Multi-Modal Image Registration with Unsupervised Deep Learning

by

Courtney K. Guo

B.S., C.S., M.I.T., 2018

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering
in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

May 2019

Signature of Author: _____
Department of Electrical Engineering and Computer Science
May 24, 2019

Certified by: _____
Adrian Dalca, Instructor of Radiology, Harvard Medical School
Thesis Supervisor
May 24, 2019

Certified by: _____
John Guttag, Professor of Electrical Engineering and Computer Science
Thesis Co-Supervisor
May 24, 2019

Accepted by: _____
Katrina LaCurts, Chair, Master of Engineering Thesis Committee

Multi-Modal Image Registration with Unsupervised Deep Learning

by Courtney K. Guo

Submitted to the Department of Electrical Engineering and Computer Science
May 24, 2019
In partial fulfillment of the requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

In this thesis, we tackle learning-based multi-modal image registration. Multi-modal registration, in which two images of different modalities need to be aligned to each other, is a difficult yet essential task for medical imaging analysis. Classical methods have been developed for single-modal and multi-modal registration, but are slow because they solve an optimization problem for each pair of images. Recently, deep learning methods for registration have been proposed, and have been shown to shorten registration time by learning a global function to perform registration, which can then be applied quickly on a pair of test images. These methods perform well for single-modal registration but have not yet been extended to the harder task of multi-modal registration. We bridge this gap by implementing classical multi-modal metrics in a differentiable and efficient manner to enable deep image registration for multi-modal data. We find that our method for multi-modal registration performs significantly better than baselines, in terms of both accuracy and runtime.

Acknowledgments

I would like to thank Adrian Dalca for advising me through my research and providing valuable suggestions and insights along the way. I would like to thank Professor John Guttag and the other members of the Data-Driven Inference Group at CSAIL for their support in the lab as well their useful feedback of my work. I'd also like to thank my parents for encouraging me throughout all of my endeavours in the past year, as well as their endless support in the 21 years prior.

Contents

List of Figures

Chapter 1

Introduction

Image registration is essential in many medical imaging tasks, including the planning and execution of surgical procedures, the diagnosis of various conditions, and the analysis of population statistics. The goal of image registration is to compute a vector field that describes the deformation of the moving image to align with the fixed image.

Different imaging techniques are sensitive to different tissues in the body. Therefore, images of different modalities might need to be registered to each other to provide complementary information. In this thesis, we develop methods for multi-modal image registration, evaluated on a dataset of brain scans acquired with MRI-T1 and MRI-T2. As shown in Figure 1.1, MRI-T1 images do well to distinguish between different healthy tissues in the brain, whereas MRI-T2 images are best for highlighting abnormal structures such as tumors.

Multi-modal registration is more difficult than single-modal registration because of the complex relationship between the intensities of corresponding structures in the two images. Tissues that are dark in images of a certain modality may be bright in images of a different modality. Therefore, one cannot use cost functions such as mean squared error or cross-correlation to compute the cost of misaligning two images that are of different modalities.

Multi-modal image registration has been implemented using traditional image registration methods, which solve an optimization problem per pair of images by minimizing the cost of misalignment and maximizing the smoothness of the deformation. Using traditional methods, single-modal image registration on a pair of 3D images can take tens of minutes to hours to complete on a CPU. Recent machine learning methods shorten single-modal registration time to a few minutes on a CPU or a few seconds on a GPU, without sacrificing registration accuracy for single-modal image registration [3]. We aim to extend this research to multi-modal image registration by using loss functions compatible with multi-modal registration. The loss functions we explore are based

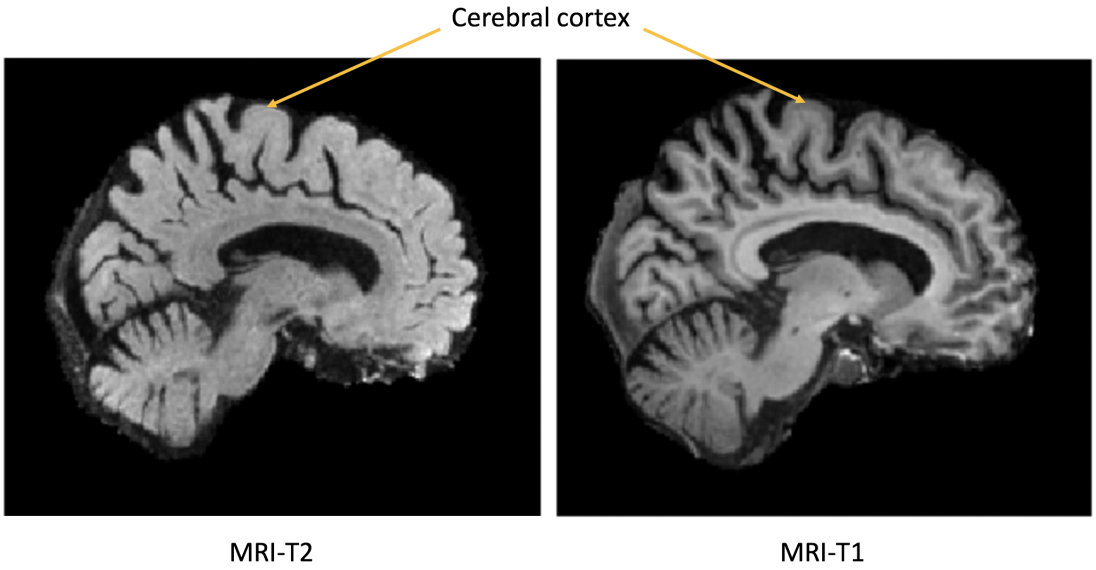

Figure 1.1. MRI-T1 and MRI-T2 brain scans of the same patient. The contrast between different tissues is significantly better in MRI-T1 than MRI-T2, most noticeably in the delineation of the the cerebral cortex.

on mutual information (MI) and the Modality-Independent Neighborhood Descriptor (MIND).

Chapter 2

Background

2.1 Medical Image Registration

There have been many methods developed for medical image registration [22]. Some of these methods concern the optimization of the deformation field, including elastic registration [2], [12], [31], b-spline registration [30], and discrete (non-differentiable) methods [15], [10]. All of these methods solve an optimization problem on a pair of images, where each pair is solved independently.

2.2 Learning-Based Registration

Instead of registering a pair of images using traditional optimization techniques, learning-based methods learn a global registration function from training data, and then apply the learned function onto pairs of images to perform registration. Many of these methods are supervised, and require ground truth deformations in order to train the model [29], [5], [32]. Since labeled data can be difficult to obtain in medical imaging, unsupervised methods have recently been proposed. Balakrishnan *et al.* [3], [4], [11] proposed an unsupervised method for image registration using a deep convolutional neural network, VoxelMorph, that takes in an image pair and outputs a deformation field to warp one image into the other. The architecture used in VoxelMorph, depicted in Figure 2.1, is similar to UNet: it has an encoder-decoder structure that has skip connections between encoder and decoder layers of the same size.

