

FAIM – A ConvNet Method for Unsupervised 3D Medical Image Registration

Dongyang Kuang
Department of Math & Stat
University of Ottawa

Tanya Schmah
Department of Math & Stat
University of Ottawa

November 26, 2018

Abstract

We present a new unsupervised learning algorithm, “FAIM”, for 3D medical image registration. Based on a convolutional neural net, FAIM learns from a training set of pairs of images, *without* needing ground truth information such as landmarks or dense registrations. Once trained, FAIM can register a new pair of images in less than a second, with competitive quality. We compared FAIM with a similar method, VoxelMorph [BZS⁺18], as well as a diffeomorphic method, uTilzReg GeoShoot [VRRRC12], on the LPBA40 and Mindboggle101 datasets. Results for FAIM were comparable or better than the other methods on pairwise registrations. The effect of different regularization choices on the predicted deformations is briefly investigated. Finally, an application to fast construction of a template and atlas is demonstrated.

1 Introduction

Image registration is a key element of medical image analysis. The spatial deformations required to optimally register images are highly non-linear, especially for regions such as the cerebral cortex, the folding patterns of which can vary significantly between individuals. Most state-of-the-art registration algorithms, including ANTs [ATS⁺11] and uTilzReg [VRRRC12], use geometric methods that are guaranteed to produce smooth invertible deformations. These algorithms are very computationally intensive and still do not generally find optimal deformations. One general problem is that the optimisation problems solved by these algorithms are highly nonconvex. Another is that they treat each pair of images to be registered *de novo*, without any learning.

A revolution is taking place in the last couple of years in the application of machine learning methods, especially convolutional neural networks, to this problem. After training, the resulting networks can register new pairs of images in milliseconds, where previous algorithms took hours, with accuracy sometimes approaching state of the art geometric methods. Supervised methods, as in [SdVB⁺17, RDH⁺17], learn from known reference deformations for training data – either

actual “ground truth” in the case of synthetic image pairs, or deformations computed by other automatic or semiautomatic methods. Unsupervised methods, as in [YHT⁺17, LF17, WKWS17, SGY⁺17, BZS⁺18], do not require reference deformations, but instead minimise some cost function modelling the goodness of registration, optionally regularized by a term constraining the deformation.

The present manuscript describes a new unsupervised image registration algorithm, FAIM¹ (for FAsT IMage registration), based on deep convolutional neural nets and specifically spatial transformer networks (STN) [JSZ⁺15]. Our algorithm is similar to the VoxelMorph algorithm of Balakrishnan et al. [BZS⁺18], however our architecture and cost function are different. We compare three algorithms: FAIM (with variations); VoxelMorph; and a geometric method, namely the geodesic shooting method in the uTilzReg package [VRRRC12]. We evaluate these methods on two datasets: LPBA40 and Mindboggle101. Both datasets have associated anatomical parcelations, which we use to evaluate the registration methods by degree of overlap of regions of interest.

2 Methods

Our idea is directly inspired by the spatial transformer network (STN) of Jaderberg et al. [JSZ⁺15], which is used to learn the proper parametrized transformation of the input feature so that later tasks such as classifications can be better performed. This kind of module is originally developed for 2D images and only affine and thin plate spline transformation were implemented. In some very recent research [SGY⁺17, BZS⁺18], this framework begins to appear in 3D medical image registration. All these works aim to find an optimal parametrized transformation $\phi : \Omega \rightarrow \mathbf{R}^3$, for image domain $\Omega \subset \mathbf{R}^3$, such that the warped volume $M \circ \phi^{-1}(x)$ from a moving/source volume $M(x)$ is well aligned to the fixed/target volume $F(x)$. In our network, we use displacement field $\mathbf{u}(x)$ to parametrize the deformation ϕ by $M \circ \phi^{-1}(x) = M(x + \mathbf{u}(x))$, which is learned through a spatial deformation module (SDM). Figure 1 shows

¹Our code is available at [Kua18]

the flow chart when the network is in training and a closer look at the SDM.

During training, the moving volume and the target volume are stacked together as the input feeding into SDM. The first layer is inspired by Google’s Inception module [SLJ⁺15]. It contains convolutional kernels of different sizes attempting to compare and capture information from different temporal scales for later registration. PReLU [HZRS15] activations are used at the end of each convolutional block except the last layer which uses linear activation to produce displacement field. This is then passed to a sampling module that generates a deformed grid to produce the warped image from the source image. We use kernel stride > 1 to reduce the volume size instead of inserting max pooling layers. Transposed convolutional layers are used for upsampling. There are three “add” skip connections between the downsampling and upsampling path.

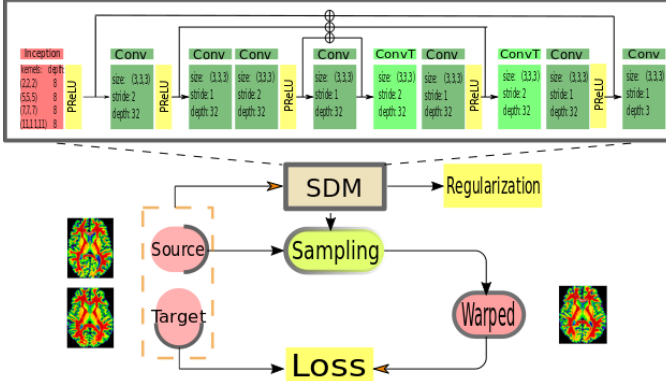


Figure 1: FAIM network architecture

Both FAIM and Voxelmorph use a convolutional neural net with bottleneck, and use the idea of a Spatial Transformer Network (STN). However there are significant differences in architecture: (i) FAIM uses an “Inception” module similar to [SLJ⁺15] right after the input concatenating convolutional results from kernels with different sizes; (ii) FAIM is less deep with 2 down/up sampling layers compared with VoxelMorph which has 4 down/up sampling layers; (iii) FAIM has “add” connections to help back propagate gradients instead of concatenation as used by VoxelMorph.

