ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

# Diffeomorphic Modeling of the Brain Aging Process

Master Thesis

Patrice Marti

Wednesday 29th May, 2019

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

Department of Computer Science, ETH Zürich

**Abstract**

Average human life expectancy has increased dramatically over the last century, leading to a significant rise in the prevalence of aging-related neurodegenerative diseases. Alzheimer's Disease in particular is one of the only leading causes of death still on the rise. As such, advancing our understanding of the brain aging process as well as early stage detection of Alzheimer's Disease onset is a important area of research. In our work, we focus on modeling the brain aging process on Magnetic Resonance Imaging (MRI) scans. To that end, we express the aging process as a diffeomorphic deformation and introduce a conditional Generative Adversarial Network (GAN) architecture to predict the future state of a brain. Specifically, our model learns a stationary velocity field which is subsequently integrated using the scaling and squaring method, for which we propose an extension to handle arbitrary time steps both in training and inference. Beyond visual inspection, we validate our model's performance by applying a pre-trained age regressor to the generated outputs. Furthermore, we use our model to predict the probabilty of patients with Mild Cognitive Impairment converting to Alzheimer's Disease.

# Contents

Chapter 1

# Introduction

The past decades have seen medical research progress at a rapid rate, resulting in a dramatic increase of the average human life expectancy [25]. 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 [7]. 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) [21] and Generative Adversarial Networks (GANs) [13], have been successfully applied to model gradual changes in image data for wide range of settings such as face aging [23], image registration [4] and style transfer [32]. In this thesis, we propose a diffeomorphic Generative Adversarial Network architecture to model the brain aging process . Specifically, we consider a model which given a T1-weighted MRI image $x$ taken at time $t_0$ predicts an image $y$ at some time $t_1$ in the future. Such a model could pave the way for a number of applications TODO(fix). For instance, existing diagnostic tools such as diagnosis classifiers operating on MRI data 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 such as the areas most strongly correlated with a condition can be gained by aggregating the model's outputs over different groups of subjects.

In the field of medical imaging and computational anatomy, diffeomorphic deformations are a popularly used model choice [6] [3]. Unlike entirely convolutional models, diffeomorphisms are constrained to transformations which are invertible and therefore topology preserving, and are thus generally better suited to model the gradual changes observed in sensitive tissues

such as the brain. Following [8], 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 [2]. In its general formulation, the scaling and squaring method is used to integrate a vector field to one specific time step. Therefore, in order to maximize the available training data as well as being able to generalize over larger time ranges, we propose a modification to the method to approximate deformations for arbitrary time steps. In turn, this allows our brain aging model to generate and be trained on image pairs with arbitrary time differences.

Finally, validating generative model outputs beyond subjective visual inspection is a notoriously difficult task. While user studies can be employed in some domains such as face aging [23], this is not a viable option for the task of brain aging. Instead, we propose to use a pre-trained age regressor applied to our generator's outputs as a more meaningful and comparable metric. We show that while an age regressor's absolute loss may be comparatively high, the relative loss between two images taken from the same subject can be significantly lower due to an error cancelling effect.

TODO introduce conversion problem

Our main contributions are:

- we model the brain aging process using diffeomorphic deformations
- we propose a modification to the scaling and squaring method for arbitrary timesteps
- we suggest the use of an pre-trained age regressor to validate our generative model's ability to predict follow-up images
- we demonstrate our model's ability to predict the Alzheimer's Disease conversion probability of patients with Mild Cognitive Impairment TODO(maybe fix)

## 1.1 Related Work

Most similar to our work in terms of the problem setting, [30] use a WGAN architecture to model the brain's aging process. They propose a UNet-derived model architecture based on [5], 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. [5] 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 [30] and [5] are entirely convolutional and do not use deformations.

Pursuing a similar goal, [24] 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 [6].

From a model perspective, the work most similar to ours is [4] [8], 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 [9]. In the domain of face aging, [23] suggest a conditional GAN [22] 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. [28] 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%, [27] also use an SVM classifier on features extracted from a stationary velocity field, which is calculated using [29] on longitudinal data up to 36 months.

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 [4], segmentation [11] and visual feature attribution [5]. Of particular interest are deformation-based models due to their ability to closely model the gradual changes observed in the context in medical imaging. Additionally, by using *diffeomorphic* deformations, the model can be limited to operations which are smooth, differentiable and invertible and, as a consequence, topology preserving. Moreover, unlike convolutional models which generally implement a single atomic transformation, diffeomorphic deformations can be interpolated at any intermediate time step, therefore generally 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 [4] [8] 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 $\Phi$ such that $x \circ \Phi$ is similar to $y$.

### 2.1.1 Voxelmorph

Dalca et al [8] 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 $\Phi : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ mapping $x$ to $y$ through the ordinary differential equation (ODE)

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

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 [2].

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 $\Phi_z$ through the ODE (2.1). The prior probability 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 $\Lambda_z$ 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 $\lambda$ denotes a parameter controlling the scale of the velocity field.

The generator's output is modelled as a gaussian distribution with mean $y$, the target image, and standard deviation $\sigma$. In other words, the warped image $x$ is interpreted as a noisy oberservation of the target image $y$

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

with $\sigma^2$ reflecting the variance of the additive noise.

A likely registration field $\Phi_z$ 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 $\psi$. 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

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

The complete loss function can be separated into two components, a reconstruction and a prior term. Furthermore, the latter can be split into a covariance and a precision term.

$$\begin{aligned} \mathcal{L}(\psi;x,y) &= -\mathbb{E}_q[\log p(x|z;y)] + \mathrm{KL}[q_\psi(z|x;y)||p(z)] \\ &= \underbrace{\frac{1}{2\sigma^2}\|y - x\circ\Phi_z\|^2}_{\text{reconstruction term}} \\ &+ \frac{1}{2}\Bigg[\underbrace{tr(\lambda D\Sigma_{z|x;y} - \log|\Sigma_{z|x;y}|)}_{\text{covariance term}} + \underbrace{\mu_{z|x;y}^T\Lambda_z\mu_{z|x;y}}_{\text{precision term}}\Bigg] \end{aligned} \tag{2.8}$$

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 $\sigma$ and $\lambda$ 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 upsampling 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 [21],

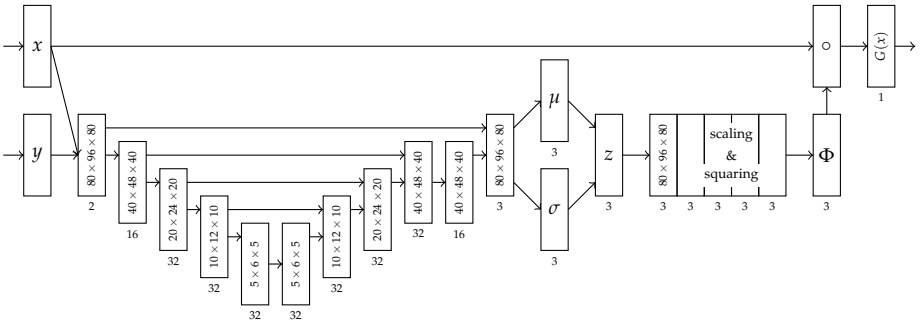



Figure 2.1: Voxelmorph for diffeomorphic image registration as proposed in [8]. The UNet-based encoder receives both $x$ and $y$ as inputs and approximates the distributon $q_\psi$ from which the velocity field $z$ is sampled. Subsequently, $z$ is integrated using a configurable number of scaling and squaring layers resulting in the final deformation field $\Phi^{(1)}$ which is then applied to the input image $x$.

which is then integrated using scaling and squaring layers newly introduced in [8] 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 $\Phi_{z_k}$ 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 [20] optimizer.

## 2.2 Adaptation for Brain Aging

While the tasks of brain registration and generative brain aging may not appear to have much in common superficially, both can be described in terms of learning a deformation function. As such, the approach used in [4] can be adapted for the brain aging setting. 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 [4] 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, therefore increasing the relative magnitude of the noise introduced as part of the reconstruction term in Equation 2.8. In turn, this lowers the model's ability to capture subtle changes which may negatively affect its performance in the brain aging setting. While this problem can be addressed by lowering the hyperparameter $\sigma$, this comes at the cost of decreased generalization as the model is forced to produce results which are progressively closer to the target image $y$.

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 scanning 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) [13].

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

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

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

$$\begin{aligned}\min_G \max_D V(G,D) = \mathbb{E}_{(x,y,t)\sim p_{data}}[\log D(x,y,t)]\,] \\ -\mathbb{E}_{(x,t)\sim p_{data}}[\,1-\log D(x,G(x),t))\,]\end{aligned} \tag{2.10}$$

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 [14].