VoxelMorph learns a transformation function ϕ to perform registration. Figure 2.2 shows the overall model: the convolutional neural network takes a pair of images as input and outputs a registration field ϕ, which is used to warp the moving image. A loss function is computed on the warped moving image and the fixed image. VoxelMorph is trained in an unsupervised fashion on a dataset of 3D MRI images, by minimizing the

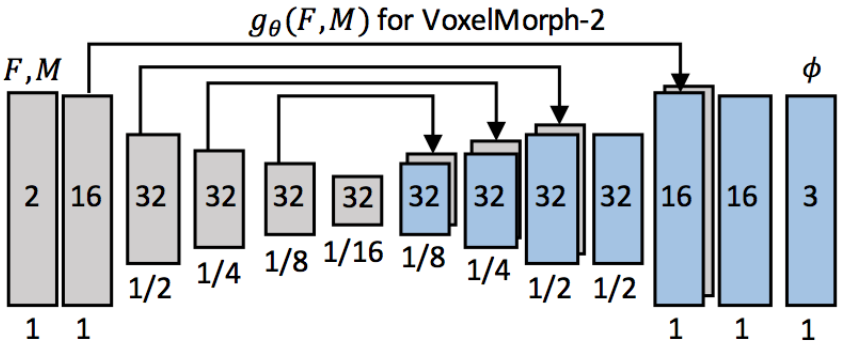

Figure 2.1. Structure of the VoxelMorph encoder-decoder model. Each rectangle represents a 3D volume. The number of channels is shown inside the rectangle, and the spatial resolution with respect to the input volume is printed underneath. Diagram taken from [3].

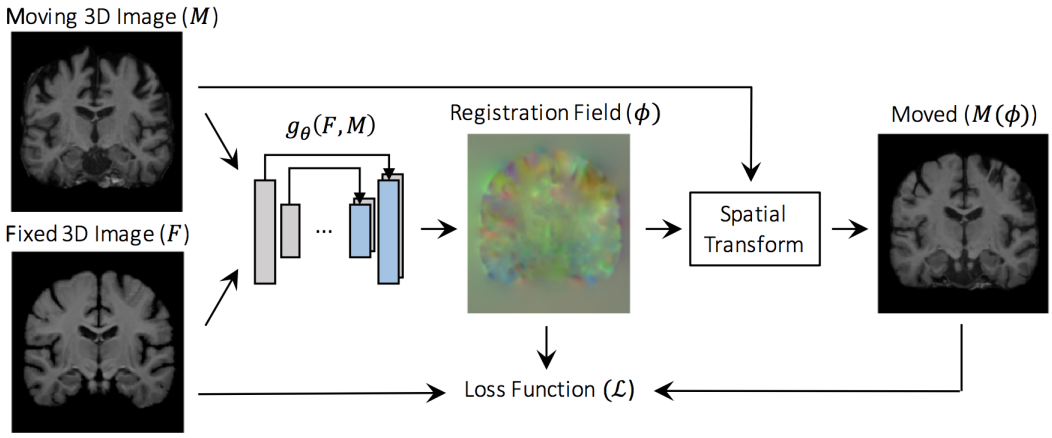

Figure 2.2. Diagram of the overall model. VoxelMorph learns a transformation ϕ that warps the moving image M to align to the fixed image F. The loss function is a function of the similarity between the warped image $M(\phi)$ and the fixed image, as well as the smoothness of the transformation ϕ. Diagram taken from [3].

following cost function:

$$\mathcal{L}(F, M, \phi) = \mathcal{L}_{sim}(F, M(\phi)) + \lambda \mathcal{L}_{smooth}(\phi), \tag{2.1}$$

where F and M are the two input images, ϕ is the deformation field that warps M to match F, $\mathcal{L}_{sim}$ is a cost function that represents how closely aligned the warped M is

to F, $\mathcal{L}_{smooth}$ is a regularization on the deformation field, and λ is the weight given to the regularization. VoxelMorph has been demonstrated to work for $\mathcal{L}_{sim}$ equal to mean squared error or cross-correlation.

Once VoxelMorph is trained, it only takes one second on a GPU or one minute on a CPU to register two images, whereas traditional methods may take on the order of hours on a CPU. This vast speedup can be explained by amortization: VoxelMorph takes on the order of days to train, but once trained it registers each pair of test images very quickly. The precomputation of training the network is what allows for the fast registration time. For single-modal image registration, VoxelMorph performs just as well as state-of-the-art methods. However, VoxelMorph has not yet been extended to multi-modal registration because it uses cost functions such as mean-squared-error and cross-correlation, which assume that the intensities in corresponding structures in the two images have a linear relationship.

■ 2.3 Mutual Information

Image registration methods try to maximize the similarity between the two images, so that the images are maximally aligned to each other. Many similarity metrics have been proposed: the most common compare voxel intensities between the two images such as L2 norm, L1 norm, and cross correlation [17]. However, to perform multi-modal registration we cannot assume that the relationship between intensities in the two images are linear.

There have been methods developed for multi-modal image registration using traditional optimization techniques. One common objective function uses mutual information (MI), a classic concept in information theory [7], defined intuitively as the amount of information one distribution gives about another. In 1997, Viola *et al.* [34] was one of the first to use MI for image alignment. Now it has become a common loss function for image registration [28].

Formally, the mutual information between two images A and B is defined as the following [28]:

$$I(A, B) = \sum_{a,b} p(a, b) \log \frac{p(a, b)}{p(a)p(b)}. \tag{2.2}$$

The probability $p(a)$ is the proportion of the voxels in image A with intensity equal to a, and the probability $p(b)$ is likewise for image B. These probabilities are calculated by constructing a histogram of pixel intensities for each of the two images. The probability

$p(a, b)$ is the joint distribution of the intensities of two images A and B: it is the proportion of pairs of corresponding pixels in the two images that have intensities equal to the tuple (a, b). Equation (2.2) is equal to the Kullback-Leiber divergence [33] between the probability distributions $p(a, b)$ and $p(a)p(b)$. Therefore, $I(A, B)$ is maximized when the two distributions are the least similar. Since $p(a)p(b)$ represents the distribution of pixel intensities if the two images were independent, $p(a, b)$ and $p(a)p(b)$ should be the least similar when the two images A and B are aligned, because it maximizes the amount of information one image gives about the other. To implement this in practice, we bin intensities to calculate the KL divergence in a discrete manner.

To do multi-modal image registration, one can use negative mutual information in place of $\mathcal{L}_{sim}(\cdot, \cdot)$ (2.1), so that we align two images by maximizing their mutual information. This can be done globally or locally: global mutual information is the mutual information of the two images, whereas local mutual information is the aggregate of the mutual information of corresponding patches of the two images, to be further explained in Section 3. Mutual information has been shown to work with traditional image registration methods but has not yet been used with learning-based registration.

■ 2.4 MIND

Another loss function that has been used for multi-modal image registration with traditional optimization techniques is a similarity metric involving the Modality Independent Neighbourhood Descriptor (MIND) [18]. MIND is a feature that describes the local patterns around each voxel. This is computed by looking at the similarity between the central patch and patches a certain distance away. The assumption behind the MIND-based similarity function is that the local patterns around a voxel should be similar even across different image modalities, so we wish to minimize the difference in the MIND features between the two images we are registering.

