adni.loni.usc.edu/wp-content/uploads/how_to_apply/ADNI_AcknowledgementList.pdf`. The authors would like to thank Raphaël Sivera for the preparation of the ADNI data.

Disclaimer: This feature is based on research, and is not commercially available. Due to regulatory reasons its future availability cannot be guaranteed.

References

1. Arsigny, V., Commowick, O., Pennec, X., Ayache, N.: A log-euclidean framework for statistics on diffeomorphisms. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 924–931. Springer (2006)
2. Balakrishnan, G., Zhao, A., et al.: An unsupervised learning model for deformable medical image registration. arXiv preprint arXiv:1802.02604 (2018)
3. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: Advances in neural information processing systems. pp. 2017–2025 (2015)
4. Jason, J.Y., Harley, A.W., Derpanis, K.G.: Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. In: European Conference on Computer Vision. pp. 3–10. Springer (2016)
5. Kingma, D.P., et al.: Semi-supervised learning with deep generative models. In: Advances in Neural Information Processing Systems. pp. 3581–3589 (2014)
6. Krebs, J., Mansi, T., Delingette, H., et al.: Robust non-rigid registration through agent-based action learning. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 344–352. Springer (2017)
7. Lorenzi, M., Ayache, N., Frisoni, G.B., et al.: LCC-Demons: a robust and accurate symmetric diffeomorphic registration algorithm. NeuroImage 81, 470–483 (2013)
8. Mueller, S.G., et al.: Ways toward an early diagnosis in alzheimers disease: the alzheimers disease neuroimaging initiative (adni). Alzheimer's & dementia: the journal of the Alzheimer's Association 1(1), 55–66 (2005)
9. Sotiras, A., Davatzikos, C., Paragios, N.: Deformable medical image registration: A survey. IEEE Transactions on Medical Imaging 32(7), 1153–1190 (2013)
10. Yang, X., Kwitt, R., Niethammer, M.: Fast predictive image registration. In: Deep Learning and Data Labeling for Medical Applications, pp. 48–57. Springer (2016)