ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Diffeomorphic Modeling of the Brain Aging Process

Master Thesis

Patrice Marti

Monday 27th May, 2019

Supervisor: Prof. Dr. J. Buhmann
Advisor: V. Wegmayr

Department of Computer Science, ETH Zürich

Abstract

Contents

Chapter 1

Introduction

The past decades have seen medical research advancing at a rapid rate, resulting in a dramatic increase of the average human life expectancy [22]. As a consequence, the number of cases of aging-related neurodegenerative diseases such as Alzheimer's Disease has increased significantly and is expected to keep rising, reaching over 100 million Alzheimer's cases by 2050 [5]. For this reason, advancing the understanding of the brain aging process as well as early prediction and treatment methods of degenerative diseases have attracted considerable research efforts. In our work, we propose a generative model to simulate the aging process on T1-weighted MRI brain scans.

Generative models, most prominently Variational Autoencoders (VAEs) [18] and Generative Adversarial Networks (GANs) [10], have been successfully applied to model gradual changes in image data for wide range of settings such as face aging [20], image registration [2] and style transfer [28]. The goal of this thesis is to apply some of these methods to the problem of brain aging. Specifically, we consider a model which given a T1-weighted MRI image x taken at time t_0 aims to predict an image y at some time t_1 in the future. If sufficiently accurate, such a model could pave the way for a number of applications. For instance, existing diagnostic tools which operate on MRI data such as diagnosis classifiers can be applied directly to the generated image, therefore leveraging decades of research in the field. Furthermore, insights into the aging process and the effects of degenerative diseases can be gained by aggregating the model's outputs over different groups of subjects.

Validating generative model outputs beyond subjective visual inspection is a notoriously difficult task. Meaningful metric

In the field of medical imaging, diffeomorphic deformations are popularly used to model biological processes [4] [?]. Unlike entirely convolutional

models, diffeomorphisms are constrained to transformations which are differentiable and invertible and therefore topology preserving, thus generally resulting in a more realistic representation while also producing more interpretable results. Following [6], we model the brain aging process as a diffeomorphic deformation field obtained by numerically integrating a stationary velocity field using the *scaling and squaring* method [**?**]. We furter propose an extending the method to yield deformations for arbitrary time steps, allowing our brain aging model to generate and be trained on image pairs with arbitrary time differences.

Our main contributions are:

- we model the brain aging process using diffeomorphic deformations
- we adapt the scaling and squaring method for arbitrary timesteps
- we validate our model's ability to predict follow-up images
- we use our model to predict the Alzheimer's Disease conversion probability of patients with Mild Cognitive Impairment

Chapter 2

Generative Diffeomorphic Deformation Models

Generative models have been successfully applied to a wide range of medical image analysis tasks such as image registration [2], segmentation [8] and visual feature attribution [3]. Of particular interest are deformation-based models due to their ability to closely model the gradual changes occuring in the context biological processes. Additionally, by using *diffeomorphic* deformations, the model can be limited to operation which are smooth, differentible and invertible. Diffeomorphisms are topology preserving which further improves the model's realism while generally also resulting in more interpretable outputs.

In this section, we discuss the general architecture of our model. We first examine the diffeomorphic brain registration model proposed in [2] [6] followed by a discussion of our adaptations for the generative brain aging task.

2.1 Diffeomorphic Image Registration

In medical imaging, deformable image registration tackles the problem of warping one image onto another. More formally, given two scans x and y, the aim is to find a deformation function Φ such that $x \circ \Phi$ is similar to y.

2.1.1 Voxelmorph

Dalca et al [6] propose a deep learning architecture to learn such a mapping for 3-dimensional MRI brain data. Formally, given x and y the model generates a stationary velocity field v which defines the deformation $\Phi : \mathbb{R}^3 \to \mathbb{R}^3$ mapping x to y through the ordinary differential equation (ODE)

$$\frac{\partial \Phi^{(t)}}{\partial t} = v(\Phi^{(t)}) \tag{2.1}$$

3

where $\Phi^{(0)} = id$ is the identity transformation and t is time. The final deformation field $\Phi^{(1)}$ is then obtained by integrating the field v over time $t = [0, 1]$, which is computed numerically using the scaling and squaring method.

In group theory, v is a member of the Lie algebra and is exponentiated to produce $\Phi^{(1)} = \exp(v)$. The collection $\{\Phi^{(t)}\}_{t \in [0,1]}$ forms a one-parameter subgroup of diffeomorphisms and therefore for any scalars t and t' we have

$$\exp((t + t')v) = \exp(tv) \circ \exp(t'v) \tag{2.2}$$

where $\circ$ is a composition map associated with the Lie group. Consequently, we can then use the recurrence

$$\Phi^{(1/2^{(t-1)})} = \Phi^{(1/2^t)} \circ \Phi^{(1/2^t)} \tag{2.3}$$

starting from $\Phi^{(1/2^T)}$ to obtain $\Phi^{(1)} = \Phi^{(1/2)} \circ \Phi^{(1/2)}$ where T is chosen such that $v \approx 0$.

The model uses a variational inference method to generate a stationary displacement field z which defines the deformation Φ_z through the ODE (2.1). The prior probabilty of z is modeled as

$$p(z) = \mathcal{N}(z; 0, \Sigma_z) \tag{2.4}$$

Spatial smoothness of z is is encouraged by letting $\Sigma_z^{-1} = \Lambda_z = \lambda L$ where Λ_z is a precision matrix, L is the Laplacian of a neighborhood graph defined as $L = D - A$, with graph degree matrix D and voxel adjacency matrix A, and λ denotes a parameter controlling the scale of the velocity field.

The target image y is interpreted as a noisy observation of the warped image x

$$p(y|z; x) = \mathcal{N}(y; x \circ \Phi_z, \sigma^2 \mathbb{I}) \tag{2.5}$$

with σ^2 reflecting the variance of the additive noise.

A likely registration field Φ_z can then obtained by sampling z from the posterior distribution $p(z|x; y)$. However, computing this distribution is intractable in this setting and hence a variational approach is used where z is sampled from an approximate posterior probability $q_\psi(z|x; y)$ parametrized by ψ. The distribution is modeled as a multivariate normal

$$q_\psi(z|x; y) = \mathcal{N}(z; \mu_{z|x,y}, \Sigma_{z|x,y}) \tag{2.6}$$

and approximated by minimizing the KL divergence

$$\min_\psi KL[q_\psi(z|x; y) || p(z|x; y)]$$
$$= \min_\psi KL[q_\psi(z|x; y) || p(z)] - \mathbb{E}_q[\log p(y|z; x)] \tag{2.7}$$

The complete loss function can be separated into three terms denoted as follows

$$\mathcal{L}(\psi; \mathbf{x}, \mathbf{y}) = -\mathbb{E}_q[\log p(\mathbf{x}|\mathbf{z}; \mathbf{y})] + \text{KL}[q_\psi(\mathbf{z}|\mathbf{x}; \mathbf{y})||p(\mathbf{z})]$$

$$= \underbrace{\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{x} \circ \Phi_z\|^2}_{\text{reconstruction term}} \tag{2.8}$$

$$+ \frac{1}{2}\left[\underbrace{tr(\lambda\mathbf{D}\Sigma_{z|x;y} - \log|\Sigma_{z|x;y}|)}_{\text{sigma term}} + \underbrace{\mu_{z|x;y}^T\Lambda_z\mu_{z|x;y}}_{\text{precision term}}\right]$$

The first term enforces similarity between the target image y and the warped source image $x \circ \Phi_z$, the second term encourages the posterior to be close to the prior $p(z)$ while the third term spatially smoothes the mean $\mu_{z|x,y}$. This effect can be shown more explicitly by rewriting the precision term as $\frac{\lambda}{2}\sum\sum_{j\in N(I)}(\mu[i] - \mu[j])^2$, where $N(i)$ denotes the set of neighbors of voxel i. Both σ and λ are treated as hyperparameters, respectively controlling the reconstruction penalty and the magnitude of the velocity field.

Network Architecture

The paramaters $\mu_{z|x,y}$ and $\Sigma_{z|x,y}$ are estimated by a convolutional neural network (CNN). The architecture, which takes x and y as input, is based on a fully convolutional 3D UNet consisting of a convolutional layer of 16 filters followed by four downsampling layers with strides of two and three up-sampling layers of 32 filters each. All convolutional layers use leaky ReLU activations with $\alpha = 0.2$ and kernels of size $3 \times 3 \times 3$. See Figure 2.1 for an illustration of the generator model.

Given $\mu_{z|x,y}$ and $\Sigma_{z|x,y}$, the subsequent layer then samples a new stationary velocity field $z_k \sim \mathcal{N}(\mu_{z|x,y}, \Sigma_{z|x,y})$ using the reparameterization trick [18], which is then integrated using newly introduced scaling and squaring layers to compute $\Phi_{z_k} = \exp(z_k)$. Specifically, one such layer performs a differentiable vector field composition, that is, given vector fields a and b, it computes $(a \circ b)(p) = a(b(p))$ for each voxel p. Note that linear interpolation is used in a as $b(p)$ generally yields a non-integer location. The recurrence in Equation 2.3 is implemented using $T = 7$ of these layers. Finally, a spatial transform layer applies the deformation field Φ_{z_k} to the source image x to obtain $x \circ \Phi_{z_k}$.

The network is implemented in Keras with a Tensorflow backend and trained end-to-end using the Adam [17] optimizer.
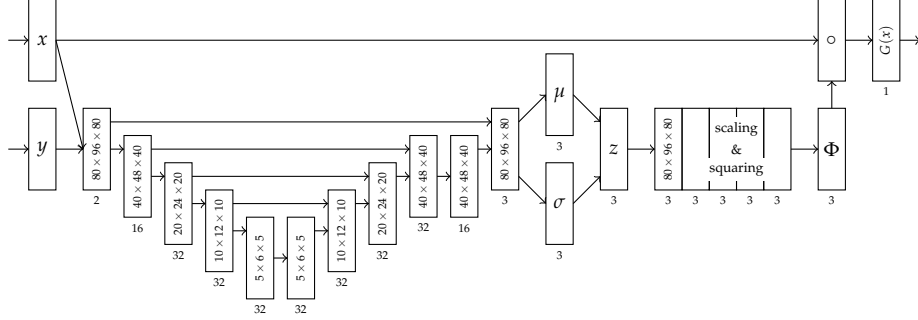
Figure 2.1: Voxelmorph for diffeomorphic image registration as proposed in [6]. The UNet-based encoder receives both x and y as inputs and approximates the distributon q_ψ from which the velocity field z is sampled. Subsequently, z is integrated using a configurable number of scaling and squaring layers resulting in the final deformation field $\Phi^{(1)}$ which is then applied to the input image x.