The total training loss is the sum of an image dissimilarity term L_{image} and a regularization term R ,

$$L_{total} = L_{image}(M, F) + \beta R(u), \quad (1)$$

where β is a hyperparameter controlling degree of regularization. The regularization term we adopted here penalizes both the size and nonsmoothness of the displacement field. The loss and regularization functions are defined in Table 1. With regard to regularization, note that many researchers [SdVB⁺17, BZS⁺18] penalize only the first derivatives of the displacement u , which promotes smoothness of predicted displacement fields during training. We also add an additional regularization term $\|u\|$ to help control the predicted deformations indirectly.

$$L_{image}(M, F): \quad 1 - CC(M \circ \phi^{-1}, F)$$

$$\text{Regularization:} \quad R(u) = w_1 \|u\|_2 + w_2 \|\nabla u\|_2$$

Table 1: Loss and regularization functions used.

In the following 2 dimensional toy example, we place a square in the center as the source image and its vertically translated image as the target. We trained our network (a 2 dimensional version) with this single pair using mean square error loss and regularization term as shown in Table 1. Different choices of weights w_1 and w_2 can lead to similar warped images that are close to the target image, but the transformation behind can be very different. As shown in Figure 2, if no penalty is placed on u , the network tends to learn the linear translation perfectly. On the other hand, if the ratio $w_1 : w_2 = 1 : 3^2$, the resulting deformation is then a nonlinear transformation that looks like waves are propagating in the vertical direction. Here, the ratio $w_1 : w_2$ seems to control the balance between the linear and nonlinear components that consists the whole transformation during training. While the same result probably could be obtained by proper tuning w_2 along and let w_1 be 0, the insight here is w_1 works like a switch. We hope to have a rough control of learned deformations through the ratio $w_1 : w_2$ by simply turning this option on instead of carefully tuning the value of w_2 . The control through the ratio over learned deformations will be further explored in our experiment with 3D brain volumes in Section 3.4.

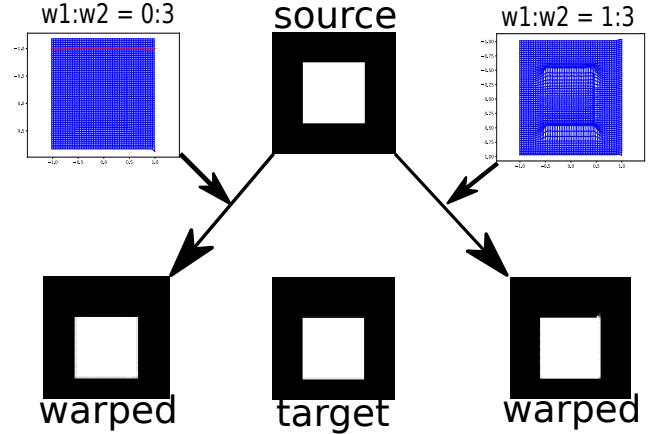


Figure 2: Different choices of weights w_1 and w_2 in regularization term affects the learned transformation. Grids shown are warped grids generated by learned displacement fields added to the original grids. Left: the red line is $y = -1$, i.e. the top line of the original grid. The network learns the whole linear translation. Right: the transformation learned is more local and nonlinear, there is no global translation learned.

²Other hyperparameters used are the same: epochs=50, $\beta = 0.005$.

3 Experiments

We compare FAIM with VoxelMorph [BZS⁺18] and the geodesic shooting method implemented in uTIlzReg [VRR12]. Each algorithm is tested on two tasks: pairwise registration, and registration to template; using two publicly-available datasets: LPBA40 [SMA⁺08], and Mindboggle101 [KT12]. Each dataset consists of T1-weighted MR images, each of which has been manually labeled with anatomical regions, and one or more atlases constructed from these labeled images.

3.1 LPBA40 dataset

The LONI Probabilistic Brain Atlas (LPBA40) dataset contains one T1-weighted MR image from each of 40 healthy young adults. After preprocessing, which included skull-stripping, all images were rigidly aligned to the MNI305 atlas. In each image was then manually labeled with 56 anatomical labels. By combining these labeled images using different registration options, three different atlases were created, which are considered as variants of “the LONI probabilistic brain atlas”. Details of data collection and processing are described in [SMA⁺08], and the resulting dataset is available at <http://www.loni.ucla.edu/Atlases/LPBA40>.

We used FLIRT from the FSL package [SBB⁺01] to linearly align these preprocessed images, along with their labels, to the ICBM152 2mm template, resulting in a set of images with size $91 \times 109 \times 91$. Finally, we normalized the intensity of each brain volume by its maximum voxel intensity.

3.2 Mindboggle101 dataset

This dataset, created by Klein et al. [KT12], is based on a collection of 101 T1-weighted MRIs from healthy subjects. Most of these images (96 out of 101) are drawn from other publicly-available datasets. The Freesurfer package (<http://www.martinos.org/freesurfer>) was used to preprocess all images, and then automatically label the cortex using its DK cortical parcellation atlas (see [Des06]). For 54 of the images, including the OASIS-TRT-20 subset, these automatic parcellations were manually edited to follow a custom labeling protocol, DKT, based on the DK protocol. The full DKT protocol has 31 cortical regions per hemisphere, while the variant DKT25, used here, combines two groups of regions per hemisphere. Details of data collection and processing, including atlas creation, are described in [KT12].