### 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 $t$. As a result of this modification, our model can predict and be trained on image pairs with arbitrary time steps greatly increasing the available training data as well as its ability to generalize over different time ranges.

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 $\varepsilon$ by scaling and squaring, followed by iterative composition of $\Phi^{(\varepsilon)}$. While this is much faster in practice, the choice of $\varepsilon$ 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 $4, 2, 1, 1/2, \ldots$ 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 approximated to within a temporal precision factor $\varepsilon$ by composing 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 steps as laid out in algorithm 1. Refer to Figure 2.2 for a visual example of one such composition.

The temporal precision $\varepsilon$, 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, $\varepsilon$ 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.

### 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.

**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**

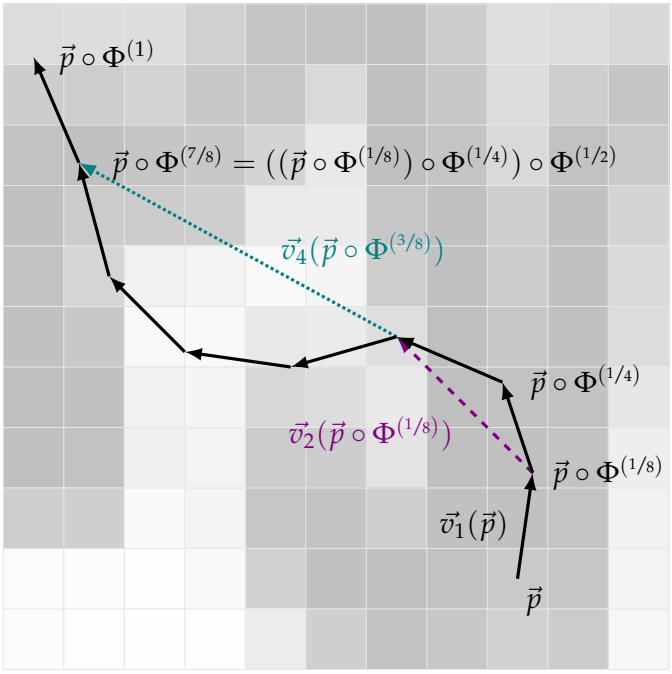



Figure 2.2: Arbitrary time step scaling and squaring with $T = 2$ squaring and $2^{T+1} = 8$ atomic steps, shown for one voxel $\vec{p}$. The deformation $\Phi^{(t)}$ can be approximated at any time step $t \in [0, 1]$ by composing a subset of intermediate deformations. Note that in practice, larger $T$ 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

While the diffeomorphic approach encourages the model to generate realistically aged $G(x) = x \circ \Phi^{(t)}$ with respect to time step $t$, we strengthen this further by using a pre-trained age regressor $R$ to estimate the apparent age of $G(x)$. In this setting, $R$ is a model which estimates a patient's age based on an MRI scan of their brain. Let $a_x$ denote a patient's age at the time of taking image $x$ and $\hat{a}_x = R(x)$ 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 inaccuracies 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.

$$\begin{aligned} &H(p, q) = -p \log q - (1 - p) \log(1 - q) \\ &(1) \quad \mathcal{L}_{dx}(x, y, C) = H(d_y, \hat{d}_{G(x)}) \\ &(2) \quad \mathcal{L}_{dx}(x, y, C) = H(\hat{d}_y, \hat{d}_{G(x)}) \end{aligned} \tag{2.14}$$

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 [5] and [30], we introduce an additional loss term intended to prevent the model from introducing drastic changes 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 difference is primarily caused by differences not related to aging. As a side note, the weak correlation between the $L_1$ difference and time step $t$ also indicates that the metric is not well suited to validate our the aging performance of our generative model.

#### 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

$$\mathcal{L}_G = \mathcal{L}_{ws} + \lambda_{kl}\ \mathcal{L}_{kl} + \lambda_{age}\ \mathcal{L}_{age} + \lambda_{dx}\ \mathcal{L}_{dx} + \lambda_{sim}\ \mathcal{L}_{sim} + \lambda_{sparse}\ \mathcal{L}_{sparse} \tag{2.17}$$

where $\mathcal{L}_{ws}$ is the generator's component of the Wasserstein loss function and $\mathcal{L}_{kl}$ consists of the covariance and precision terms from Equation 2.8. All $\lambda$ in the objective as well as $\lambda_{prior}$ in Equation 2.8 are treated as model hyperparameters.

### 2.2.4 Network Architecture

Based on architectures applied to similar problems such as [30] and [5], 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,

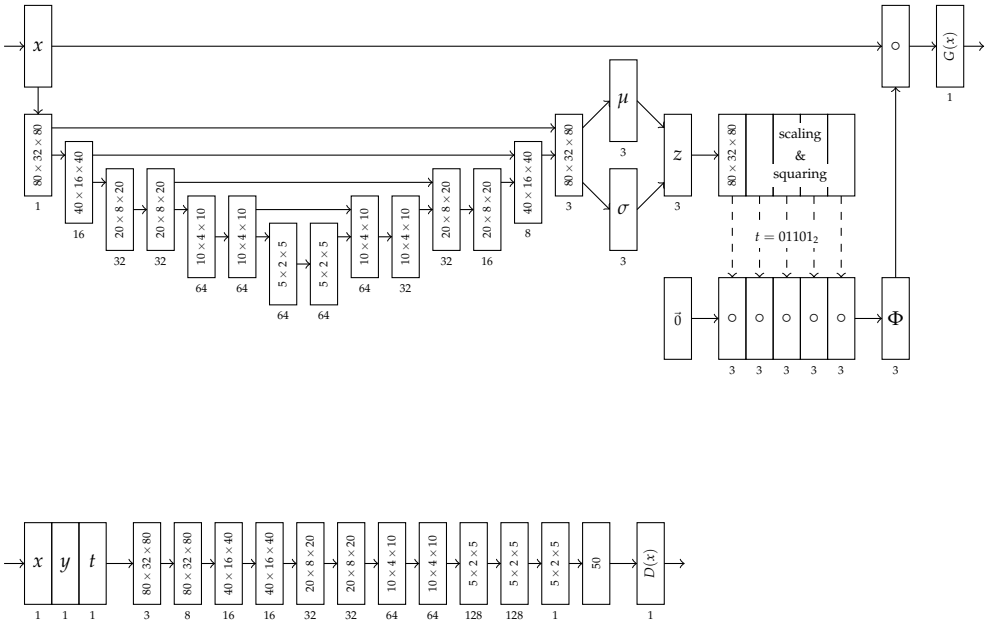






Figure 2.3: Overview of our generator and critic network architectures. The outputs of the squaring layers are applied selectively depending on the value of $t$. In the critic, depending on the step of the GAN training procedure, input $y$ is either a real or a generated image.

extracted from the UNet, to capture differences that are independent of time step $t$ such as misalignments introduced during scanning or preprocessing. More specifically, we fork the UNet's decoder into two distinct paths and bypass the scaling and squaring layers for one of them, resulting in two deformation maps to be applied to the input image $x$. 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.

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 $t$.

As shown by [30], the ability to generate realistic predictions is beneficial in the early detection of Alzheimer's Disease onset. Generative models are of particular interest since existing diagnostic tools, such as diagnosis classifiers operating on MRI scan, can be directly applied to predictions $\hat{y}$ 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 $\Delta$. In this context, we distinguish between progressive cases (pMCI) for which the diagnosis converts within $\Delta$, 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 $\Delta$, that is

$d_i$ = MCI for all visits $v_i$ and $\max_{v_i \in V(s)} t_i - \min_{v_i \in V(s)} t_i \geq \Delta$, where $V(s)$ is the set of visits of subject $s$.

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 phase, 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 [5], 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. Finally, since our model is probabilistic, multiple different predictions for the same input image $x$ can be generated, which 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) [15]

and Australian Imaging Biomarkers and Lifestyle (AIBL) [12] 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[1] 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 [17], using the `flirt` [19] [16] and `bet`[2] [26] [18] 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`[3] [31] 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 as shown in [10]. 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 approach 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.