MIND is parameterized by a distance vector $\mathbf{r}$ and patch size p. To compute a MIND-based loss function, we wish to look at the similarity of 6-neighborhood patches to a central patch, depicted in Figure 2.3. We first define D_p as a similarity metric on a pair of image patches:

$$D_p(I, \mathbf{x_1}, \mathbf{x_2}) = \frac{1}{|P|} \sum_{\mathbf{t} \in P} (I(\mathbf{x_1} + \mathbf{t}) - I(\mathbf{x_2} + \mathbf{t}))^2. \tag{2.3}$$

Here, I is an image, $\mathbf{x_1}$ and $\mathbf{x_2}$ are two locations in the image, and P is the set of dis-

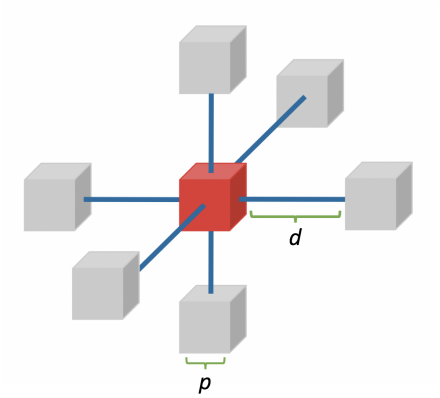

Figure 2.3. Visualization of the 6-neighborhood patches used in $\mathcal{L}_{\text{MIND}}$, parameterized by a patch size p and a distance d. Diagram adapted from [19].

placements from a voxel in a patch of size $p \times p \times p$ to the center of the patch. Therefore, D_p computes the mean squared difference between two image patches centered at $\mathbf{x_1}$ and $\mathbf{x_2}$, with patch size $p \times p \times p$. Then we define MIND as follows:

$$\text{MIND}(I, \mathbf{x}, \mathbf{r}) = \exp\left(\frac{-D_p(I, \mathbf{x}, \mathbf{x}+\mathbf{r})}{V(I, \mathbf{x})}\right). \tag{2.4}$$

Here, I is an image, $\mathbf{x}$ is a location in the image, $\mathbf{r}$ is a distance vector, and $V(I, \mathbf{x})$ is an estimate of the local variance. We let MIND be a Gaussian function of D_p to allow MIND to have a low response when patches are dissimilar and a high response when patches are similar.

To construct a MIND-based loss function for image registration, we take the mean of absolute differences between the MIND features of the two images we wish to align. Therefore, we define:

$$\mathcal{L}_{sim}(A, B) = \frac{1}{|R|} \sum_{\mathbf{r} \in R} |\text{MIND}(A, \mathbf{x}, \mathbf{r}) - \text{MIND}(B, \mathbf{x}, \mathbf{r})|, \tag{2.5}$$

where A and B are the fixed image and warped moving image respectively, and R is a set of six displacement vectors of length d, with two in each x, y, or z direction. MIND has been shown to work with traditional image registration methods [18], [19] but has not yet been used with learning-based registration.

Chapter 3

Methods

We aim to extend VoxelMorph to be able to tackle multi-modal image registration. To do this, we will experiment with different loss functions to be used as $\mathcal{L}_{sim}(\cdot, \cdot)$. Mutual information and MIND have both been shown to work as multi-modal loss functions in traditional registration methods, so we will implement these loss functions to work with learning-based methods.

3.1 Differentiable Mutual Information

As mentioned in Section 2, mutual information is a loss function that has been shown to work with multi-modal image registration. To use mutual information with VoxelMorph, it must be approximated in a differentiable way. Intuitively, each voxel should contribute continuously to a range of histogram bins, instead of contributing only to the bin it falls into. We will use Parzen windowing [28], which calculates $P(x)$ as follows: given a set of n samples S, each sample s contributes to $P(x)$ with a function of its distance to x:

$$P_S(x) = \frac{1}{n} \sum_{s \in S} W(x - s). \tag{3.1}$$

We will use a Gaussian function as the weighting function W, with a parameter σ:

$$W(x - s) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-s)^2}{2\sigma^2}}. \tag{3.2}$$

To compute the MI between two images A and B, we need to compute each image's intensity distribution, as well as the two images' joint distribution. The intensity distribution of a single image A is given by $P_A(x)$. To compute the joint distribution, we

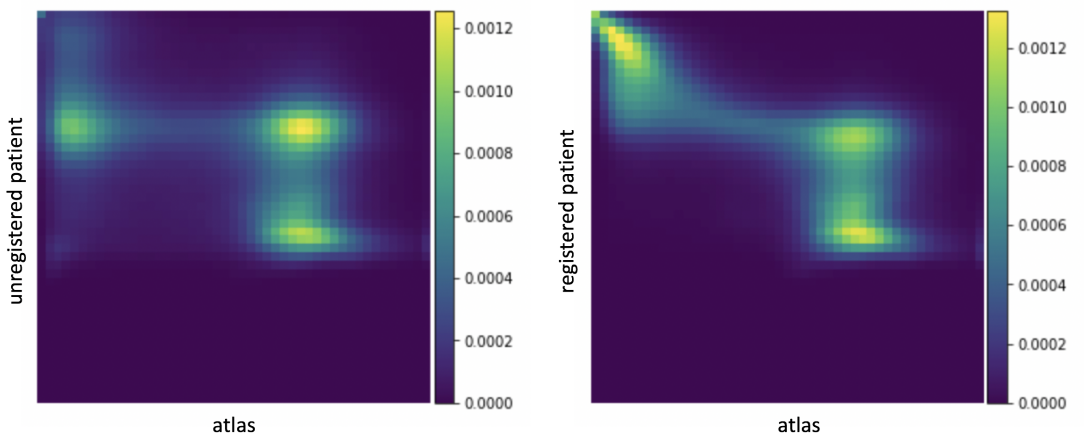

Figure 3.1. Heat maps that depict the joint distribution $p(a, b)$ between two images A and B. Left: joint distribution of the atlas and a patient scan before registration. Right: joint distribution of the atlas and a patient scan, after the patient scan has been registered to the atlas by maximizing MI.

treat each sample as a pair of intensities of corresponding locations in the two images:

$$P_{A,B}(x, y) = \frac{1}{n} \sum_{(a,b)\in(A,B)} W(x - a)W(y - b). \tag{3.3}$$

To calculate the mutual information of the two images using (2.2), we evaluate $P_A(x)$ and $P_B(x)$ at k equally-spaced intensity bin centers, and $P_{A,B}(x, y)$ at all k^2 pairs of bin centers. The parameter k determines the number of bins in the histogram of each image, affecting the accuracy of this approximation of mutual information.

Figure 3.1 shows a visualization of the joint distribution of two images, $p(a, b)$, evaluated using $k = 48$ bins. Each square at (i, j) in the heat map represents the proportion of voxel locations that have intensity i in image A and intensity j in image B, approximated using Parzen windowing as described above. The heat map on the left depicts the distribution $p(a, b)$ for two unregistered images, whereas the right depicts $p(a, b)$ for two images registered with a MI-based loss function. Intuitively, the MI of the two images after registration is larger because the heat map has more concentrated bright spots, which implies that the intensity of voxels in one image give more information about the intensity of corresponding voxels in the other image.