In the present paper, we used brain volumes from the following three named subsets of Mindboggle101, for a total of 62 volumes:

- NKI-RS-22: “Nathan Kline Institute / Rockland sample”
- and NKI-TRT-20: “Nathan Kline Institute / Test-retest”
- OASIS-TRT-20: “Open Access Series of Imaging Studies”.

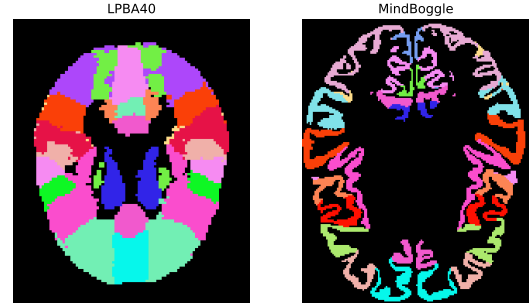


Figure 3: Samples of ROI labels provided by LPBA40 atlas(left) and Mindboggle’s OASIS-TRT-20 jointfusion atlas (right). Though regions that has parts in both left and right hemispheres are labeled differently, they are annotated here in one color for better visibility.

Each image in Mindboggle101 dataset has a dimension of $182 \times 218 \times 182$, we truncated the marginal reducing the size to $144 \times 180 \times 144$. These images are already warped to MNI152 space. We also normalized the intensity of each brain volume by its maximum voxel intensity.

Figure 3 shows one slice of annotated labels provided from each dataset. Labels used in Mindboggle101 data set are cortex surface labels. Their geometrical complexity leads to more challenging registration tasks, especially for neural network approaches which usually generate deformations that are less smooth compared to diffeomorphic methods.

3.3 Evaluation Methods

We divide each dataset into sets of training and test images, and use these to form training and test sets of *pairs* of images. For LPBA40, the training set consists of all ordered pairs from the *first* 30 subjects (870 pairs in total), and the test set consists of all ordered pairs from the last 10 subjects (90 pairs in total). For Mindboggle101, the training set consists of all ordered pairs from the union of the NKI-RS-22 and NKI-TRT-20 subsets, (1722 pairs in total), and the test set consists of all ordered pairs from the OASIS-TRT-20 subset (380 pairs in total).

For each dataset, we train FAIM and VoxelMorph on all pairs of images from the training set, and then evaluate all methods on pairs of images from a test set. Our model is implemented with Keras [C⁺15] with Tensorflow [ABC⁺16] backend. The Adam optimizer [KB14] is used. When not otherwise specified, both networks are trained on our training set with the same hyperparameters: learning rate = 10^{-4} , epochs = 10, $\beta = 1$, $w_1 = 0$, $w_2 = 1$. FAIM has 233,683 trainable parameters, which is fewer than VoxelMorph’s 259,675.

The uTIlzReg geodesic shooting method, “GeoShoot”, does not require training, however it is very slow. We used the following options, leaving other hyperparameters at their default settings:

- For LPBA40: 60 iterations at full resolution with kernel specified by "M_Gauss_easier 4 20";
- For Mindboggle101: 20 iterations at half resolution followed by 30 iterations at full resolution, both with "-alpha 0.0003 -M_Gauss_easier 4 8".

For Mindboggle, registering a single pair of images with these settings took approximately 9 hours on an Intel Xeon Gold CPU.

We evaluate each method on two tasks: pairwise registration, and registration to a template. Registration quality is primarily evaluated using the Dice score,

$$\text{Dice}(M, F) = 2 \frac{|M \cap F|}{|M| + |F|} \quad (2)$$

to measure the degree of overlap between corresponding regions in the parcellations associated with each image. We also assess smoothness of the deformations ϕ by the Jacobian determinant $\det(\nabla\phi(x))$.

3.4 Results

3.4.1 Pair-wise registration

We compare FAIM, VoxelMorph and uTilzReg GeoShoot on both datasets.

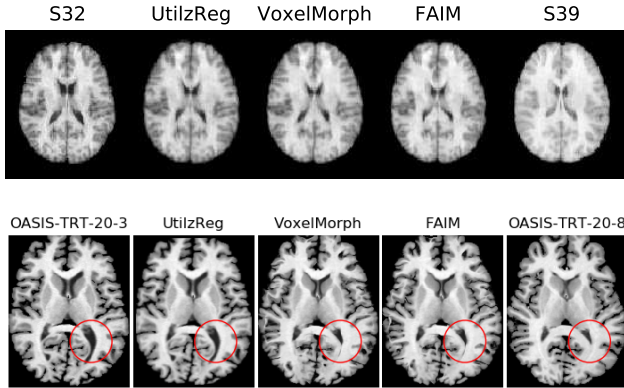


Figure 4: Pairwise registration results for test volumes from LPBA40 (top row) and Mindboggle101 (bottom row). In each row, the leftmost image is an axial slice from the moving/source volume, and the rightmost image is the same slice from the target volume. The other images show deformations of the moving volume produced by three algorithms, as labeled.

Figure 4 shows examples of the deformations produced by the three algorithms, for both datasets. In the bottom row (Mindboggle101), note that the deformations produced by FAIM and VoxelMorph are larger than the one produced by uTilzReg. In particular, note the red circled region: all three methods “know” that the ventricle needs to shrink, however only FAIM and VoxelMorph shrink it sufficiently to match the target well.