[1] Reference: MNI152_T1_1mm

[2] Parameters: `-v -f 0.3 -g -0.1`

[3] Parameters: `-t 1 -n 3 -l 20 -I 4 -O 4 -B`

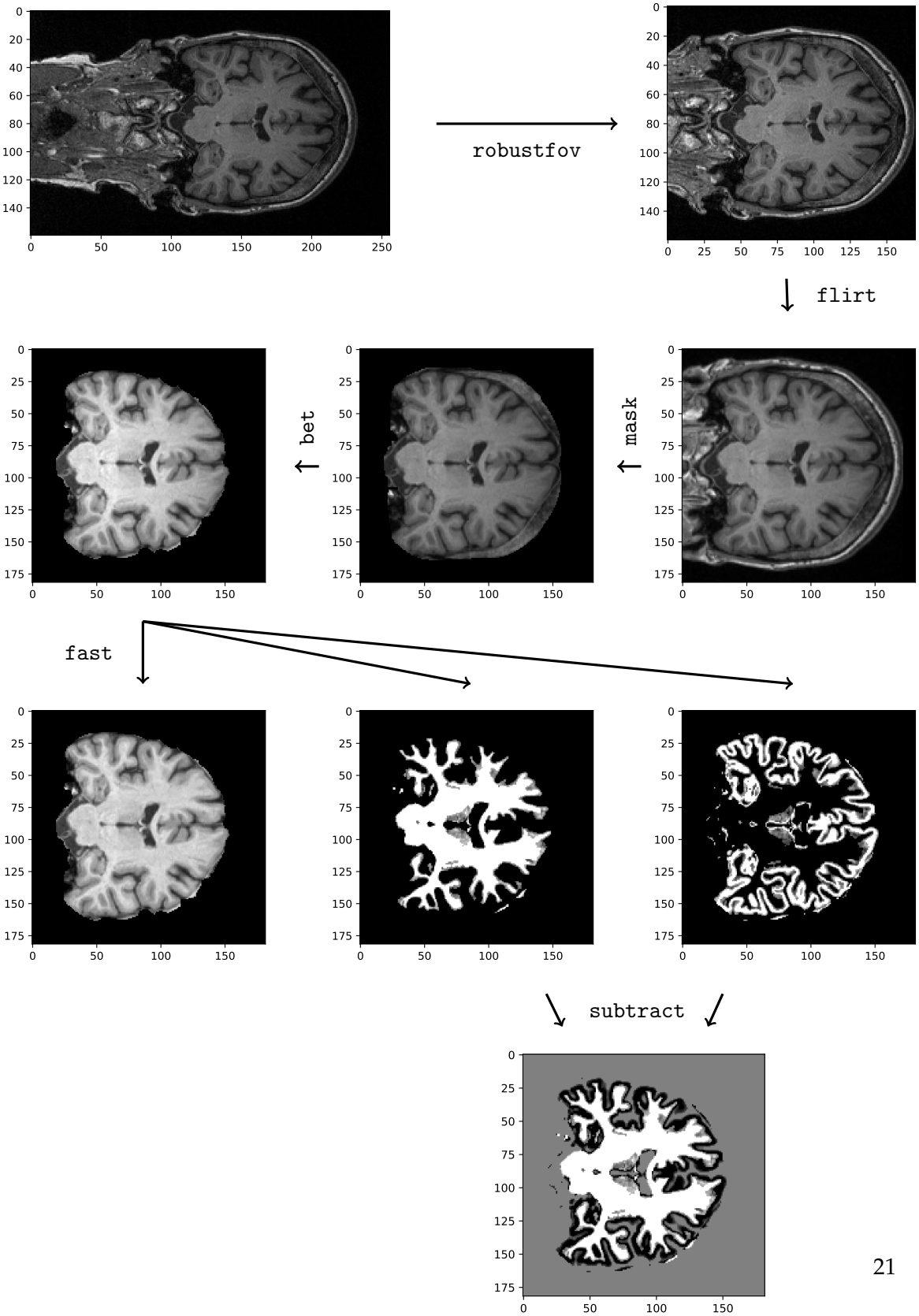





Figure 4.1: Preprocessing pipeline visualized for a sample image.

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 number 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 ½ 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 $B$, e.g. pairs and single images, with $i \neq j$, the intersection $\mathcal{S}_i^A \cap \mathcal{S}_j^B = \varnothing$ is guaranteed to be empty. In addition to these five splits, we keep a separate $\mathcal{S}_{conv}$ containing all images of subjects which are contained in the MCI conversion data set explained below.

#### 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.

| 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.

### 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:

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.

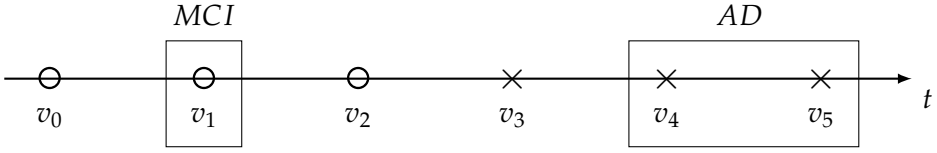
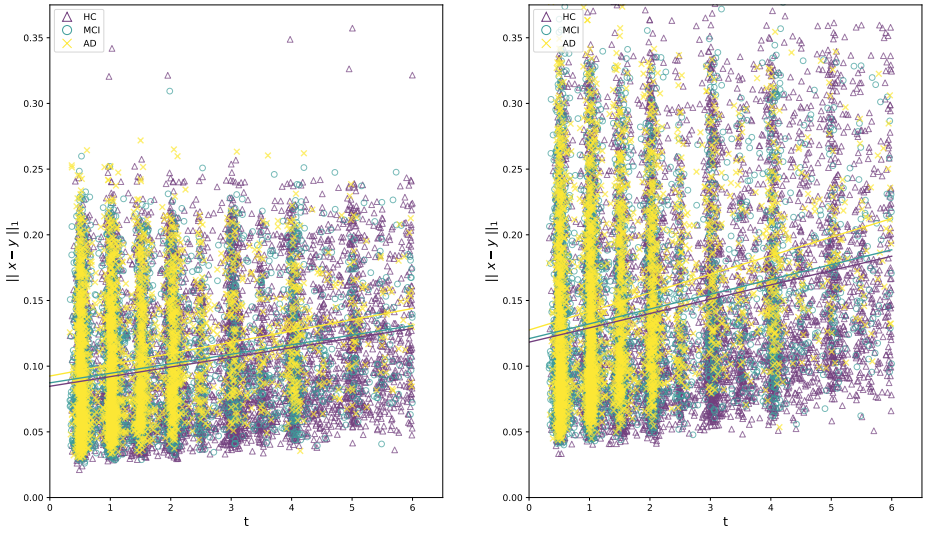



Figure 4.2: Illustation of MCI and AD visits, ○ = MCI, × = AD

### 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 time steps with median of 1.53, mean of 2.00 and a standard deviation of 1.47 years. To mitigate this, we calculate sample weights $w = (|t - \bar{t}| + 1)^{1/2}$, where $\bar{t}$ is the mean over all time steps $t$. 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 $v_j$

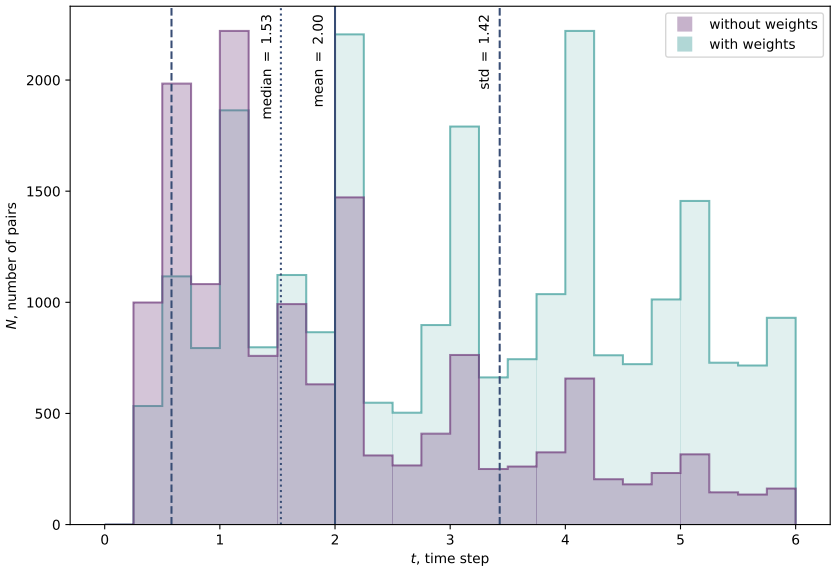



Figure 4.4: The histogram for time step $t$ across all pairs in the pairs data set. To mitigate the bias towards smaller $t$, we calculate sample weights to be used during training of our generator models.

to be $\Delta = 4$ based on the available data as well as previous work in [30]. In general, multiple viable image pair combinations exist for each subject. We prioritize matching $\Delta$ followed by centering the point in time where the diagnosis change occurs within $\Delta$. 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.