Mutual information can be used as a loss function in two ways: global and local. Global mutual information is just the mutual information of two images, whereas local

mutual information is the average of the mutual information of corresponding patches of the two images. To use a global MI-based loss function, we define the following:

$$\mathcal{L}_{sim}(A, B) = -I(A, B). \tag{3.4}$$

To use a local MI-based loss function, we define the following:

$$\mathcal{L}_{sim}(A, B) = -\frac{1}{|P|} \sum_{P \in \mathcal{P}} I(A(P), B(P)), \tag{3.5}$$

where P is a set of patches. In both equations, we take the negative MI because MI is maximized when the two images are aligned, and we want to minimize our loss function.

Global MI is based on intensity distributions over the entire image. So, given two voxels that have the same intensity but are spatially far away from each other, global MI will treat them the same manner. To allow us to consider voxels with similar intensity only if they are spatially close together, we need to use local MI instead. Because local MI is able to capture spatial information, we intuitively expect local MI to perform better but potentially at a cost of overfitting.

■ 3.2 Vectorized MI

To use global and local MI with VoxelMorph, they must be implemented in a vectorized manner. Vectorized operations run vastly faster than looped operations, and since MI loss must be computed every time a pair of images is passed through the network, we can only use vectorized operations to implement MI or else VoxelMorph will not train in a timely manner.

■ 3.2.1 Vectorized Global MI

To compute the global MI of two images A and B, we first compute matrices I_A and I_B, each of which have shape num_voxels $\times$ num_bins, which represent how much each voxel contributes to each intensity bin as given by equation (3.2). This can be done by first reshaping and tiling the image to have num_voxels $\times$ num_bins dimensions, subtracting the intensities of the bin centers, and then doing element-wise operations to compute the Gaussian function. Then to compute the distributions $p(a)$ and $p(b)$, we compute the mean of I_A and the mean of I_B, both along axis 0. To compute the joint distribution $p(a, b)$, we simply evaluate $\frac{1}{n}(I_A)^T \cdot I_B$, where n is the number of voxels in each image, as this will give us exactly the sum in Equation (3.3). This is summarized

in the pseudocode in Algorithm 1.

$A, B \leftarrow \text{reshape}(A, -1), \text{reshape}(B, -1)$
$I_A \leftarrow \exp(\text{square}(A[:, \text{new\_axis}] - \text{bin\_centers}[\text{new\_axis}, :]))$
$I_B \leftarrow \exp(\text{square}(B[:, \text{new\_axis}] - \text{bin\_centers}[\text{new\_axis}, :]))$
$I_A, I_B \leftarrow \text{normalize}(I_A, \text{axis} = 1), \text{normalize}(I_B, \text{axis} = 1)$
$P_A, P_B \leftarrow \text{mean}(I_A, \text{axis} = 0), \text{mean}(I_B, \text{axis} = 0)$
$P_{AB} \leftarrow \text{dot}(I_A^T, I_B)/n$
$P_A P_B \leftarrow \text{dot}(P_A^T, P_B)$
$MI \leftarrow \text{sum}(P_{AB} * \log(P_{AB}/P_A P_B + \epsilon))$

Algorithm 1. Pseudocode for global MI.

■ 3.2.2 Vectorized Local MI

To compute the local MI of two images A and B, we first reshape each image into patches of size $p \times p \times p$. We then reshape further to get arrays A' and B', each of size num_patches $\times$ p^3, where num_patches $= \frac{n}{p^3}$. Note that this implementation implies that the patches do not overlap. To get overlapping patches one could create multiple copies of each image and reshape them to create offset patches, but for the sake of memory usage we did not create overlapping patches. Once we get A' and B', we treat the first dimension as batch size, and proceed with the same computations as those for global MI except everything is now batched.

■ 3.3 Vectorized MIND

In order to use a MIND-based loss function with VoxelmMrph, we need to implement MIND in a vectorized manner. Given an image A, a MIND feature is computed by a Gaussian function of the mean squared difference between a central patch of A and one of its six neighboring patches some fixed distance away. To do this in a vectorized fashion, we first shift image A by distance d in one of the six directions, and subtract the original image A. Then, we can take the element-wise square of each copy and apply a convolution with patch size $p \times p \times p$ and kernel uniformly equal to $\frac{1}{p^3}$. This results an image where the value at location x is the mean squared difference between the patch shifted by d in the chosen direction and the patch centered at x. To compute the Gaussian function on the mean squared difference, we simply apply element-wise operations such as squaring and dividing. To compute the MIND-based loss function, we need to compute the MIND feature in every one of the six directions, and then take

the absolute difference in the MIND features of the two images, averaged over all n voxels and the six directions.

Chapter 4

Experiments

4.1 Dataset

In order to evaluate our methods for multi-modal image registration, we need a multi-modal dataset. We use a large-scale, multi-site, multi-study dataset of 14,000 MRI-T1 images taken from eight publicly available datasets: ADNI [27], OASIS [23], ABIDE [25], ADHD200 [26], MCIC [16], PPMI [24], HABS [8], and Harvard GSP [21]. This dataset was first used by Dalca *et al.* [9] and is also used by Balakrishnan *et al.* [3] The dataset is split into approximately 6000 train, 3000 validate, and 5000 test images. Images were preprocessed with standard procedures such as affine spatial normalization and brain extraction, and were segmented with FreeSurfer [14]. Manual quality control was applied to catch large errors in segmentation. We also use a dataset of MRI-T2 images, taken from the ADNI [27] dataset. We model our task as registering patient T1 images to a T2 atlas. Since we did not have a T2 atlas and constructing one is difficult, we chose one of the T2 scans as our "atlas," and modeled our task as trying to register T1 scans to this T2 "atlas."

Although our methods are unsupervised and therefore don't require segmentations, we use segmentation accuracy as a proxy for evaluating image registration accuracy, so we need segmentations for our T2 atlas in order to evaluate our methods. The T2 scans are not segmented, but all the T1 scans are, so we take the same patient's T1 scan and affinely register it to the T2 scan. Then, we warp the segmentations of the T1 scan with that affine transformation, to get segmentations for the T2 scan. The affine transformation can be computed with ANTs-MI, which is software for multi-modal registration that uses traditional optimization algorithms to maximize MI. Since both scans are from the same subject, we expect that an affine registration will be fairly accurate. There is no way to evaluate this because we do not have T2 segmentations, but we can visualize the resulting segmentation in Figure 4.1. On the left we have ground