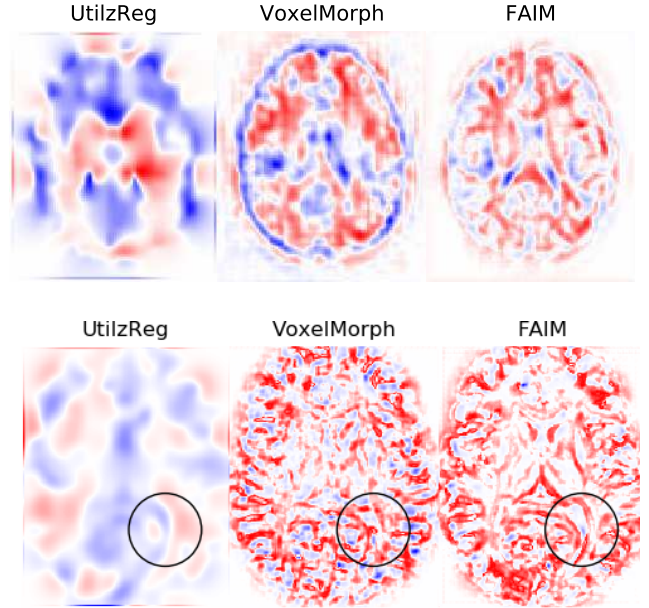


Figure 5: Pairwise registration results for LPBA40 (top) and Mindboggle101 (bottom); within each group, all images correspond to the same slice of the same test volume. Each image is a map of the Jacobian determinant of $\phi(x)$, which measures local growth (red) or shrinkage (blue).

Figure 5 shows maps of the Jacobian determinant of the deformation $\phi(x)$, for the same example brain volumes used in Figure 4. Note that uTilzReg, with the regularization used here, produces very different and much smoother transformations than either of the neural network methods. The effects of different levels of regularization are explored below.

Figure 6 shows our main result: mean Dice scores for all methods, over several large regions: each cortical lobe, plus three more regions for LPBA40. Most of these large regions are composed of a group of ROIs (regions of interest) from the parcellation; we calculate Dice scores for each ROI separately and then compute the mean across the group. The distribution of these mean scores is shown in the figure. Two sets of results for FAIM are shown: the first after training with the learning rates described above; and the second after training with a different learning rate: 10^{-3} for the LPBA40 dataset and 5×10^{-4} for the Mindboggle-101 dataset. The first set of results (shown in red) allow a fair comparison with VoxelMorph, while the second set (shown in purple) illustrates that Dice scores can be improved by further tuning hyperparameters.

We performed paired t-tests on the FAIM and VoxelMorph results on both test sets, with the null hypotheses “ H_0 : there is no mean difference in Dice scores in the specified region”. For the LPBA40 test set, FAIM’s results are significantly better ($p < 0.01$) than VoxelMorph in the following regions: Frontal Lobe, Parietal Lobe, Temporal Lobe, Cingulate Lobe and Hippocampus; while there was no significant difference for the Occipital Lobe and Putamen. For the Mindboggle101 test set, FAIM significantly outperformed VoxelMorph for all regions

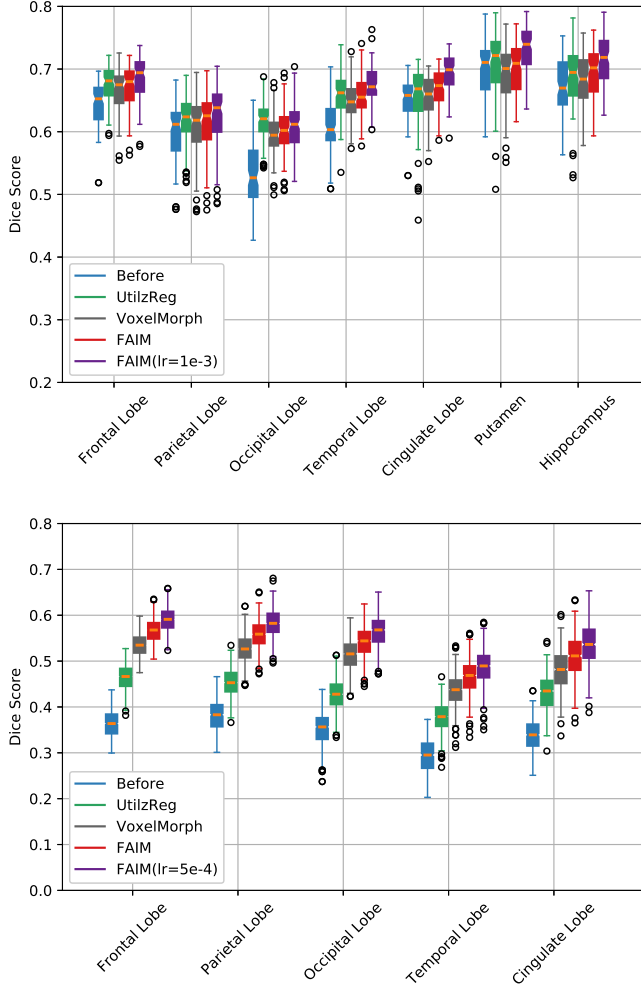


Figure 6: Mean Dice scores of different methods on selected regions. Each point is the mean Dice score averaged over corresponding ROI labels per registration pair. For example, we average the Dice score for left and right Putamen to get the data visualized in that column.

considered, and in fact this was true for every one of the 380 image pairs.

3.4.2 The effect of w_1

As we saw earlier in the 2D toy example, different deformations may result in similar performance measured visually by warped images or Dice score on labeled regions. In order to investigate how does the deformation change with choices of w_1 in more complex situations with real dataset, we conducted the experiment as below. With separate models trained using different w_1 ³ on LPBA40 training set, we record their mean Dice scores at different regions and visualize the determinant map of the Jacobian of their predicted deformations warping test subject 32 to subject 39. Table 2 and table 3 records the

³Other parameters are remained the same.

mean Dice score at different regions while using the different trained models to warp images pairs in the test set. The table suggest changing w_1 from 0 to 1 does not affect region overlapping much for most of the regions considered except for Occipital Lobe whose Dice score drops down by about 3% for both network when $w_1 = 1$. On the other hand, Figure 7 shows predicted deformation changes obviously with different choices of w_1 during training. The pattern is similar to the 2d toy example: as w_1 becomes larger, the deformation gets less smooth and large deformations will tend to occur at the boundaries of the image.