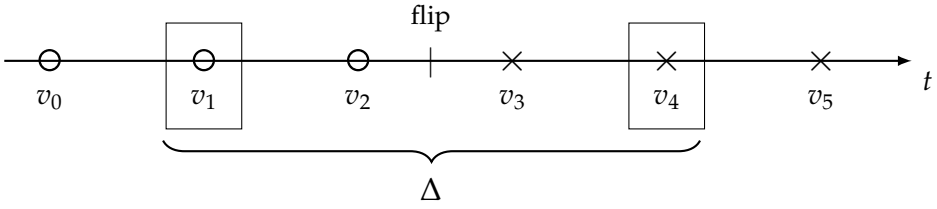



Figure 4.5: Illustration of a pMCI image pair, ○ = MCI, × = AD

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 obtain an estimation of the age label, and by extension the time step, by applying a pre-trained age regressor to our model's outputs. 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 of which eight use 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 on 50'000 batches of 32 samples each optimizing the absolute mean error as its objective function. Using our 5-fold data split, we perform cross validation which yields a mean loss of 3.86 years with a standard deviation of 3.05 (see Table 5.1). Figure 5.1 shows the estimated age label $\hat{a}_x$ 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 data set's mean age of 75 years with a tendency to be 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 $(x, y)$. As disucussed in section 2.2.3, we expect errors to partially cancel out in this setting and thus evaluate the regressor's performance on our data set of image pairs to confirm this hypothesis. As

before, we perform 5-fold cross validation resulting in a mean of X.XX and a standard deviation of Y.YY, and report the detailed results in Table 5.1. Figure 5.2 compares the absolute losses for $x$ and $y$ and shows the error cancelling effect. Furthermore, we also estimate the age difference $\hat{a}_y - \hat{a}_x$ for one split and visualize it against the ground truth time step in Figure 5.3. Using linear regression on all data points, we obtain an intercept of 0.13 and a slope of 0.61, indicating that while the age regressor is capable of distinguishing different time steps, its estimates tend to be too low. This can be explained by closer examination of Figure 5.1 which demonstrates that since the estimate $\hat{a}_x$ tends to the mean for labels futher away from it, relative age differences are subject to a shrinking 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.

| Split | absolute age | | | | age difference | |
|---|---|---|---|---|---|---|
| | absolute | | squared | | absolute | squared |
| | $\mu$ | $\sigma$ | $\mu$ | $\sigma$ | | |
| $\mathcal{S}_0$ | 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 our age regressor model on the tasks of predicting the age label of a single image as well as estimating the age difference for a pair of images. We also compare models trained using absolute and squared loss objectives.

### 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

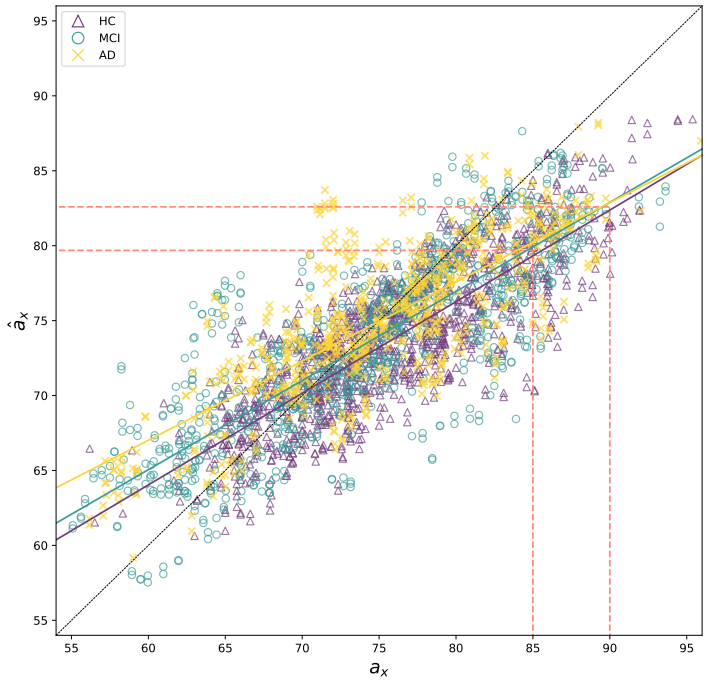



Figure 5.1: Age regressor estimates on 2937 validation samples from our base image data set. Note how the predicitions tend to the mean, with a slope of 0.58 compared to the target of 1 indicated in black. The subject's diagnosis does not appear to have a significant impact on the performance. Indicated in red, we observe that as we move away from the mean, the predicted relative time differences between two images shrink as a consequence of the estimator's tendency to the mean.

our training and validation sets to a subset of images we consider *firmly MCI* or *firmly AD*. Note that we exclude all subjects in the healthy control group HC and use a binary classifier focussing on the 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 minimized 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, yielding in an accuracy of 70.10% and an $F_1$-score of 70.60%. 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 select the run with the highest accuracy. Each model is trained on 10'000

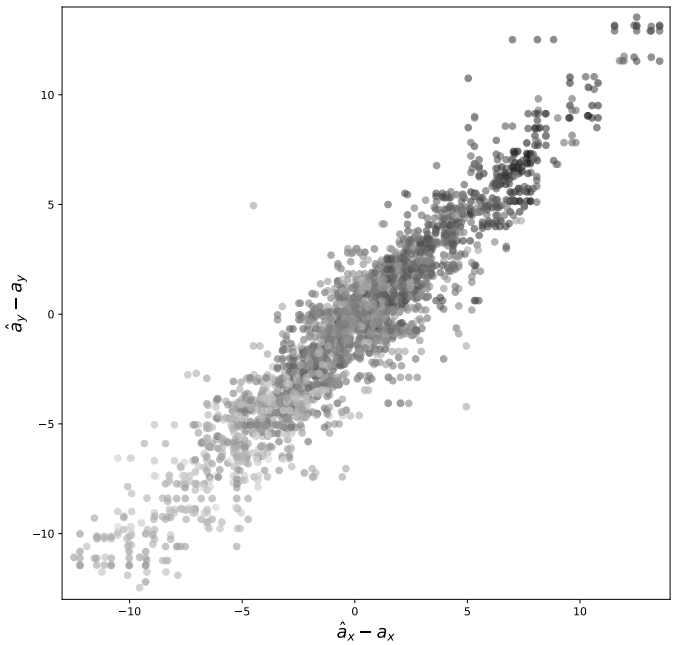


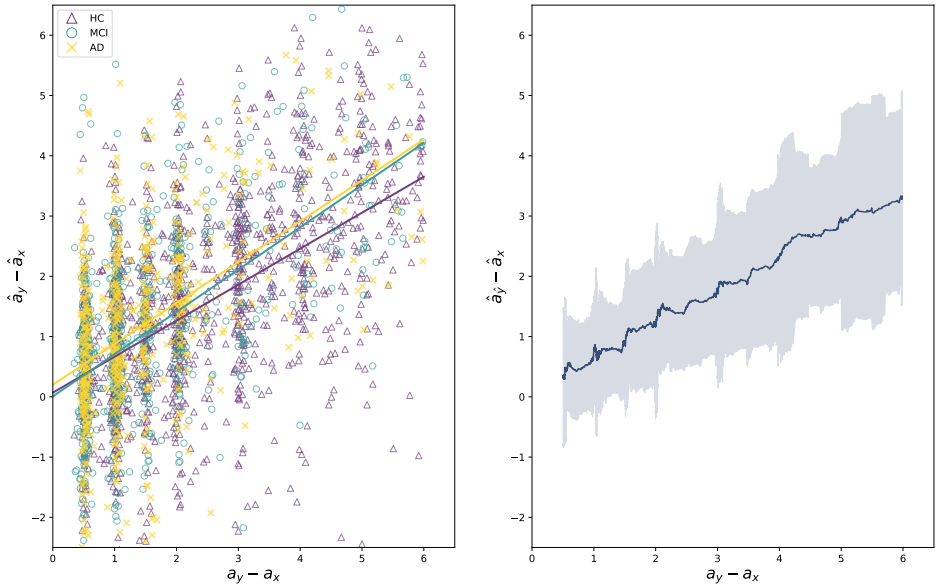

Figure 5.2: Visualization of the error cancelling effect on 2651 validation image pairs from our pairs data set. Each axis represents the difference between the predicted and true age labels for one image. We observe that while the absolute error for a prediction on a single image is quite significant with a mean of 3.58, 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. The brightness of each data point corresponds to the averaged true age labels of the image pair, revealing that the absolute loss is tied to the age label.

batches of 32 samples each.

## 5.3 Diffeomorphic Models

### 5.3.1 Synthetic Data

To validate our modifications to the Voxelmorph 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




(a) diagnoses (b) mean and standard deviation

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 0.13 and a slope of 0.61. 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.