2.2 Adaptation for Brain Aging

While the tasks of brain registration and generative brain aging may not appear to have much in common at first, both can be described in terms of learning a deformation function. As such, the approach used in [2] can presumably be adapted to suit brain aging as well. However, while there are simililarites, a number of key differences in the problem settings require modifications to the model design.

Most importantly, the brain registration task as defined in [2] and described above is an unsupervised learning problem where both the source image x and the target image y are available in the prediction step. Conversely, since the goal of the brain aging task is to predict the future state of x, the aged target image y is only available in training and therefore cannot be a part of the model's input.

Furthermore, the learned deformations for the brain aging task can be expected to be much smaller in scale and therefore, the noise introduced as part of the reconstruction term in Equation 2.8 may have a negative effect on the model performance.

Finally, while intermediate deformations $\Phi^{(t)}$ for time steps $t \notin \{0, 1\}$ are not of primary interest in the brain registration task, the ability to predict a brain image $G(x) = x \circ \Phi_z^{(t)}$ for arbitrary t promises valuable insights into the progression of neurodegenerative diseases as well as the brain's aging process in general. Furthermore, the ability to train on image pairs over a large range of different time steps is also beneficial as the number of image pairs for any particular fixed t is very limited. Moreover, training on

a continuous range of time steps as opposed to a limited number of fixed intervals should result in improved generalization.

2.2.1 Adversarial Loss

As described above, the model input is restricted to the source image x and, without access to y, predicting differences between the source x and target y that are not related to aging, such as artifacts introduced during scaning or preprocessing (e.g. skull remnants or misalignment), is virtually impossible. As a consequence, our loss function should be invariant to such changes, yet this is not the case for the reconstruction term. Moreover, the term introduces image noise which can be problematic given the small scale of aging related changes.

Therefore, we opt to replace the reconstruction loss term in Equation 2.8 with an adversarial loss component. We realize this by adding a secondary critic network to the architecture which is trained alongside the generator in an adversarial fashion. Effectively, this tranforms the model into a Generative Adversarial Network (GAN) [10].

In the adversarial setting, a generative model G and a discriminative model D are engaged in a minimax game, in which the generator aims to produce outputs that to the discriminator are indistinguishable from samples drawn from a real data distribution p_{data}. More formally, a GAN optimizes the objective

$$\min_G \max_D V(G,D) = \mathbb{E}_{x \sim p_{data}(x)} \big[\, D(x) \, \big] - \mathbb{E}_{z \sim p_z(z)} \big[\, 1 - D(G(z))) \, \big] \qquad (2.9)$$

where $D(x)$ is a probability and z is usually sampled from a latent distribution. However, in the brain aging setting the goal is to transform a source image x in a way that resembles the actual aging process and therefore we get the revised objective

$$\min_G \max_D V(G,D) = \mathbb{E}_{(x,y,t) \sim p_{data}} \big[\, D(x,y,t) \, \big]$$
$$- \mathbb{E}_{(x,t) \sim p_{data}} \big[\, 1 - D(x, G(x), t)) \, \big] \qquad (2.10)$$

where the image pair (x,y) and the corresponding age difference t are samples from the real data distribution. Note that in order to avoid the issue of mode collapse, where the generator outputs the same image for all inputs, the discriminator also observes x. Furthermore, to enable the discriminator to discern pairs with differing time steps, we additionally pass t as an input. Both G and D are implemented as neural networks which are trained in an alternating fashion.

We use a variation of the original GAN known as Wasserstein GAN (WGAN) [1] in which the discriminator D is replaced by a critic with real-valued outputs instead of probabilities. The critic is limited to the set of 1-Lipschitz functions, which is enforced by imposing a gradient penalty as proposed in [11].

2.2.2 Arbitrary Time Step Training and Prediction

The scaling and squaring method as described in section 2.1 is fixed to one specific time step t determined by the model configuration as well as the training data. As mentioned above, this is not necessarily an issue in the case of image registration but highly undesirable for the brain aging task. Therefore, in this section we propose an extension to the scaling and squaring method enabling integration of the stationary velocity field v to arbitrary time steps t.

One straightforward approach is to abandon the scaling and squaring method in favor of iterative composition

$$\Phi^{(t)} = \underbrace{\Phi^{(1/2^T)} \circ \ldots \circ \Phi^{(1/2^T)}}_{\lceil 2^T \times t \rceil \text{ times}} \tag{2.11}$$

where 2^T is the scaling factor and t is the desired time step. Given a large enough T, this method can handle any positive time step with arbitrary precision, however very quickly at the cost of computional unfeasibility. Similarly, we could use a two step approach, calculating the deformation $\Phi^{(\varepsilon)}$ for some time step ε by scaling and squaring, followed by iterative composition of $\Phi^{(\varepsilon)}$. While this is much faster in practice, the choice of ε represents a trade-off between precision, data availability and computational viability.

In addition to the final deformation field $\Phi^{(1)}$, the recurrence also yields intermediate deformations $\{\Phi^{(1/2^t)}\}_{t \in 1..T}$ at no additional computational cost. For instance, the computation of a deformation field corresponding to a time step of 8 years additionally yields the deformations for (and therefore the ability to predict and train on) time steps of $4, 2, 1, 0.5, \ldots$ years. While this represents an improvement, the benefits are relatively minor as we are still limited to a small and very specific set of time steps.

However, from the properties of one-parameter subgroups in Equation 2.2 we know that any two given deformations $\Phi^{(t)}$ and $\Phi^{(t')}$ can be composed to obtain $\Phi^{(t+t')} = \Phi^{(t)} \circ \Phi^{(t')}$. It follows that for any time step $t \in [0, 1)$, the corresponding deformation $\Phi^{(t)}$ can be represented as a composition of deformations from a subset $\mathcal{S}^{(t)} \subset \{\Phi^{(1/2^t)}\}_{t \in 1..T}$ of intermediate deformations

$$\Phi^{(t)} = \bigcirc_{\Phi^{(i)} \in \mathcal{S}^{(t)}} \Phi^{(i)} \tag{2.12}$$

In other words, $\{\Phi^{(1/2^s)}\}_{s \in 1..T}$ can be interpreted as a set of vectors that span the space of all deformations $\Phi^{(t)}$ for $t \in [0, 1)$, where each $\Phi^{(t)}$ is uniquely represented by a binary vector in this space. Intuitively speaking, this is analogous to how any positive integer can be expressed in $base_2$ as the sum over a set of powers of 2. The deformation is computed iteratively over all squaring step as laid out in algorithm 1. Refer to Figure 2.2 for a visual example of one such composition.

The temporal precision ε, i.e. the smallest difference in time steps representable by the model, is determined by the number of squaring steps T as well as the maximum time step t_{max} used during training. Specifically, ε is the time step corresponding to the smallest deformation field $\Phi^{(1/2^T)} = v/2^T$ and therefore $\varepsilon = t_{max}/2^T$. For instance, given $t_{max} = 6$ years and $T = 7$, $\varepsilon = 0.046$ years or approximately 17 days.

In practice, the efficiency of the computation can be improved by computing only the intermediate deformations up to the largest step required for the composition of $\Phi^{(t)}$. Note also that predictions for time steps $t > 1$ can be generated by dynamically increasing the number of squaring layers during inference.

Algorithm 1: Scaling and Squaring for arbitrary time step

---

input : v velocity field
          t time step $\in [0, 1)$
          T number of squaring steps
output: d,   deformation field

$bits \leftarrow \texttt{floor}\ (t\ \texttt{<<}\ T)$
$d \leftarrow 0$
$v \leftarrow v\ /\ 2^T$

for bit **in** $bits$ **do**
   if $bit = 1$ **then**
       $d \leftarrow d + \texttt{transform}\ (v, d)$
   end
    $v \leftarrow v + \texttt{transform}\ (v, v)$
end

2.2.3 Additional Loss Terms

In addition to the adversarial loss we also examine four additional loss terms and their effects on the model performance.
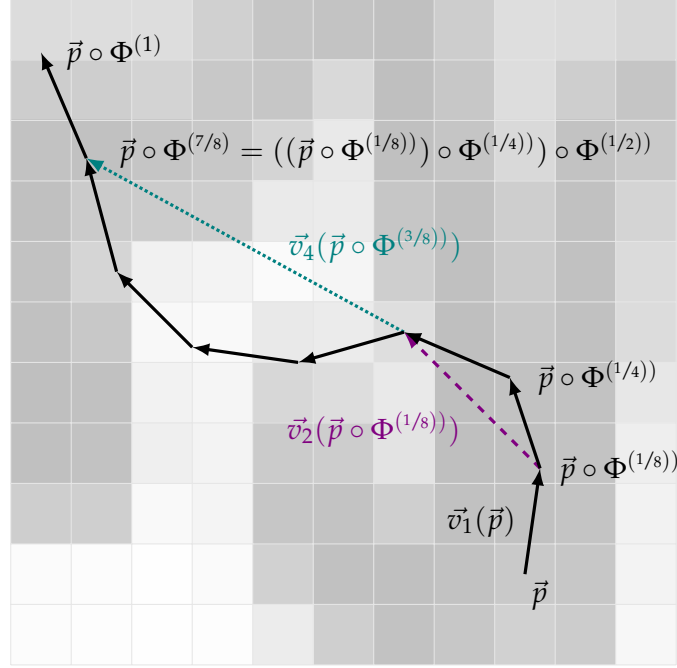
Figure 2.2: Arbitrary time step scaling and squaring with $T = 2$ squaring and $2^{T+1} = 8$ atomic steps, shown for one voxel $\vec{p}$. The deformation $\Phi^{(t)}$ can be approximated at any time step $t \in [0, 1]$ by composing a subset of intermediate deformations. Note that in practice, larger T are used resulting in an exponentially higher number of atomic steps and therefore a better approximation, e.g. $T = 7$ yielding $2^8 = 256$ atomic steps. We calculate the deformation for all voxels $\vec{p}$ in parallel.

Age Regressor

To encourage our model to generate realistically aged $G(x) = x \circ \Phi^{(t)}$ with respect to time step t, we use a pre-trained age regressor R to estimate the apparent age of $G(x)$. Let a_x denote a patient's age at the time of taking image x and $\hat{a}_x = A(x)$ denote the age as estimated by the age regressor on x. As a side note, for the generator we generally assume $t \in [0, 1]$ normalized by $\max_{(x,y) \in \mathcal{D}_{train}} a_y - a_x$, the maximum time step occuring in the training data, and therefore $t_{(x,y)} \neq a_y - a_x$ in general.