Region	0	0.001	0.01	0.1	1
Frontal Lobe	0.665	0.664	0.667	0.669	0.663
Parietal Lobe	0.607	0.613	0.612	0.613	0.611
Occipital Lobe	0.596	0.600	0.596	0.586	0.563
Temporal Lobe	0.648	0.643	0.645	0.644	0.632
Cingulate Lobe	0.656	0.652	0.657	0.660	0.654
Putamen	0.691	0.690	0.692	0.700	0.701
Hippocampus	0.682	0.674	0.676	0.689	0.673

Table 2: Mean Dice scores on different regions with VoxelMorph trained on different values of w_1 .

Region	0	0.001	0.01	0.1	1
Frontal Lobe	0.671	0.670	0.672	0.675	0.665
Parietal Lobe	0.616	0.614	0.614	0.617	0.612
Occipital Lobe	0.602	0.601	0.600	0.596	0.568
Temporal Lobe	0.656	0.655	0.656	0.655	0.637
Cingulate Lobe	0.669	0.669	0.672	0.676	0.659
Putamen	0.700	0.699	0.701	0.705	0.701
Hippocampus	0.691	0.695	0.689	0.698	0.681

Table 3: Mean Dice scores on different regions with FAIM trained on different values of w_1 . Note that every entry in this table is greater than or equal to the corresponding entry in Table 2, showing that FAIM outperforms VoxelMorph consistently.

3.4.3 The effect of w_2

We conducted similar experiments with models trained with different choices of w_2 while sharing other parameters. Figure 8 confirms that larger w_2 will result in smoother deformation for both network architectures considered. Table 4 and table 5 records mean Dice scores at each considered region when using trained models to warp image pairs in the test set. From the tables, we saw 1) all Dice scores generally get improvements as w_2 becomes larger; 2) FAIM have larger Dice score for all regions considered under different w_2 .

This change of smoothness level in the predicted deformation by varying w_2 mimics the pattern when one changes the Gaussian kernel widths used in uTilzReg GeoShoot(see figure 9). There is no direct correspondence between these kernel

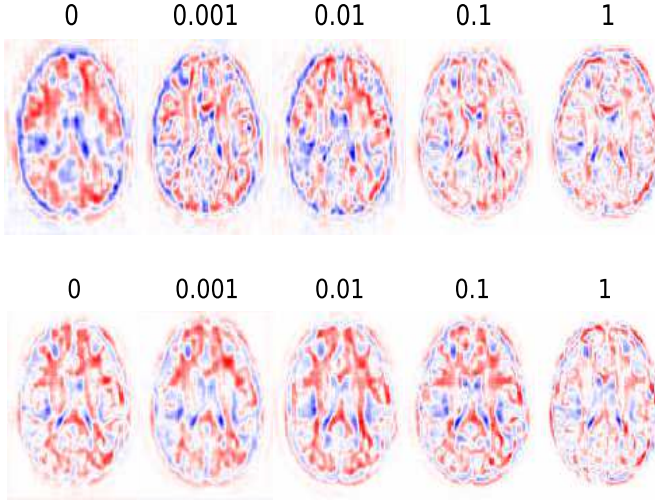


Figure 7: The effect on the Jacobian map of training with different choices of w_1 while w_2 is fixed at 1. The slices shown are taken at the same location shown in Figure 4. Top row: S32 warped to S39 in the LPBA40 dataset with VoxelMorph. Bottom row: the same pair but with FAIM.

Region	0.5	1	2	4	8
Frontal Lobe	0.658	0.665	0.667	0.669	0.667
Parietal Lobe	0.604	0.607	0.609	0.610	0.608
Occipital Lobe	0.591	0.596	0.603	0.605	0.597
Temporal Lobe	0.641	0.648	0.649	0.652	0.649
Cingulate Lobe	0.648	0.656	0.653	0.663	0.659
Putamen	0.689	0.691	0.686	0.700	0.703
Hippocampus	0.666	0.682	0.678	0.689	0.691

Table 4: Mean Dice scores on different regions with VoxelMorph trained on different values of w_2 .

widths and either w_1 or w_2 . However we observe that changing any of these parameters affects the smoothness of the Jacobian map. Note also that the Jacobian maps produced by uTilzReg are smoother and generally very different than those produced by the neural network methods.

3.4.4 One application: fast atlas creation

On application of FAIM is to use a trained network to quickly create templates and atlases from target populations. We demonstrate template creation on the LPBA40 dataset, and atlas creation on both LPBA40 and Mindboggle101 datasets. For LPBA40, we use all 40 subjects in the dataset, for comparability with the LPBA40 atlas. For Mindboggle101, we use only the 20 subjects in the OASIS-TRT-20 subset, for comparability with the “jointfusion” atlas provided with Mindboggle. Note that in the latter case all subjects used to create the atlas are “test” subjects that were not used by FAIM for training.