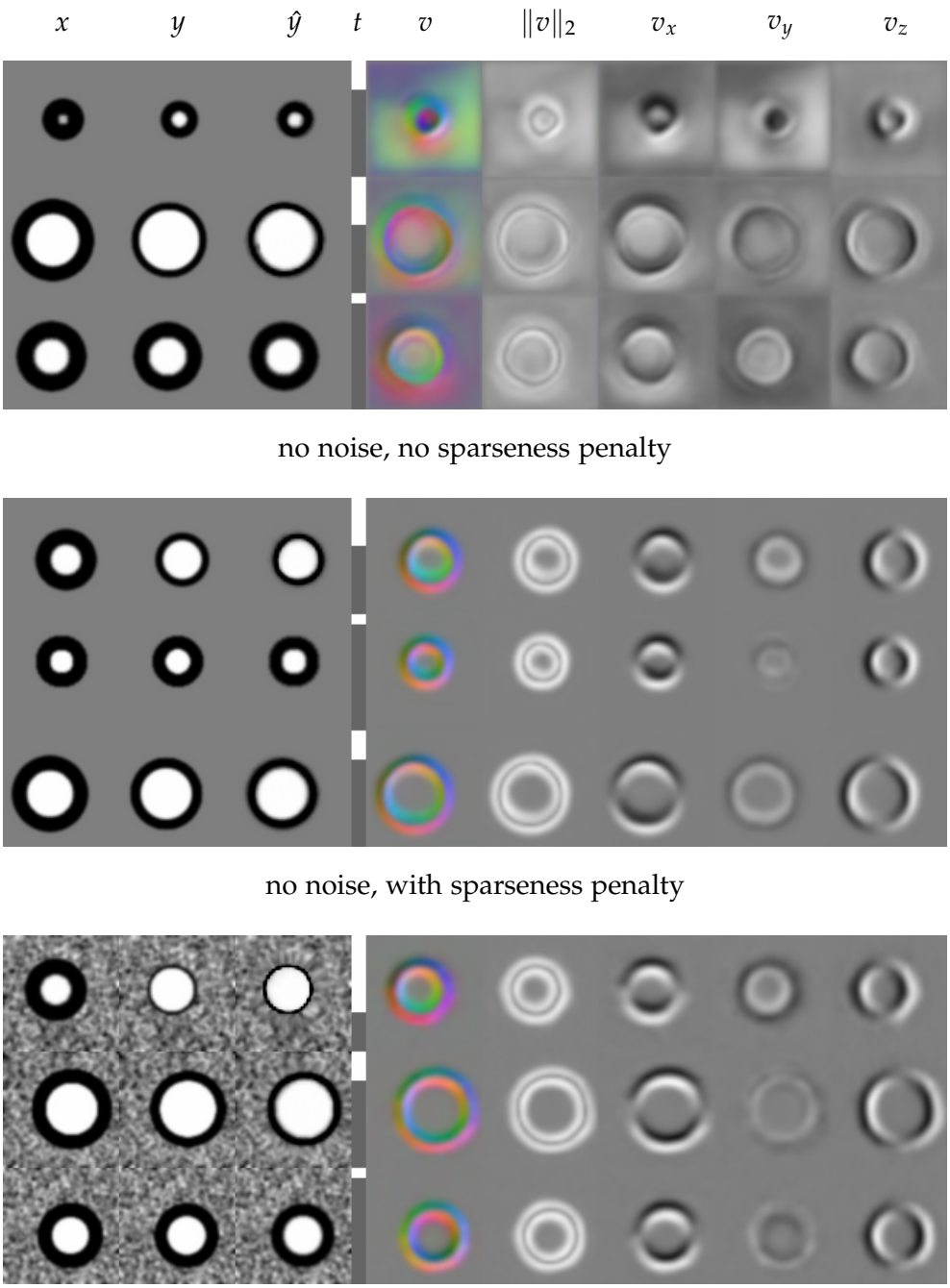
the deformation field in areas of little change. Note that neither the age regressor nor the diagnosis classifier are used in any of these experiments as there are no corresponding features in the synthetic data samples.

### 5.3.2 MRI Data

We train the brain aging model in a number of different configurations and on different data splits, as listed in Table 5.3.

For the remaining hyperparameters, we use $\lambda_{kl} = 10$, $\lambda_{prior} = 25$, $\lambda_{sim} = 200$ and $\lambda_{sparse} = 0$ due to negative performance impacts even for small sparseness loss weights. During training, the velocity field $v$ is integrated using up to $T = 7$ squaring layers, 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 the regressor and the classifier used as loss terms in configurations 2, 3, and 4 are trained on $\mathcal{S}_{valid}$ and evaluated on $\mathcal{S}_{train}$ in order to prevent




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.

| Split | Run 1 | | 2 | | 3 | | 4 | | 5 | |
|---|---|---|---|---|---|---|---|---|---|---|
| | acc | $F_1$ | acc | $F_1$ | acc | $F_1$ | acc | $F_1$ | acc | $F_1$ |
| $\mathcal{S}_0$ | 68.58 | 66.76 | 69.08 | 66.76 | **70.74** | **69.50** | 67.81 | 65.39 | 63.74 | 63.51 |
| $\mathcal{S}_1$ | **70.32** | **72.07** | 66.13 | 68.13 | 68.10 | 70.26 | 69.95 | 72.01 | 70.07 | 71.84 |
| $\mathcal{S}_2$ | 67.33 | 70.24 | 67.56 | 69.58 | 65.33 | 67.97 | **67.78** | **69.79** | 64.89 | 67.76 |
| $\mathcal{S}_3$ | **71.68** | **71.03** | 70.14 | 69.86 | 70.26 | 69.94 | 68.13 | 67.31 | 70.50 | 68.52 |
| $\mathcal{S}_4$ | 66.71 | 67.60 | 66.58 | 67.59 | **69.97** | **70.59** | 68.15 | 68.56 | 64.88 | 64.28 |

Table 5.2: Cross validated accuracy and $F_1$-score of our classifier models.

| # | $\lambda_{age}$ | $\lambda_{dx}$ | $\mathcal{S}_{train}$ | $\mathcal{S}_{valid}$ | $\mathcal{S}_{conv}$ | Pairs |
|---|---|---|---|---|---|---|
| 1 | - | - | $\mathcal{S}_1\,\mathcal{S}_2\,\mathcal{S}_3\,\mathcal{S}_4$ | $\mathcal{S}_0$ | $\mathcal{S}_{conv}$ | any $\rightarrow$ any |
| 2 | 100 | - | $\mathcal{S}_1\,\mathcal{S}_3$ | $\mathcal{S}_2\,\mathcal{S}_4$ | $\mathcal{S}_0$ | any $\rightarrow$ any |
| 3 | - | 100 | $\mathcal{S}_1\,\mathcal{S}_3$ | $\mathcal{S}_2\,\mathcal{S}_4$ | $\mathcal{S}_{conv}$ | any $\rightarrow$ MCI/AD |
| 4 | 100 | 100 | $\mathcal{S}_1\,\mathcal{S}_3$ | $\mathcal{S}_2\,\mathcal{S}_4$ | $\mathcal{S}_0\,\mathcal{S}_{conv}$ | 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 model configurations.

them from overfitting on the generator training data.

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 model until convergence, in the the case of configurations 1, 5 and 6, and until $\mathcal{L}_{age}$ or $\mathcal{L}_{dx}$ display signs of overfitting. Following the training procedure in [13], 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. We present three sample generator outputs in Figure 5.5.

#### Follow-Up Prediction

As our first experiment, we predict follow-up images using the actual time steps in the pairs data set. Using the pre-trained age regressor, we evaluate the performance on this task for configurations 1 and 2 and present the results in Figure 5.6.

TODO(show example)

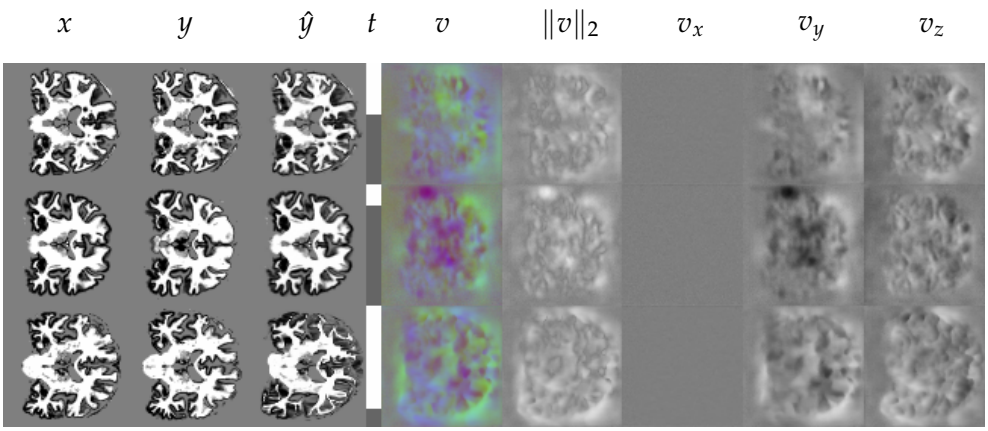



Figure 5.5: Model outputs generated using configuration 1 on validation image pairs. Each row represents one sample output consisting of the original, the target, and the generated image, the time step $t$ scaled to $[0, 1)$, the velocity field $v$ and its magnitude, as well as its separate dimensions..