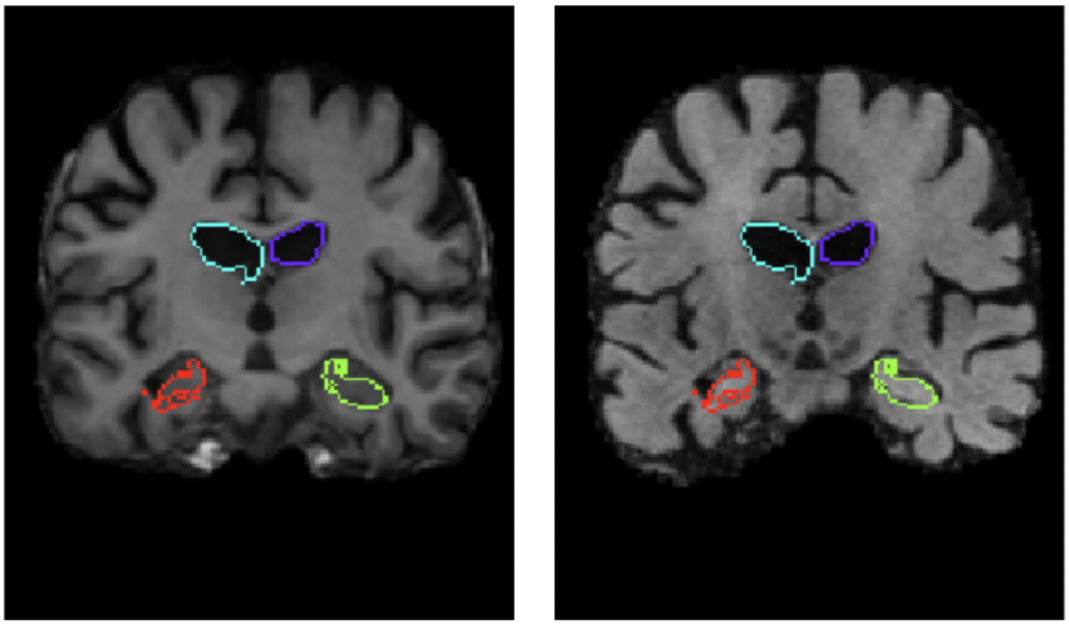

Figure 4.1. Left: original T1 scan and segmentation, affinely warped to the T2 scan. Right: T2 scan overlaid with warped T1 segmentation. Top two structures are ventricles, and the bottom two are the hippocampi.

truth segmentation of the T1 scan, displayed after linearly warping to match the T2 scan. On the right we have the T2 scan overlaid with the warped T1 segmentation. This is just one slice of the entire brain scan, but we can see that the ventricles (top two structures) seem to be nearly perfectly segmented, and the hippocampi (bottom two structures) are almost identical in both images as well.

As a last pre-processing step, we make sure that our new T2 "atlas" is affinely registered to all the T1 patients, which makes it easier to train our model.

■ 4.2 Evaluation Metrics

To evaluate image registration methods, we evaluate the segmentations resulting from the atlas-based registration. When a patient image is registered to an atlas, we get a deformation field that warps the patient image to the atlas. By warping the patient's segmentation with that deformation field, we get a segmentation of the atlas. We can then evaluate the accuracy of that segmentation by comparing it to the ground

truth segmentation of the atlas. Therefore, we are essentially evaluating our image registration procedure using a proxy, which is the segmentation accuracy. The Dice score [13] is a metric for segmentation accuracy of a single structure:

$$Dice(A, B) = \frac{|A \cap B|}{|A| + |B|},$$

where A is the set of pixels belonging to the structure in the predicted segmentation and B is the set of pixels belonging to the structure in the ground truth segmentation. The Dice score calculates the percentage of overlap between the two segmentations.

To evaluate our proposed methods, we evaluate their Dice scores on our multi-modal dataset. We identified 30 different brain regions that had at least 100 voxels in volume in all the test subjects, so we compute the Dice score for each patient as the average of the Dice score for each of those 30 structures. We will evaluate our proposed methods against two baselines for traditional multi-modal image registration, ANTs and NiftyReg, as well as upper baselines for learning-based methods on a single-modal dataset.

4.3 Baselines

4.3.1 Lower Baselines

We use Symmetric Normalization (SyN) as one of our baselines, implemented with the Advanced Normalization Tools (ANTs) software. We use MI loss computed with 48 intensity bins. We also use the following parameters, which were found to be optimal in [3]: SyN step size of 0.25, Gaussian parameters (9, 0.2), and 201x201x201 for the convergence parameters. We use the NiftyReg package for our second baseline, also with MI as the loss metric.

4.3.2 Upper Baselines

Since single-modal registration is an easier problem, we use VoxelMorph trained with mean-squared-error loss (VM-MSE) for T1-T1 registration as an upper baseline. We trained VM-MSE using the T1 scan of the patient we chose for the T2 "atlas," in order to account for decrease in Dice loss caused by the lack of a clean T2 atlas.

	VM-MSE on T1-T1	VM-MSE on T1-T1 (patient atlas)	VM-MI (global) on T1-T2	VM-MI (local) on T1-T2	VM-MIND on T1-T2	ANTs-MI on T1-T2	NiftyReg-MI on T1-T2
Average Dice	0.763	0.760	0.686	0.698	0.694	0.613	0.660
Std (50 patients)	0.015	0.029	0.033	0.043	0.036	0.051	0.035

Figure 4.2. Average Dice score over 30 brain regions and 50 test patients, for upper baselines, our methods, and lower baselines.

■ 4.4 VoxelMorph Implementation

We use the TensorFlow [1] and Keras [6] frameworks to implement our deep convolutional neural network. We use the NEURON package [20] to perform 3D spatial transformations. To train VoxelMorph, we use the ADAM optimizer with a learning rate of 10^{-5} for MI and $5 * 10^{-5}$ for MIND. Code for VoxelMorph can be found at voxelmorph.csail.mit.edu.

■ 4.5 Results

To evaluate a method for image registration on a single patient, we compute the Dice score of the resulting segmentation produced by a trained model, averaged over the 30 chosen brain structures. We present the results of each of our methods on 50 test patients in Figure 4.2. Our methods achieve significantly worse Dice score than our upper baselines, by more than 0.06 Dice. However, our best methods do much better than both of our baselines, with an increase of more than 0.03 Dice. Our methods also have a lower standard deviation than the lower baselines. We note that using a patient scan as the atlas reduced Dice slightly between our two upper baselines, so acquiring a segmented T2 atlas would likely improve the Dice scores for our methods.

Within our methods, VoxelMorph trained with local MI performs the best, though VoxelMorph trained with MIND does almost as well. Voxelmorph trained with global MI performs 0.01 Dice worse than VoxelMorph with local MI, which is as expected because global MI isn't able to use spatial information about the voxels in the image.

For single-modal registration, the baselines ANTs and NiftyReg take tens of minutes to a few hours to perform the registration, but for multi-modal registration both baselines run in just a few minutes. This suggests that the deformation fields computed are not much different from zero. Changing MI parameters and convergence parameters