Our template creation method follows [ZF14]: given a spec-

Region	0.5	1	2	4	8
Frontal Lobe	0.671	0.671	0.674	0.676	0.676
Parietal Lobe	0.612	0.616	0.613	0.617	0.617
Occipital Lobe	0.599	0.602	0.603	0.608	0.605
Temporal Lobe	0.656	0.656	0.657	0.658	0.659
Cingulate Lobe	0.669	0.669	0.671	0.675	0.678
Putamen	0.695	0.700	0.701	0.710	0.712
Hippocampus	0.695	0.691	0.686	0.696	0.698

Table 5: Mean Dice scores on different regions with FAIM trained on different values of w_2 . Note that every entry in this table is greater than or equal to the corresponding entry in Table 4, showing that FAIM outperforms VoxelMorph consistently.

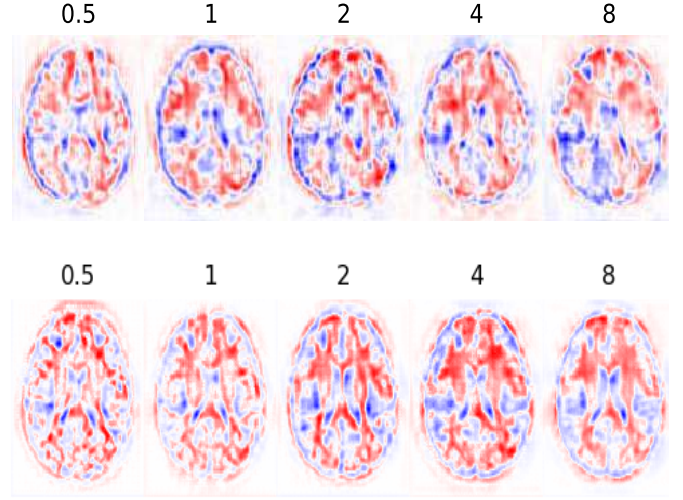


Figure 8: The effect on the Jacobian map of training with different choices of w_2 while w_1 is fixed at 0. The slices shown are taken at the same location shown in figure 4. Top row: S32 warped to S39 in LPBA40 dataset with VoxelMorph. Bottom row: same pair but with FAIM.

ified population $\{I_1, I_2, \dots, I_N\}^4$, we construct an average brain I by consecutively updating I till convergence with formula:

$$I = \frac{\sum_{k=1}^N I_k \circ \phi_k |D\phi_k|}{\sum_{k=1}^N |D\phi_k|}. \quad (3)$$

The warped image $I_k \circ \phi_k$ can be directly predicted by our network instead of performing a geodesic shooting algorithm in the original paper.

We apply this method to compute an average brain with all the 40 subjects in LPBA40 dataset, with the result shown in Figure 10. The algorithm was deemed to converge when the mean absolute change of voxel intensities of two consecutive iterations of I are less than 0.005. Note that the average brain created by FAIM appears to have more local details than the corresponding LPBA40 template (FLIRT ver-

⁴already linearly aligned to certain coordinates.

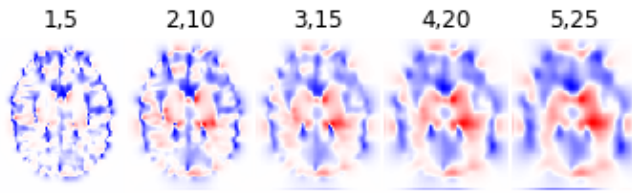


Figure 9: The effect on the Jacobian map of different Gaussian kernel widths (minimum, maximum) specified on the GeoShoot command line; for example the fourth from the left was produced using “M.Gauss.easier 4 20”. The slices shown correspond to those in Figures 7 and 8.

sion) provided with the dataset. In one sense, this is expected, since the FLIRT version of LPBA40 atlas was produced using only affine registration. However it shows that FAIM is accurate enough to produce useful templates very quickly.

The Mindboggle data set does not come with an average brain for the OASIS-TRT-20 subset, so we did not show an analogous comparison here. Instead, we use the OASIS-Atropos-30 template (in MNI152 space) provided with the dataset.

For each dataset, once we have a template, all subjects’ brain images are warped (i.e. registered) to the template, using FAIM, and the resulting warp is also applied the associated anatomical labels for each subject. Finally, at each voxel in the template, the labels from all of the warped subjects are converted to a probability distribution on labels at that location. Figure 11 shows the probability of the “winning” (i.e. most likely) label at each location using with a heat map overlaid on top of the average brain for LPBA40. The same figure shows the corresponding map for the LPBA40 atlas; the two are almost identical. The mean Dice score calculated between their winning labels is 92.6%. Notice that the average brain created using FAIM does not have brain stem and cerebellum, this is because subject images we used in LPBA40 FLIRT data set does not have them included. The network did not “see” the two parts during training, but due to the smoothness of learned deformations, it still can predict their locations when doing prediction.

The top row of Figure 12 shows the analogous comparison of label probabilities for OASIS-TRT-20 dataset. The atlas provided by Mindboggle uses OASIS-Atropos-30 as the template brain and uses the “joint fusion” algorithm [WSD⁺13] provided with ANTs to create label probabilities. The result is then warped to MNI152 space. Instead of using complex joint fusion algorithm, the probability label created using FAIM here still follows the simple procedure as in LPBA40 but with the OASIS-Atropos-30 template (warped to MNI152 space) as the average brain.

Their winning labels from the two atlases (FAIM and Mindboggle) have a mean Dice score of 80.7%. The reason the lower Dice score here is that label probabilities generated by using FAIM looks less certain (more cold regions) and spreads more spatially compared with Mindboggle’s. The bottom row

of figure 12 shows some winning labels by stacking the labels from Mindboggle’s atlas (green) and labels constructed using FAIM (blue). One can see that the former is almost entirely contained within the latter. We expect that if joint fusion algorithm were used together with FAIM warping the image, the result will look as good as using diffeomorphic methods for image warping.