### Fixed Time Step Prediction

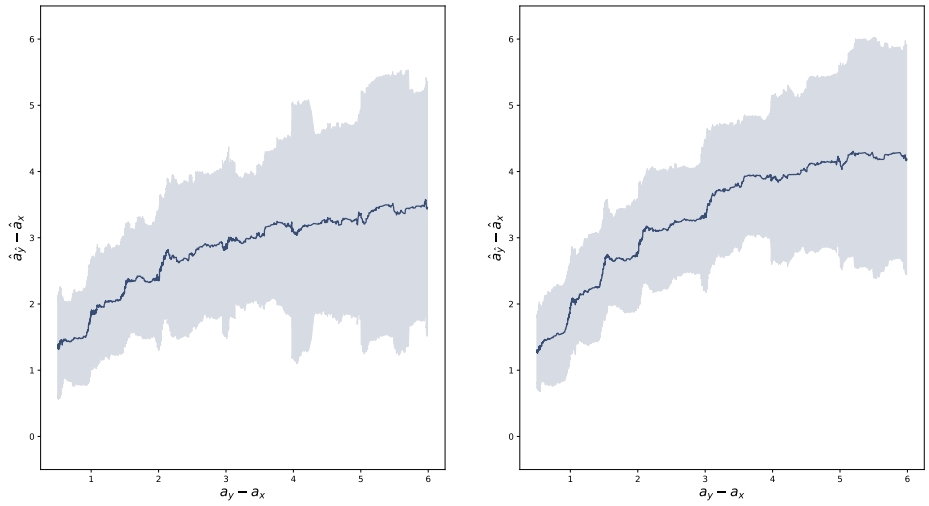
To evaluate the model's ability to generate follow-up images at fixed time steps, we predict images at $t \in \{1, 2, 4, 6, 8\}$ years and estimate the age labels using the age regressor. The results for configurations 1 and 2 are presented in Table 5.4 and visualized in Figure 5.7. We run configuration 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. Both configurations are capable of producing older looking images and the age regre

| # | $t = 1$ | | 2 | | 4 | | 6 | | 8 | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $\mu$ | $\sigma$ | $\mu$ | $\sigma$ | $\mu$ | $\sigma$ | $\mu$ | $\sigma$ | $\mu$ | $\sigma$ |
| 1 | 1.90 | 0.95 | 2.60 | 1.36 | 3.30 | 2.24 | 3.51 | 3.01 | 3.35 | 3.69 |
| 2 | 2.08 | 0.79 | 3.02 | 1.10 | 4.01 | 1.74 | 4.66 | 2.30 | 5.02 | 2.78 |

Table 5.4: Mean and standard deviation of the estimated age labels on outputs generated for various different time steps.

### 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 data sets which are correspondingly limited. We qualitatively and quantitatively ob-




(a) Configuration 1 (b) Configuration 2

Figure 5.6: Rolling mean and standard deviation of the age labels predicted by the age regressor on images generated using configurations 1 and 2. For comparison, Figure 5.3 shows the results on real image pairs.

serve higher mean deformation magnitudes for configuration 6 and visually compare the effects of the models applied to a healthy subject in Figure 5.9 and a subject affected by AD in Figure 5.8.

#### Long-Term Prediction

As mentioned in section 3.2, our model architecture allows generating images for very large time steps, such as 50 years, at reasonable computational cost. However, while the velocity field predicted by the generator is time-invariant in theory, its accuracy is highly dependent on the distribution of time steps occuring in the training data. In practice, using time steps around and beyond the maximum step in the training data very quickly result in unrealistic looking outputs. We generate images at time steps of $t \in \{2, 4, 6, 8, 10\}$ years for a random sample using configuration 1 and present the results in **??**. Superior results could likely be obtained by using an iterative approach similar to [30], using multiple smaller time steps in sequence. TODO(fig)

#### Conversion Prediction

As described in section 3.1, we evaluate the performance of our generative model on the MCI conversion prediction task. Given an image pair $(x, y)$

we generate $G(x)$ with time step $t = 4$ using our generative model and use a diagnosis classifier to predict $p_{AD}(G(x))$ and use thresholding to predict whether the subject is progressive or stable.

Keeping in mind the imbalanced nature of the conversion data set, we use balanced accuracy and $F_1$-score as our metrics

$$acc = \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right) \tag{5.1}$$

$$F_1 = 2\left(\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}\right) \tag{5.2}$$

where

$$\text{precision} = \frac{TP}{TP+FP}$$

$$\text{recall} = \frac{TP}{TP+FN}$$

and $TP$ = true positives, $TN$ = true negatives, $FP$ = false positives and $FN$ = false negatives.

Given a set of images $\{x\}$ from our conversion data set, we independently calculate the balanced conversion accuracy and the $F_1$-score as follows:

1. We use a diagnosis classifier to predict the probabilities $p_{AD}(x)$ for each $x$
2. We split the conversion data set into 5 balanced splits $\{\mathcal{T}_i\}_{i \in 0 \,..\, 4}$ and calculate $t_i$ for each split as the threshold which maximizes the target metric on $\{\mathcal{T}_j\}_{i \neq j}$.
3. We calculate the metrics for each split $\mathcal{T}_i$ using the corresponding $t_i$ and take the mean over all splits
4. We repeat steps 2 and 3 five times and take the mean for both metrics
5. We repeat steps 1 to 4 for each of the five best classifier in Table 5.2 and again take the mean

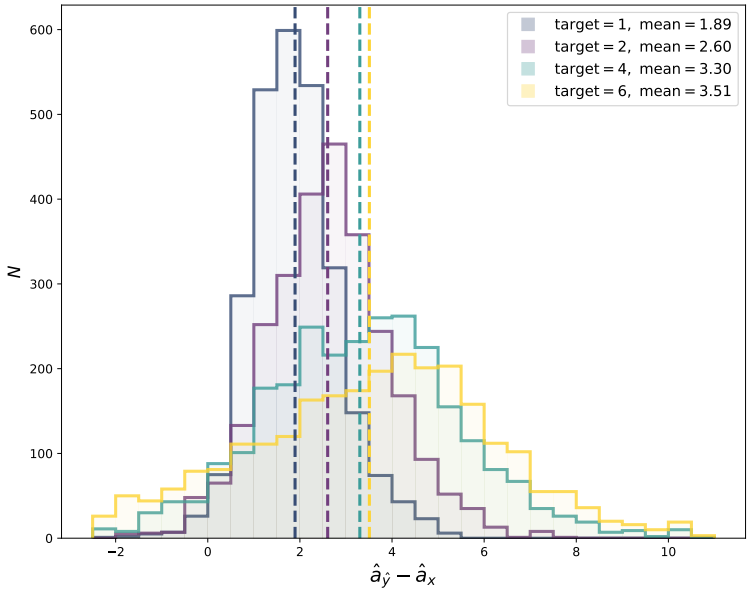
We calculate the balanced accuracy and $F_1$-score on the set of base images $x$ as a baseline and similarly on the set of target images $y$ as an upper bound. Finally, we calculate the metrics for images generated using configurations 1 to 4 and compare the results in Tables 5.5 and 5.6.

| Config | | Accuracy | | Classifier | | | | |
|---|---|---|---|---|---|---|---|---|
| | | $\mu$ | $\sigma$ | 1 | 2 | 3 | 4 | 5 |
| $x$ | | 63.81 | 1.67 | 64.17 | 65.41 | 64.18 | 61.06 | 64.25 |
| $y$ | +12.09 | 75.90 | 0.90 | 74.52 | 76.17 | 76.69 | 75.80 | 76.31 |
| 1 | +2.39 | 66.20 | 1.52 | 65.52 | 65.94 | 65.57 | 65.60 | 68.38 |
| 1* | +2.95 | 66.75 | 2.33 | 69.76 | 65.83 | 69.02 | 64.59 | 64.58 |
| 2 | +0.17 | 63.98 | 3.84 | 68.40 | 60.62 | 68.53 | 60.15 | 62.22 |
| 2* | +2.62 | 66.43 | 3.74 | 67.92 | 71.00 | 68.97 | 61.17 | 63.11 |
| 3 | +0.90 | 64.71 | 2.21 | 65.38 | 67.98 | 62.73 | 63.01 | 64.46 |
| 4 | +0.90 | 64.71 | 2.97 | 63.37 | 66.44 | 68.88 | 63.03 | 61.83 |