We consider two different possible loss terms

$$
\begin{aligned}
&(1) \quad \mathcal{L}_{age}(x, y, R) = |(a_y - a_x) - (\hat{a}_{G(x)} - a_x)| = |a_y - \hat{a}_{G(x)}| \\
&(2) \quad \mathcal{L}_{age}(x, y, R) = |(\hat{a}_y - \hat{a}_x) - (\hat{a}_{G(x)} - \hat{a}_x)| = |\hat{a}_y - \hat{a}_{G(x)}|
\end{aligned}
\tag{2.13}
$$

with (1) using ground truth labels whenever available and (2) using the regressor throughout. We hypothesize (2) to be superior due to inaccura-

cies of the age regressor cancelling out. This assumption is supported by our experimental results in section 5.1 and consequently, we use (2) for our model.

Diagnosis Classifier

Similar to the age regressor, we also add a loss term based on a diagnosis classifier C to encourage the model to understand and distinguish between different diagnoses. Let d_x denote the ground truth diagnosis label assigned to x (with 0 = MCI, 1 = AD) and $\hat{d}_x$ denote the classifier's estimated probability of the brain in image x being affected by AD. As before, we examine two possible cross entropy loss terms between ground truth labels and the estimated probabilities respectively and implement (2), following the same reasoning used in selecting the age regressor loss term.

$$H(p,q) = -p \log q - (1-p) \log(1-q)$$

$$(1) \quad \mathcal{L}_{dx}(x,y,C) = H(d_y, \hat{d}_{G(x)})$$

$$(2) \quad \mathcal{L}_{dx}(x,y,C) = H(\hat{d}_y, \hat{d}_{G(x)})$$

(2.14)

In addition to its use in this loss term, the classifier also serves as our baseline for the conversion prediction experiment as described in section 3.1.

Similarity Loss

Similar to [3] and [**?**], we introduce an additional loss term intended to enforce similarity and identity preservation between the original image x and the warped image $x \circ \Phi^{(t)}$ by imposing an L_1 loss on their difference

$$\mathcal{L}_{sim}(x,G) = \|x - G(x)\|_1 \tag{2.15}$$

Note that we choose not to scale the similarity loss with respect to time step t. This is based on the observation that while the L_1 difference does increase for larger t, as depicted in Figure 4.3, it does so rather slowly, indicating that the L_1 difference is primarily caused by differences not related to aging. As a side note, this also means that the L_1 metric is not very well suited to validate our generative model outputs.

Sparseness Loss

Finally, we encourage sparseness of the velocity field by imposing an L_1 loss on its magnitude

$$\mathcal{L}_{sparse}(x,G) = \|\mu_z\|_1 \tag{2.16}$$

Note that while this loss term acts as a regularizer, the primary motivation for sparseness is to improve the interpretability of the deformation field by discouraging displacements with very little or no effect at all.

Complete Objective

To summarize, we obtain the complete objective for the generator as follows

$$\mathcal{L}_G = \mathcal{L}_{ws} + \lambda_1\, \mathcal{L}_{kl} + \lambda_2\, \mathcal{L}_{age} + \lambda_3\, \mathcal{L}_{dx} + \lambda_4\, \mathcal{L}_{sim} + \lambda_5\, \mathcal{L}_{sparse} \qquad (2.17)$$

where $\mathcal{L}_{ws}$ is the generator's component of the Wasserstein loss function and $\mathcal{L}_{kl}$ consists of the sigma and precision terms from Equation 2.8. Furthermore, we treat $\{\lambda_i\}_{i \in 1\,..\,5}$ as hyperparameters.

2.2.4 Network Architecture

Based on architectures applied to similar problems such as [**?**] and [3], we expect the brain aging problem to be a more difficult task compared to brain registration as implemented in Voxelmorph. Therefore, we significantly increase the complexity of the UNet model as shown in Figure 2.3. Moreover, we experiment with a low resolution time-invariant deformation component, extracted from the UNet, to capture differences that are independent of time step t such as misalignments introduced during scanning or preprocessing. Separating the deformation into two components is desirable as it allows ignoring changes independent of time and therefore aging during inference on unseen data. However, in preliminary experiments, this component appears to overpower the time-dependent deformations resulting in poor model performance and, as a consequence, is not included in our final model architecture.

Figure 2.3: Overview of our generator and critic network architectures. The outputs of the squaring layers are applied selectively depending on the value of t. In the critic, depending on the step of the GAN training procedure, input y is either a real or a generated image.

Chapter 3

Applications

Our primary goal is to design a generative model G capable of learning and simulating the aging process of the brain. Given an input image x, we can then use the trained model G to generate a predicted future state of the brain $\hat{y}^{(t)} = G(x) = x \circ \Phi^{(t)}$ for any time step t.

Being able to generate realistic predictions might be beneficial in the early detection of Alzheimer's onset, in particular since existing diagnostic tools can be directly applied to predicted $\hat{y}$ without any necessary adaptations. Furthermore, the resulting deformation fields may yield insights into the progression and specific changes of neurodegenerative diseases.

3.1 Conversion Prediction

Early prediction of Alzheimer's Disease onset is an important area of Alzheimer's research, with one particular interest being the Mild Cognitive Impairment (MCI) conversion problem. Given data about a patient diagnosed with MCI at some visit v_i, our goal is to predict the probabilty of that patient's diagnosis converting to AD over a given period of time Δ. In this context, we distinguish between progressive cases (pMCI) for which the diagnosis converts within Δ, and stable cases (sMCI) for which it does not.

More specifically, a case is considered *progressive* if there exists a pair of visits (v_a, v_b) at times (t_a, t_b) with $t_b - t_a \geq \Delta$ and diagnoses $d_a = \text{MCI}$ and $d_b = \text{AD}$. Moreover, we require that the diagnosis does not revert after v_b, that is $d_i = \text{AD}$ for all visits v_i with $t_i > t_b$.

Conversely, a case is considered *stable* if its diagnosis does not change across the entire data set and its visits span a time frame of at least Δ, that is $d_i = \text{MCI}$ for all visits v_i and $\max_t t - \min_t t \geq \Delta$.

Using our model, we can generate $\hat{y}^{(\Delta)} = x \circ \Phi^{(\Delta)}$ and use this prediction to estimate the probability of a conversion occuring.

3.2 Long-Term Prediction

Another interesting application is to generate predictions for larger time steps. While we don't expect the model's predictions to be particulary accurate in this setting, especially for time steps $t \gg 1$, i.e. time steps significantly exceeding the maximum time step occuring in training data, long-term predictions can be helpful in highlighting areas of significant change as well as in visualizing how the aging process of a healthy brain differs from that of a brain affected by AD. Since every additional squaring layer doubles the maximum time step t for which we can generate an image, our model can produce outputs for very large steps at little additional computational cost.

3.3 Feature Attribution

Finally, similar to [3], our generator has potential applications in the area of visual feature attribution, that is, highlighting the parts of an image which are most strongly correlated to one of its labels, e.g. the subject's diagnosis TODO(can we say this? Does this make sense?). For instance, by training the model on different subsets of our data, such as exclusively AD or HC cases, we can model the different progressions and visualize their effects either on a single image or aggregated over subsets of our data. Moreover, our model is probabilistic which allows us to produce multiple different predictions for the same input image x and helps in getting an understanding of our model's uncertainty.

Chapter 4

Data

4.1 Synthetic Data

In order to validate our architecture, we first train and evaluate our model on a synthetic data set designed to yield easily interpretable results while still being similar in structure to the preprocessed brain data.

Each sample consists of a pair (x_i, y_i) of $80 \times 96 \times 80$ images, containing a spherical shell with a value of -1 on its shell and 1 in its interior. We randomize both the sphere's radius and position within the image, and sample $t_i \sim \mathcal{U}(0, 1)$, the time step between x_i and y_i. The shell's thickness decreases from x_i to y_i, where the thickness in y_i is defined as $d_{y_i} = (1 - t)d_x$, with d_x identical for all x_i. We explore two different backgrounds, a constant value of 0 as well as smoothed gaussian noise identical for x_i and y_i as shown in Figure 5.4.

We generate a total of 10'000 samples, using 60% of the data set for training and 20% for validation and testing each.

4.2 MRI Data

To train and validate our brain aging models, we use T1-weighted 3D MRI brain scans. We obtain a large data set of raw images with corresponding subject and image meta data from publically available sources and apply a preprocessing pipeline in order to extract, align and segment the brain tissue. Finally, we generate multiple different data sets tailored to our specific experiments.

4.2.1 Data Sources

We use a data set consisting of 19'480 brain MRI scans obtained from the publicly available Alzheimer's Disease Neuroimaging Initiative (ADNI) [12]

and Australian Imaging Biomarkers and Lifestyle (AIBL) [9] studies. The study data was collected on a total of 9976 visits over a time period of 15 years involving 2794 subjects.

The dimensions of the raw scans depend on the type and model of scanner used and therefore vary slightly, with a median of $240 \times 256 \times 170$. Furthermore, depending on a subject's study group assignment, images are taken at a field strengths of 1.5T or 3T.

4.2.2 Image Data Preprocessing

Our data processing pipeline consists of three primary steps:

- Registration

- Extraction

- Segmentation

Firstly, in the registration step we align the raw images to a common reference atlas using linear transformations with 12 degrees of freedom. Secondly, we extract the brain from from the surrounding non-brain tissue in what is known as skull stripping or alternatively brain extraction. Both steps are performed utilizing the FSL toolkit [14], using the `flirt` [16] [13] and `bet` [23] [15] commands respectively. Thirdly, we segment each voxel into one of three classes, White Matter (WM), Gray Matter (GM) and Cerebrospinal Fluid (CSF) while simultaneously correcting a scanner-related image artifact known as the bias field using FSL's `fast` [27] command. The results of this operation are three voxel-wise probability maps for the different classes and we then proceed to subtract the WM map from the GM map while dropping the CSM map. This results in a new image with a number of potentially benefitial properties, where all voxel values are restricted to the range [-1, 1] and can be directly compared across different images. Note that the MR imaging process captures relative intensity differences and as a consequence, direct comparison of absolute values is in general not possible for raw or even unit gaussian normalized data. Furthermore, the operation enhances the structural contrast and removes low level variance in the image. We choose this apprach based on the assumption, that most of the information relevant to the brain aging process is contained in the structural changes of the segmentation, with smaller differences in intensity most likely representing noise. Figure 4.1 shows the entire preprocessing pipeline and all its intermediate steps applied to one sample from our data set.