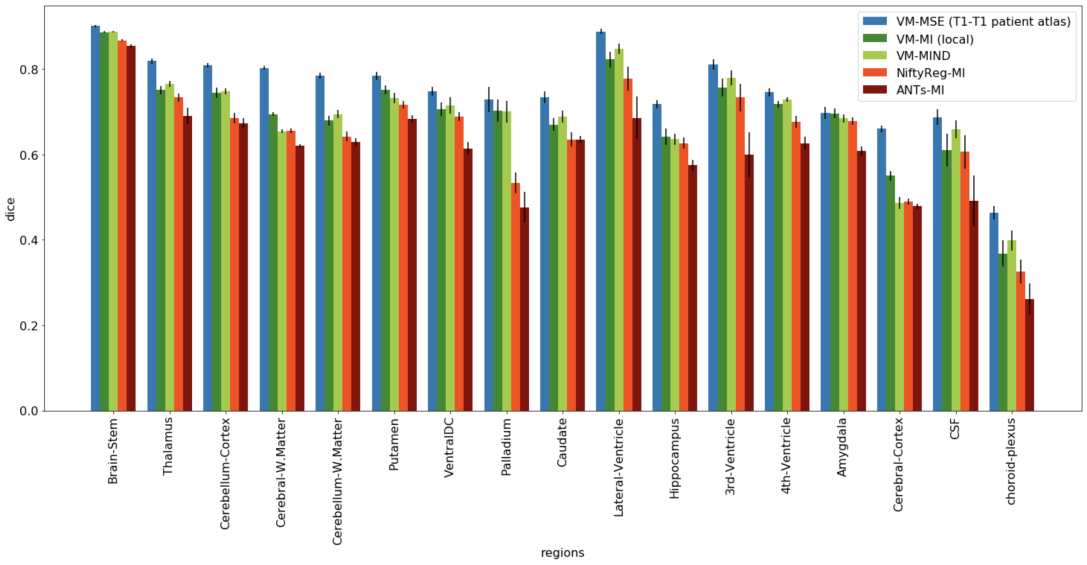

Figure 4.3. Dice score per brain region averaged over 10 validation patients, for upper baselines, our methods, and lower baselines.

did not improve the result. Since these baselines are widely cited, we can only conclude that multi-modal registration is a hard problem.

We take a look at how the Dice score is broken down by brain region. Figure 4.3 shows the Dice score per brain region averaged over 10 validation patients, for five methods: an upper baseline, VM-MI (local), VM-MIND, and both lower baselines. There is variation between brain regions, with the ventricles and brain stem being the easiest regions to segment (at around 0.9 Dice for our methods), and the choroid plexus being the hardest region to segment (at less than 0.4 Dice for our methods). However, this variation is exhibited in the same manner across all five methods: within almost every brain region, we have the same hierarchy of the upper baseline having a higher Dice than our methods which in turn have a higher Dice than the lower baselines.

In addition to achieving a better Dice score, the learning-based models also perform faster than the ANTs and NiftyReg baselines. On a CPU, VoxelMorph-based models take an average of 57 seconds to register a pair of images, compared to 155 seconds for ANTs and 99 seconds for NiftyReg.

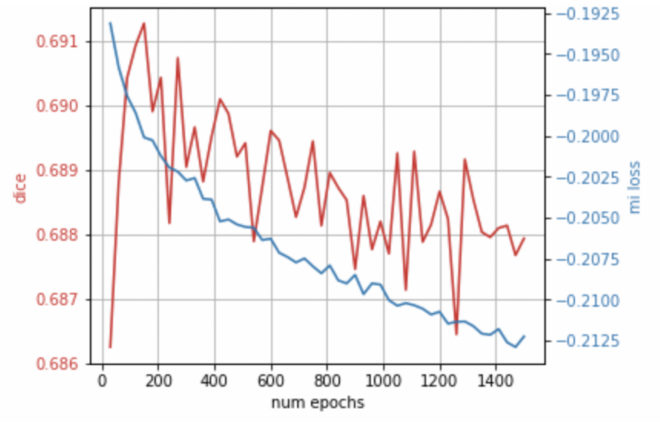

Figure 4.4. Dice score and MI loss on 10 validation subjects, as the model trains for 1500 epochs.

■ 4.6 Parameter Search

Each of the VoxelMorph models trained with multi-modal loss functions has parameters that need to be tuned. Each model needs a tuned regularization parameter that determines the weight of $\mathcal{L}_{smooth}$, the smoothness of the deformation field, in the loss function. There are also other parameters such as patch size that are specific to each loss function.

■ 4.6.1 Local MI

While training the local MI model, we observed signs of overfitting. Figure 4.4 shows the Dice score and the local MI loss on 10 validation subjects as the model trains for 1500 epochs, where an epoch is defined as 100 iterations of gradient descent. The model (as with all the models shown in this section) is initialized with weights from a model trained with global MI, to reduce the number of epochs needed to train the model. From the graph, we see that the validation Dice score increases in the first 150 epochs but steadily decreases afterwards, for a total decrease of 0.003 Dice. Though the Dice score decreases, the validation MI loss also decreases, meaning that the model is overfitting to MI loss and that MI loss may not be the best proxy for Dice loss. In order to train an optimal model, we enforce early stopping at 200 epochs.

To tune the regularization parameter (λ), we train each model with a different λ for 200 epochs, and plot the Dice score. Figure 4.5 shows the validation Dice on 50 subjects for different values of λ. A regularization parameter of 0 yields a very low Dice score

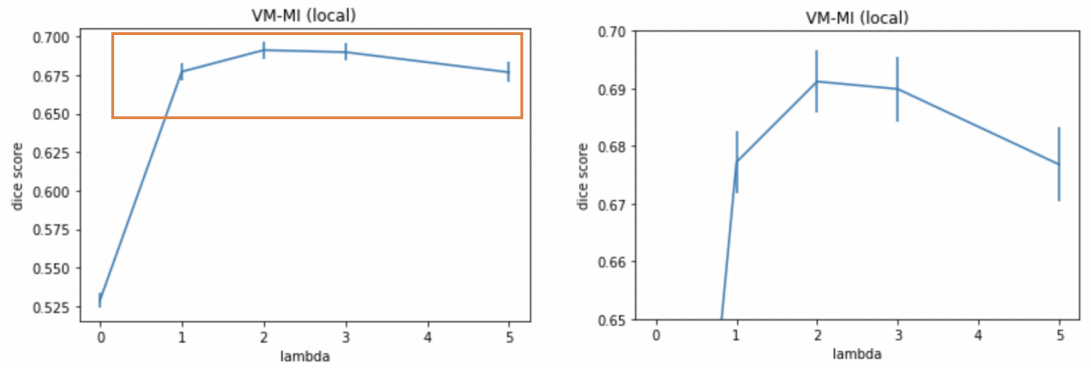

Figure 4.5. Validation Dice on 50 subjects after 200 epochs, for different values of the regularization parameter. The graph on the right is zoomed in from the graph on the left.

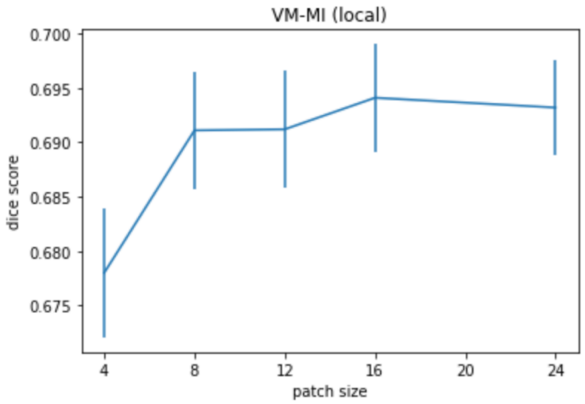

Figure 4.6. Validation Dice on 50 subjects after 200 epochs, for different patch sizes.

of 0.53. Zooming in to ignore $\lambda = 0$, we see that the model is very sensitive to the λ parameter. For example, doubling the parameter from 1 to 2 results in more than 0.01 Dice increase, and roughly doubling it again results in more than a 0.01 Dice decrease. We see that $\lambda = 2$ yields the best model.