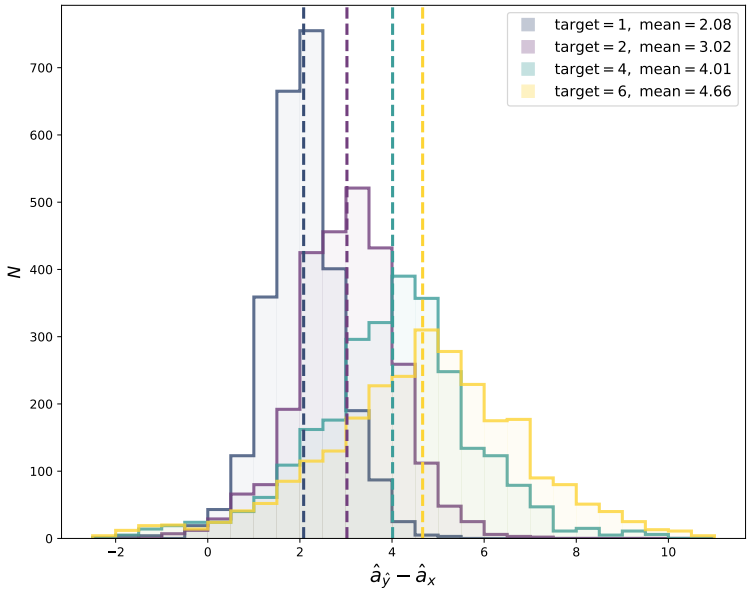
Table 5.5: TODO conv accuracy

| Config | | $F_1$-Score | | Classifier | | | | |
|---|---|---|---|---|---|---|---|---|
| | | $\mu$ | $\sigma$ | 1 | 2 | 3 | 4 | 5 |
| $x$ | | 54.76 | 2.77 | 52.48 | 56.36 | 55.33 | 51.49 | 58.14 |
| $y$ | +14.61 | 69.37 | 1.14 | 67.77 | 69.71 | 70.26 | 68.93 | 70.20 |
| 1 | +3.97 | 58.73 | 2.08 | 57.58 | 59.08 | 56.93 | 58.72 | 61.34 |
| 1* | +5.07 | 59.83 | 2.77 | 63.10 | 60.32 | 61.81 | 58.34 | 55.60 |
| 2 | +1.17 | 55.93 | 4.03 | 61.24 | 51.55 | 59.31 | 52.03 | 55.52 |
| 2* | +4.42 | 59.18 | 3.75 | 61.22 | 62.82 | 61.72 | 53.04 | 57.09 |
| 3 | +2.39 | 57.15 | 2.87 | 59.50 | 60.74 | 54.22 | 55.80 | 55.50 |
| 4 | +0.93 | 55.69 | 4.81 | 55.51 | 58.24 | 62.04 | 53.76 | 48.86 |

Table 5.6: TODO conv F1




(a) Configuration 1




(b) Configuration 2

Figure 5.7: Results of the fixed time step experiments for configurations 1 and 2. We predict follow-up images at time steps 1, 2, 4 and 6 and present the results as a histogram. Note that the standard deviation increases significantly for larger steps but does so less strongly for configuration 2 which uses an age regressor in its loss function.

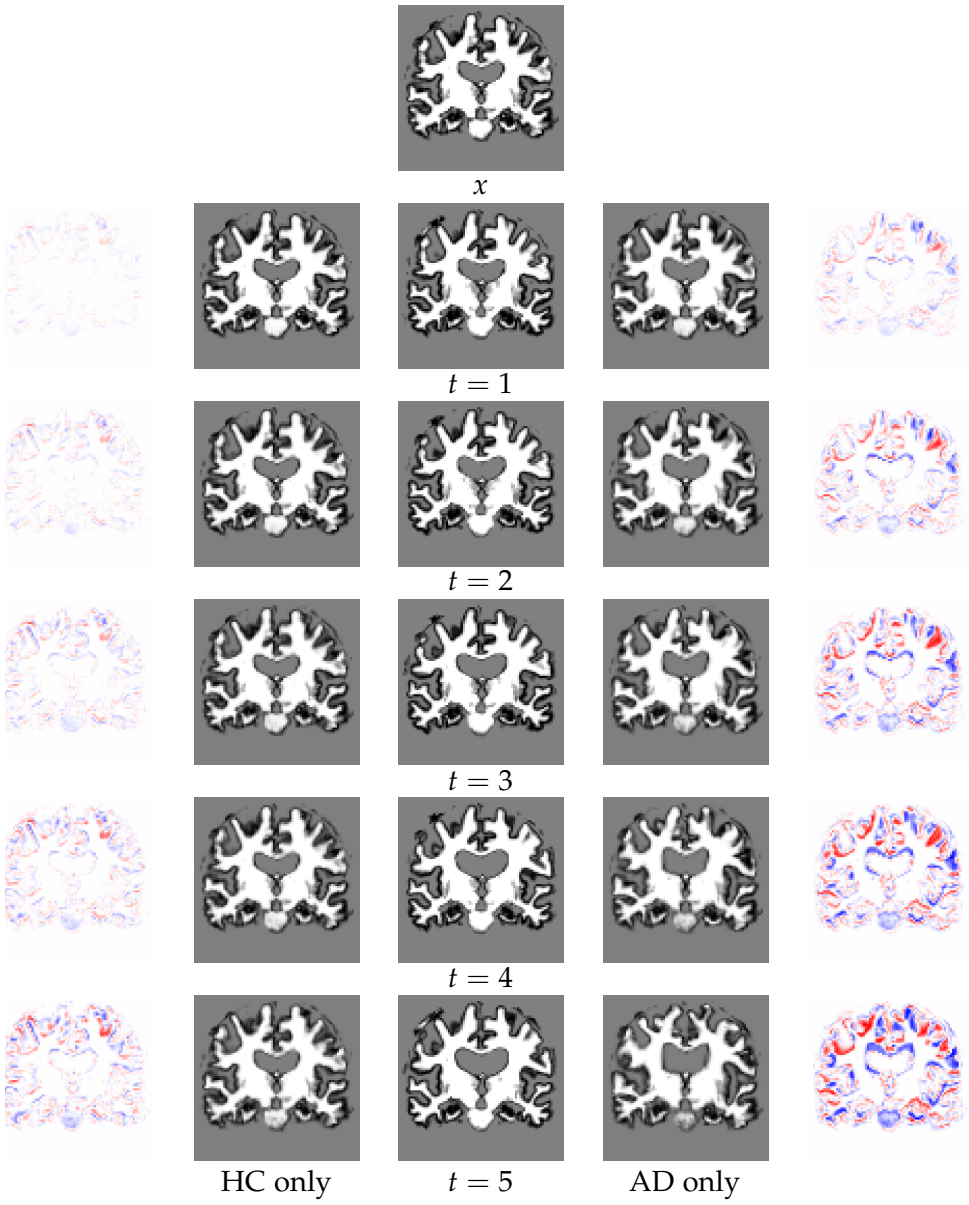



Figure 5.8: The center column consists of a series of real images from a **subject diagnosed with Alzheimer's Disease**, taken at one-year intervals steps, top to bottom. The two columns to the left show the predicted images at the same time steps using a model trained exclusively on healty patients, 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.