Note that since the primary output of our preprocessing pipeline is based entirely on segmentation masks, one could combine the T1-weighted scans with data from different brain imaging modalities such as T2-weighted MRI data or proton density (PD) scans, therefore drastically increasing the num-
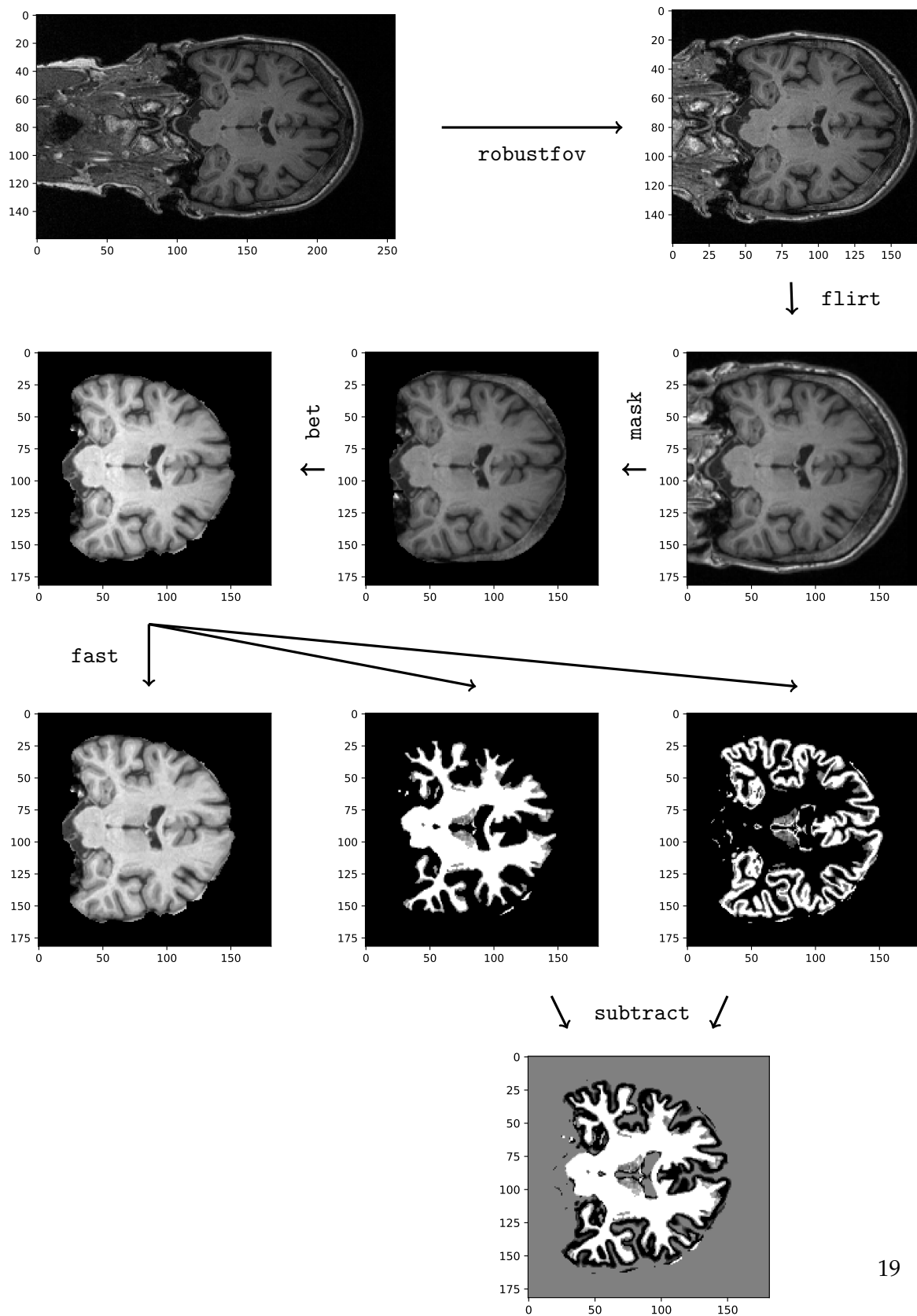
Figure 4.1: Preprocessing pipeline visualized for one sample

ber of possible data sources. However, we do not validate or pursue this idea in the context of this thesis.

For computational reasons, we also perform downsampling with a factor of 0.5 followed by cropping to keep only the center 32 coronal slices, resulting in a final shape of $80 \times 32 \times 80$. However, note that our architecture uses 3D convolutions throughout and therefore can be trained on the full-size data if desired, albeit at a significant computational penalty.

4.2.3 Data Splitting

In order to run our experiments, we generate a number of data sets for the different settings. For the sake of notational brevity and conciseness, let s denote one subject in our data set and let v_i^s denote the i-th visit of subject s, with $V(s)$ denoting the temporally ordered set of visits $v_i^s \in V(s)$ of subject s, for $i \in 1 .. |V(s)|$. To improve readability, we generally omit s unless required. For redundancy, MRI scans are usually performed twice resulting in two separate but very similar images for the same visit. Moreover, images taken at different magnetic field strength levels are available for some subjects. As a consequence, a visit v_i typically consists of multiple images $x_{i,k} \in I(v_i)$ along with the corresponding image meta data. Finally, the examdate t_i and the subject age a_i at time t_i as well as the diagnosis d_i are available for most visits.

Our data sets are divided into five equal splits $\{\mathcal{S}_i\}_{i \in 0 .. 4}$ of 20% each which are used in various different configurations detailed in the corresponding experiment's section. We perform this split on a subject basis and do so globally, in the sense that if a subject appears in a specific data set, it is always assigned to the same split. In other words, for any pair of splits $\mathcal{S}_i^A$ and $\mathcal{S}_j^B$ taken from data sets A and B with $i \neq j$, the intersection $\mathcal{S}_i^A \cap \mathcal{S}_j^B = \varnothing$ is guaranteed to be empty.

Base Image Set

The *Base Image Set* forms the foundation for all other data sets. It consists of all images $x_{i,k}$ and the meta data for the corresponding visits v_i for which t_i and a_i are obtainable. The primary use for this set is in the training of our age regressor models.

MCI/AD Set

The *MCI/AD Set* consists of the images of all visits v_i^s for which the diagnosis $d_i^s \in \{MCI, AD\}$ and we have high confidence in the label, defined as follows:

Split	N	S	HC		MCI		AD		Age	
			N	S	N	S	N	S	mean	std
$\mathcal{S}_0$	2944	503	1286	274	961	172	697	137	75.1	7.4
$\mathcal{S}_1$	2905	504	1198	266	1018	181	689	139	75.6	7.2
$\mathcal{S}_2$	3202	506	1435	269	1036	178	731	132	76.3	7.4
$\mathcal{S}_3$	3294	504	1563	256	1025	192	706	136	75.6	7.4
$\mathcal{S}_4$	2947	506	1242	264	992	189	713	147	75.1	7.6
$\mathcal{S}_{conv}$	4188	271	174	14	3184	271	830	98	75.2	7.1
All	19480	2794	6898	1343	8216	1183	4366	789	75.5	7.4

Table 4.1: Overview of the base image set. *N* refers to the number of separate images and *S* to the number of distinct subjects. Note that for any split, the sum of subjects over all diagnoses generally exceeds the total number of subjects, since one subject may have images with different diagnoses.

We consider a visit v_i^s *firmly MCI* if d_i^s as well as both the diagnoses of the previous and following visit d_{i-1}^s and d_{i+1}^s are *MCI*. Implicitly, this also means that we only consider subjects with at least three visits.

Conversely, for a visit v_i^s to be considered *firmly AD*, we require that both d_i^s and d_{i-1}^s, the current and previous diagnoses, are *AD*.

Note that following these definitions, it is possible for one subject to have visits in both the MCI *and* AD group, see Figure 4.2 for an illustrated example. An overview of the data set is shown in Table 4.2.
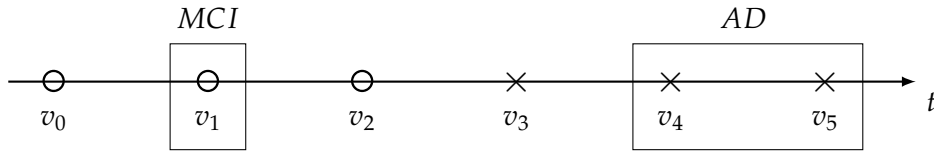

Figure 4.2: Illustation of MCI and AD visits, ○ = MCI, × = AD

Image Pairs Set

The *Image Pairs Set* consists of pairs of images (x_i, x_j) of subject s at two different visits v_i and v_j. Of particular importance is the time step $t_j - t_i$ between v_i and v_j. We limit the maximum time step to 6 years for computational reasons explained in subsection 2.2.2. As visualized in Figure 4.4, the data set is biased towards smaller t with median of 1.53, mean of 2.00 and

a standard deviation of 1.47. To mitigate this, we calculate sample weights $w = (|t - \bar{t}| + 1)^{1/2}$, where $\bar{t}$ is the mean over all t. The resulting distribution is shown in Figure 4.4. We also visualize the L_1 difference between x and y for all pairs in Figure 4.3.
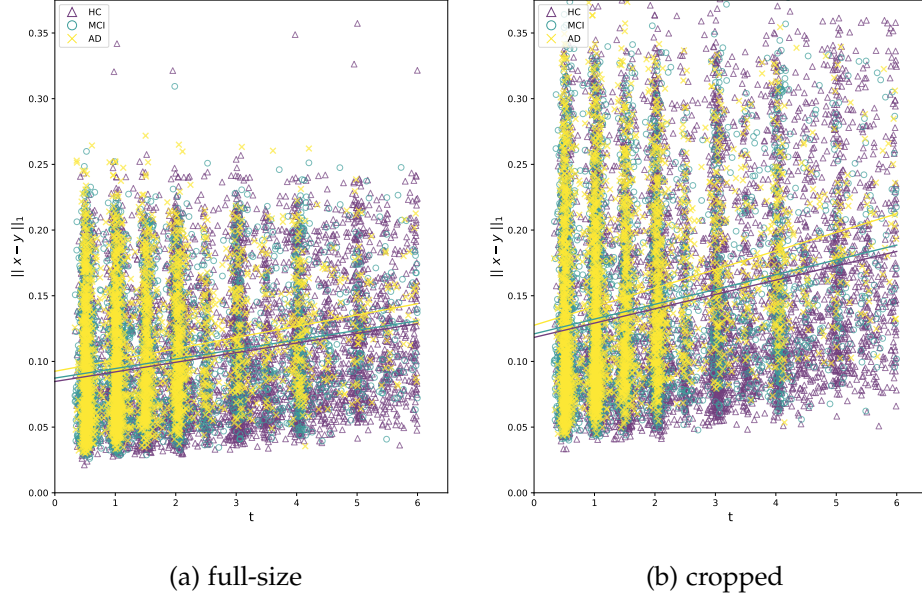

(a) full-size

(b) cropped

Figure 4.3: L_1 difference between image pairs (x, y) from our data set for both the full-size and the cropped scans. While the difference increases with t, it does so rather slowly with slopes of 0.007 and 0.011 respectively.