After tuning the regularization parameter, in Figure 4.6 we explore the effect of the patch size used in the computation of local MI on Dice score. Again, each model is trained for 200 epochs and evaluated on 50 validation subjects. A patch size of 4 performs significantly worse than larger patch sizes, but patch sizes 8 through 24 have comparable results.

Local MI is also parameterized by the number of intensity bins used as well as the σ

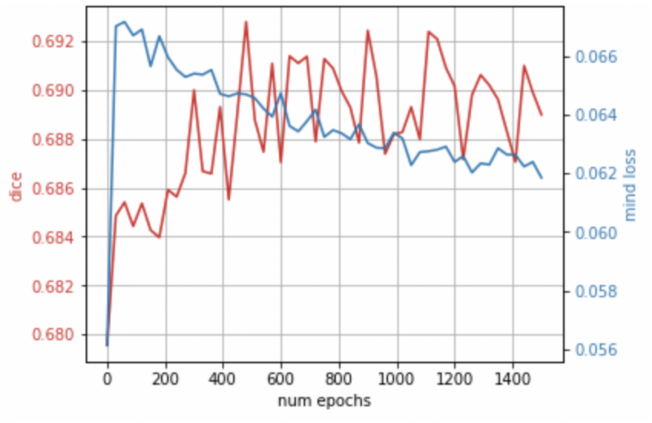

Figure 4.7. Dice score and MIND loss on 10 validation subjects, as the model trains for 1500 epochs.

used in the Gaussian function in Parzen windowing. We used 32 intensity bins, which was the largest number of bins possible due to GPU memory constraints, and a σ equal to half of each bin size. Since there are so many parameters, we could not do a thorough grid search across all of them since each combination of parameters takes at least half a day to train. Our final results for VoxelMorph trained with local MI use a patch size of 12, 32 intensity bins, a regularization parameter of 2, early stopping at 200 epochs, and a learning rate of 1e-5.

4.6.2 MIND

To train VoxelMorph with a MIND-based loss function, we plot validation Dice and MIND loss as a function of number of epochs trained. As with VM-MI, we initialize the model weights with those from a pretrained model, and use the same initialization for every VM-MIND model we train. Figure 4.7 shows the Dice score and MIND loss on 10 validation subjects, as the model trains for 1500 epochs. We see a steady rise in Dice score for the first 600 epochs, which then plateaus. It is interesting to note that VM-MIND does not overfit, whereas VM-MI (local) does. Since the model does not improve after a certain number of epochs, we will train every VM-MIND model for only 700 epochs.

To tune the regularization parameter, we train each model with different values of λ for 700 epochs, and plot the resulting Dice scores. Figure 4.8 shows the validation Dice on 50 subjects for different values of λ. As with VM-MI, when $\lambda = 0$ we get a very low Dice. We zoom in to a Dice score range of 0.65-0.70, and see that the model is also

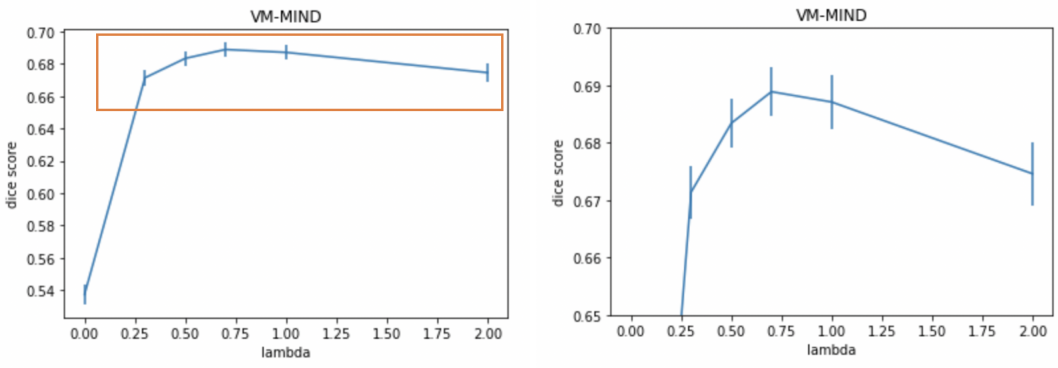

Figure 4.8. Validation Dice on 50 subjects after 700 epochs, for different values of the regularization parameter. The graph on the right is zoomed in from the graph on the left.

fairly sensitive to λ, with factors of 2 in λ resulting in changes of more than 0.01 Dice.

MIND is parametrized by patch size as well as patch distance. We used a patch size of 3 and patch distance of 2, which are parameters suggested by Heinrich *et al.* for a similar method [19]. As with VM-MI, since there are so many different parameters we could not do a thorough grid search of all them. Our final results for VoxelMorph trained with MIND use a patch size of 3, a patch distance of 2, a regularization parameter of 0.7, stopping at 700 epochs, and a learning rate of 5e-5.

Chapter 5

Discussion

We have successfully extended VoxelMorph, a learning-based method for image registration, to tackle the problem of multi-modal image registration. VoxelMorph trained with MI-based and MIND-based loss functions perform significantly better than the ANTs and NiftyReg baselines for multi-modal registration, where performance is measured by Dice score. This is a stronger result than VoxelMorph for single-modal registration, which performed comparably to baselines [3].

Part of the reason that our methods perform much better than the baselines is that both baselines do poorly on a multi-modal dataset, at a decrease of 0.10 Dice from single-modal to multi-modal. This was quite surprising because both packages are widely used, suggesting that multi-modal registration is a difficult task. Our learning-based methods also suffered a decrease of Dice (around 0.06) when applied on a single-modal task compared to a multi-modal task, but this decrease was much less than the 0.10 exhibited in the baselines.

While we only looked at a T1-T2 dataset, our method does not depend on those two modalities and can be used in general for any multi-modal dataset. Since our method does not depend on the structure of the brain, it can potentially be used for other medical imaging applications as well. We investigated a classical metric (MI) as well a newer metric (MIND) that both have been used in traditional methods for multi-modal registration, but we welcome further exploration of loss functions that VoxelMorph can be used with, to improve multi-modal image registration.

Bibliography

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, and M. Isard. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[2] Ruzena Bajcsy. Multiresolution elastic matching. *Computer Vision, Graphics, and Image Processing*, 45(1):132, 1989. doi: 10.1016/0734-189x(89)90082-0.

[3] G. Balakrishnan, A. Zhao, M. R. Sabuncu, J. Guttag, and A. V. Dalca. An unsupervised learning model for deformable medical image registration. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern*, pages 9252–9260, 2018.

[4] Guha Balakrishnan, Amy Zhao, Mert R. Sabuncu, John Guttag, and Adrian V. Dalca. Voxelmorph: A learning framework for deformable medical image registration. *IEEE Transactions on Medical Imaging*, page 11, 2019. doi: 10.1109/tmi.2019.2897538.