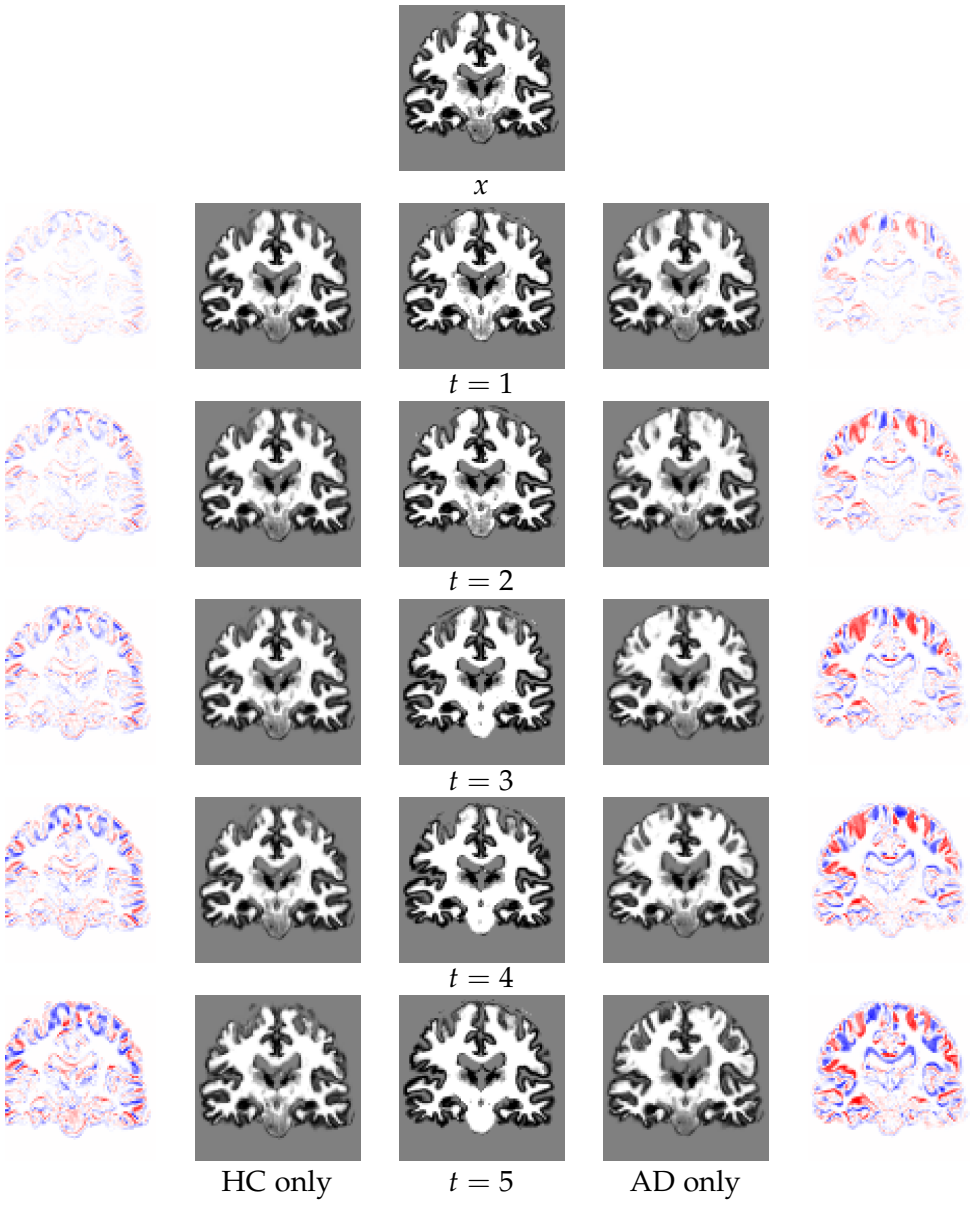



Figure 5.9: The center column consists of a series of real images from a **healthy subject**, taken at one-year intervals steps, top to bottom. The two columns to the left show the predicted images at the same time steps using a model trained exclusively on healty patients, as well as the difference 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.

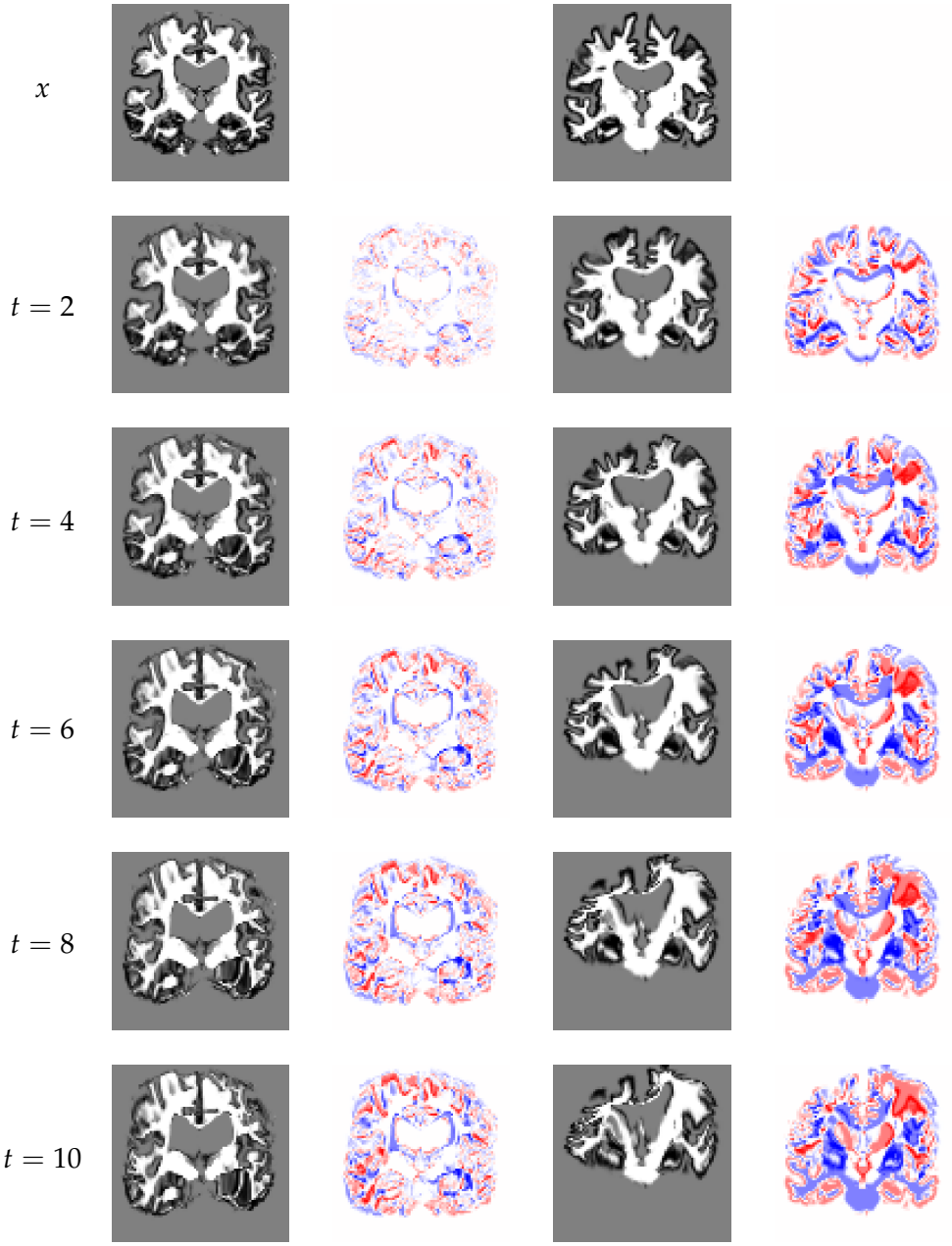



Figure 5.10: Sequence of a base image $x$ and images generated for time steps of 2, 4, 6, 8 and 10 years, as well as the corresponding difference maps to the base image. We pick two samples which represent common outcomes in our data sets and use configuration 1 to generate the predictions.

Chapter 6

# Discussion

Using the age regressor as part of the loss objective has a clear impact on the model's performance, however, the same does not appear to be the case for the diagnosis classifier. While this may a consequence of the more limited amount of training data available to these configurations, the diagnosis classifier also appears to be very sensitive to both data splits as well as its starting conditions and therefore, might not be suitable to be used as a target in the objective function. Most probably, this is a consequence of the fairly limited number of training samples available to the classifier, which could be addressed by incorporating image data from other studies. As touched upon in subsection 4.2.2, our use of segmentation-derived images opens up the possibility of using different image modalities such as PD or T2-weighted MRI scans, greatly expanding the number of viable data sources. We also note that downsampling the images may have a significant negative performance impact on the classifier and that therefore, better results could be obtained using architectures operating on full-size and uncropped images. Beyond more extensive and higher resolution image data sets, additional data modalities such as cognitive test results as well as biomarkers could be incorporated. Furthermore, both the generator and the classifier might benefit from being trained on longitudinal data, that is, data from multiple visits.

Due to the comparatively high number of components in our architecture, attributing variations in the model's performance to changesof its architecture is non-trivial. In particular, the large number of hyperparameters combined with training durations of up to two days renders systematic parameter tuning computationally infeasible. On the other hand, we also observe that some of the hyperparameters have limited to no effect, suggesting that the architecture could be simplified.

Finally, we address the issue of validating generative outputs by using an age regressor and find this to be a meaningful metric. However, less abstract

and more image based objective metrics might be interesting to explore in the context of brain aging models.

Chapter 7

# Conclusion

# 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] Jeffrey De Fauw, Joseph R Ledsam, Bernardino Romera-Paredes, Stanislav Nikolov, Nenad Tomasev, Sam Blackwell, Harry Askham, Xavier Glorot, Brendan O'Donoghue, Daniel Visentin, et al. Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature medicine*, 24(9):1342, 2018.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

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

[18] 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.

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

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

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

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

[23] 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.

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

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

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

[27] 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.

[28] 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.

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

[30] Viktor Wegmayr, Maurice Hoerold, and Joachim M. Buhmann. Generative aging of brain mri for early prediction of mci-ad conversion. *2019 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2019)*, 2019 in press.

[31] 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.

[32] 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.