MCI Conversion Set

The *MCI Conversion Set* consists of image pairs (x_i, x_j) of progressive and stable MCI subjects according to the definitions in section 3.1. Adding to these constraints, for a subject to be considered pMCI we further require a minimum of two visits diagnosed as MCI and AD each. Furthermore, the subject's diagnosis may not revert from AD to MCI at any point in time.

Following the notation in 3.1, we choose the time step between v_i and v_j to be $\Delta = 4$ based on the available data as well as previous work in [?]. In general, multiple viable image pair combinations exist for each subject. We prioritize matching Δ followed by centering the point in time where the diagnosis change occurs within Δ. Figure 4.5 shows one such pair of visits for a pMCI subject. In total, the conversion set contains 271 pairs of images from 271 subjects, of which 98 are pMCI and 173 are sMCI.
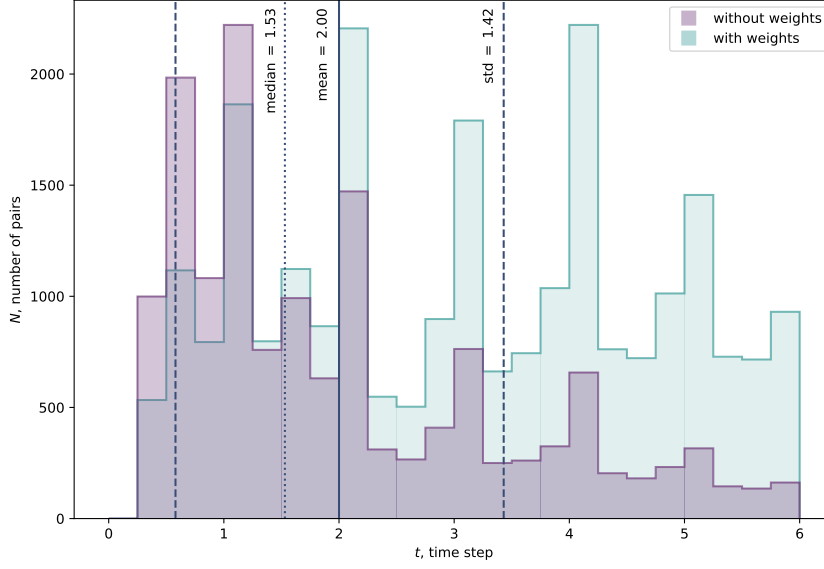
Figure 4.4: The historgram for time step t across all pairs in the data set. To mitigate the bias towards smaller t, we calculate sample weights to be used during training.
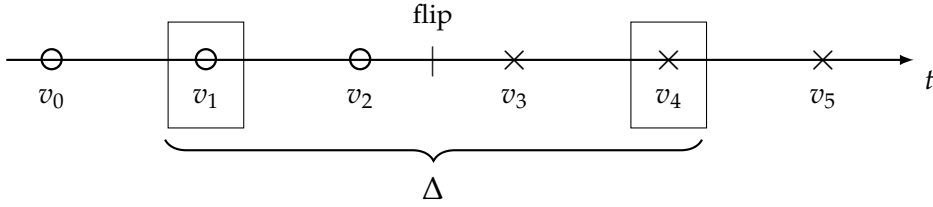

Figure 4.5: Illustration of a pMCI image pair, ○ = MCI, × = AD

Note that due to its use in the model validation, this data set represents a separate independent split, that is, subjects which are part of the MCI Conversion Set do not occur in any other split.

Split	N	S	MCI		AD		MCI ∩ AD		Age	
			N	S	N	S	N	S	mean	std
$\mathcal{S}_0$	787	148	391	82	396	71	40	5	74.4	7.6
$\mathcal{S}_1$	814	148	408	85	406	71	71	8	76.4	7.9
$\mathcal{S}_2$	909	150	451	82	458	72	45	4	77.0	7.9
$\mathcal{S}_3$	845	148	426	83	419	71	75	6	76.0	7.9
$\mathcal{S}_4$	767	150	372	89	395	72	90	11	74.9	8.0
All	4122	744	2048	421	2047	357	321	34	75.8	7.9

Table 4.2: Overview of the MCI/AD data set. N refers to the number of separate images and S to the number of distinct subjects. For MCI ∩ AD, N refers to the number of images from subjects for which we have images in both groups.

	All		HC		MCI		AD	
	N	S	N	S	N	S	N	S
HC	7339	674	6739	649	495	110	105	27
MCI	5399	918	469	83	3698	737	1232	331
AD	2019	421	1	1	43	18	1975	416
All	14757	1789	7209	673	4236	794	3312	646

Table 4.3: Overview of the image pairs set, showing the number of pairs for all combinations of diagnoses. Rows correspond to the first image, columns to the second. N refers to the number of separate images and S to the number of distinct subjects.

Chapter 5

Experiments

5.1 Age Regressor

Validating the performance of a generative model is a hard problem in general. Beyond visual inspection of the outputs, we also pre-train an age regressor, which given an image x produces an estimated age label $\hat{a}_x$, and use it to estimate the age label, and by extension the time step, of outputs generated by our model. Furthermore, the regressor is also included in the generator loss function as described in section 2.2.3. Given its importance in the validation of our generative model, we evaluate the regressor's performance on a number of different tasks.

The regressor is implemented as a 3D CNN with nine layers, eight of which using batch normalization, and trained using the Adam optimizer with $\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999$ and $\varepsilon = 0.0001$.

Absolute Error

First, we train the regressor for 30'000 batches of 32 samples each optimizing the absolute mean error as its objective function. Using our 5-fold data split, we perform cross validation and present the results in **??**. Figure 5.1 shows the estimated age $\hat{a}_x$ against the ground truth label a_x for one validation split. We note that the estimator tends to the mean, that is, its estimates are most accurate around the mean age across the data set of 75 years, and are low for subjects above the mean and high for subjects below the mean. Given the linear nature of the loss function, this is to be expected.

However, in our generative model, rather than estimating the absolute age of an image, the age regressor is used to estimate the relative age difference $a_y - a_x$ for an image pair (x, y). Therefore, we further evaluate its performance on this task using our data set of image pairs. As before, we perform 5-fold cross validation and report the results in Table 5.1. Figure 5.2 com-
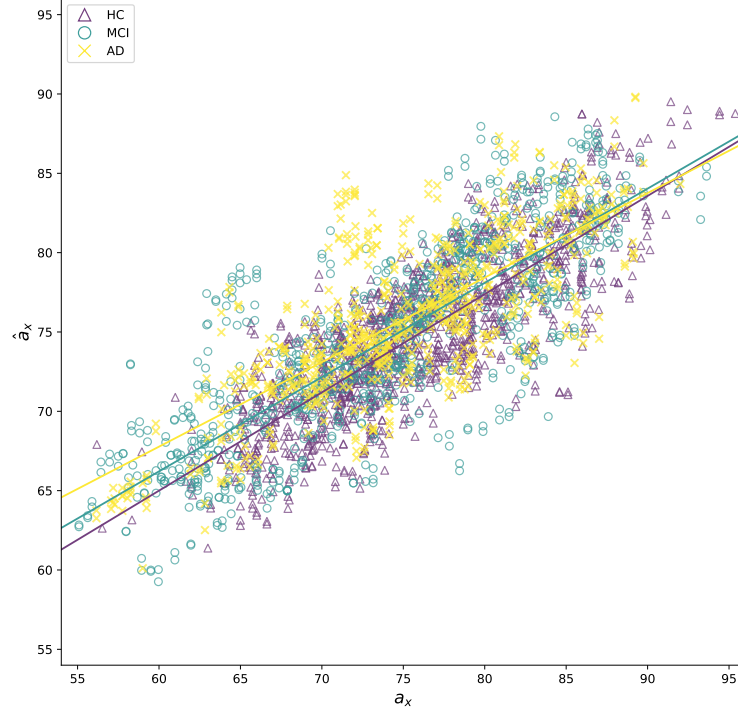
Figure 5.1: Age regressor estimates on 2937 validation samples from our base image data set. Note how the predicitions tend to the mean, with a slope of 0.58 compared to the target of 1. The subject's diagnosis does not appear to have a major impact on the performance.

pares the absolute losses for x and y and shows the error cancelling effect discussed in section 2.2.3. Furthermore, we also estimate the age difference as $\hat{a}_y - \hat{a}_x$ for one split and visualize it against the ground truth time step in Figure 5.3. Using linear regression on our data points, we obtain an intercept of 0.03 and a slope of 0.67, indicating the estimated age difference tends to be too small for increasing t. This can be explained by closer examination of Figure 5.1 which shows that as the estimate $\hat{a}_x$ tends to the mean for labels futher away from it, relative age differences are subject to a shrinkage effect.

Note that since we are performing these experiments on real image pairs, the results represent an upper bound for the performance of our generative model.
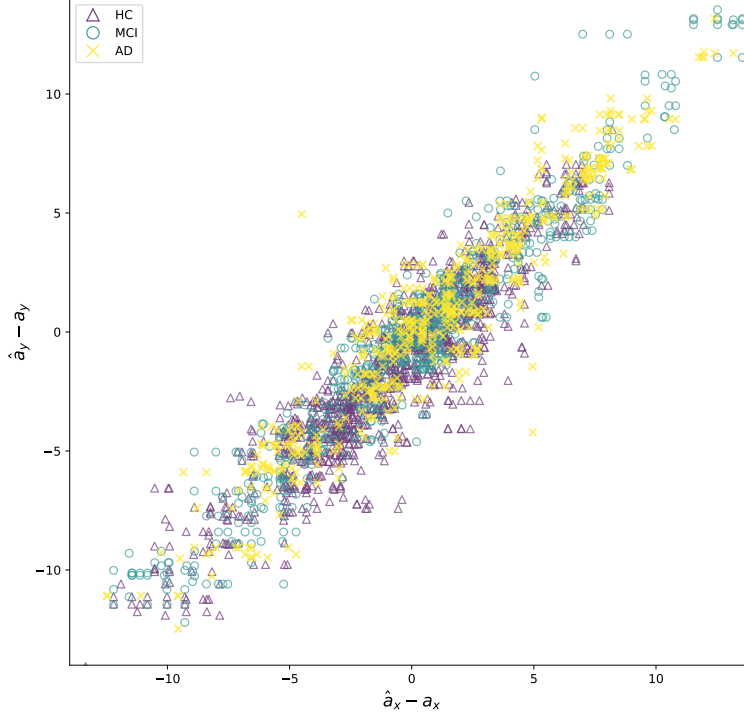
Figure 5.2: Age regressor estimates on 2651 validation samples from our pairs data set, with each axis representing one of the two images. While the absolute age error for a single image is quite significant with a mean of 3.39, the estimate of the relative age difference between two images from the same subject is considerably more accurate with a mean error of 1.21.