[5] Xiaohuan Cao, Jianhua Yang, Jun Zhang, Dong Nie, Minjeong Kim, Qian Wang, and Dinggang Shen. Deformable image registration based on similarity-steered cnn regression. *Medical Image Computing and Computer Assisted Intervention MICCAI 2017 Lecture Notes in Computer Science*, page 300308, 2017. doi: 10.1007/978-3-319-66182-7_35.

[6] F. Chollet. Keras, May 2019. URL https://github.com/fchollet/keras.

[7] T. M. Cover and Joy A. Thomas. *Elements of information theory*. Wiley-Interscience, 2006.

[8] A. Dagley, M. LaPoint, W. Huijbers, T. Hedden, D. G. McLaren, J. P. Chatwal, K. V. Papp, R. E. Amariglio, D. Blacker, and D. M. Rentz. Harvard Aging Brain Study: dataset and accessibility. *NeuroImage*, 2015.

[9] A. V. Dalca, J. Guttag, and M. R. Sabuncu. Anatomical Priors in Convolutional Networks for Unsupervised Biomedical Segmentation. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern*, pages 9290–9299, 2018.

[10] Adrian V. Dalca, Andreea Bobu, Natalia S. Rost, and Polina Golland. Patch-based discrete registration of clinical brain images. *Patch-Based Techniques in Medical Imaging Lecture Notes in Computer Science*, page 6067, 2016. doi: 10.1007/978-3-319-47118-1_8.

[11] Adrian V. Dalca, Guha Balakrishnan, John Guttag, and Mert R. Sabuncu. Unsupervised learning for fast probabilistic diffeomorphic registration. *Medical Image Computing and Computer Assisted Intervention MICCAI 2018 Lecture Notes in Computer Science*, page 729738, 2018. doi: 10.1007/978-3-030-00928-1_82.

[12] Christos Davatzikos. Spatial transformation and registration of brain images using elastically deformable models. *Computer Vision and Image Understanding*, 66(2): 207222, 1997. doi: 10.1006/cviu.1997.0605.

[13] L. R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.

[14] B. Fischl. Freesurfer. *NeuroImage*, 62(2):774–781, 2012.

[15] Ben Glocker, Nikos Komodakis, Georgios Tziritas, Nassir Navab, and Nikos Paragios. Dense image registration through mrfs and efficient linear programming. *Medical Image Analysis*, 12(6):731741, 2008. doi: 10.1016/j.media.2008.03.006.

[16] R. L. Gollub, J. M. Shoemaker, M. D. King, T. White, S. Ehrlich, S. R. Sponheim, V. P. Clark, J. A. Turner, B. A. Mueller, and V. Magnotta. The MCIC Collection: a shared repository of multi-modal, multi-site brain image data from a clinical investigation of schizophrenia. *Neuroinformatics*, 11(3):367–388, 2013.

[17] Joseph Hajnal, D. J. Hawkes, and Derek Hill. *Medical image registration*. CRC Press, 2001.

[18] M. P. Heinrich, M. Jenkinson, M. Bhushan, T. Matin, F. V. Gleeson, S. M. Brady, and J. A. Schnabel. MIND: Modality independent neighbourhood descriptor for multi-modal deformable registration. *Med. Image Anal.*, 16(7):1423–1435, 2012.

[19] M. P. Heinrich, M. Jenkinson, Papiez B. W, S. M. Brady, and J. A. Schnabel. Towards Realtime Multimodal Fusion for Image-Guided Interventions Using Self-similarities. *Medical Image Computing and Computer Assisted Intervention*, LNCS (8149):187–194, 2013.

[20] Michael Hines. Neuron and python. *Frontiers in Neuroinformatics*, 3, 2009. doi: 10.3389/neuro.11.001.2009.

[21] A. J. Holmes, M. O. Hollinshead, T. M. OKeefe, V. I. Petrov, G. R. Fariello, L. L. Wald, B. Fischl, B. R. Rosen, R. W. Mair, and J. L. Roffman. Brain Genomics Superstruct Project initial data release with structural, functional, and behavioral measures. *Scientific Data*, 2, 2015.

[22] J.b.antoine Maintz and Max A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1):136, 1998. doi: 10.1016/s1361-8415(01)80026-8.

[23] D. S. Marcus, T. H. Wang, J. Parker, J. G. Csernansky, J. C. Morris, and R. L. 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(9):1498–1507, 2007.

[24] K. Marek, D. Jennings, S. Lasch, A. Siderowf, C. Tanner, T. Simuni, C. Coffey, K. Kieburtz, E. Flagg, and S. Chowdhury. The Parkinson Progression Marker Initiative (PPMI). *Progress in Neurobiology*, 95(4):629–635, 2011.

[25] A. Di Martino, C.-G. Yan, Q. Li, E. Denio, F. X. Castellanos, K. Alaerts, J. S. Anderson, M. Assaf, S. Y. Bookheimer, and M. Dapretto. The Autism Brain Imaging Data Exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular Psychiatry*, 19(6):659–667, 2014.

[26] M. P. Milham, M. Mennes D. Fair, and S. H. Mostofsky. The ADHD-200 Consortium: a model to advance the translational potential of neuroimaging in clinical neuroscience. *Frontiers in Systems Neuroscience*, 6:62, 2012.

[27] S. G. Mueller, M. W. Weiner, L. J. Thal, R. C. Petersen, C. R. Jack, W. Jagust, J. Q. Trojanowski, A. W. Toga, and L. Beckett. Ways toward an early diagnosis in Alzheimers disease: the Alzheimers Disease Neuroimaging Initiative (ADNI). *Alzheimer's & Dementia*, 1(1):55–66, 2005.

[28] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. Mutual-information-based registration of medical images: A survey. *IEEE Trans. Med. Imag.*, 22(8):986–1004, 2003.

[29] Marc-Michel Roh, Manasi Datar, Tobias Heimann, Maxime Sermesant, and Xavier Pennec. Svf-net: Learning deformable image registration using shape matching. *Medical Image Computing and Computer Assisted Intervention MICCAI 2017 Lecture Notes in Computer Science*, page 266274, 2017. doi: 10.1007/978-3-319-66182-7_31.

[30] D. Rueckert, L.i. Sonoda, C. Hayes, D.l.g. Hill, M.o. Leach, and D.j. Hawkes. Nonrigid registration using free-form deformations: application to breast mr images. *IEEE Transactions on Medical Imaging*, 18(8):712721, 1999. doi: 10.1109/42.796284.

[31] Dinggang Shen and C. Davatzikos. Hammer: hierarchical attribute matching mechanism for elastic registration. *Proceedings IEEE Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA 2001)*. doi: 10.1109/mmbia.2001.991696.

[32] Hessam Sokooti, Bob De Vos, Floris Berendsen, Boudewijn P. F. Lelieveldt, Ivana Igum, and Marius Staring. Nonrigid image registration using multi-scale 3d convolutional neural networks. *Medical Image Computing and Computer Assisted Intervention MICCAI 2017 Lecture Notes in Computer Science*, page 232239, 2017. doi: 10.1007/978-3-319-66182-7_27.

[33] Igor Vajda. Theory of statistical inference and information. 01 1989.

[34] P. Viola and W.M. Wells. Alignment by Maximization of Mutual Information. *nternational Journal of Computer Vision*, 24:137–154, 1997.