The whole process including the average brain construction and creation of each label’s probability map only took about 6 min for LPBA40 dataset and 12 min for OASIS-TRT-20 dataset with a single Tesla P100 GPU and unoptimized codes.

4 Discussion

We have developed a new unsupervised learning algorithm, FAIM, for 3D medical image registration. We compared it experimentally to VoxelMorph [BZS⁺18] and uTilzReg GeoShoot [VRR⁺12], on the LPBA40 and Mindboggle101 datasets, using pairwise registration of individual brain volumes.

The FAIM algorithm produced significantly better registrations (as measured by label overlap) than VoxelMorph on all Regions of Interest (ROIs) for the Mindboggle101 dataset. For the LPBA40 dataset, FAIM produced registrations that were significantly better than VoxelMorph in some ROIs and not significantly different in others. This pattern is consistent across varying regularization parameters. Further, and surprisingly, both neural network algorithms significantly outperformed uTilzReg GeoShoot on Mindboggle101. However, this was at the cost of large and non-smooth local volume changes, as illustrated in Figure 5. The striking differences shown in that figure suggest that all three methods could be improved by more careful regularization. Finally, we demonstrated an application of FAIM to fast template and atlas construction, on both the LPBA40 and Mindboggle datasets.

References

- [ABC⁺16] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016. 3
- [ATS⁺11] Brian B Avants, Nicholas J Tustison, Gang Song, Philip A Cook, Arno Klein, and James C Gee. A reproducible evaluation of ants similarity metric performance in brain image registration. *Neuroimage*, 54(3):2033–2044, 2011. 1
- [BZS⁺18] Guha Balakrishnan, Amy Zhao, Mert R Sabuncu, John Guttag, and Adrian V Dalca. An unsupervised learning model for deformable medical image registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9252–9260, 2018. 1, 2, 7
- [C⁺15] François Chollet et al. Keras. <https://keras.io>, 2015. 3

- [Des06] Rahul S Desikan, S3gonne f, fischl b, quinn bt, dickerson bc, blacker d, et al: An automated labeling system for subdividing the human cerebral cortex on mri scans into gyral based regions of interest. *Neuroimage*, 31(3):968–80, 2006. 3
- [HZRS15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015. 2
- [JSZ⁺15] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015. 1
- [KB14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 3
- [KT12] Arno Klein and Jason Tourville. 101 labeled brain images and a consistent human cortical labeling protocol. *Frontiers in neuroscience*, 6:171, 2012. 3
- [Kua18] Dongyang Kuang. Code used in the paper. <https://github.com/dykuang/Medical-image-registration>, 2018. 1
- [LF17] Hongming Li and Yong Fan. Non-rigid image registration using fully convolutional networks with deep self-supervision. *arXiv preprint arXiv:1709.00799*, 2017. 1
- [RDH⁺17] Marc-Michel Rohé, Manasi Datar, Tobias Heimann, Maxime Sermesant, and Xavier Pennec. Svf-net: Learning deformable image registration using shape matching. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 266–274. Springer, 2017. 1
- [SBB⁺01] Stephen Smith, Peter R Bannister, Christian Beckmann, Mike Brady, Stuart Clare, David Flitney, Peter Hansen, Mark Jenkinson, Didier Lebovici, Brian Ripley, et al. Fsl: New tools for functional and structural brain image analysis. *NeuroImage*, 13(6):249, 2001. 3
- [SdVB⁺17] Hessam Sokooti, Bob de Vos, Floris Berendsen, Boudewijn PF Lelieveldt, Ivana Išgum, and Marius Staring. Nonrigid image registration using multi-scale 3d convolutional neural networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 232–239. Springer, 2017. 1, 2
- [SGY⁺17] Siyuan Shan, Xiaoqing Guo, Wen Yan, Eric I Chang, Yubo Fan, Yan Xu, et al. Unsupervised end-to-end learning for deformable medical image registration. *arXiv preprint arXiv:1711.08608*, 2017. 1
- [SLJ⁺15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 2
- [SMA⁺08] David W Shattuck, Mubeena Mirza, Vitria Adisetiyo, Cornelius Hojatkashani, Georges Salamon, Katherine L Narr, Russell A Poldrack, Robert M Bilder, and Arthur W Toga. Construction of a 3d probabilistic atlas of human cortical structures. *Neuroimage*, 39(3):1064–1080, 2008. 3
- [VRRC12] François-Xavier Vialard, Laurent Risser, Daniel Rueckert, and Colin J Cotter. Diffeomorphic 3d image registration via geodesic shooting using an efficient adjoint calculation. *International Journal of Computer Vision*, 97(2):229–241, 2012. 1, 2, 7
- [WKWS17] Shaoyu Wang, Minjeong Kim, Guorong Wu, and Dinggang Shen. Scalable high performance image registration framework by unsupervised deep feature representations learning. In *Deep Learning for Medical Image Analysis*, pages 245–269. Elsevier, 2017. 1
- [WSD⁺13] Hongzhi Wang, Jung W Suh, Sandhitsu R Das, John B Pluta, Caryne Craige, and Paul A Yushkevich. Multi-atlas segmentation with joint label fusion. *IEEE transactions on pattern analysis and machine intelligence*, 35(3):611–623, 2013. 7
- [YHT⁺17] Inwan Yoo, David GC Hildebrand, Willie F Tobin, Wei-Chung Allen Lee, and Won-Ki Jeong. ssemnet: Serial-section electron microscopy image registration using a spatial transformer network with learned features. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 249–257. Springer, 2017. 1
- [ZF14] Miaomiao Zhang and P Thomas Fletcher. Bayesian principal geodesic analysis in diffeomorphic image registration. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 121–128. Springer, 2014. 6