Squared Error

We also examine the performance of the same architecture optimizing the mean squared error, with very similar results presented in Table 5.1

5.2 Diagnosis Classifier

We pre-train a diagnosis classifier to discriminate between images labeled as MCI and AD respectively. Given the gradual transition between the two diagnoses, we use the *MCI/AD Set* described in section 4.2.3, which limits our training and validation sets to a subset of images for which we have increased confidence in the diagnosis labels. Note that we exclude all subjects in the healthy control group HC and use a binary classifier focussing on the
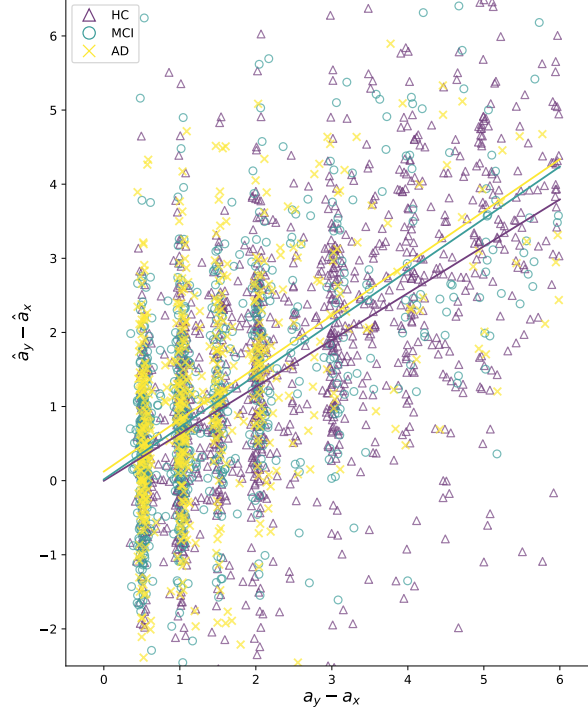
Figure 5.3: Age regressor estimates on 2651 validation samples from our pairs data set, comparing the estimated age difference to the ground truth time step. Linear regression yields an intercept of 0.03 and a slope of 0.67. Note that the stripe patterns forming along the *x*-axis are a consequence of the study scheduling which mandates follow-up visits in intervals of six or twelve months.

more subtle distinctions between the effects of MCI and AD.

The classifier is implemented as a 3D CNN identical in structure to the age regressor. Softmax cross entropy is used as the objective function and minimizes using the Adam optimizer with $\alpha = 0.0001, \beta_1 = 0.9, \beta_2 = 0.999$ and $\varepsilon = 0.01$ for increased training stability. We perform cross validation using the 5-fold data split. Due to high variance in the classifier's accuracy, we train the model five times for every fold in order to avoid suboptimal local minima and report the best results. Each training runs consists of 10'000 batches of 32 samples each.

Split	absolute		squared	
	μ	σ	μ	σ
$\mathcal{S}_0$	3.58	2.92	3.62	2.96
$\mathcal{S}_1$	3.69	2.93	3.86	2.97
$\mathcal{S}_2$	4.07	3.01	4.09	2.97
$\mathcal{S}_3$	4.07	3.07	3.89	2.90
$\mathcal{S}_4$	3.90	3.31	4.06	3.32
	3.86	3.05	3.90	3.02

Table 5.1: Cross validated mean and standard deviation of the absolute error for the age regressor model, comparing absolute and squared loss objectives.

Split	accuracy		F_1-Score	
	μ	σ	μ	σ
$\mathcal{S}_0$	3.58	2.92	3.62	2.96
$\mathcal{S}_1$	3.69	2.93	3.86	2.97
$\mathcal{S}_2$	4.07	3.01	4.09	2.97
$\mathcal{S}_3$	4.07	3.07	3.89	2.90
$\mathcal{S}_4$	3.90	3.31	4.06	3.32
	3.86	3.05	3.90	3.02

Table 5.2: Cross validated mean and standard deviation of the absolute error for the age regressor model, comparing absolute and squared loss objectives.

5.3 Diffeomorphic Models

5.3.1 Voxelmorph

To establish a baseline, we first train a variation of the Voxelmorph architecture described in subsection 2.1.1 with minimal modifications to support the brain aging task. These modifications include removing the target image y from the generator input as well as extending the scaling and squaring method to produce output deformations $\Phi^{(t)}$ for arbitrary t as described in subsection 2.2.2.

Note that we do not perform any hyperparameter tuning which might result in a significant improvement in performance.

5.3.2 Synthetic Data

To validate our modifications to the architecture, we first train the model on 1'000 batches of 4 samples each from the synthetic data set described in section 4.1. Visual inspection of the results, presented in Figure 5.4 confirms the model's ability to learn and integrate over a stationary velocity field to generate deformation fields for variable time steps t. Furthermore, we demonstrate the effectiveness of the sparseness penalty in suppressing the deformation field in areas of little change. Note that neither the age regressor nor the diagnosis classifier are used in any of these experiments as there are no corresponding features in the synthetic data samples.

5.3.3 MRI Data

We train the model in a number of different configurations and on different data sets listed in Table 5.3.

#	λ_{age}	λ_{dx}	$\mathcal{S}_{train}$	$\mathcal{S}_{valid}$	$\mathcal{S}_{test}$	Pairs
1	-	-	$\mathcal{S}_1\,\mathcal{S}_2\,\mathcal{S}_3\,\mathcal{S}_4$	$\mathcal{S}_0$	$\mathcal{S}_{test}$	any $\rightarrow$ any
2	25	-	$\mathcal{S}_1\,\mathcal{S}_3$	$\mathcal{S}_2\,\mathcal{S}_4$	$\mathcal{S}_0$	any $\rightarrow$ any
3	-	25	$\mathcal{S}_1\,\mathcal{S}_3$	$\mathcal{S}_2\,\mathcal{S}_4$	$\mathcal{S}_{test}$	any $\rightarrow$ MCI/AD
4	25	25	$\mathcal{S}_1\,\mathcal{S}_3$	$\mathcal{S}_2\,\mathcal{S}_4$	$\mathcal{S}_0\,\mathcal{S}_{test}$	any $\rightarrow$ MCI/AD
5	-	-	$\mathcal{S}_1\,\mathcal{S}_2\,\mathcal{S}_3\,\mathcal{S}_4$	$\mathcal{S}_0$	-	MCI/AD $\rightarrow$ AD
6	-	-	$\mathcal{S}_1\,\mathcal{S}_2\,\mathcal{S}_3\,\mathcal{S}_4$	$\mathcal{S}_0$	-	HC $\rightarrow$ HC

Table 5.3: Overview of the generator configurations.

For the remaining hyperparameters, we use a relatively modest L_1 similarity loss weight $\lambda_{sim} = 200$ and drop the sparseness loss altogether due to negative performance impacts even for small λ_{sparse}. During training, the velocity field v is integrated using up to $T = 7$ squaring steps, resulting in a maximum number of 256 atomic steps. In combination with the maximum time step of 6 years in our pairs data set, this corresponds to a temporal precision of 8.6 days.

Note that for the training of the regressor as well as the classifier used as loss terms in configurations 2, 3, and 4, $\mathcal{S}_{train}$ and $\mathcal{S}_{valid}$ are swapped. While this separation isn't required to ensure the validity of the results, a regressor or classifier pre-trained on the generator training data will predict age and diagnosis labels which are essentially identical to the real labels, therefore negating the error cancelling effect discussed in section 2.2.3. Indeed, using this configuration results in impressive looking but overly confident results as seen in TODO(plot with slope 1)

| x | y | $\hat{y}$ | t | v | $\|v\|_2$ | v_x | v_y | v_z |

no noise, no sparseness penalty

no noise, with sparseness penalty
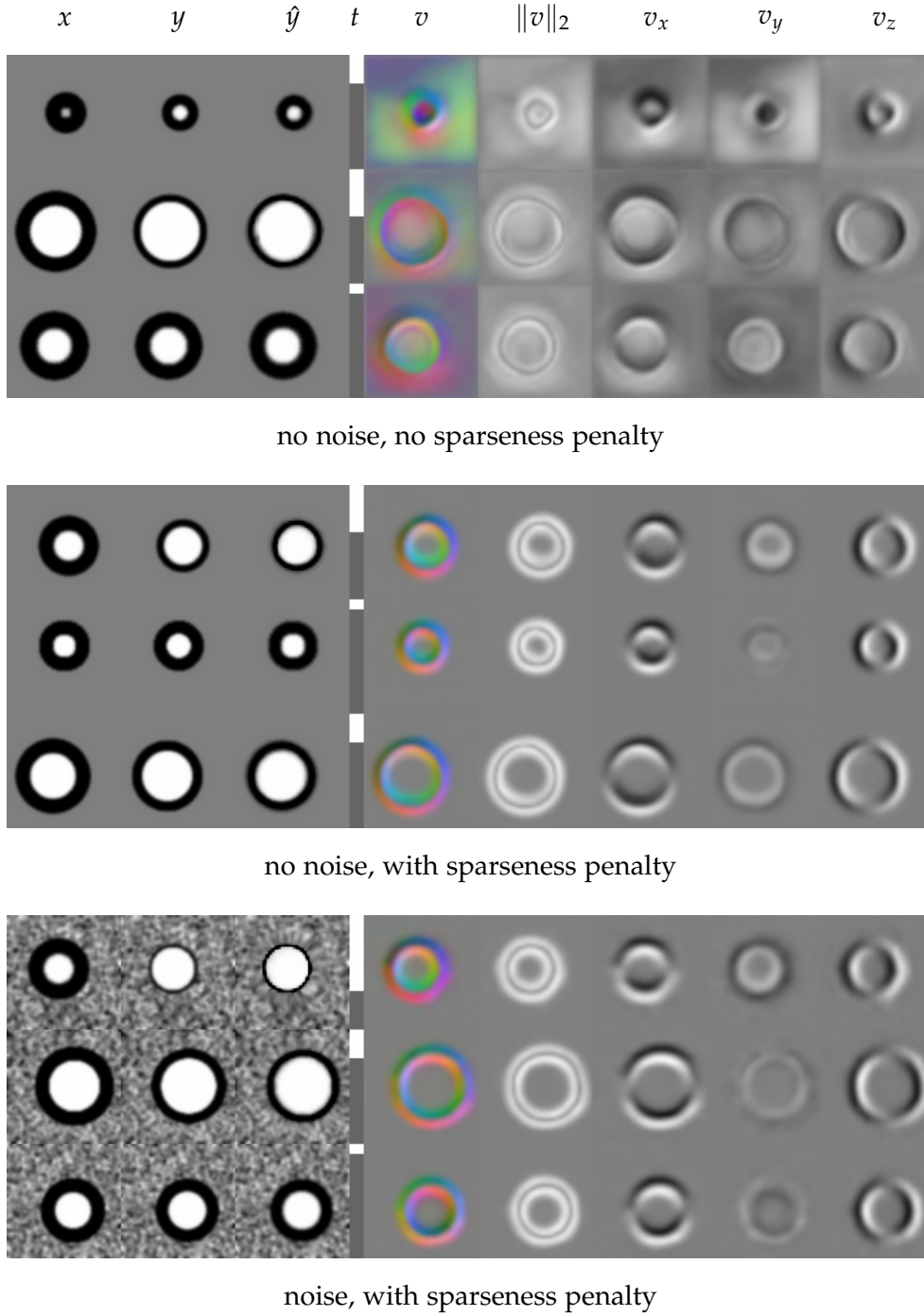
noise, with sparseness penalty

Figure 5.4: Model outputs generated using the synthetic data set. We train the model three times, on noisy or solid backgrounds, with or without applying a sparseness penalty. Each row represents one sample output consisting of the original, the target, and the generated image, the time step t scaled to $[0, 1)$, the velocity field v and its magnitude, as well as its separate dimensions. Note how the sparseness penalty leads to a more interpretable deformation field.
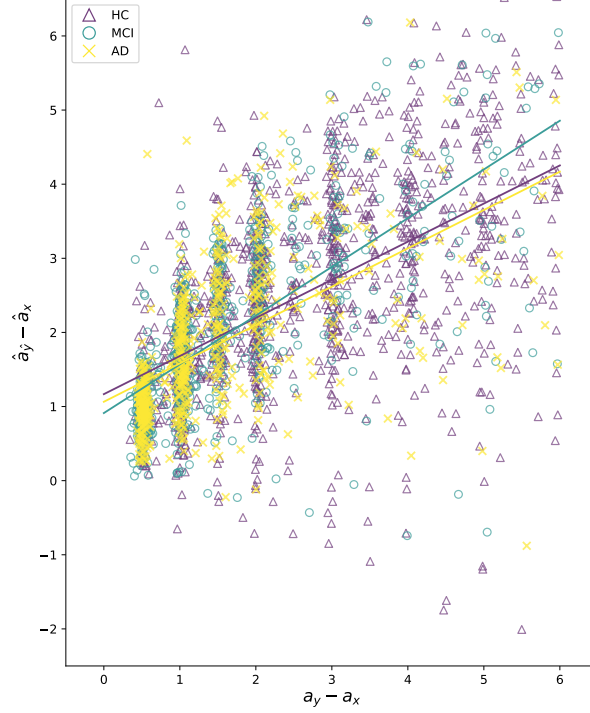
31

Figure 5.5: Age regressor estimates on 2651 validation samples $(x, \hat{y})$ with $\hat{y} = G(x)$ from our pairs data set, comparing the estimated age difference to the ground truth time step. Linear regression yields an intercept of 0.03 and a slope of 0.61. TODO Compare to figure with real diffs.

The model is trained using the Adam optimizer with $\alpha = 0.0001$, $\beta_1 = 0.0$, $\beta_2 = 0.9$ and $\varepsilon = 10^{-7}$. Limited by GPU memory, we use batch size of 8 and train the models for a varying number of steps depending on the configuration. Following the training procedure in [10], we alternate between training the critic and the generator on five and one batches respectively. Each such training step runs in roughly 5 seconds per batch or 0.6 seconds per sample.

Follow-Up Prediction

Our first experiment is to predict follow-up images using the actual time steps in our pairs data set and we evaluate the performance on this task on configurations 1 and 2. We visually inspect the results compare predicted time step

TODO(show example)

Fixed Time Step Prediction

To evaluate the model's ability to generate follow-up images at fixed time steps, we predict images at $t \in \{1, 2, 4, 6, 8\}$ years and estimate the age labels using the age regressor. The results for configurations 1 and 2 are presented in Table 5.4 and visualized for configuration 2 in Figure 5.6. Note that due to the age regressor's tendency to the mean, these results should be compared to Figure 5.3 rather than the ground truth labels. We evaluate configuration 1 at 45'000 training steps, where the model converges, and configuration 2 at 15'000 steps where $\mathcal{L}_{age}$ is minimal on the validation data.

While both configurations result in very similar predicted mean age, the standard deviation improves noticeably for configuration 2, indicating a positive effect of the age regressor loss term.

#	$t = 1$		2		4		6		8	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
1	1.90	0.95	2.60	1.36	3.30	2.24	3.51	3.01	3.35	3.69
2	1.93	0.85	2.64	1.11	3.24	1.67	3.53	2.22	3.62	2.79

Table 5.4: Results of the fixed time step experiment.

Feature Attribution

To visualize the differences in aging between healthy subjects and subjects affected by Alzheimer's Disease, we train configurations 5 and 6 on limited data sets. AD/HC only, compare magnitude of flow differs (maybe numbers, but the finding is qualitative)

Long-Term Prediction

Some plots long term, doesn't work very well because we are using deformations, might better using a hybrid iterarive approach like wegmayr

Conversion Prediction

5x classifier, from cross val 5x threshold each xr = baseline yr = upper bound
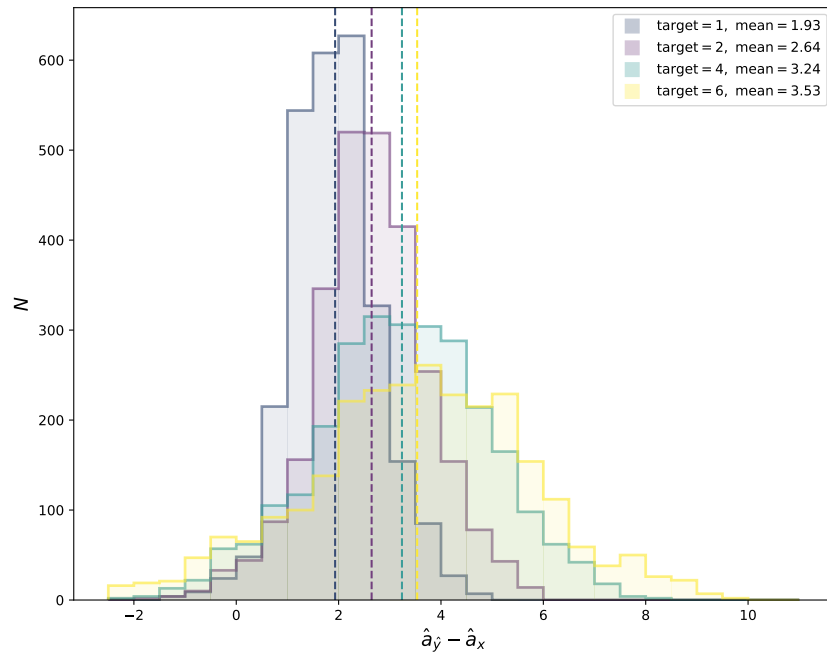
F1 score, accuracy

Figure 5.6: TODO must be compared with slope.
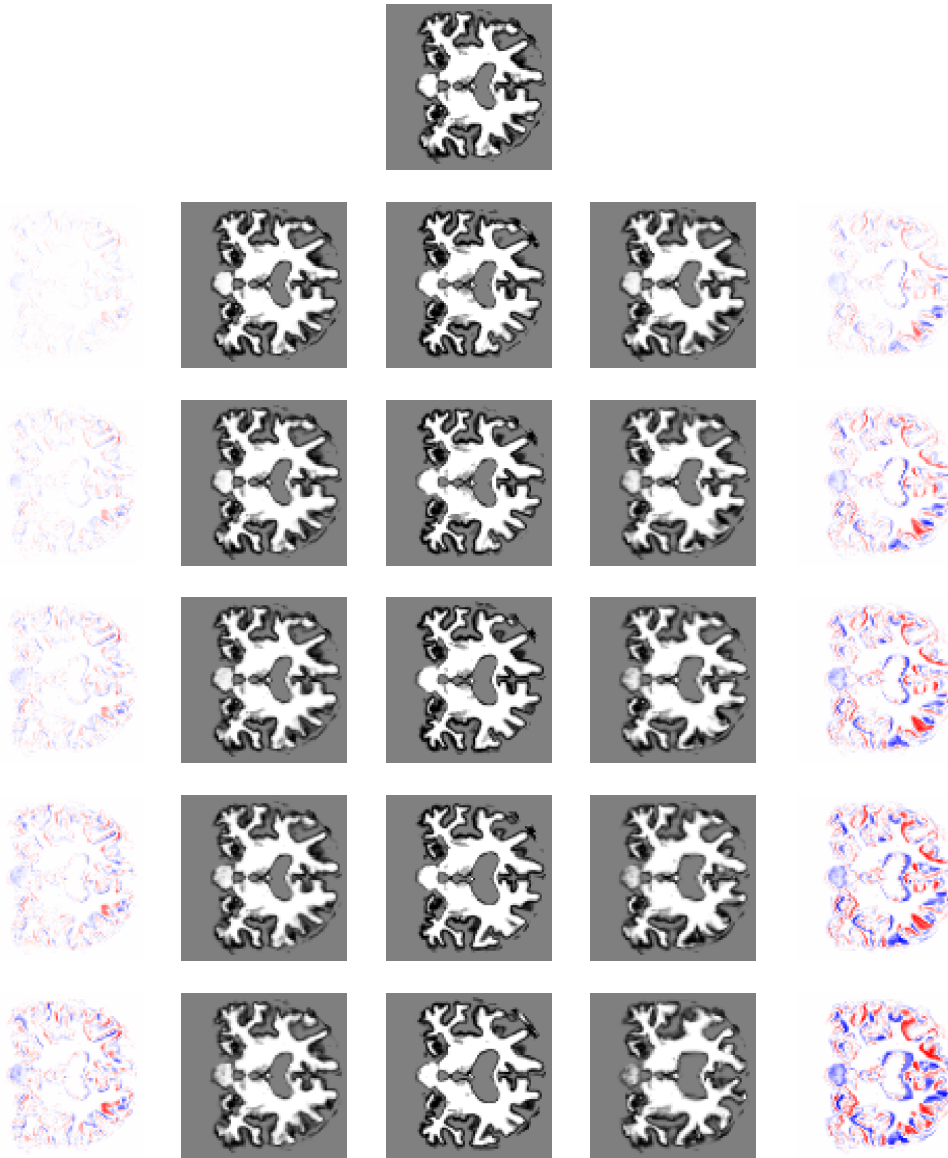
Figure 5.7: The center column consists of a series of real images from a **subject diagnosed with Alzheimer's Disease**, taken at one-year intervals steps, top to bottom. The two columns to the left show the predicted images at the same time steps using a model trained exclusively on healty patients, as well as the difference maps with respect to the base image. Similarly, the two columns to the right show the predicted images using a model exclusively trained on patients affected by Alzheimer's Disease.