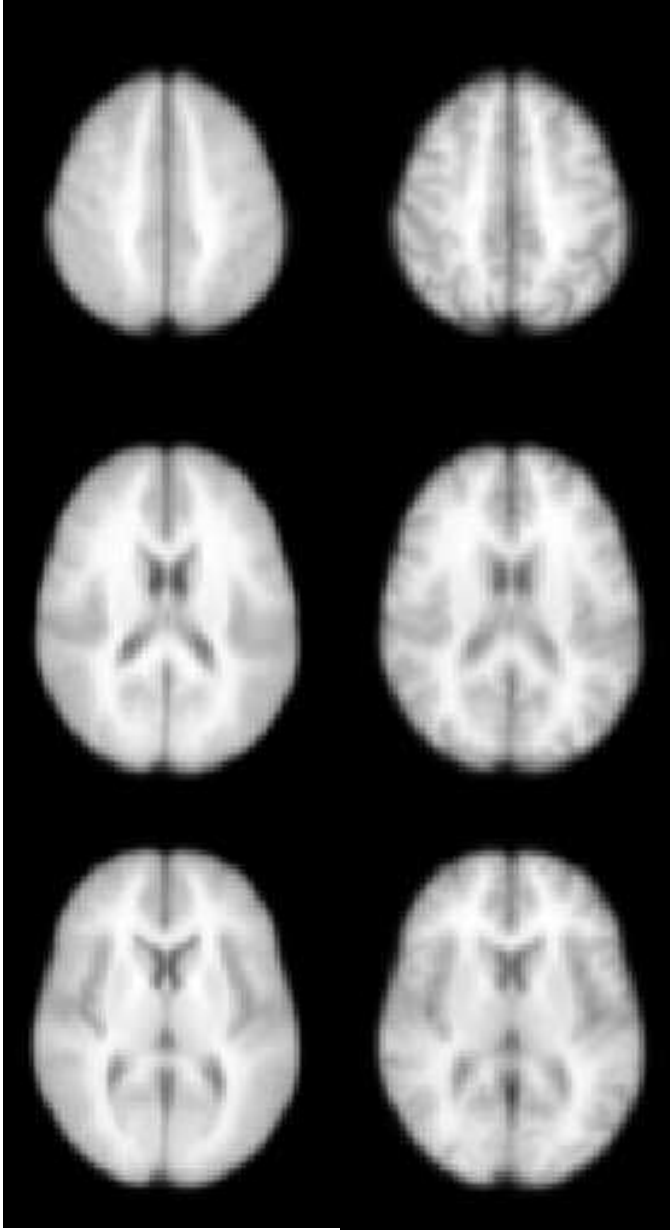


Figure 10: The average brain (3 slices at different locations) provided by the FLIRT version of the LPBA40 atlas (left) and the one constructed using FAIM (right).

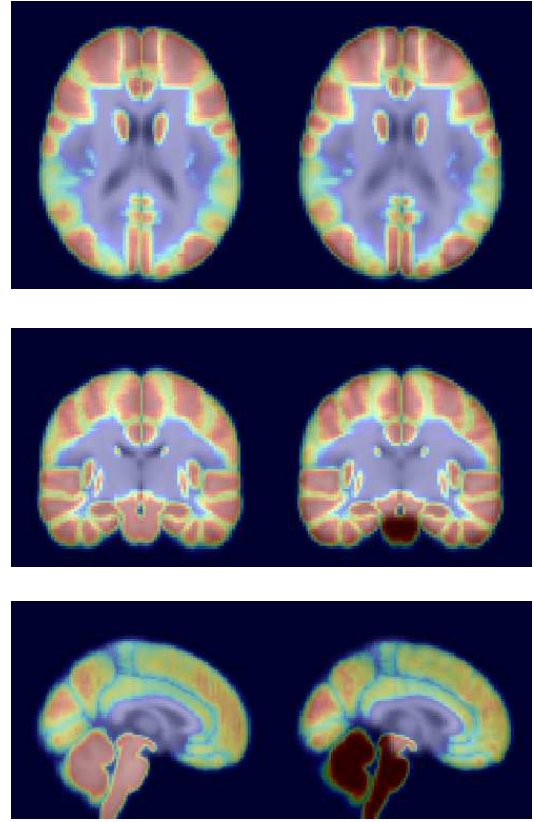


Figure 11: Maximum likelihood provided by LPBA40 atlas (left) and the one constructed using FAIM (right)

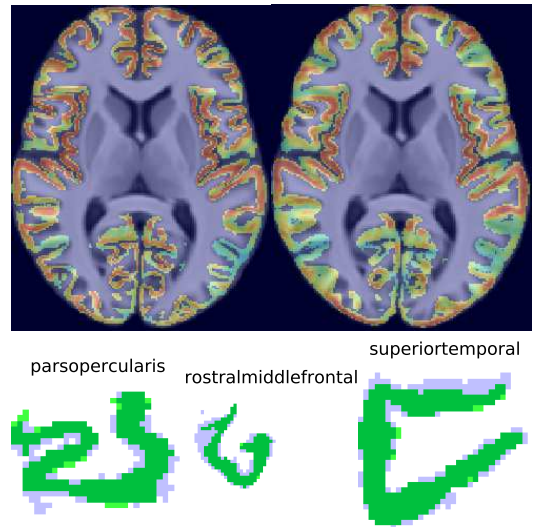


Figure 12: Top Row: Maximum likelihood of labels provided by Mindboggle's jointfusion atlas (left) and the one constructed using FAIM (right) . Bottom Row: Overlay visualization of some winning labels taken at the same slice, winning labels from Mindboggle's jointfusion atlas (Green) and winning labels constructed by FAIM (blue).