Figure 5.8: The center column consists of a series of real images from a **healthy subject**, taken at one-year intervals steps, top to bottom. The two columns to the left show the predicted images at the same time steps using a model trained exclusively on healty patients, as well as the difference maps with respect to the base image. Similarly, the two columns to the right show the predicted images using a model exclusively trained on patients affected by Alzheimer's Disease.
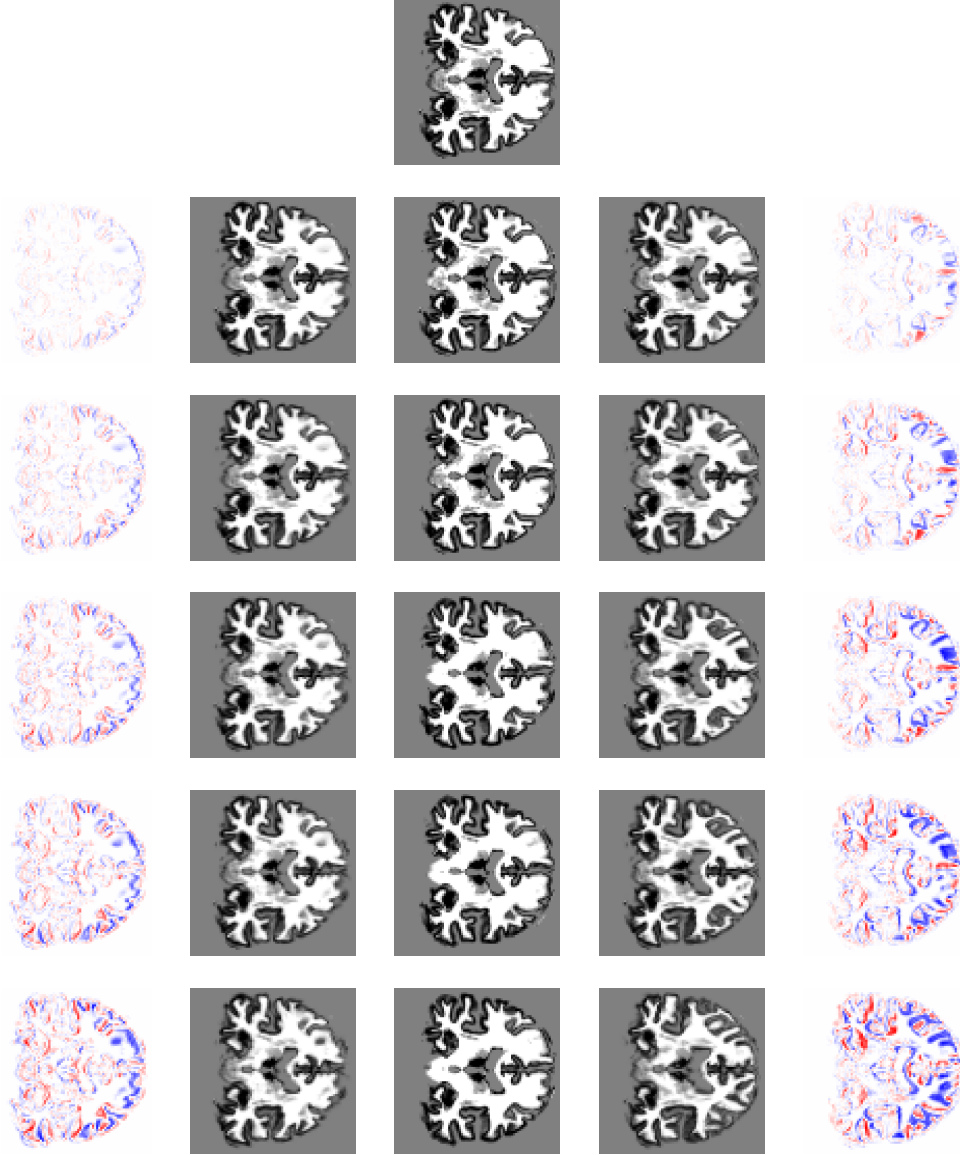
Chapter 6

Related Work

Most similar to our work in terms of the problem setting, [**?**] use a WGAN architecture to model the brain's aging process. They propose a UNet-derived model architecture based on [3], which is trained and applied iteratively to obtain predictions for different time steps. In addition to modeling the differences in the aging process between healthy subjects and subjects affected by Alzheimer's Disease, they report cautiously positive results for the task of conversion prediction. [3] use a WGAN on 3D MRI brain data to perform visual feature attribution and apply it to generate image-specific effect maps of Alzheimer's Disease. In contrast to our work, the models used in [**?**] and [3] are entirely convolutional and do not use deformations.

Pursuing a similar goal, [21] use a combination of recursive and convolutional neural networks to predict a sequence of deformations based on, and then applied to, a baseline image to obtain follow-up predictions at different time steps. The model is trained on the first image of a subject as well as the sequence of diffeomorphic vector momenta for each additional image which are generated using the LDDMM framework [4].

From a model perspective, the work most similar to ours is [2] [6], which forms the basis of our work both in terms of the model design as well as its implementation. While the architecture was initially designed for unsupervised image registration, it has since been adapted to the problem of unsupervised segmentation [7]. In the domain of face aging, [20] suggest a conditional GAN [19] architecture similar to ours using an age regressor in the loss function.

As previously mentioned, early predictions of Alzheimer's Disease onset has been the target of considerable research efforts. [25] propose an SVM classifier distinguishing subjects with stable and progressive MCI and report an accuracy of 78.2%. In contrast to our work, the features are extracted from longitudinal MRI brain data collected over a period of up to 18 months.

Reporting an accuracy of 92%, [24] also use an SVM classifier on features extracted from a stationary velocity field, which is calculated using [26] on longitudinal data up to 36 months.

Chapter 7

Discussion

Architecture with many components, hard to pinpoint problem origin

Large number of hyperparameters, high training time

Interpretability of WGAN loss

Generative model output validation is hard, regressor as

Chapter 8

Conclusion

appendix

Bibliography

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[2] Vincent Arsigny, Olivier Commowick, Xavier Pennec, and Nicholas Ayache. A log-euclidean framework for statistics on diffeomorphisms. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 924–931. Springer, 2006.

[3] John Ashburner. A fast diffeomorphic image registration algorithm. *Neuroimage*, 38(1):95–113, 2007.

[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*, 2019.

[5] Christian F Baumgartner, Lisa M Koch, Kerem Can Tezcan, Jia Xi Ang, and Ender Konukoglu. Visual feature attribution using wasserstein gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8309–8319, 2018.

[6] M Faisal Beg, Michael I Miller, Alain Trouvé, and Laurent Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International journal of computer vision*, 61(2):139–157, 2005.

[7] Ron Brookmeyer, Elizabeth Johnson, Kathryn Ziegler-Graham, and H Michael Arrighi. Forecasting the global burden of alzheimer's disease. *Alzheimer's & dementia*, 3(3):186–191, 2007.

[8] Adrian V Dalca, Guha Balakrishnan, John Guttag, and Mert R Sabuncu. Unsupervised learning for fast probabilistic diffeomorphic registration. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 729–738. Springer, 2018.

[9] Adrian V Dalca, Evan Yu, Polina Golland, Bruce Fischl, Mert R Sabuncu, and Juan Eugenio Iglesias. Unsupervised deep learning for bayesian brain mri segmentation. *arXiv preprint arXiv:1904.11319*, 2019.

[10] Hao Dong, Guang Yang, Fangde Liu, Yuanhan Mo, and Yike Guo. Automatic brain tumor detection and segmentation using u-net based fully convolutional networks. In *annual conference on medical image understanding and analysis*, pages 506–517. Springer, 2017.

[11] Kathryn A Ellis, Ashley I Bush, David Darby, Daniela De Fazio, Jonathan Foster, Peter Hudson, Nicola T Lautenschlager, Nat Lenzo, Ralph N Martins, Paul Maruff, et al. The australian imaging, biomarkers and lifestyle (aibl) study of aging: methodology and baseline characteristics of 1112 individuals recruited for a longitudinal study of alzheimer's disease. *International Psychogeriatrics*, 21(4):672–687, 2009.

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[13] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777, 2017.

[14] Clifford R Jack Jr, Matt A Bernstein, Nick C Fox, Paul Thompson, Gene Alexander, Danielle Harvey, Bret Borowski, Paula J Britson, Jennifer L. Whitwell, Chadwick Ward, et al. The alzheimer's disease neuroimaging initiative (adni): Mri methods. *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 27(4):685–691, 2008.

[15] Mark Jenkinson, Peter Bannister, Michael Brady, and Stephen Smith. Improved optimization for the robust and accurate linear registration and motion correction of brain images. *Neuroimage*, 17(2):825–841, 2002.

[16] Mark Jenkinson, Christian F Beckmann, Timothy EJ Behrens, Mark W Woolrich, and Stephen M Smith. Fsl. *Neuroimage*, 62(2):782–790, 2012.

[17] Mark Jenkinson, Mickael Pechaud, Stephen Smith, et al. Bet2: Mr-based estimation of brain, skull and scalp surfaces. In *Eleventh annual meeting of the organization for human brain mapping*, volume 17, page 167. Toronto., 2005.

[18] Mark Jenkinson and Stephen Smith. A global optimisation method for robust affine registration of brain images. *Medical image analysis*, 5(2):143–156, 2001.

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[20] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[21] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[22] Sveinn Palsson, Eirikur Agustsson, Radu Timofte, and Luc Van Gool. Generative adversarial style transfer networks for face aging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 2084–2092, 2018.

[23] Sharmin Pathan and Yi Hong. Predictive image regression for longitudinal studies with missing data. *arXiv preprint arXiv:1808.07553*, 2018.

[24] Max Roser. Life expectancy. *Our World in Data*, 2019. https://ourworldindata.org/life-expectancy.

[25] Stephen M Smith. Fast robust automated brain extraction. *Human brain mapping*, 17(3):143–155, 2002.

[26] Zhuo Sun, Martijn van de Giessen, Boudewijn PF Lelieveldt, and Marius Staring. Detection of conversion from mild cognitive impairment to alzheimer's disease using longitudinal brain mri. *Frontiers in neuroinformatics*, 11:16, 2017.

[27] Kim-Han Thung, Chong-Yaw Wee, Pew-Thian Yap, and Dinggang Shen. Identification of progressive mild cognitive impairment patients using incomplete longitudinal mri scans. *Brain Structure and Function*, 221(8):3979–3995, 2016.

[28] Tom Vercauteren, Xavier Pennec, Aymeric Perchant, and Nicholas Ayache. Diffeomorphic demons: Efficient non-parametric image registration. *NeuroImage*, 45(1):S61–S72, 2009.

[29] Yongyue Zhang, Michael Brady, and Stephen Smith. Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE transactions on medical imaging*, 20(1):45–57, 2001.

[30] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.